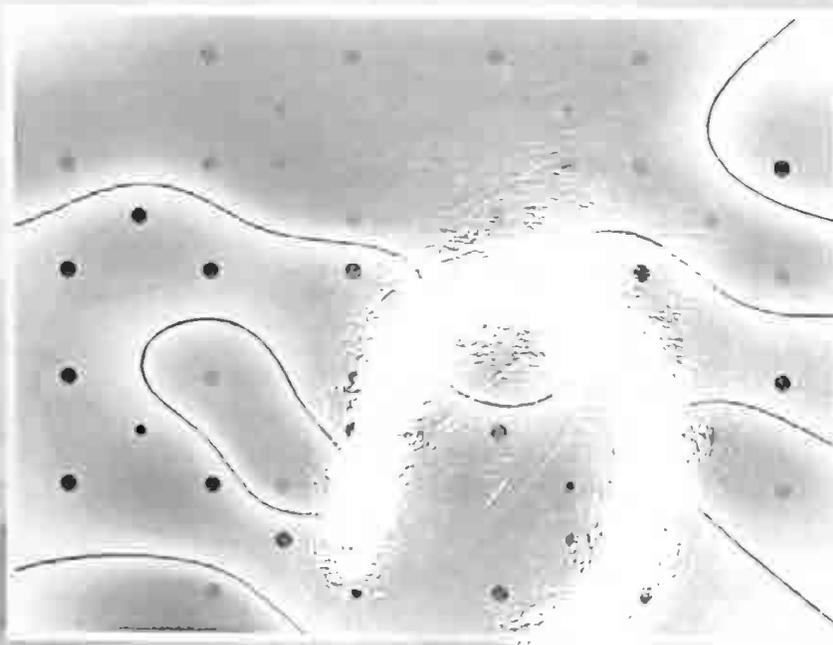


Avances Recientes en Sistemas Inteligentes

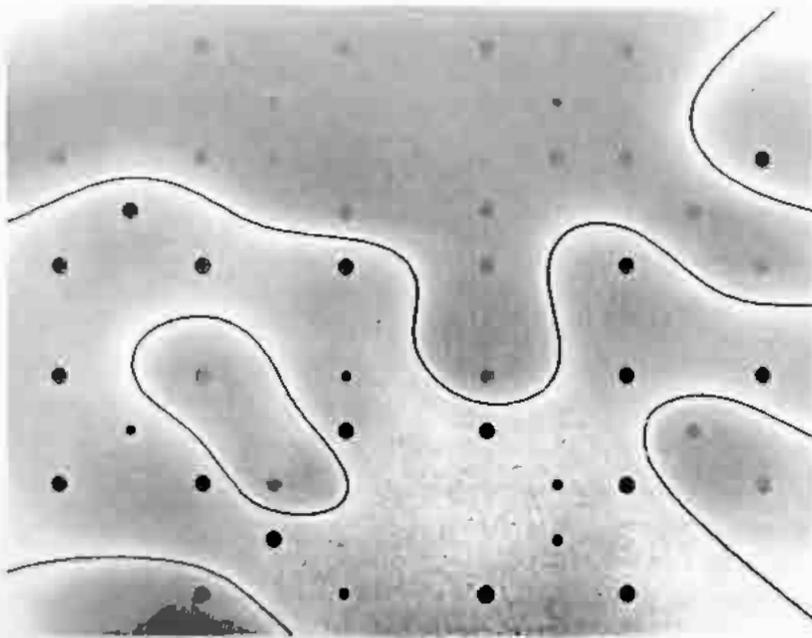


Sociedad Mexicana de Inteligencia Artificial



Miguel González Mendoza
Óscar Herrera Alcántara
Editores

Avances Recientes en Sistemas Inteligentes



Sociedad Mexicana de Inteligencia Artificial

ISBN: 978-607-95367-3-2

No ID Tributaria: SMI871231

PRIMERA EDICIÓN: 2011

Se prohíbe la reproducción total o parcial de esta obra
-incluido el diseño gráfico tipográfico y de portada-,
sea cual fuere el medio, electrónico o mecánico,
sin el consentimiento por escrito del autor.

**MIGUEL GONZÁLEZ MENDOZA
ÓSCAR HERRERA ALCÁNTARA
México, D. F.**

**DISEÑO DE PORTADA: Ágora Medios, S.A. de C.V.
PROCESOS EDITORIALES: Ágora Medios, S.A. de C.V.**

**Impresiones / Printing: 500
Impreso en México / Printed in Mexico**

Prefacio

El esfuerzo aportado por los autores, de manera coordinada con la Sociedad Mexicana de Inteligencia Artificial (SMIA), que publican en Avances recientes en sistemas inteligentes, ha estado vinculado con la necesidad de dar a conocer la investigación y enseñanza de la Inteligencia Artificial en universidades e instituciones de enseñanza superior Mexicanas en lo particular e iberoamericanas en lo general, y se promueve como la publicación de trabajos de investigación derivados de tesis o proyectos, terminados o en proceso, en Español.

Esta publicación contiene 26 artículos seleccionados siguiendo un proceso de revisión de validez internacional. El libro abarca las siguientes áreas de interés:

- Aprendizaje Máquina y Reconocimiento de Patrones
- Sistemas Multi-Agente e IA Distribuida
- Visión por Computadora y Procesamiento de imágenes
- Sistemas Inteligentes Híbridos y Redes Neuronales
- Lógica Difusa, Incertidumbre y Razonamiento Probabilístico
- Sistemas de Tutores Inteligentes
- Procesamiento del Lenguaje Natural
- Sistemas Basados en Conocimientos
- Algoritmos Evolutivos y otros algoritmos inspirados de la naturaleza

Agradecemos a todas las personas involucradas en la organización de esta obra. En primer lugar a los autores de los artículos que constituyen este libro: su calidad y su compromiso con los trabajos de investigación y desarrollo tecnológico que dan valor a esta publicación. Agradecemos a los miembros del Comité Revisor por su profesional trabajo en la evaluación, así como en la selección de los artículos.

Esperamos que los lectores del libro encuentren en él una vía efectiva para descubrir los fundamentos que llevan a la solución de problema basados en inteligencia artificial, así como sus diversas aproximaciones y enfoques que han permitido avanzar en el desarrollo de sistemas inteligentes.

Octubre de 2011
México

Miguel González Mendoza
Oscar Herrera Alcántara

Indice

Asistente de Conducción para Carreteras Basado En Visión Computacional Embebido En FPGA	1
<i>Ricardo Acevedo Ávila, Miguel González Mendoza and Andrés David García García</i>	
Generación de Código libre Multiplataforma de Máquinas de Soporte Vectorial	11
<i>García Gamboa Ariel Lucien, Miguel Gonzalez-Mendoza, Alfonso Calderón-Chávez and Francisco Javier Calderón-Chávez</i>	
Desarrollo de un modelo difuso para diagnóstico de lesiones precursoras de cáncer cérvico uterino	18
<i>David Bertani, Reyna Karem Dominguez Hernandez, Alberto Alfonso Aguilar Lasserre, Blanca Estela Gonzalez Sanchez and Patricia Espíndola Cisneros</i>	
Metaheurístico híbrido: Cómputo evolutivo y recocido simulado	28
<i>Sergia Alex Caballero-Caballero, Jaime Mora-Vargas and Ivo Erasmo Buzón-Cantera</i>	
Esteganografía en Imágenes usando el Factor de Escala para las wavelets ortogonales	40
<i>Blanca E. Carvajal-Gómez, Francisco J. Gallegos-Funes, José L. López-Bonilla and Alberto J. Rosales-Silva</i>	
Estimación Inicial de Habilidad en Evaluadores Adaptables	50
<i>Félix Castro, Joel Suárez and Raúl Chirinos</i>	
Razonamiento Intencional y No-Monotonicidad	59
<i>José Martín Castro-Manzano, Axel Barceló-Aspeitia and Alejandro Guerra-Hernández</i>	
Caracterización de la Maduración Ósea mediante Conjuntos Aproximados y Conjuntos Difusos	69
<i>José Alejandro Cid and Fernando López</i>	
Lenguaje de Definición, Manipulación y Minería de datos	79
<i>Cristina Díaz and Oscar Herrera</i>	
Clasificación Translingüe de Documentos usando Similitudes del Conjunto Objetivo	88
<i>Adelina Escobar-Acevedo, Manuel Montes-Y-Gómez, Luis Villaseñor-Pineda and Rafael Guzmán-Cabrera</i>	
SmartFUT: Desarrollo de un simulador de futbol con aprendizaje por refuerzo y un clasificador de jugadores con SVM y redes neuronales	96
<i>Oscar Franco and Barbara Buendia</i>	

Reconocimiento de Actividades Mediante el Acelerómetro de un Celular .	103
<i>Enrique A. García Ceja and Ramón F. Brena Pinero</i>	
Interoperabilidad de bases de datos heterogéneas soportada con mecanismos semánticos	113
<i>David Gonzalez, Miguel Gonzalez-Mendoza, Neil Hernandez-Gress, Sal- vador Antonio Gutiérrez-Ayala and Hugo Vázquez-Palco</i>	
Preguntas de Definición para Generación Automática de Exámenes a partir de un Texto en Español	123
<i>Alejandro Gonzalez Zeta, Yulia Ledeneva, René Arnulfo García Hernández, Citlalih Gutiérrez Estrada, Rafael Cruz Reyes and Sergio Díaz Zagal</i>	
Cálculo de FS-Testores Difusos Utilizando Algoritmos Genéticos	132
<i>María Esther Guevara Cruz, Saturnino Job Morales Escobar, Ivonne Rodríguez Pérez and José Francisco Martínez Trinidad</i>	
Optimización de Parámetros de una Línea de Estampado Usando Redes Neuronales y Estrategias Evolutivas	142
<i>Susana Gutierrez, Giovanni Lizarraga and Marco Jimenez</i>	
Aplicación de Redes Neuronales Artificiales para la predicción de la Saturación de Agua en Yacimientos	152
<i>Dante Hernandez</i>	
Aplicación de Algoritmos Genéticos a la Compresión de Imágenes con Evolets	157
<i>Oscar Herrera and Román Mora</i>	
Deteccion de hipovigilancia en un automovilista: Utilizacion de wavelets para el analisis del angulo del volante y posicion lateral	167
<i>Rodolfo Ibarra-Orozco Ibarra-Orozco, Neil Hernandez-Gress, Miguel Gonzalez- Mendoza, Victor Ramos, Juan Carlos Lopez-Pimentel and Chistian Pon-Gomez</i>	
Modelo de Evaluación de Puntos Compromiso para la Banca en México ..	177
<i>Victor Mata Mata</i>	
Algoritmos Fuzzy C-Means robustos basados en un estimador-RML para segmentar imágenes ruidosas	185
<i>Dante Mujica and Francisco Javier Gallegos-Funes</i>	
Sistema de agentes para control y protección de políticas de seguridad aplicadas a equipos de cómputo con ambiente Windows-Tecnología NT ..	195
<i>Pedro Alfonso Rojas López, Mario Alberto Medina Celaya and Guadalupe Cota Ortiz</i>	

Hacia la mejora de los algoritmos SVM y SNN para arquitectura GPGPU-CUDA.....	207
<i>Israel Tabarez Paz, Neil Hernández Gress and Miguel González Men- doza</i>	
Análisis del historial climático del Valle de Toluca con minería de datos .	213
<i>Juan Carlos Torres Arellano, Rosa María Valdovinos Rosas, Dario Calderon and Roberto Alejo</i>	
Fermat: Una Red Social de Aprendizaje Inteligente para Matemáticas ...	223
<i>Ramon Zatarain, Lucía Barrón, Armando Beltrán, Franceli Cibrian and Marsia Quiroz</i>	
Methods for Evaluation of Word Sense Disambiguation	232
<i>Alisa Zhila and Alexander Gelbukh</i>	

Comité Revisor

Sánchez López Abraham	Computer science department, University of Puebla
Moises Alencastre-Miranda	ITESM-CSF
García Gamboa Ariel Lu-	ITESM-CEM
cién	
Gustavo Arroyo Figueroa	IIE
Rosario Baltazar	Instituto Tecnológico de León
Maricela Bravo	UAM Azcapotzalco
Carlos Alberto Brizuela	CICESE Research Center
Maria Victoria Carreras	Panamerican University
Oscar Castillo	Tijuana Institute of Technology
Felix Castro Espinoza	CITIS-UAEH
Eduardo Castro-Puello	ITESM-CVA
Jair Cervantes	Cinvestav
Juan Frausto-Solis	ITESM-CVA
Olac Fuentes	U Texas
Sofia N. Galicia-Haro	Facultad de Ciencias, UNAM
Andres David Garcia Gar-	ITESM-CEM
cia	
Daniel Alejandro Garcia	Universidad Veracruzana
Lopez	
Iztli Horacio Alberto García	UAEM
Alexander Gelbukh	CIC-IPN
David Gonzalez	ITP-ITESM
Jesus A. Gonzalez	National Institute for Astrophysics, Optics, and Electronics
Miguel Gonzalez-Mendoza	ITESM-CEM
Fernando Gudiño Peñaloza	UNAM
Yasmin Hernandez	Instituto de Investigaciones Electricas
Neil Hernandez Gress	ITESM
Arturo Hernandez-Aguirre	Centre for Research in Mathematics
María De Lourdes	UAEM
Hernández-Prieto	
Oscar Herrera	UAM Azcapotzalco
Rodolfo Ibarra-Orozco	ITESM-CEM
Ana Lilia Laureano-Cruces	Departamento de Sistemas-Universidad Autonoma Metropolitana-Azcapotzalco
Yulia Ledeneva	UAEM
Wulfrano-Arturo Luna-	Universidad Veracruzana
Ramírez	
Juan Carlos López Pimentel	UP Tuxtla Gutierrez
Antonio Marin Hernandez	Universidad Veracruzana
Luis Felipe Marin Urias	Universidad Veracruzana
Lourdes Martínez	ITESM-CEM

Mauricio Martínez Medina	IPN
Efrén Mezura-Montes	Laboratorio Nacional de Informatica Avanzada (LAINIA A.C.)
Raul Monroy	Tecnologico de Monterrey, Campus estado de México
Jaime Mora-Vargas	ITESM-CEM
Saturnino-Job Morales-Escobar	UAEM
Rafael Murrieta	Centro de Investigacion en Matematicas
Lourdes Muñoz Gómez	ITESM
Antonio Neme	Universidad Autonoma de la Ciudad de Mexico
Elvia Palacios	Facultad de Ciencias, UASLP
Yoseba Peña Landaburu	DEUSTO
Maricela Quintana	ITESM-CEM
Jorge Adolfo Ramirez Uresti	Tecnologico de Monterrey, Campus Estado de Mexico
Alonso Ramirez-Manzanares	Departamento de Matematicas, Universidad de Guanajuato
Juan-Manuel Ramos-Arreguin	Universidad Tecnológica de San Juan del Río
José Federico Ramírez Cruz	Instituto Tecnológico de Apizaco
Orion Fausto Reyes-Galaviz	Universidad Autonoma de Tlaxcala
Carlos A Reyes-Garcia	Instituto Nacional de Astrofisica Optica y Electronica
Mariano Rivera	CIMAT
Dafne A. Rosso-Pelayo	PEMEX
Grigori Sidorov	CIC-IPN
Joel Suárez Cansino	Centro de Investigación en Tecnologías de Información y Sistemas, CITIS, Universidad Autónoma del Estado de Hidalgo
Raul A. Trejo-Ramírez	UAEM
Benjamin Valdés-Aguirre	ITESM-CEM
Edgar Vallejo	ITESM-CEM
Aremy Olaya Virrueta Gordillo	UP Chiapas
Julio Weissman Vilanova	Universidad de Sonora
Francisco Javier Zaragoza Martinez	Universidad Autonoma Metropolitana Azcapotzalco

Asistente de Conducción para Carreteras Basado En Visión Computacional Embebido En FPGA

Ricardo Acevedo-Ávila¹, Miguel González-Mendoza¹,
Andrés-David García-García¹.

¹Department of Electrical and Electronic Engineering
Instituto Tecnológico y de Estudios Superiores de Monterrey, Campus Estado de
México.

Atizapán de Zaragoza, Estado de México, México
{A00468443, mgonza, garcia.andres}@itesm.mx

Abstract. En este artículo presentamos el diseño de un asistente de conducción en carreteras basado en visión computacional cuyo objetivo principal es el de asistir al conductor proporcionando información vital sobre el entorno y el estado del vehículo. El sistema se propone como una plataforma de computación embebida. Debido a las limitaciones físicas, de memoria y poder de cómputo, los algoritmos existentes para el procesamiento de imágenes orientados a computadoras personales (PC) no son adecuados para esta aplicación. En su lugar, algoritmos simples y eficientes deben ser desarrollados para encajar en esta arquitectura embebida. El asistente de conducción presenta tres funcionalidades principales: detección de carriles, detección de obstáculos y detección de cambio de carril, todas diseñadas para su implantación en hardware FPGA.

Keywords: Sistemas Embebidos, Sistemas Inteligentes, Asistente a la Conducción, Procesamiento de Imagen, Visión de Computadora, Detección de Carriles, Detección de Obstáculos, Detección de Cambio de Carril, Desarrollo de Hardware, FPGA.

1 Introducción

Los accidentes de tráfico son un grave problema socio económico y una de las diez mayores causas de muerte en seres humanos a nivel mundial [1], este problema también conlleva un gasto continuo de parte de todos los actores involucrados (industria, gobierno y sociedad) para tratar de solventarlo. La investigación en seguridad vehicular resulta esencial para resolver este problema.

Los Sistemas de Asistencia a la Conducción basados en visión de computadora presentan una tecnología orientada a mejorar la seguridad vial. Estos sistemas utilizan la infraestructura carretera existente en conjunto con plataformas de computación poderosas para correr muchos algoritmos de procesamiento de imágenes distintos (reconocimiento de camino, detección de vehículos y carriles, seguimiento, etc.).

El objetivo principal de este documento es el de demostrar como implantar las funcionalidades de un asistente de conducción basado en visión computacional completo en hardware FPGA. La tarjeta de desarrollo utilizada inicialmente en este trabajo es la Altera Cyclone II con el circuito FPGA EP2C35F672C6N [2].

Este circuito se ha seleccionado, principalmente, por su bajo costo, componentes integrados y ambiente de prototipaje ofrecido. Debido a la complejidad de los algoritmos de procesamiento de imágenes, resulta necesario desarrollar modelos simplificados restringidos a los recursos ofrecidos por la tarjeta de desarrollo. Como se mencionó anteriormente, el sistema propuesto se compone de tres principales algoritmos (o módulos) de procesamiento de imágenes:

a) Detección de carriles. b) Detección de cambio de carril. c) Detección de obstáculos.

La detección de carriles es utilizada para modelar el camino y definir sus límites. También se utiliza para detectar eventos del tipo cambio de carril, eventos que tal vez ocurran inadvertidamente para el conductor del vehículo. Recientemente Nissan ha desarrollado un sistema de advertencia de cambio de carril basado en esta tecnología [3].

Dicha detección es especialmente difícil en entornos urbanos, debido a vehículos en movimiento, árboles, peatones, edificios, sombras y otras fuentes de ruido existentes en la escena [4]. Nosotros presentamos un método rápido y eficiente de detección orientado únicamente a entornos de carretera. Utilizamos filtros y algoritmos diseñados en hardware que tomen ventaja de este ambiente simplificado.

La detección de obstáculos es otra funcionalidad vital para la seguridad vehicular inteligente. Nuestra preocupación principal es que este tipo de detección es, típicamente, una tarea de cómputo lenta en aplicaciones de tiempo real, así que será necesario desarrollar un método de detección alternativo que sea rápido pero efectivo.

Todos los módulos del sistema propuesto se han diseñado como componentes de hardware hechos a medida para esta aplicación. El componente central de este sistema es un corrector de distorsión de perspectiva; su principal objetivo es obtener una imagen en "vista aérea" de la escena, haciendo uso de un mapeo inverso de perspectiva (*IMP - Inverse Perspective Mapping*) [5]. Después, corriendo un algoritmo de software embebido simple, es posible determinar el número de cambios de carril y tener conocimiento de la presencia de un objeto al frente del vehículo.

La solución propuesta forma parte de una primera iteración del sistema. Para explorar la viabilidad de este diseño, pruebas experimentales se han realizado con datos de escenarios reales. Aún es necesaria una etapa de optimización y ajuste de recursos al circuito FPGA utilizado, pero la funcionalidad básica de cada componente/módulo ha mostrado resultados satisfactorios.

2 Diseño del Sistema

La transformada de Hough es un método de detección de líneas y círculos [6] constantemente utilizado en la detección de carriles [7] debido a su robustez a cambios de iluminación, sin embargo, la complejidad de esta técnica [8] demanda un poder de cómputo significativo, más allá de las restricciones de hardware de esta aplicación. Es posible simplificar el modelo de detección de líneas si se extrae la distorsión de perspectiva introducida por la cámara utilizando IMP. La Figura 1 muestra la imagen distorsionada por la lente de la cámara, también se muestra la imagen real que puede ser obtenida si un método rectificador de perspectiva es

aplicado. Es importante observar que esta operación es, en esencia, un mapeo de píxeles entre dos planos (el plano de la imagen y el plano del mundo real).

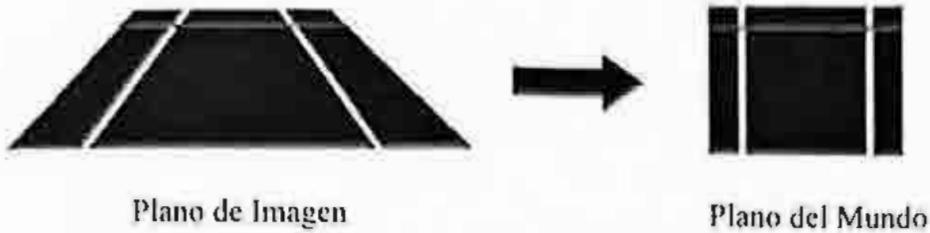


Fig 1. Distorsión en el plano de imagen y el plano del mundo.

El método utilizado por el rectificador de perspectiva requiere 4 puntos no colineales para crear el mapeo entre los dos planos. Los 4 puntos corresponden a cada una de las 4 esquinas del trapecio formado en el plano de la imagen y sus posiciones equivalentes en el plano del mundo real.

Estos puntos generan un conjunto de 8 ecuaciones simultáneas [9]. El sistema de ecuaciones puede resolverse para generar un par de matrices que contengan las posiciones exactas de los píxeles dentro del plano del mundo real. La distorsión de perspectiva puede modelarse utilizando una transformación de geometría proyectiva, representada en la ecuación (1), donde \bar{X} y \bar{X}' son 3 vectores que representan un punto, y \bar{H} es una matriz homogénea no singular de dimensiones 3x3.

$$\bar{X}' = \bar{H}\bar{X}. \quad (1)$$

Con las correspondencias de los 4 puntos mencionados anteriormente, es posible extraer 8 ecuaciones para generar la matriz H, como se muestra en la ecuación (2).

$$\begin{bmatrix} x_0 & y_0 & 1 & 0 & 0 & 0 & -x_0x'_0 & -y_0x'_0 \\ x_1 & y_1 & 1 & 0 & 0 & 0 & -x_1x'_1 & -y_1x'_1 \\ x_2 & y_2 & 1 & 0 & 0 & 0 & -x_2x'_2 & -y_2x'_2 \\ x_3 & y_3 & 1 & 0 & 0 & 0 & -x_3x'_3 & -y_3x'_3 \\ 0 & 0 & 0 & x_0 & y_0 & 1 & -x_0y'_0 & -y_0y'_0 \\ 0 & 0 & 0 & x_1 & y_1 & 1 & -x_1y'_1 & -y_1y'_1 \\ 0 & 0 & 0 & x_2 & y_2 & 1 & -x_2y'_2 & -y_2y'_2 \\ 0 & 0 & 0 & x_3 & y_3 & 1 & -x_3y'_3 & -y_3y'_3 \end{bmatrix} * \begin{bmatrix} h_{11} \\ h_{12} \\ h_{13} \\ h_{21} \\ h_{22} \\ h_{23} \\ h_{31} \\ h_{32} \end{bmatrix} = \begin{bmatrix} x'_0 \\ x'_1 \\ x'_2 \\ x'_3 \\ y'_0 \\ y'_1 \\ y'_2 \\ y'_3 \end{bmatrix} \quad (2)$$

Es posible resolver este sistema de ecuaciones lineales utilizando cualquier software de cálculo matemático. Los resultados de la rectificación de perspectiva se muestran en la Figura 2.

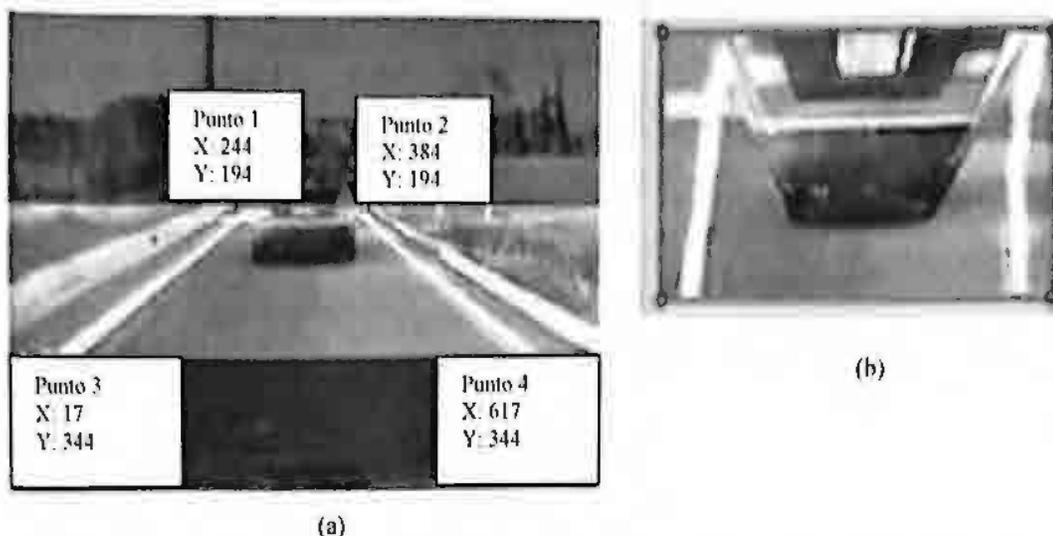


Fig. 2. (a) Imagen de la Cámara (b) Imagen Rectificada. Note los cuatro puntos de correspondencia en cada imagen.

2.1 Implantación en FPGA

Una vez que se ha obtenido la matriz H, las posiciones de cualquier pixel en el plano del mundo real pueden ser obtenidas para cualquier pixel dentro del plano de la imagen. Esta información puede almacenarse en memoria de solo lectura (ROM). Esto significa que, en la arquitectura hardware propuesta, cada pixel existente en la imagen distorsionada puede re ubicarse en una nueva posición, de acuerdo a la información almacenada en ROM, como se muestra en la Figura 3.

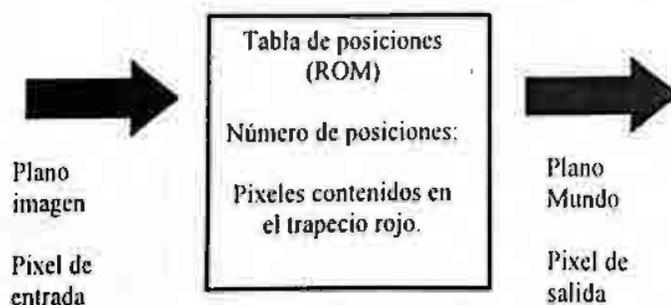


Fig. 3. Cada pixel de entrada es re-ubicado a una nueva posición de acuerdo a la información almacenada en la tabla ROM.

Para implantar esta idea en hardware FPGA real, al menos 3 componentes son necesarios: memoria ROM para almacenar la información de las posiciones de la imagen rectificada, memoria RAM para almacenar cada imagen (cuadro) de entrada y un controlador RAM/ROM que se encargue de sincronizar la transmisión de datos entre los dos bloques de memoria. El controlador RAM/ROM se diseña mediante un modelo de máquinas de estados finitos con cuatro principales estados:

- S0: Inicialización de señales internas.

- S1: Petición de datos a ROM.
- S2: Datos de ROM recibidos, petición a RAM.
- S3: Datos de RAM recibidos, datos enviados a contenedor final. (En este caso, un controlador VGA para visualizar la imagen en un monitor de computadora).

2.2 Extracción de las coordenadas del carril

Una vez que se ha rectificado la imagen, es necesario extraer las coordenadas de las dos líneas en el eje horizontal. La Figura 4 muestra la rectificación ideal del camino, obtenida a través de la transformación de perspectiva.

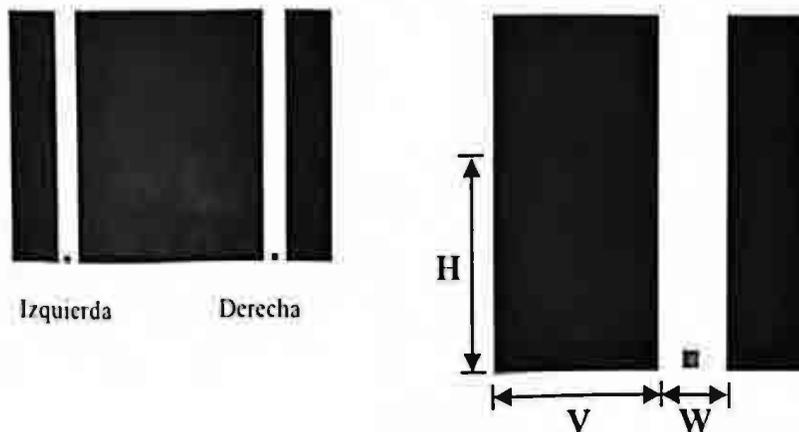


Fig. 4. Imagen Rectificada Ideal. Cada centroide del carril está marcado con un cuadro rojo.

Cada una de las líneas blancas puede representarse como dos rectángulos perfectos, con ancho "w" y altura "h". Si se considera a "v" como la distancia desde la esquina inferior izquierda hasta el inicio de la línea blanca, entonces, la coordenada central "C_t" de este carril puede calcularse como:

$$C_t = V + \frac{W}{2}. \quad (3)$$

En una imagen rectificada real (Figura 5) las dos líneas difícilmente se asemejan a un par de rectángulos perfectos. Es posible procesar cada fila de la imagen de manera independiente y, después, computar un promedio por fila para aproximar un valor de coordenada final.

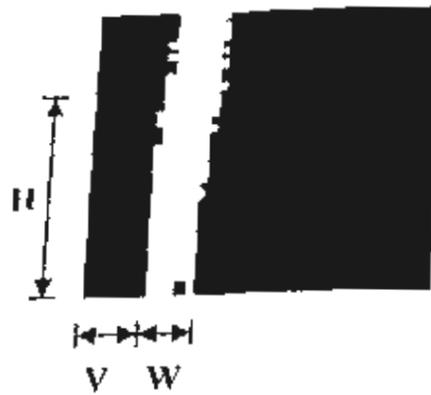


Fig. 5. Imagen Rectificada Real.

Considere que cada fila de la imagen se procesa de izquierda a derecha. Primero, es necesario contar cada pixel negro y almacenar el resultado en una variable acumulador " V_n ". Los pixeles blancos también son contados y el valor final almacenado en otra variable acumulador llamada " W_n ". Si una fila completa (320 pixeles) se ha procesado, entonces una variable contadora de filas " H " debe incrementarse en una unidad. Si el número de filas procesadas ha alcanzado un límite " H " predefinido, el procesamiento ha concluido y es necesario calcular el promedio de cada variable acumulador:

Para píxeles negros:

$$V_t = \frac{V_n}{H} \quad (5)$$

Para píxeles blancos:

$$W_t = \frac{W_n}{H} \quad (6)$$

El centroide del carril, en el eje horizontal, puede calcularse como:

$$C_x = V_t + \frac{W_t}{2} \quad (7)$$

Es importante recordar que se busca una implantación 100% hardware de este algoritmo, así que debemos idear alguna manera de computar las divisiones en las ecuaciones (5), (6) y (7). Si únicamente se procesan 16 ($H = 24$) filas, las divisiones en las ecuaciones (5) y (6) pueden expresarse como desplazamientos hacia la izquierda de cuatro posiciones. Para la ecuación (7) el desplazamiento es de solo una posición.

2.3 Procesamiento de datos

Después de calcular ambos centroides, finalmente es posible determinar si un carril está presente en la imagen, si el vehículo está cambiando de carril, o si un obstáculo se encuentra frente al vehículo. Para detectar obstáculos es posible binarizar el área entre los dos carriles, si un obstáculo está presente, un conjunto (blob) de píxeles blancos de área considerable se obtendrá en la imagen binaria. Podemos trabajar con la hipótesis que establece que entre más cerca se encuentre el objeto, más área cubrirá

en la imagen capturada por la cámara. Si el blob de píxeles excede una cierta distancia de seguridad, como se muestra en la Figura 6, podremos alertar al conductor de la presencia de un obstáculo justo al frente del vehículo.



Fig. 6. Detección de obstáculos simple. La imagen binaria muestra el objeto como un conjunto de píxeles blancos. después, es sencillo rodear este blob de píxeles con un rectángulo de color. El resultado final de la detección se muestra en el espacio de color RGB.

Para detectar el cambio de carril, el procedimiento es un poco más difícil, pues es necesario rastrear ambos carriles a través del tiempo; esto se logra definiendo una "ventana de procesamiento". La ventana de procesamiento evaluará los cambios ocurridos en 10 cuadros de video, después, un algoritmo simple concluirá si el vehículo, en efecto, ha cambiado de carril. La idea presentada es sencilla: Dividamos la parte baja de la imagen en dos grandes áreas: Izquierda (Izq.) y Derecha (Der.). Dividamos de nuevo estas dos áreas en dos mitades, al final obtendremos las cuatro sub áreas mostradas en la Figura 7. Cada centroide correspondiente cada línea cruzará cada región de la imagen en un momento dado, dependiendo de la dirección del vehículo.

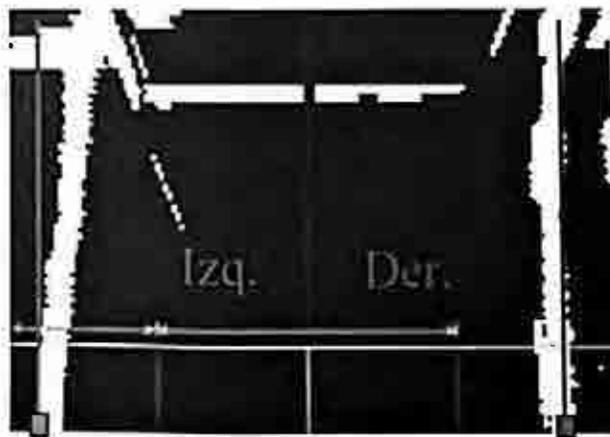


Fig. 7. Detección de cambio de carril. Note las cuatro sub áreas en la parte baja de la imagen. Las líneas rojas verticales representan cada uno de los carriles.

Si el centroide izquierdo se mueve hacia la derecha y, después, el centroide derecho desaparece de la escena, probablemente, un cambio hacia la izquierda ha ocurrido. Si por el contrario, el centroide derecho se mueve hacia la izquierda y el centroide

izquierdo eventualmente desaparece de la escena, podemos inferir que un cambio hacia la derecha ha ocurrido. No obstante, necesitamos un cierto valor de umbral para estar completamente seguros de que el vehículo realmente está cambiando de dirección. Definimos ese umbral como la mínima distancia que ambos centroides deben cruzar, representado como las dos mitades de las áreas izquierda y derecha en la Figura 8 (líneas rojas). Previamente, definimos una ventana de procesamiento de 10 cuadros. Si notamos que ambos centroides caen más allá de la distancia mínima de umbral en al menos 6 de 10 cuadros, podremos concluir con certeza que un cambio de dirección real (cambio de carril) está ocurriendo en ese instante de tiempo. El algoritmo encargado de supervisar estas condiciones se ha implantado con un programa simple de software embebido escrito en C, que corre sobre un CPU NIOS II suave de 32-bit diseñado específicamente para FPGAs de Altera.

La configuración de dicho procesador se enlista de la siguiente manera:

- Núcleo Estándar.
- 20 KBytes de memoria en el chip.
- UART JTAG para comunicación y depuración.

3 Resultados

Es importante notar que el sistema muestra resultados después de haber procesado 10 fotogramas, de esa manera podemos concluir que, si la detección se ha presentado en al menos 6 fotogramas, estamos frente a una detección positiva. El vehículo de prueba cambió de carril de manera aleatoria a lo largo de todo el video. La Figura 8 muestra los resultados de la detección de un cambio hacia la derecha obtenidos con el sistema propuesto.

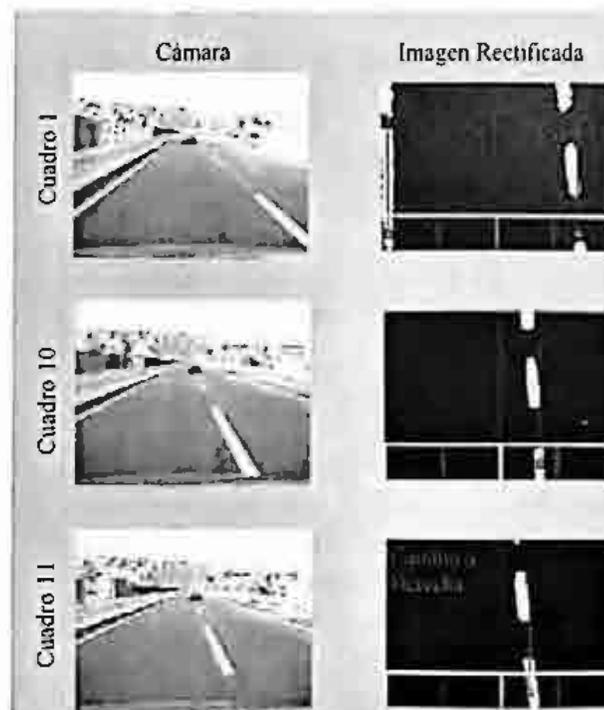


Fig. 8. Pruebas de cambio de carril.

El sistema detecta activamente e identifica correctamente el carril y el cambio de dirección siempre y cuando las marcas en el camino se encuentren correctamente pintadas y la cámara correctamente posicionada dentro del vehículo de prueba. Las pruebas presentaron un alto grado de precisión para cada secuencia de video. Sin embargo, es preciso notar que durante las pruebas no ocurrió ningún fenómeno de oclusión, por lo tanto se esperaba un desempeño óptimo del sistema. La implantación esencial del sistema se puede observar en la Figura 9.

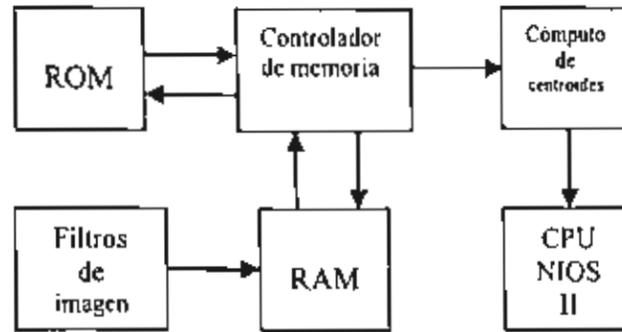


Fig. 9. Arquitectura central del Sistema (Simplificado)

Se han incluido etapas de pre-procesamiento compuestas de filtros de erosión-dilatación, promedio y binarización antes de efectuar la corrección de perspectiva para dotar de robustez al sistema ante posibles variaciones de luz. La Tabla 1 resume el consumo de recursos del diseño proceso, en su primera iteración, el sistema trabaja a una frecuencia máxima de 44.69 MHz.

Tabla 1. Consumo de recursos del sistema

Circuito FPGA	Consumo de Recursos		
	LE	Bits de Memoria	F _{max} (MHz)
Propuesto	4,166	568,854	44.69
EP2C35	33,216	483 840	44.69
EP2C70	68,416	1125 K	44.69

El circuito EP2C35 FPGA de Altera se utilizó como plataforma de pruebas, sin embargo, los bits de memoria (bloques de memoria) requeridos en este diseño exceden los ofrecidos por este circuito en particular. El diseño completo puede optimizarse aún más para minimizar el número de bloques de memoria requeridos o bien, implantarse en un circuito EP2C70, de capacidad mayor. Como se muestra en la Tabla 1, utilizando esta última opción los requisitos de memoria son completamente cumplidos sin problemas.

4 Conclusiones

En este trabajo proponemos el diseño e implantación de un Sistema de Asistencia a la Conducción sobre una plataforma de hardware embebida. Esta propuesta presenta una solución real a una necesidad de mejorar la seguridad vial de nuestra sociedad. Empleamos un modelo simplificado del camino utilizando una transformación de perspectiva orientada a explotar los recursos y la configuración de hardware de esta aplicación. Uno de los componentes cruciales de este sistema es la técnica IPM, que sustituye a otras técnicas computacionalmente complejas para detección del camino. Como se observa en la ecuación (2), esta solución puede independizarse de los parámetros intrínsecos y extrínsecos de la cámara, siempre y cuando la cámara se posicione correctamente en el vehículo de prueba. Esta solución promete ser factible y fiable de acuerdo a los experimentos llevados a cabo. Aún es posible realizar ciertas optimizaciones, como trabajo futuro, y será posible utilizar este sistema en una aplicación completamente implantada a bordo del vehículo. Durante el desarrollo de este sistema, se obtuvo un tiempo de procesamiento de 0.008147 segundos por cuadro, tras optimizar los componentes, se espera una mejora significativa en el tiempo de procesamiento de cada cuadro.

5 Agradecimientos

Los autores de este trabajo desean expresar su gratitud al Dr. Cuauhtémoc Carbajal, Dr. Alfredo Santana y Dr. Sadegh Babaii, por su continua ayuda en las discusiones culminadas en la elaboración de este trabajo.

6 Referencias

- [1] The World Health Organization "The top 10 causes of Death" <http://www.who.int/mediacentre/factsheets/fs310/en/index.html> 07 May 2011.
- [2] Cyclone II Literature Wbsite by Altera: <http://www.altera.com/literature/lit-eye2.jsp> 02 May 2011.
- [3] Nissan's "All Around Collision Free" Prototype To Demonstrate Advanced Accident Avoidance Systems At Its World Congress.
<http://www.nissannews.com/newsrelease.do?sessionId=EBF6F1B4AEC92778E31524B6C91816E7?&id=626&mid=1> 23 April 2011.
- [4] Aly, M. "Real Time Detection Of Lane Markers In Urban Streets". Computer Vision Lab, Electrical Engineering, California institute of Pasadena, 2008.
- [5] Bertozzi, M. "Stereo Inverse Perspective Mapping: Theory and Applications" Dipartimento di Ingegneria dell'Informazione, Università di Parma, Parma, Italy, 1998.
- [6] Coifman, B. "A Real-Time Computer Vision system for Vehicle Tracking and Traffic Surveillance" Institute of Transportation Studies, University of California, Berkeley, California, 1998.
- [7] Mc. Donald, J. "Application of the Hough Transform to Lane Detection and Following on High Speed Roads" Signals & Systems Group, Department of Computer Science, National University of Ireland, Maynooth, Ireland, 2001.
- [8] Yang, G., "Computer Vision Hough Transform" Department of Computing, Imperial College London, 2005.
- [9] Hartley, R., Zisserman, A. "Multiple View Geometry in Computer Vision" Second Edition, Cambridge University Press, Cambridge, U.K., 2003.

Generación de Código Libre Multiplataforma de Máquinas de Soporte Vectorial

Ariel García Gamboa¹, Miguel González Mendoza¹, Alfonso Calderón-Chávez²,
Francisco Javier Calderón-Chávez²

¹ITESM-CEM. Carretera Lago de Guadalupe Km. 3.5
Atizapán de Zaragoza, Estado de México, C.P. 52926, México

²ITSA. Carretera Apatzingán-Aguilla Km. 3.5
Apatzingán, Michoacán, A.P. 49, México

Abstract. El presente trabajo presenta un estudio del algoritmo de las Máquinas de Soporte Vectorial para optimizarlo implementando el método de "Caching", creado y utilizado en el algoritmo de SMO ("Sequential Minimal Optimization"). De igual forma, se implementan diferentes estrategias de inicialización para mejorar el tiempo de optimización con vectores de soporte candidatos en algoritmo "chunking". Una vez depurado, el proyecto se distribuirá como código libre multiplataforma teniendo así una cobertura amplia de las SVM's para su mayor implementación dentro de empresas e instituciones.

1 Introducción

Las Máquinas de Soporte Vectorial (SVM's) son un algoritmo ampliamente utilizado para resolver problemas de clasificación, regresión y estimación de densidad. El algoritmo fue desarrollado por Vapnik, [1], y ha sido objeto de estudio y aplicación de diversos investigadores en múltiples áreas del conocimiento. El éxito de este algoritmo se debe principalmente a los siguientes aspectos: a) la capacidad de generalización del algoritmo, es decir la capacidad de categorizar datos no conocidos mediante el aprendizaje de un número reducido de ejemplos, b) su eficacia al clasificar correctamente datos con ruido. Sin embargo, el uso de este algoritmo se ha visto limitado por el tiempo necesario para realizar el aprendizaje de problemas con una gran cantidad de datos dado que la complejidad crece cuadráticamente con el tamaño de la entrada, lo que dificulta resolver problemas de gran escala. Existen diferentes trabajos cuyo objetivo es reducir el tiempo de entrenamiento de las SVM's, [2], [7], [8]. Estas mejoras han reducido el tiempo de entrenamiento de las SVM's, pero el tiempo de entrenamiento sigue siendo un área de mejora para el algoritmo.

En este documento se muestra la combinación de algunas estrategias para la reducción del tiempo de entrenamiento de las SVM's. En primer lugar, se utiliza la estrategia desarrollada por Platt, [2], quien diseñó un algoritmo iterativo que optimiza

por pares de variables y que, además, almacena los datos ya calculados en iteraciones pasadas, de tal forma que antes de calcular un nuevo valor para la optimización, se verifica si ya existe y se utiliza. Esta estrategia fue llamada "Caching" y reduce de considerablemente los cálculos necesarios de cada iteración y el tiempo de entrenamiento global. Esta estrategia se utilizó para optimizar pares de variables y no ha sido utilizada para el entrenamiento de las SVM's siguiendo una estrategia de descomposición "Chunking", por lo que aquí se propone utilizar esta estrategia para la construcción de la matriz hesiana en las diferentes iteraciones de una optimización tradicional.

En segundo lugar, la estrategia de "Caching" se combinará con una estrategia heurística de selección de los vectores de soporte, [8]. Esta estrategia está basada en principios geométricos y el objetivo principal es encontrar los vectores de soporte a priori, de tal forma que al realizar el entrenamiento de la SVM ya se conozcan los Vectores de Soporte y se entrene con un conjunto reducido de datos que sólo contenga los vectores solución. Esta estrategia también ha disminuido el tiempo de entrenamiento y su combinación con "Caching" permitirá utilizar el algoritmo en problemas de gran escala.

Este documento está organizado de la siguiente forma: primero se muestra brevemente la formulación matemática de las SVM's. Luego, da una breve explicación de la estrategia de "Caching" utilizada en la optimización de las SVM's. En tercer lugar se muestra la heurística para identificar los Vectores de Soporte a priori. En cuarto lugar se muestran los resultados obtenidos al combinar la estrategia heurística con "Caching" y finalmente se presentan las conclusiones.

2 Las Maquinas de Soporte Vectorial

Dado un conjunto de datos de entrenamiento x_1, x_2, \dots, x_n la SVM se encargará de dividir los datos en dos clases diferentes $y_i \in \{1, -1\}$, por medio de un hiperplano de margen máximo de separación. La formulación matemática del problema es la siguiente:

$$\max_{\alpha} \sum_{i=1}^n \alpha_i - \frac{1}{2} \sum_{i=1}^n \sum_{j=1}^n y_i y_j k(x_i, x_j) \alpha_i \alpha_j$$

Sujeto a:

$$\sum_{i=1}^n y_i \alpha_i = 0$$

$$0 \leq \alpha_i \leq c$$

Donde las α_i mayores o iguales que cero corresponden a los Vectores de soporte del problema. La constante c es un valor de penalización que permite tratar problemas con ruido y la función K corresponde a una función *Kernel* la cual permite mapear los datos de entrada de su espacio original a un espacio característico de dimensión mayor en el que el problema puede ser separado linealmente. Esto último permite resolver problemas del tipo no linealmente separables sin necesidad de hacer un mapeo explícito de los datos a una dimensión superior.

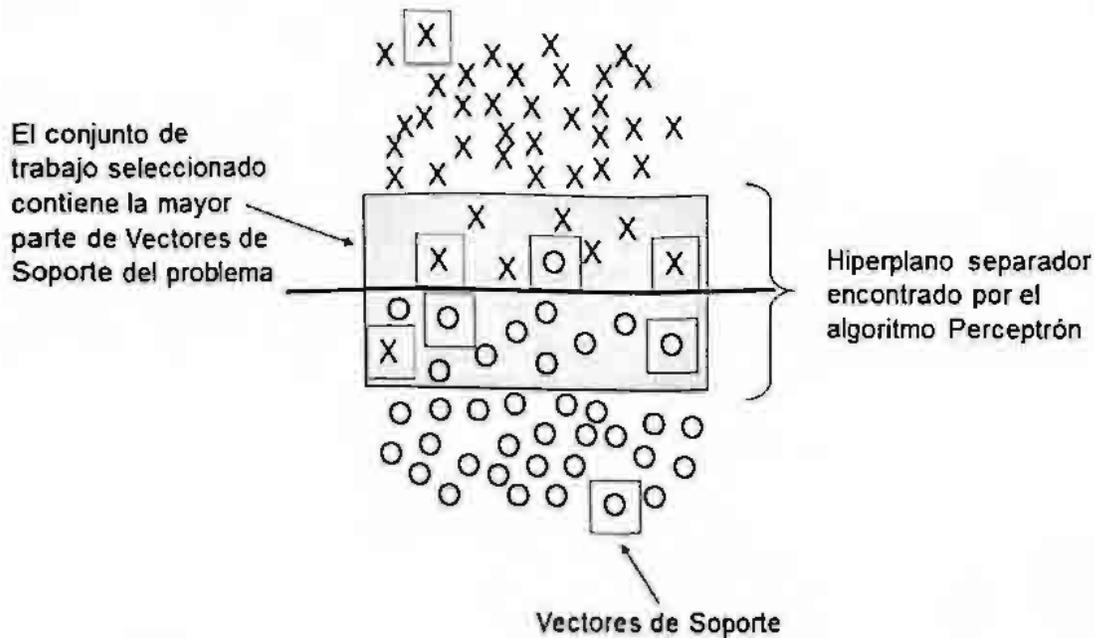


Fig. 1. Heurística para la identificación de los Vectores de Soporte a priori

3 Metodología propuesta

Se propone combinar dos estrategias para reducir el tiempo de entrenamiento de las SVM's. La primera estrategia es una heurística para identificar a priori los Vectores de Soporte, mientras que la segunda estrategia es la estrategia propuesta en [2] llamada "Caching". En las siguientes secciones se explica en detalle cada una de las estrategias.

3.1 Heurística para identificar los Vectores de Soporte a priori

Dado que los vectores de soporte son aquellos datos que se encuentran sobre el margen de separación y que el entrenamiento de las SVM's se puede realizar únicamente utilizando el subconjunto de los Vectores de Soporte, es posible afirmar que si éstos se conocen a priori, entonces se puede realizar el entrenamiento con dicho subconjunto y así disminuir el tiempo requerido. El principal problema de esta técnica es conocer a priori el subconjunto que contiene los vectores de soporte. En esta sección se muestra una heurística que aproxima la solución al identificar un subconjunto que contiene una buena proporción de Vectores de Soporte. La heurística

para identificar a priori el subconjunto de los vectores de soporte consiste en lo siguiente:

1. Utilizar un algoritmo de clasificación que encuentre un hiperplano de separación cercano al encontrado por las SVM's. Para este paso se ha seleccionado utilizar el algoritmo Perceptrón.
2. Identificar, utilizando una métrica euclidiana, los patrones más cercanos al hiperplano separador y formar un subconjunto de trabajo. Dicho subconjunto contiene una buena proporción de los vectores de soporte del problema, por lo que es utilizado para realizar el aprendizaje de las SVM's. Esta heurística para identificar a priori los Vectores de Soporte se muestra en la Figura 1.

Algoritmo Perceptrón.

El algoritmo Perceptrón, [4], es uno de los principales procedimientos de aprendizaje de clasificadores lineales, [5], [6]. El aprendizaje en dicho algoritmo es un aprendizaje supervisado e incremental. Este algoritmo inicia con un conjunto de pesos aleatorios w_1 un conjunto de entradas x_1, x_2, \dots, x_n y un conjunto de etiquetas o de salidas deseadas d_1, d_2, \dots, d_n . El algoritmo de aprendizaje del Perceptrón consiste en una regla de corrección del error que se encarga de ajustar los pesos de la red de manera proporcional al error $e_1 = d_1 - y_1$. La regla de aprendizaje de los nuevos pesos corresponde a $w_2 = w_1 + \Delta w_1 = w_1 + \eta(d_1 - y_1)x_1$. Esta regla se aplica para cada ejemplo de la base de datos. La constante η es llamada paso de aprendizaje y es una magnitud del incremento del vector de pesos δw . El valor de η puede hacer variar el número de iteraciones que el algoritmo necesita para llegar a la convergencia. Este algoritmo asegura la convergencia en un número finito de iteraciones para problemas linealmente separables. El algoritmo Perceptrón se muestra en la Tabla 1.

Table 1. Algoritmo Perceptrón que encuentra los Vectores de Soporte

<ol style="list-style-type: none"> 1. Inicializar el vector de pesos w y el término b. 2. Establecer el valor del paso de aprendizaje η 3. Mientras exista un $i: i \in N$, tal que $f(x_i) \neq y_i$ <ol style="list-style-type: none"> a. Calcular el valor de la función $f(x_i) = \text{signo}((w \cdot x_i) + b)$ b. Calcular el valor de la función de error $e_i = d_i - y_i$ c. Si $e_i \neq 0$, entonces Actualizar los valores de w y b con: $\Delta w = w_{old} + \eta(d_i - y_i) \cdot x_i$ $\Delta b = b_{old} + \left(\frac{\eta}{2}\right) y_i$ 4. Regresar al punto 3

3.2 Uso de "Caching" en la creación de la matriz Hesiana

El cache es típicamente comprendido como una pequeña porción de memoria que es más rápida que la memoria normal. En este documento, la estrategia "Caching" se refiere a una pequeña tabla que almacena los valores de la matriz Hesiana que son calculados en cada iteración. El objetivo es utilizar dichos cálculos cuando es necesario en lugar de rehacer las operaciones, lo que finalmente consume menos tiempo computacional que realizar el cálculo completo.

Resolver de forma iterativa el problema de optimización cuadrático de las SVM's, utilizando un método del conjunto activo, involucra que en cada iteración se identifiquen un conjunto de variables, aquellas que se convierten en activas y aquellas que se relajan. Las variables activas se utilizan para calcular la matriz Hesiana y realizar la optimización. Durante las diferentes iteraciones, las variables que antes fueron activas se pueden relajar o pueden seguir en el conjunto activo, por lo que hay variables que ya se calcularon en iteraciones anteriores. Es así que surge la idea de almacenar las variables que se mantienen en el conjunto activo, "Caching", de tal forma que en cada iteración ya no es necesario calcularlas, sino tomarlas de una pequeña porción que las almacena.

Para facilitar la comprensión de la estrategia "Caching" se ha preparado un ejemplo que muestra dos iteraciones en la creación de la matriz Hesiana.(ver Tabla 2)

1.000	0.560	0.056	-0.100
0.560	1.000	0.310	-0.050
0.056	0.310	1.000	-0.050
-0.100	-0.050	-0.050	1.000

Table 2. Primera iteración en la creación de la Hesiana

1.000	0.560	0.056	0.003	-0.100	0.003
0.560	1.000	0.310	0.500	-0.050	0.005
0.056	0.310	1.000	0.560	-0.050	0.050
0.003	0.500	0.560	1.000	-0.003	0.010
-0.100	-0.050	-0.050	-0.003	1.000	-0.310
0.003	0.005	0.050	0.010	-0.310	1.000

Table 3. Segunda iteración en la creación de la Hesiana

Como se puede observar en las tablas 2 y 3, en cada iteración del algoritmo existen elementos de la matriz Hessiana que no es necesario volver a calcular puesto que ya se han calculado en iteraciones anteriores. Así, se reduce el tiempo computacional utilizado en cada iteración de la optimización de las SVM's.

4 Resultados obtenidos

Para probar la eficacia de la estrategia propuesta se seleccionaron diferentes bases de datos de clasificación. Las bases de datos fueron seleccionadas del repositorio UCI Machine Learning, [3]. Las bases de datos utilizadas fueron: Fonemas, Adult y Shuttle. Se seleccionaron bases de datos cuya complejidad (no linealidad) permitiera evaluar la estrategia propuesta.

Se realizaron diferentes pruebas y se determinó que la función Kernel de mejor desempeño fue el Kernel RBF con un valor de gamma igual a 0.5 y una constante de penalización C con valor de 1000.

Los resultados que se muestran en la Tabla 4 fueron obtenidos combinando la heurística para seleccionar los Vectores de Soporte con la estrategia de Caching. Además, se coloca el tiempo de entrenamiento utilizando únicamente la heurística (sin Caching). Como se puede observar, se mejora el tiempo de entrenamiento en las tres bases de datos seleccionadas para las pruebas con lo cual se comprueba la hipótesis de este trabajo: la combinación de estrategias permite mejorar el tiempo de entrenamiento de las SVM's.

	No. Datos	V. S.	Iteraciones	Heurística	Heurística + Caching
Fonemas	1027	326	8	3.7297 seg	3.7 seg
Adult	1478	903	9	29.2773 seg	17.4724 seg
Shuttle	13633	558	14	76.6192 seg	35.4312 seg

Table 4. Resultados obtenidos combinando la heurística con la estrategia Caching

Chunking	Cálculos totales	Cálculos con Caching	Cálculos recuperados con caching
10%	2527	928	1599
30%	3312	929	2383
50%	1717	924	793
80%	1266	946	320
100%	990	946	44

Table 5. Total de Cálculos en la creación de la Hessiana con la estrategia de "Caching"

5 Conclusiones

En este documento se ha mostrado una estrategia para reducir el tiempo de entrenamiento de las SVM's. La estrategia propuesta combina dos estrategias ya

existentes. La primera es una heurística geométrica que permite identificar los Vectores de Soporte del problema a priori, con lo que es posible entrenar con un grupo reducido de datos y en consecuencia se disminuye el tiempo de entrenamiento. La segunda consiste en almacenar de forma temporal aquellos cálculos más utilizados en la creación de la matriz Hessiana necesaria para optimizar las SVM's. Se probó la estrategia en 3 bases de datos y se mostró que el tiempo de entrenamiento sí se reduce con la estrategia propuesta.

6 Bibliografía

1. Vapnik Vladimir. *The Nature of Statistical Learning Theory*. Springer, 2nd edition. 1999.
2. C. Platt John. *Fast training of support vector machines using sequential minimal optimization*. In Bernhard Scholkopf, Christopher J. C. Burges, Alexander J. Smola, editors. *Advances in Kernel Methods: Support Vector Learning*. The MIT Press. 1998.
3. A. Asunción and D. J. Newman. *UCI machine repository*. 1997.
4. Rosenblatt. F. *The perceptron: A probabilistic model for information storage and organization in the brain*. *Psychological review*, 65(6):386–408, 1958.
5. A. Kowalczyk. *Maximal margin perceptron*. *Advances in Neural Information Processing Systems*, pages 75–114. 1999.
6. U. Ahmad, A. Gavrilov, Lee Sungyoung, and Lee Young-Koo. *Modular multilayer perceptron for wlan based localization*. In *Neural Networks, 2006. IJCNN '06. International Joint Conference on*, pages 3465–3471, 2006.
7. Ariel García Gamboa and Miguel González Mendoza: A Comparison of Different initialization Strategies to Reduce the Training Time of Support Vector Machines. In: Duch, W.K., J.; Oja, E.; Zadrozny, S. (ed.): ICANN, Vol. 3697. LNCS, Warsaw, Poland (2005) 613–618
8. Ariel García-Gamboa. Miguel González-Mendoza: Support Vector Optimization through Hybrids: Heuristics and Math Approach. MICAI 2009: 237-245

Desarrollo de un modelo difuso para diagnóstico de lesiones precursoras de cáncer cérvico uterino

Bertani David¹; Domínguez R. Karem¹; González Blanca E.¹; Aguilar Alberto A.¹; Espíndola Patricia.²

¹Instituto Tecnológico de Orizaba

{d_bertani, karem_dguez, aaguilaral, blancags67, patricia_ec7}@hotmail.com

²Hospital Regional de Río Blanco

Resumen. Existen muchos tipos de cáncer, uno de los que más preocupa principalmente a las mujeres, es el cáncer cérvico uterino (CaCu), que al igual que la mayoría de ellos, si es detectado a tiempo tiene una alta probabilidad de cura. En el presente trabajo de investigación se describe el desarrollo de un modelo de lógica difusa que basado en los factores de riesgo y hallazgos colposcópicos, disminuye la subjetividad existente entre los diferentes puntos de vista de los médicos colposcopistas. El presente trabajo representa una alternativa en la toma de decisiones para diagnosticar lesiones precursoras de cáncer. El resultado proporcionado por el modelo muestra una efectividad del 90% de coincidencia con el diagnóstico emitido por el médico especialista.

Palabras clave: Lógica Difusa, Colposcopia, Cáncer Cérvico Uterino.

1. Introducción

El cáncer de cuello uterino es el segundo más frecuente en la mujer, tras el cáncer de mama. De acuerdo con la Organización Mundial de la Salud (OMS), la incidencia de cáncer cervical en el mundo hasta el año 2008 fue de 530,232 casos nuevos y 275,008 defunciones al año ^[1]. En México, donde ha estado en marcha un programa de tamizaje durante más de 20 años, se han evitado menos del 13 por ciento de los casos potencialmente prevenibles. El Instituto Mexicano del Seguro Social (IMSS) diagnóstica 15 mil casos de CaCu anualmente, de acuerdo con su comunicado No. 227 (SSA, 2009). Así mismo, el CaCu es la primera causa de muerte por neoplasias en mujeres mayores a 25 años. En el Hospital Regional de Río Blanco (HRRB) de la ciudad de Río Blanco, Veracruz, el diagnóstico de CaCu es primordialmente establecido por el departamento o clínica de displasias, mediante un examen llamado Colposcopia, que es un procedimiento médico en el cual se utiliza un microscopio especial (llamado colposcopio) para observar dentro de la vagina y examinar detenidamente el cuello uterino (la abertura del útero, o matriz). Este proceso le permite al médico identificar problemas en el cérvix de la paciente antes de realizar la prueba de

Papanicolau que lo verifique. Sin embargo, la clínica de displasias se enfrenta con un inconveniente: la apreciación poco uniforme de los criterios en el momento de emitir un diagnóstico, ya que la confiabilidad de la prueba radica en la experiencia del médico ginecólogo. Como parte de los estudios a esta problemática se han realizado aportaciones importantes como la de Flores et al. 2005 ^[2], cuyo trabajo tiene como objetivo encontrar patrones temporales de textura para detectar lesiones precursoras de cáncer analizando imágenes de videos de colposcopia. Sierra Torres et al, 2006 ^[3], evalúan los principales factores de riesgo asociados a lesiones intraepiteliales escamosas de alto grado, mediante la realización de una encuesta y análisis estadístico en el departamento de Cauca, Colombia. Rodríguez et al. 2010 ^[4], presenta una metodología que extrae datos de imágenes colposcópicas y éstos alimentan a un sistema de lógica difusa para identificar el tipo de lesión precursora de cáncer de una paciente. Srinivasan et al, 2006 ^[5], presentan un sistema de segmentación automática y clasificación de lesiones de cérvix mediante procesamiento de imágenes para reducir la variación que producen los métodos de evaluación manual (apreciación del médico) que determinan la localidad para la toma de biopsia de diagnóstico. Pouletaut et al, 2001 ^[6], presentan una herramienta integrada de análisis para ayudar a los ginecólogos a realizar un diagnóstico colposcópico, mediante la selección interactiva de diferentes atributos para segmentación de imágenes colposcópicas.

Este trabajo presenta el desarrollo de un sistema experto basado en lógica difusa. Para empezar se describe la perspectiva médica, donde se explica de manera breve la manifestación del cáncer cérvico uterino, su evolución y sus factores de riesgo, así como una breve descripción de lo que es la colposcopia. Posteriormente, se describe el desarrollo del sistema y finalmente las últimas dos secciones muestran las pruebas, resultados y conclusiones.

2. Perspectiva médica

2.1 Manifestación de Cáncer

El cáncer es el crecimiento tisular patológico originado por una proliferación continua de células anormales. Puede originarse a partir de cualquier tipo de célula de los diferentes tejidos del organismo. En su evolución natural, estas lesiones pueden regresar, permanecer o progresar a cáncer. Sin embargo, la lenta evolución de la enfermedad y la accesibilidad de células del cérvix para su estudio, proporciona el tiempo necesario para detectar y erradicar la enfermedad, si el diagnóstico se realiza oportunamente ^[7].

2.2 Colposcopia

Muchos de los sistemas de exploración y diagnóstico emplean la colposcopia, un sistema óptico de visión ampliada del cuello uterino. La colposcopia se define como un medio de investigación con el cual es posible reconocer, delimitar y diagnosticar

los diferentes aspectos normales y anormales del exocérvix ^[7] El análisis de la imagen colposcópica permite detectar cambios patológicos y el grado de lesión de una paciente.

2.3 Factores de Riesgo

Existen varios factores de riesgo para desarrollar CaCu, considerado el virus de papiloma humano (VPH) como uno de los principales, el cual comúnmente se disemina a través de las relaciones sexuales. Otros factores deben coincidir con el VPH -se notifica una prevalencia del 38% de infección-, entre estos se encuentran: el inicio temprano de relaciones sexuales, la edad actual de la paciente, el número de hijos nacidos por conducto vaginal, múltiples parejas sexuales, tabaquismo y/o adicciones, uso de métodos anticonceptivos hormonales y finalmente antecedentes heredo-familiares de cáncer, ^[8].

2.4 Hallazgos colposcópicos

Los hallazgos colposcópicos están clasificados básicamente en normales, anormales y sugerentes de cáncer invasor, éstos se identifican dentro del proceso de análisis colposcópico (ver figura 1). Dentro de los anormales, se encuentran cambios menores (Lesión Intraepitelial de Bajo Grado o NIC I) y cambios mayores (Lesión Intraepitelial de Alto Grado o NIC II-III) ^[9].

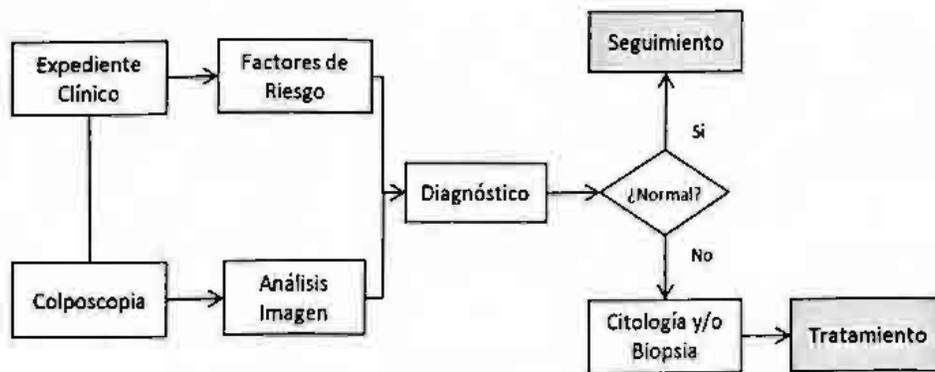


Figura 1. Secuencia de análisis colposcópico.

3. Desarrollo del Modelo de Lógica Difusa

La inteligencia artificial (IA) abarca en la actualidad una gran variedad de subcampos que van desde áreas de propósito general a otras más específicas como la demostración de teoremas matemáticos y el diagnóstico de enfermedades (sistemas expertos). Un sistema experto se puede definir como aquel programa de computadora que contiene el conocimiento de un especialista humano que ha sido entrenado en un determinado campo de aplicación. ^[10]

El modelo de lógica difusa que se desarrolla en esta investigación se basa en dos partes fundamentales: la toma de datos extraídos del expediente clínico de la paciente, los cuales son de relevancia para el médico al momento de emitir un diagnóstico y la segunda parte de la investigación esta sustentada en la imagen colposcópica de la paciente, la cual proporciona una puntuación determinada por el índice de Reid (RCI), que facilita una predicción colposcópica del diagnóstico histológico. Los datos obtenidos en ambas partes fueron considerados como variables de entrada al modelo. Esto puede se puede observar en la figura 2.

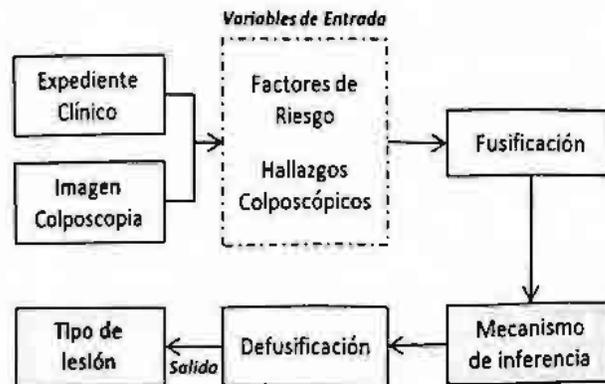


Figura 2. Modelo de Lógica Difusa.

3.1 Definición de Variables de lingüísticas

Expediente clínico e imagen colposcópica

Cada conjunto difuso fue definido previamente con la colaboración de la Dra. Patricia Espíndola Cisneros, ginecóloga del HRRB, con base a su experiencia y criterio. Para el caso de la imagen colposcópica se consideraron cuatro rangos en este conjunto difuso, para lo cual fue necesaria la participación de los especialistas del HRRB. Las variables se muestran en la tabla 1.

Cabe mencionar que estos rangos para la variable IC fueron determinados con los especialistas del hospital y con base a la "Terminología Colposcópica Actual" Ginecol Obstet Mex 2005; 73:653-60. En la tabla 2 se presenta la terminología para determinar los hallazgos colposcópicos.

En la tabla 2, cada especialista asigna una puntuación en cada imagen observada. La suma de dicho puntaje se toma como variable de entrada al modelo difuso. En la figura 3 se pueden observar los diferentes hallazgos o alteraciones colposcópicas.

Tabla 1. Conjuntos difusos del expediente clínico.

Variable de entrada	Descripción	Rangos
IVSA (Inicio de vida sexual)	Edad a la que la paciente inició su vida sexual.	Mayor riesgo, menor de 18 años. Riesgo medio, mayor de 17 años y menor de 19. Menor riesgo, mayor de 18 años.
E (Edad)	Las alteraciones citológicas aparecen con mayor frecuencia entre los 25 y 40 años, edades a las que pueden ser detectadas tempranamente para ser tratadas de acuerdo a su grado de malignidad. Medisur, 2008.	Jove, menor de 24 años. Adulta, mayor de 18 años y menor de 42. Mayor, mayor de 35 años.
P (paridad)	Número de embarazos llevados a término y que han dado a luz por vía vaginal. Las mujeres con 2 o 4 embarazos tienen 2.6 veces más riesgo de contraer CaCu que aquellas que no han dado a luz. Muñoz N, 2002.	Nula, 0 hijos. Habitual, de 1 a 7 hijos. Multiparidad, más de 3 hijos.
Pa (parejas)	Las mujeres con mayor número de parejas sexuales tienen mayor riesgo de contraer virus de transmisión sexual.	Habitual, menor a 5 parejas. Muchas, de 3 a 9 parejas. Excesivas, mayor a 7 parejas.
T (Tabaquismo)	Estudios demuestran que el riesgo para las mujeres que fuman se duplica el de las no fumadoras. ACCP, 2004.	Sí, 1 a 3 cigarrillos al día. No, 0 a 1 cigarrillo.
H (Uso de hormonales)	Indica si la paciente utiliza algún método de planificación familiar basado en hormonas sintéticas.	Sí No
IC (Imagen Colposcópica)	Valor asignado por el médico de acuerdo al índice de Reid.	Normal, de 0 a 2 puntos. Anormal, 1 a 4 puntos. Sospechoso, 3 a 13 puntos. Cancer In Situ, 12 a 14 puntos.

Fuente: Factores de riesgo de cáncer cérvico uterino invasor en mujeres mexicanas y Ginecol Obstet Mex 2005; 73:653-60.

Tabla 2. Terminología para hallazgos colposcópicos.

1. Normales
i. Epitelio escamoso original (0)
ii. Epitelio columnar o cilíndrico (0)
iii. Zona acetoblanc brillante mal definido (1)
2. Anormales
i. Epitelio acetoblanc plano (2)
ii. Mosaico fino (1)
iii. Puntilleo fino (1)
3. Sospechosos (mínimo de 2 aspectos)
i. Epitelio acetoblanc denso (3)
ii. Mosaico burdo (4)
iii. Puntilleo burdo (4)
4. Cáncer Invasor (mínimo de 3 aspectos)
i. Cambio acetoblanc denso (4)
ii. Puntilleo grueso (4)
iii. Vasos atípicos (5)
iv. Bordes con relieve (5)

Fuente: Terminología Ginecol Obstet Mex 2005; 73:653-60

DIAGNOSTICOS COLPOSCOPICOS

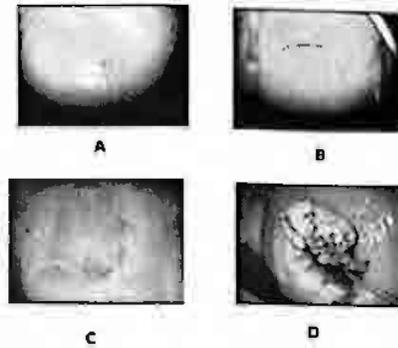


Figura 3. Imagen colposcópica. A) Sin Alteraciones. B) Alteración inflamatoria. C) Lesión intraepitelial de bajo grado. D) lesión intraepitelial de alto grado.

3.2 Fusificación

El desarrollo del modelo se realizó en Matlab, se eligió un sistema tipo Mamdani¹ y se procedió a fijar los parámetros básicos para cada variable de entrada y la variable de salida, al igual que las reglas difusas, que como se mencionó antes, se establecieron de acuerdo al criterio del experto. Los intervalos y tipos de cada conjunto difuso se aprecian en la tabla 3.

En la siguiente fase del proceso se asignan o calculan los valores que representen el grado de membresía o pertenencia para todos los conjuntos difusos previamente definidos sobre cada variable de entrada. Para ejemplificar lo anterior se presenta la variable E (edad de la paciente) en la figura 4.

Las variables lingüísticas de entrada se consideraron de acuerdo a los factores de mayor riesgo considerados por los médicos especialistas. La variable de salida consta de cuatro conjuntos difusos descritos anteriormente en la tabla 2. La variable de salida puede ser apreciada en la figura 5.

¹ El modelo Mamdani se caracteriza por la utilización de reglas difusas las cuales son un conjunto de proposiciones del tipo IF –THEN que modelan la problemática que se quiere resolver. Una regla difusa tiene la forma: *if (x is A) and (y is B) then (z is C)*^[10].

Tabla 3. Discretización difusa de las variables de entrada.

Variable de entrada	Intervalo	Tipo	Etiqueta
IVSA (Inicio de vida sexual)	[0 0 17 18]	Trapmf	Mayor riesgo
	[17 18 19]	Trimf	Riesgo medio
	[18 19 30 30]	Trapmf	Menor riesgo
E (Edad)	[0 0 18 24]	Trapmf	Joven
	[18 24 35 42]	Trapmf	Adulta
	[35 42 60 70]	Trapmf	Mayor
P (paridad)	[0 0 1]	Trimf	Nula
	[0 1 3 7]	Trapmf	Habitual.
	[3 7 7 7]	Trapmf	Multiparidad
Pa (parejas)	[0 0 3 5]	Trapmf	Habitual
	[3 5 7 9]	Trapmf	Muchas
	[7 9 9 10]	Trapmf	Excesivas
T (Tabaquismo)	[0 3 3]	Trimf	Sí
	[0 0 1]	Trimf	No
H (Uso de hormonales)	[0 1 1]	Trimf	Sí
	[0 0 1]	Trimf	No
IC (Imagen colposcópica)	[0 0 9 13]	Trapmf	SA
	[9 13 16 21]	Trapmf	AI
	[16 21 25 29]	Trapmf	LIBG
	[25 29 40 40]	Trapmf	LIAG

Fuente: Tabla creada por los investigadores.

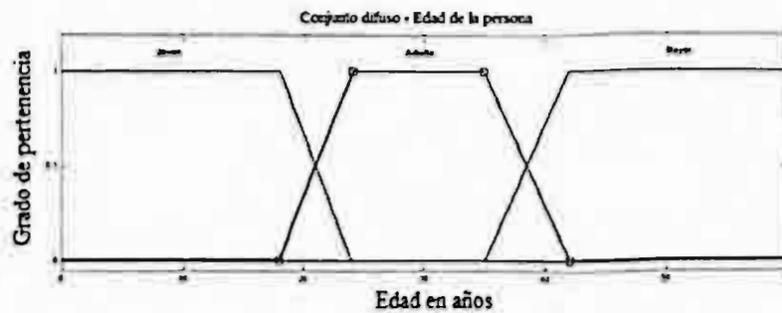


Figura 4. Variable E

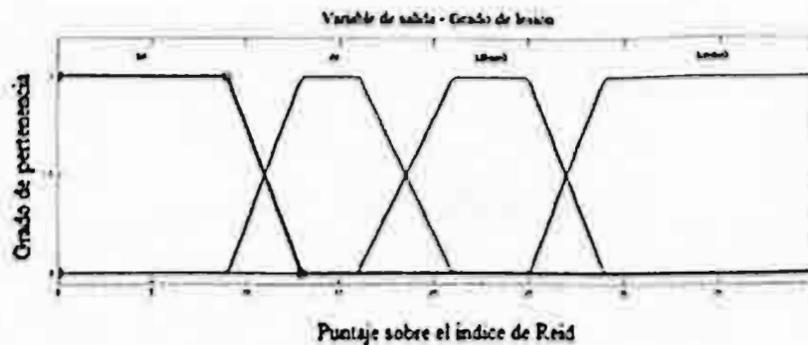


Figura 5. Variable de salida del modelo difuso

3.3 Reglas de inferencia

Las reglas de inferencia se generaron por producto cartesiano, obteniendo un total de 1290 reglas. Ver tabla 4.

Tabla 4. Reglas de inferencia del modelo.

No. Regla	Puntaje imagen	Inicio de Vida sexual	Edad	Paridad	No. Prejuz sexuales	Adicciones	uso de Hormonales	Salida del Sistema
781	2	Menor riesgo	Adulta	Nula	Excesivas	Si	Si	AI
334	5	Mayor riesgo	Joven	Nula	Excesivas	Si	No	LIBG
884	9	Menor riesgo	Joven	Habitual	Muchas	No	No	LIBG
901	12	Menor riesgo	Adulta	Nula	Habitual	Si	Si	LIBG
994	12	Mayor riesgo	Joven	Habitual	Excesivas	Si	No	LIAG
1107	17	Riesgo Medio	Joven	Multiparidad	Habitual	No	Si	LIAG
1243	15	Menor riesgo	Adulta	Habitual	Muchas	No	Si	LIBG

Fuente: reglas creadas por los investigadores y en conjunto con especialistas del HRRB.

3.3 Defusificación

El proceso de defusificación de este modelo fue realizado bajo el método de centroide de Mamdani. De acuerdo al resultado que arrojen las reglas de inferencia, se determinan el valor de salida en puntos, los cuales son ponderados por los conjuntos difusos de la variable de salida. Los conjuntos se determinaron de forma trapezoidal ya que los especialistas determinaron un nivel de certeza igual a 1 en cada conjunto. Los extremos de cada conjunto muestran difusidad en el diagnóstico, ya que pertenecen en cierto grado a un conjunto y su complemento a otro conjunto subsecuente. Dentro de la variable de salida se consideraron cuatro diagnósticos principales, los cuales representan los conjuntos difusos de respuesta al sistema (figura 5). Los tipos de lesión obtenidos como respuesta al sistema son SA, AI, LIBG Y LIAG. De este modo, el sistema sugiere el tipo de lesión presente en la paciente, siendo el médico quien toma la decisión final para emitir un diagnóstico.

4. Pruebas y Resultados

Esta metodología ha sido probada en imágenes colposcópicas, las cuales fueron seleccionadas de pacientes que acuden por primera vez al HRRB a realizarse un estudio de colposcópico. Cada caso cuenta con 1 o 2 imágenes colposcópicas según lo considere el médico, y 6 factores de riesgo obtenidos del expediente clínico que se le genera. Los casos fueron tomados de acuerdo a una muestra aleatoria de los casos registrados de marzo 2010 a marzo 2011, proporcional al número de casos por padecimiento. Utilizando la fórmula (1), con error permitido del 1% y un índice de confiabilidad del 99%, de una población de 247 casos de pacientes que acudieron por primera vez al estudio de colposcopia en el año señalado anteriormente, se obtuvieron 71 casos para prueba. Lo que significa un 28.7 % del total de la población de estudio.

$$n = \frac{n'}{1+n'/N} \quad (1)$$

Donde:

$$n' = S^2 / \sigma^2, \quad \sigma^2 = 1.0 \quad \text{y} \quad S^2 = p(1-p)$$

De los casos seleccionados aleatoriamente, se selecciona la mejor imagen para el análisis y los factores de riesgo registrados en el historial clínico de la paciente, así como la puntuación que se le otorga a la imagen de acuerdo a lo que observa el médico colposcopista. Estos factores de riesgo son registrados por el sistema experto como se muestra en la figura 6.



Figura 6. Sistema experto desarrollado para el HRRB.

Se comparan los resultados obtenidos después de aplicar este método a los diferentes casos y el diagnóstico previo realizado por los médicos especialistas, donde se muestra que, en un 90% de los casos, el diagnóstico emitido por el sistema es el mismo que el otorgado por el médicos, quienes consideran que la eficacia del sistema es alta en comparación con el de Rodríguez F et al. [4] Aunque la colposcopia es un método de detección, la certificación diagnóstica de la lesión cervical se obtiene mediante el examen citológico e histológico de acuerdo al grado de lesión que se observe en la colposcopia. Sin embargo, no se cuenta con los resultados de dichos exámenes para los casos de estudio, por lo que no es posible verificar los diagnósticos en falsos positivos o falsos negativos.

Por otra parte, cuando una lesión es señalada como lesión de bajo grado o alto grado, se requiere hacer una biopsia para confirmarla. Como parte del procedimiento, el especialista (de manera muy subjetiva) localiza el área de mayor pigmentación donde considera que la lesión es mayor y toma la muestra. Para reducir la subjetividad en este proceso se complementa con análisis de imagen capaz de identificar el área de lesión en la imagen colposcópica con el objetivo de señalar el lugar de mayor anomalía y sirva de referencia para la toma de la biopsia en caso de ser requerida como se muestra en la figura 6.

5. Conclusiones y Recomendaciones

El objetivo principal de esta investigación es ayudar en el diagnóstico oportuno del cáncer cérvico uterino a nivel colposcopia. Para cada anomalía detectada, el sistema experto emite un nivel de lesión. Con base en los resultados obtenidos, la metodología proporciona un soporte para un diagnóstico clínico más detallado. Este sistema tiene un importante impacto en hospitales con recursos limitados de inversión en modernas tecnologías, ya que el sistema sólo requiere de un software comercial. Cabe mencionar que el modelo diseñado también es capaz de identificar la zona de lesión y sugerir el lugar óptimo para la toma de la biopsia. Actualmente, en el Instituto Tecnológico de Orizaba, se desarrollan dos trabajos por parte de los autores de este artículo, que cubren las áreas de citología e histología como sistemas complementarios en el diagnóstico de cáncer cérvico uterino.

Referencias

- [1] Organización Mundial de la Salud (OMS) Proyecto GLOBOCAN 2008.
- [2] Flores. A.B. Robles. L.A. Tepalt. R.M.M. Aragon. J.D.C "Identifying precursory cancer lesions using temporal texture analysis". Computer and Robot, Vision 2005. Proceedings. The 2nd Canadian Conference on. 9-11 May 005. ISBN: 0-7695-2319-6
- [3] Carlos H, Sierra Torres, Maria P. Acosta Aragón y Leonora Orejuela Aristizabal. "Papiloma virus y factores asociados a la Neoplasia Intraepitelial Cervical de Alto grado en Cauca, Colombia" Revista de Salud Pública, mayo, año/vol. 8 Universidad Nacional de Colombia, Bogotá Colombia. 2006
- [4] Fabiola Rodríguez Machorro. "Desarrollo de un modelo difuso para diagnosticar el nivel de lesión precursora de cáncer cérvico uterino mediante la extracción de características de imágenes colposcópicas y análisis de factores de riesgo". Instituto Tecnológico de Orizaba, 2010.
- [5] Yeshwanth Srinivasan, Fei Gao, Bhakti Tulpule, Shuyo Yang, Sunanda Mitra y Brian Nutter. "Segmentation and classification of cervix lessions by pattern and texture analysis". Int. J. Intelligent Systems Technologies and Applications, Vol 1. Nos ¾. 2006.
- [6] Claude, I. Pouletaut, P. Huault, S. Boulanger. J.C. "Integrated color and texture tools for colposcopic image segmentation", Biomechanic & Biomed. Eng., Univ. de Technol. De Compiègne. 7-10 October 2001. ISBN: 0-7803-6725-1
- [7] Lewis, Merle J., "Análisis de la situación del cáncer cervicouterino en América Latina y el Caribe " Organización de Panamericana de la Salud (OPS), pp. 40, 2004.
- [8] Laura Leticia Tirado-Gómez, Alejandro Mohar-Betancourt, Malaquias López-Cervantes, Alejandro García-Carrancá, Francisco Franco-Marina, Guilherme Borges, "Factores de riesgo de cáncer cervicouterino invasor en mujeres mexicanas". 21th International Papillomavirus Conference en la Ciudad de México, D.F., 22-26 de febrero de 2004. salud pública de México/vol.47, no.5, septiembre-octubre de 2005.
- [9] Vargas Hernández VM y Ruiz Moreno JA, "Terminología Colposcópica actual" Ginecol Obstet Mex 2005; 73:653-60.
- [10] Martín de Brío; Sanz Molina., "Redes Neuronales y Sistemas Difusos" alfaomega, 2ª edición. Colombia 2005

Metaheurístico híbrido: Cómputo evolutivo y recocido simulado

Sergio Alex Caballero-Caballero, Jaime Mora-Vargas, and Ivo Erasmo Buzón-Cantera

Tecnológico de Monterrey Campus Estado de México

Resumen En la literatura se han propuesto una amplia variedad de técnicas metaheurísticas para resolver los problemas de Calendarización de horarios. Actualmente, el interés por los metaheurísticos híbridos se ha incrementado entre los investigadores de optimización combinatoria. En este sentido, este trabajo presenta el desarrollo y la aplicación de una técnica híbrida de optimización basada en cómputo evolutivo y recocido simulado para la solución del problema de calendarización de horarios.

Keywords: Metaheurísticos, Cómputo evolutivo, Recocido Simulado

1. Introduction

El cómputo evolutivo, más concretamente los algoritmos genéticos, y el recocido simulado han sido ampliamente estudiados. Desde su aparición, los primeros en los años 60 y en la década de los 80 los segundos, se han utilizado en la resolución una gran cantidad y variedad de problemas combinatorios complejos. Ambos algoritmos pertenecen a la familia de los Metaheurísticos y son algoritmos de búsqueda, en los cuales hay que balancear entre la exploración y la explotación del espacio de soluciones. En los últimos años ha habido un gran avance tecnológico en el área de computación tanto en velocidad como en facilidad de operación, parece lógico desarrollar híbridos que buscan en la unión la obtención de mejores resultados. Al realizar la hibridación es importante complementar las buenas características de cada uno para obtener un buen balance entre la exploración y la explotación. El híbrido propuesto es un algoritmo genético que incluye un operador de mejora individual. De esta forma los individuos de la población son mejorados genéticamente mediante el recocido simulado para luego ser cruzados y obtener nuevos individuos.

Para poder evaluar el algoritmo híbrido propuesto contra sus padres se utiliza un problema combinatorio. El problema seleccionado es el problema de calendarización de horarios. Se escogió este problema por ser un problema combinatorio, ampliamente estudiado y poder contar con un banco de problemas suficientemente grande y de reconocida complejidad. Una característica en los problemas de calendarización de horarios es que permite fácilmente modificar la dificultad del problema en relacionada con la cantidad de restricciones del problema.

La siguiente sección presenta una descripción del problema de calendarización de horarios (timetabling). La sección 3 presenta el estado del arte en la resolución del problema de calendarización de horarios con cómputo evolutivo y recocido simulado. La sección 4 presenta el algoritmo genético propuesto y la sección 5 presenta el algoritmo recocido simulado utilizados en este trabajo. La sección 6 presenta el algoritmo híbrido desarrollado y propuesto. La sección 8 presenta las instancias del problema de calendarización de horarios analizadas.

2. Descripción del problema

Los problemas de calendarización de horarios (TTP, por sus siglas en inglés) son un tipo específico de problemas de optimización que pueden estar sujetos a gran cantidad de restricciones y ser difíciles de resolver. El problema general de calendarización de horarios consiste en asignar un conjunto de eventos a un número limitado de periodos de tiempo, minimizando las violaciones a un conjunto de restricciones [2]. Las restricciones asociadas a un problema particular usualmente se clasifican en dos tipos [16]. Las restricciones duras son aquellas restricciones que bajo ninguna circunstancia deben ser violadas. Es necesario el cumplimiento de este tipo de restricciones para que una solución sea factible. Las restricciones suaves son condiciones deseables pero no es esencial cumplirlas. De hecho, usualmente es imposible satisfacer todas las restricciones suaves en un problema dado. En consecuencia, el TTP es un problema de optimización que busca la combinación de eventos que satisfaga todas las restricciones duras y que a su vez maximice la cantidad de restricciones suaves satisfechas. El TTP es un problema NP-duro [4] [16]. No se conoce ningún algoritmo polinomial que puede resolver este tipo de problema [14]. Esto conduce a tiempos de cómputo de tal magnitud que los hace no aplicables en problemas prácticos.

La 2ª competencia internacional de horarios (2ª^d international timetabling competition, 2008) dentro de la Serie Internacional de Conferencias sobre la Práctica y Teoría de calendarización de horarios Automatizado, PATAT (International Series of Conferences on the Practice and Theory of Automated Timetabling), es una competencia de algoritmos para la resolución de un conjunto de instancias del TTP [15], en particular del Post Enrolment based Course Timetabling Problem (PEPCTP). El PEPCTP consiste en un conjunto E de eventos que deben ser programados en 45 periodos de tiempo (5 días de 9 horas cada uno), un conjunto R de salones en los cuales tienen lugar los eventos, un conjunto S de estudiantes que asisten a los eventos, un conjunto F de características presentes en los salones y requeridas por los eventos, un conjunto D que indica los periodos disponibles para un evento dado y un conjunto P que indica condiciones de precedencia entre los eventos (de tal forma que ciertos eventos deben ocurrir antes que otros). Cada estudiante asiste a varios eventos y cada salón tiene capacidad de estudiantes limitada. Un horario factible es aquél en el cual todos los eventos han sido asignados a un periodo de tiempo y a un salón de tal manera que se satisfacen las siguientes restricciones duras [15] :

- Ningún estudiante asiste a más de un evento al mismo tiempo.

- El salón es lo suficientemente grande para todos los estudiantes asistentes y satisface todas las características requeridas por el evento.
- Solo hay un evento en cada salón en cualquier periodo de tiempo.
- Los eventos solo pueden ser asignados a periodos de tiempo disponibles.
- Respetar las condiciones de precedencia entre los eventos.

Además, se penaliza equitativamente por la ocurrencia de las siguientes violaciones a las restricciones suaves:

- Un estudiante tiene una clase en el último periodo del día.
- Un estudiante tiene más de dos clases consecutivas.
- Un estudiante tiene una sola clase en el día.

El PEPC^{TP} es un problema tan retador que incluso obtener una solución factible, una solución que respete las restricciones duras, es difícil. El objetivo de este trabajo no es obtener un algoritmo para resolver óptimamente el PEPC^{TP} (mejor que cualquier otro algoritmo conocido) sino en comparar tres algoritmos para lo cual se toma el PEPC^{TP} como elemento de comparación.

3. Estado del arte

Los algoritmos genéticos (GA por sus siglas en inglés) se enmarcan en el área del cómputo evolutivo y son métodos de búsqueda basados en la mecánica de la selección y genética natural [6]. Los algoritmos genéticos fueron inventados por John Holland en los 60s y desarrollados por Holland, alumnos y colegas de la Universidad de Michigan en los 60s y 70s [11]. Los algoritmos genéticos son un método estocástico de búsqueda global que imita la evolución biológica natural. Los GA operan sobre una población de soluciones potenciales aplicando el principio de supervivencia del más apto para producir mejores soluciones. En cada generación, se crea un nuevo conjunto de soluciones mediante el proceso de selección de individuos según su nivel de aptitud en el dominio del problema y reproduciéndolos usando operadores prestados de la genética natural. Este proceso lleva a la evolución de poblaciones de individuos mejor adaptados que los individuos de quienes fueron creados, al igual que en la adaptación natural. Entre las aplicaciones de algoritmos genéticos al TTP podemos citar [12], [5] y [10]. Paccher y otros [12] describen dos enfoques para resolver el TTP empleando algoritmos genéticos. Los métodos permiten producir no sólo horarios factibles sino horarios considerados buenos respecto a alguna función de evaluación especificada. Los métodos transforman el espacio de búsqueda a uno en el cual la proporción de soluciones factibles es considerablemente mayor. Este nuevo espacio de búsqueda es explorado por un algoritmo genético. Fernandes y otros [5] describen un método para generar horarios escolares empleando un nuevo operador denominado mutación de malos genes que permite mejorar la velocidad del algoritmo y los resultados. Mahdi y Zainuddin [10] describen una herramienta de optimización de un algoritmo genético con control adaptativo de parámetros. Su investigación muestra que controlando las configuraciones de

los parámetros de los operadores genéticos es posible mejorar la calidad de los horarios.

Desde que Kirkpatrick y otros [9] introdujeron hace un poco más de 20 años el concepto de Recocido Simulado (SA, por sus siglas en inglés), esta metaheurística ha demostrado ser una herramienta exitosa para resolver una amplia gama de problemas de optimización combinatoria. Su nombre proviene de la idea en la que está basado, simular el enfriamiento de materiales (proceso denominado recocido) [3]. El recocido simulado es el más antiguo entre los metaheurísticos y, con seguridad, uno de los primeros algoritmos que tienen una estrategia explícita para salir de los mínimos locales. La idea fundamental es permitir movimientos que resultan en soluciones de peor calidad que la solución actual (movimientos ascendentes) con el propósito de escapar del mínimo local. La probabilidad de hacer tal movimiento disminuye durante la búsqueda. Entre las aplicaciones del recocido simulado al TTP están los artículos de Abramson [1] y Wright [17]. En el trabajo de Abramson [1] se establecen elementos, cada uno de los cuales corresponde a una tupla formada por una clase de estudiantes, un profesor, una asignatura y un salón de clases. Se define una función de costo, se genera una solución inicial asignando los elementos a un período elegido aleatoriamente, y se calculan el costo y la temperatura inicial. La condición de paro exige que el costo sea cero, o bien que el costo no cambie en cierto número de iteraciones. Wright [17] establece la función de costo como una combinación lineal del conflicto entre las preferencias de enseñanza y las insuficiencias departamentales. La búsqueda procede sistemáticamente a través de los vecinos sin aleatoriedad. Se emplea un esquema de enfriamiento geométrico, mientras que el fin del proceso se fija después de cierta cantidad de iteraciones para asegurar que estén cerca del óptimo.

4. Algoritmo genético propuesto

Esta sección describe el algoritmo genético utilizado para resolver el PEPCITP. El cromosoma utilizado es una matriz en la cual los renglones representan los periodos de tiempo y las columnas representan los salones, de tal manera que los elementos de la matriz correspondan a los eventos programados y tiene el valor cero o vacío si no tiene un evento asignado [7]. La figura 1 representa el algoritmo genético propuesto, cabe resaltar que es un algoritmo básico.

4.1. Función de aptitud

Como proponen Pacchter y otros [13], la función de aptitud evalúa la calidad de un horario para poder comparar dos soluciones dadas. El primer paso para resolver el PEPCITP es obtener un horario factible, pero esto en ocasiones no se puede lograr. Por lo tanto, se considera un horario semi-factible aquel que respeta las restricciones duras, pero en el horario no están presentes todos los eventos. Por lo tanto la función aptitud es la suma de los estudiantes de los eventos omitidos en el horario. De tal forma que dado un horario completo, en el

cual todos los eventos están presentes, pero no factible se realiza un proceso para convertirlo en un horario semi-factible y poder obtener el valor de la función de aptitud. Los problemas del PEPC'TP tienen una gran complejidad y la obtención de soluciones factibles con algoritmos poco especializados no es posible en un tiempo adecuado.

4.2. Operador de cruce

La matriz del cromosoma se transforma en una matriz línea de tal forma que los primeros elementos corresponden a los periodos de tiempo del primer día y luego los periodos del segundo día y así hasta tener completa la matriz línea cromosoma. El operador de cruce empleado parte de dos individuos originales, P1 y P2, y genera otros cuatro, H1, H2, H3 e H4. Inicialmente, se selecciona aleatoriamente un punto de cruce (el mismo para ambos padres). Se copia la primera parte de P1 a H1 e H2 y la segunda parte de P1 a H3 e H4 y se copia la primera parte de P2 a H3 e H4 y la segunda parte de P2 a H1 e H2. Luego se asignan los eventos faltantes de manera aleatoria en los periodos de tiempo libres en la primera parte en H1 y H3 y en la segunda parte en H2 y H4. Finalmente de los seis individuos obtenidos se seleccionan los 2 mejores para la siguiente generación.

4.3. Operador de mutación

El operador de mutación empleado requiere que se generen aleatoriamente dos puntos y una longitud dentro del individuo, luego se intercambia la sección de longitud aleatoria a partir del primer punto con la sección de la misma longitud que empieza en el segundo punto. El procedimiento descrito se muestra en la figura 2.

Para determinar la probabilidad de mutación se evaluaron diferentes valores de probabilidad 0.001, 0.005, 0.01, 0.05, 0.1 y 0.15. Como se observa en la figura 3, la probabilidad de 0.15 es aquella que proporciona mejores resultados en términos de la función objetivo para el mejor generacional. En consecuencia ese es el valor empleado en el presente trabajo.

5. Algoritmo de Recocido Simulado

Esta sección presenta el algoritmo recocido simulado para resolver el PEPC'TP y se muestra en la figura 4. El algoritmo empieza generando una solución inicial, s , (obtenida de forma aleatoria, asignando los eventos aleatoriamente en el horario) e inicializando el parámetro de temperatura T . Luego se repite el ciclo interior hasta que se satisface la condición de terminación. Una solución s' del vecindario $N(s)$ se muestrea aleatoriamente. Se calcula el valor de la función aptitud para la solución s ($f(s)$) y para la solución s' ($f(s')$). s' reemplaza a s si $f(s') > f(s)$ (en caso de maximización) o, en caso que $f(s') \leq f(s)$ con una

probabilidad que es una función del valor de T y $f(s') - f(s)$. La probabilidad se calcula con la distribución de Boltzmann.

$$P = \exp\left(\frac{f(s') - f(s)}{T}\right) \quad (1)$$

Una vez terminado el ciclo interno se realiza el ciclo externo que es controlado por el factor de equilibrio térmico. La temperatura T disminuye durante el proceso de búsqueda, así al principio de la búsqueda la probabilidad de aceptar movimientos ascendentes es alta y decrece gradualmente convergiendo a un algoritmo de búsqueda local.

6. Algoritmo híbrido propuesto

Esta sección presenta el algoritmo híbrido desarrollado y utilizado para resolver el PEPCTP, el cual se presenta en la figura 5. El algoritmo básicamente es incluir en el algoritmo genético un paso en el cual se aplica el recocido simulado a cada individuo de la población, previamente a los operadores genéticos.

Principalmente, al incluir el recocido simulado en el algoritmo genético para cada individuo de la población se realiza una mejora genética individual. De esta forma se busca mejorar los individuos y la velocidad del proceso evolutivo propio del algoritmo genético. Hay que resaltar que la mejora genética se basa en vecindades diferentes a las utilizadas en el algoritmo genético y permite considerar individuos diferentes que difícilmente se pueden obtener por cruce o mutación.

En consecuencia, la exploración y explotación del algoritmo híbrido es mejor. El recocido simulado al no tener un método para aumentar la temperatura, se identifica claramente primero una fase de exploración y posteriormente una fase de explotación. Sin embargo el algoritmo genético en cada generación se mezcla la explotación con el operador de cruce y la exploración con el operador de mutación, pero no se realizan de manera muy continuada como en el caso del recocido simulado.

7. Experimentación

Las instancias utilizadas para comparar los algoritmos provienen de la 2ª competencia internacional de horarios (2008). Se seleccionaron los problemas del 1 al 8 [15]. En el cuadro 1 se presentan las características de cada problema. Por cada problema se realizaron 6 replicas. Las replicas se realizaron en una computadora con procesador Intel Core2 Duo CPU T7300 a 2.00 GHz cada uno y con una memoria RAM de 3.00 GB en el sistema operativo Windows 7 Professional SP1. Los algoritmos se desarrollaron en el mismo software y lenguaje de programación (Matlab v2009a).

Es importante resaltar que para realizar una comparación justa entre los tres algoritmos los parámetros de los mismos se ajustan para que el tiempo de trabajo sea el mismo. En realidad se modifican los algoritmos, para que el

criterio de paro sea un tiempo de proceso igual para todos. Los experimentos tienen un tiempo de proceso de 10 minutos por replica. La población en el algoritmo genético es de 100 individuos, la probabilidad de mutación es del 0.15% y el número de generaciones fue alrededor de 200. En el algoritmo de recocido simulado la temperatura inicial y final son respectivamente de 10⁴0 y 0.1, el valor de α es 0.989 y el valor de Maxiter es de 150. En el algoritmo híbrido se definieron los siguientes parámetros: Temperatura inicial: 500, temperatura final: 5, α : 0.8, Maxiter: 20, Número de individuos: 8, Probabilidad de mutación: 0.1 y generaciones: se realizan alrededor de 40. El cuadro 2 presenta los resultados numéricos de los experimentos. Se observan con fondo gris los mejores resultados obtenidos por problema, el algoritmo híbrido obtiene el mejor resultado en cinco problemas y el algoritmo genético en los tres restantes.

La figura 6 presenta una gráfica del valor promedio y mejor valor de los algoritmos genético e híbrido respecto de la generación. La figura 6 presenta un mejor desempeño del algoritmo híbrido, esto se debe a que en cada generación se realiza una búsqueda más extensa. La figura 7 presenta la gráfica del mejor valor de la población de los 6 experimentos de los algoritmos genético e híbrido respecto del tiempo de proceso. En ambas figuras se representa el problema 1. En la figura 7 el algoritmo híbrido también presenta mejores resultados que el algoritmo genético, sobretodo se observa una rápida mejora de la solución en los primeros minutos de proceso. En la figura 7 y en el cuadro 2 se observa que no se presentan grandes desviaciones entre los experimentos de un mismo algoritmo. Este efecto también se observa para los otros problemas y en el algoritmo de recocido simulado.

8. Trabajo futuro y conclusiones

En el presente trabajo se propone un algoritmo híbrido para resolver el TTP. El algoritmo híbrido es comparado con un algoritmo genético propuesto y un algoritmo de recocido simulado. Los resultados obtenidos muestran un mejor desempeño del algoritmo híbrido. Se observa que el tiempo de proceso es relativamente corto y no alcanzan una fase de estabilidad típica en ellos. Por ello es importante realizar experimentos con un mayor tiempo de proceso. Es muy posible que el algoritmo híbrido al contar con varias estrategias para definir las soluciones vecinas obtenga mejores soluciones.

Considerando los resultados obtenidos y con el objeto de enriquecer la investigación efectuada, se sugiere realizar el siguiente trabajo futuro: 1) Implantar y evaluar diferentes criterios de vecindad en el recocido simulado. 2) Considerar otros operadores de cruce y mutación en los algoritmos genéticos. 3) Considerar métodos analíticos de sintonización de parámetros. 4) Adicionalmente, evaluar otras instancias del PEPTP y evaluar otros problemas combinatorios.

Referencias

1. Abramson, D., Constructing school timetables using simulated annealing: sequential and parallel algorithms. *Management Science*, 37-1, 98-113 (2001).

2. Burke, E.K. and Petrovic, S., Recent research directions in automated timetabling. *European Journal of Operational Research*, 140-2, 266-280 (2002).
3. Dowsland, K. y Díaz, B., Diseño de heurísticas y fundamentos del recocido simulado. *Revista Iberoamericana de Inteligencia Artificial*, 20, 34-52 (2001).
4. Even, S., Atai, A. y Shamir, A., On the complexity of timetabling and multicommodity flow problems. *SIAM Journal of Computation*, 5-4, 691-703 (1976).
5. Fernandes, C., Caldeira, J., Melicio, F. y Rosa, A. High school weekly timetabling by evolutionary algorithm. *Proceedings of the 1999 ACM Symposium on Applied Computing*, 344-350 (1999).
6. Goldberg, D.E., *Genetic Algorithms in Search, Optimization and Machine Learning*. Addison Wesley (1989).
7. Gómez, J.L., *Algoritmos Genéticos con Diversidad Forzada para la resolución del problema Timetabling Educativo*. Tesis de maestría, Instituto Tecnológico y de Estudios Superiores Monterrey, Campus Cuernavaca, México (2005).
8. Gonzalez, C., *Educational timetabling resuelto con Recocido Simulado y modelado bajo una arquitectura Web. Generalización y Aplicación Práctica: Casos PATAT y UADY*. Tesis de maestría, Instituto Tecnológico y de Estudios Superiores Monterrey, Campus Cuernavaca, México (2005).
9. Kirkpatrick, S., Gelatt, C. D. y Vecchi, M. P., Optimization by Simulated Annealing. *Science*, 220-4598, 671-680 (1983).
10. Mahdi, O. y Zaimuddin, R., Using a genetic algorithm optimizer tool to generate good quality timetables. *Proceedings of the 10th IEEE International Conference on Electronics, Circuits and Systems (ICECS 2003)*, 1300-1303 (2003).
11. Mitchel, M., *An Introduction to Genetic Algorithms*. The MIT Press, (2001).
12. Paechter, B., Cumming, A., Luchian, H. y Perinc, M., Two solutions to the general timetable problem using evolutionary methods. *Proceedings of the IEEE World Congress in Computational Intelligence*, 300-305 (1994).
13. Paechter, B., Rossi-Doria, O., Blum, C., Knowles, J., Sampels, M. y Socha, K., A local search for the timetabling problem. *Proceedings of the Fourth International Conference on the Practice and Theory of Automated calendarización de horarios*, 124-127 (2002).
14. Papadimitriou, C.H. y Steiglitz, K., *Combinatorial Optimization. Algorithms and Complexity*. Dover Publications, Inc. (1982).
15. PATAT, 2nd international timetabling competition. [http:// www.cs.qub.ac.uk/itc2007/index.htm](http://www.cs.qub.ac.uk/itc2007/index.htm). Consultado el 5 /Agosto/ 2011 (2011).
16. Schaer, A.A., A survey of automated timetabling. *Artificial Intelligence Review*, 87-127 (1999).
17. Wright, M., School timetabling using heuristic search. *Journal of the Operational Research Society*, 47-3, 347-357 (1996).

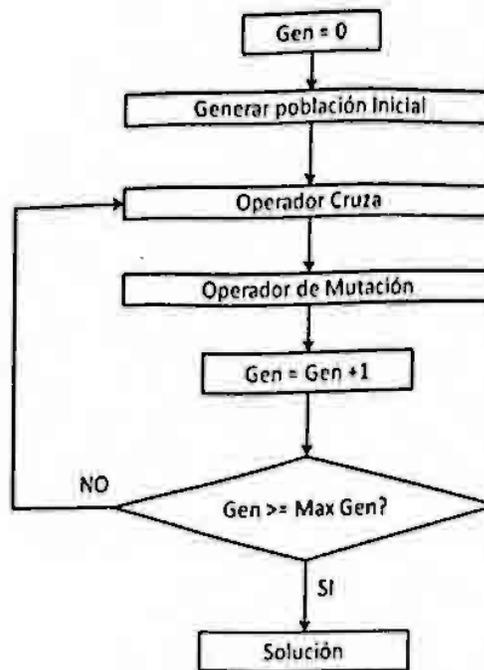


Figura 1. Esquema del algoritmo genético utilizado para el PEPCTP

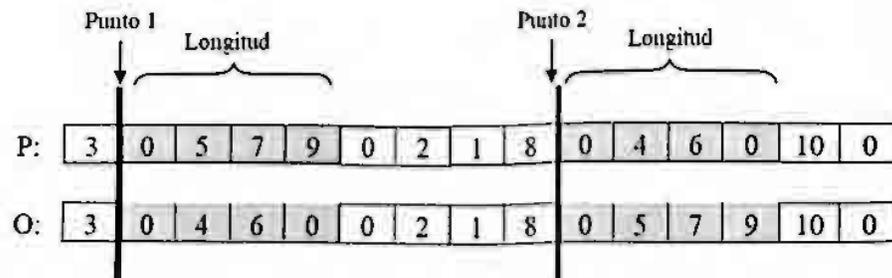


Figura 2. Operador de mutación

Problema	Número de eventos	Número de salones	Número de características	Número de estudiantes
1	400	10	10	500
2	400	10	10	500
3	200	20	10	1000
4	200	20	10	1000
5	400	20	20	300
6	400	20	20	300
7	200	20	20	500
8	200	20	20	500

Cuadro 1. Problema 1 a 8 del PEPCTP

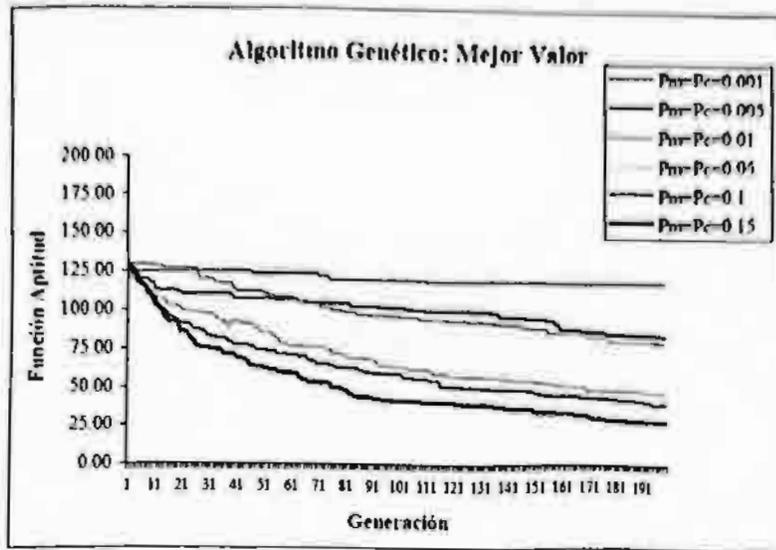


Figura 3. Algoritmo genético: Mejor valor para diferentes probabilidades mutación

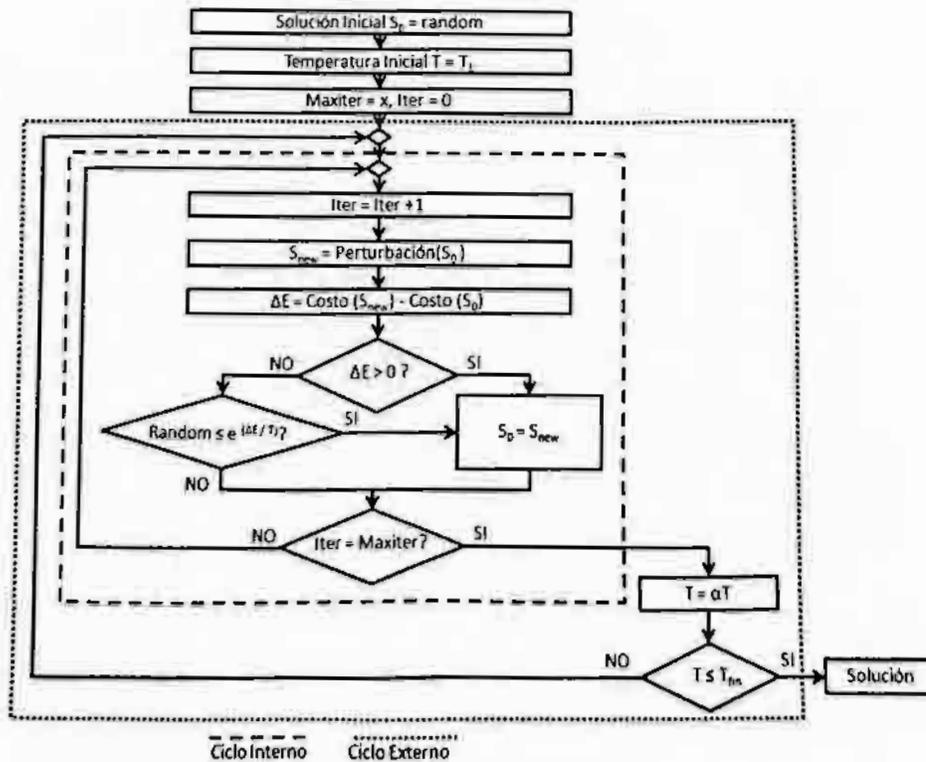


Figura 4. Esquema del algoritmo recocido simulado

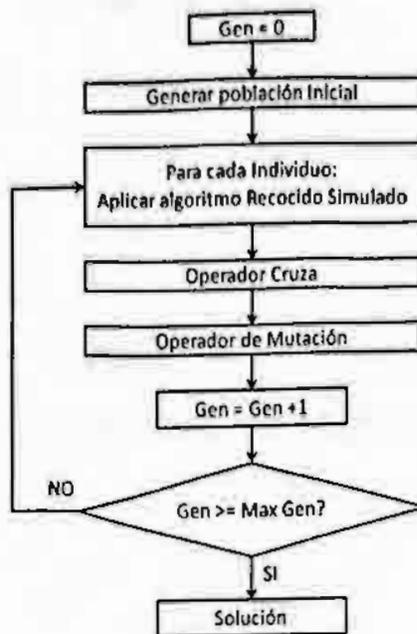


Figura 5. Esquema del algoritmo híbrido

	Problema	1	2	3	4	5	6	7	8
Genético	Promedio	6,896	7,065	6,336	6,808	4,139	4,436	3,214	3,422
	Desv est.	169	119	136	216	107	55	142	205
	Mejor	6,597	6,859	6,165	6,438	4,039	4,376	2,967	3,149
Recocido Simulado	Promedio	6,819	7,089	7,602	7,141	4,017	4,205	3,953	4,343
	Desv est.	161	68	174	180	50	106	206	306
	Mejor	6,586	6,996	7,474	6,847	3,970	4,033	3,713	4,079
Híbrido	Promedio	5,621	5,811	7,102	6,486	3,707	4,076	4,196	4,208
	Desv est.	99	121	293	243	82	160	65	128
	Mejor	5,512	5,649	6,797	6,091	3,597	3,876	4,114	4,026

Cuadro 2. Resultados de los algoritmos

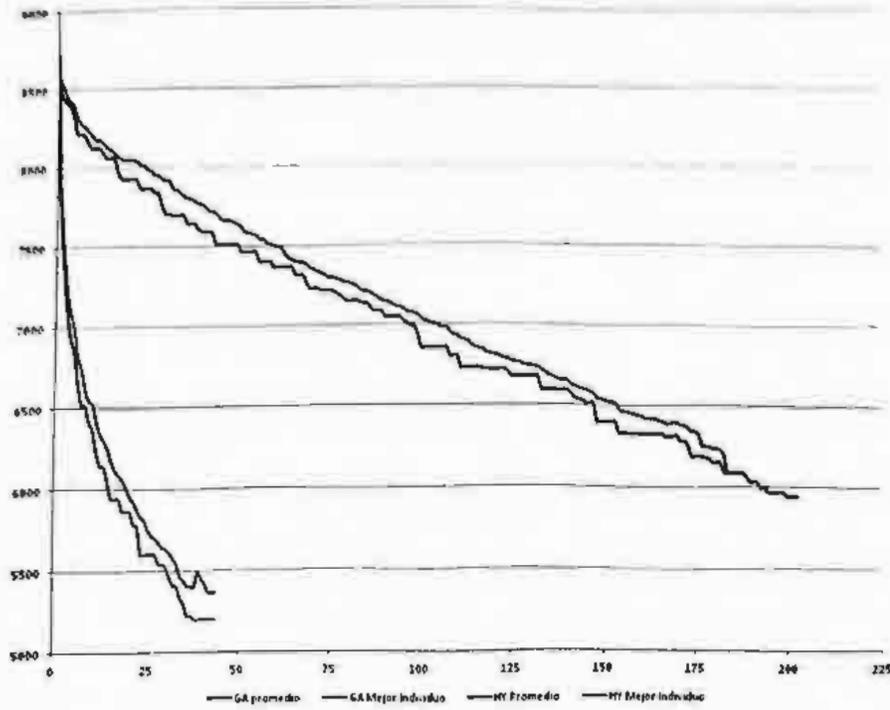


Figura 6. Avance de los experimentos respecto de la generación

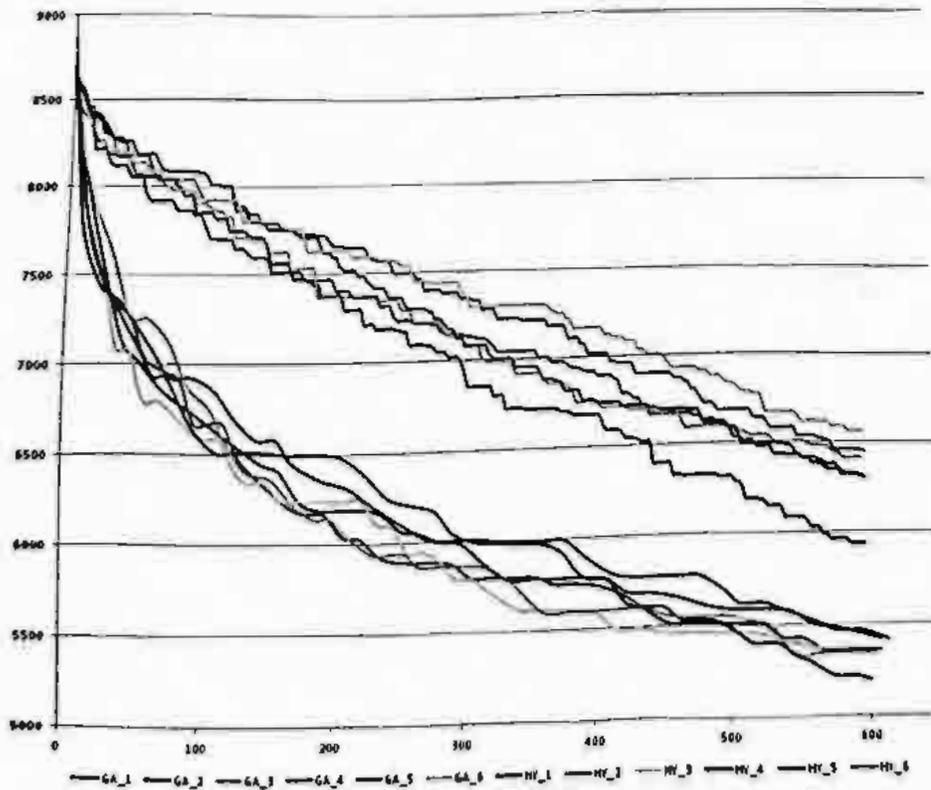


Figura 7. Avance de los experimentos respecto del tiempo

Esteganografía en Imágenes usando el Factor de Escala para las wavelets ortogonales

Blanca E. Carvajal-Gómez¹, Francisco J. Gallegos-Funes¹, José L. López-Bonilla¹,
Alberto J. Rosales-Silva¹

Instituto Politécnico Nacional

¹Escuela Superior de Ingeniería Mecánica y Eléctrica,

Av. IPN s/n, ESIME SEPI- Electronica, Lindavista, 07738, México D. F., Mexico
becarvajalg@gmail.com, {fgallegosf, jlopezb, arosales}@ipn.mx

Resumen. Cuando las imágenes en diferentes modelos de color son procesadas para la implementación de algoritmos esteganograficos, es importante el estudio de la calidad de la imagen anfitriona y de la imagen recuperada, ya que usualmente son usados filtros digitales, deformando visiblemente las imágenes. Al aplicar algoritmos esteganograficos se emplean medidas las cuales muestran que tan deformadas fueron la imagen anfitriona y la imagen recuperada. Proponemos un factor de escala en el dominio wavelet el cual depende del número de bits de la imagen para así poder ajustar la estegoimagen a la imagen original.

Palabras clave: modelos de color, esteganografía, factor de escala, dominio wavelet.

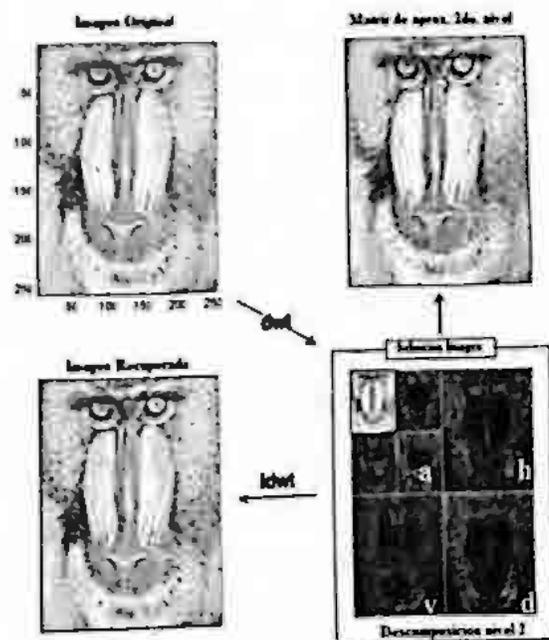
1 Introducción

El ocultamiento de la información es el proceso de agregar información o datos dentro de archivos digitales como: música, video y/o imágenes [1]. En la esteganografía, la información oculta no tiene relación con la imagen anfitriona. En la imagen anfitriona se debe de tener un mínimo de calidad, ya que cualquier relieve, color o algún pixel fuera de lugar de manera aparente puede causar sospecha, siendo susceptible a la extracción de la información oculta sin previa autorización. Una de las características más importantes es la recuperación total de la información oculta. La esteganografía en nuestros días es de vital importancia ya que es una herramienta de apoyo a la protección de los derechos de autor, que permita la autenticación de dichas obras, así como a la distribución y uso de manera legal de esta. La imagen anfitriona con la información oculta es conocida como estegoimagen. Típicamente, existen dos tipos para realizar una estegoimagen: en el dominio espacial y en el dominio de la frecuencia. Para el dominio espacial el método más común es conocido como: Bit Menos Significativo (LSB), el cual es la modificación del bit menos significativo en cada pixel de la imagen [2]. En el dominio de la frecuencia la Transformada Discreta

de Fourier (DFT), la Transformada del Coseno Discreto (DCT), y la Transformada Wavelet Discreta (DWT) son usadas para transformar los valores espaciales de los pixeles en coeficientes de frecuencia [3]. Varios factores pueden causar cierta sensibilidad al ojo humano como: la luminosidad, banda de frecuencia, textura, y/o proximidad a un bordes [4]. Aplicando el factor de escala la energía generada por la imagen anfitriona preserva a la estoimagen lo más próximo posible a la imagen original, eliminando cualquier alteración visual.

2 Método Propuesto

El algoritmo propuesto [5,6] para la implementación esteganografica está basado sobre la DWT. En la evaluación de algoritmos de compresión en imágenes digitales, el PSNR es el más usado frecuentemente para evaluar la calidad de estas imágenes [7,8]. Si aplicamos la DWT, la resolución de la imagen es dividida dentro de 4 submatrices llamadas: aproximaciones (a), horizontales (h), verticales (v) y diagonales (d), ver Figura 1. Cada sub matriz es una copia de la original pero a diferentes niveles de frecuencia el cual aporta ciertos niveles de energía [9,10]. Del diagrama de bloques de la figura 2, el primer paso se refiere a la doble descomposición wavelet aplicada a la imagen a ocultar y una simple descomposición wavelet para la imagen anfitriona. Obteniendo las 4 sub matrices (a), (h), (v) y (d), se elige a la sub matriz (h), para el ocultamiento de la imagen. La elección por consiguiente es debido a que los bordes son considerados como áreas naturales de ruido las cuales se puede ocultar la imagen. Posteriormente para la imagen a ocultar es elegida la sub matriz (a), ya que esta componente de la descomposición wavelet contribuye al 90% de la energía total de la imagen, siendo esta una copia fiel de la original a una menor escala. En la segunda etapa del algoritmo esteganografico, el bloque se refiere a la obtención de la desviación estándar σ_R , de esta manera se aplica criterio de inserción [5]. Finalmente, el último paso es aplicar la Transformada Wavelet Discreta Inversa (IDWT).



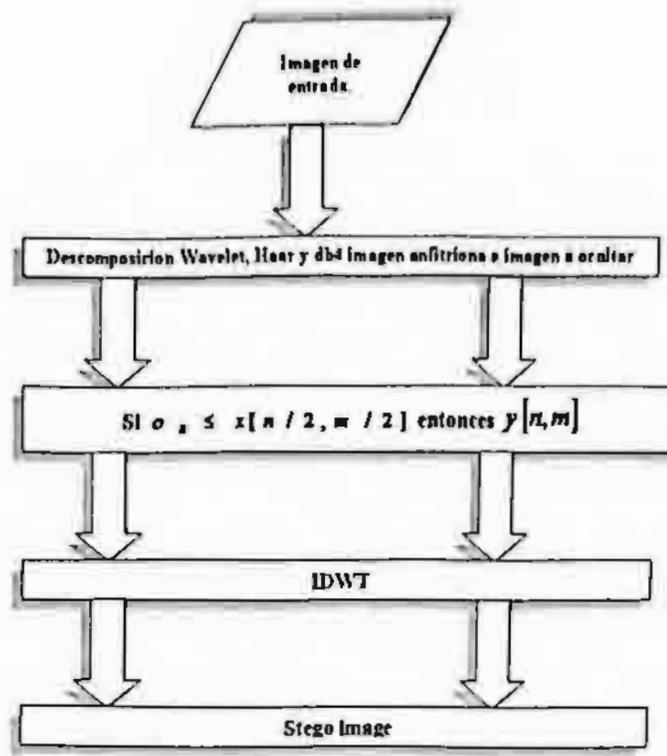


Fig 2. Diagrama a bloques del algoritmo esteganografico.

La DWT está íntimamente ligada con el análisis multiresolución, esto es, se puede observar una señal o una imagen en diferentes niveles de frecuencia [1], el cual permite conocer la imagen y facilitar el cálculo y manejo, cuando la familia de las wavelet es ortogonal [12-14]. Esta familia de wavelets ψ puede obtenerse moviendo j y dilatando i ec. (1):

$$\left\{ \psi_{j,i}(t) = \frac{1}{\sqrt{2^j}} \psi\left(\frac{t-2^j i}{2^j}\right) \right\}_{j,i \in \mathbb{Z}^2} \quad (1)$$

esta es una base ortonormal de $L^2(\mathbb{R})$. Estas wavelets ortogonales transportan información acerca de los cambios de la imagen para la resolución 2^{-j} . La función de aproximación f con una resolución 2^{-j} especifica la malla de muestreo discreta, el cual provee un promedio local de f en una vecindad proporcional al tamaño de f para 2^j . Las wavelets $\psi_{j,i}(t)$ son generados de la misma función madre wavelet con diferente escala j y lugar i , pero teniendo la misma forma. El factor de escala $j > 0$ es frecuentemente usado. La wavelet es dilatada cuando $j > 1$, y es contraída cuando $j < 1$. De esta manera, diferentes rangos de frecuencias son cubiertos al cambiar el valor de j . Para valores muy grandes del parámetro j corresponden para frecuencias de menor valor, o a escalas grandes de $\psi_{j,i}(t)$. Valores pequeños de j corresponden para frecuencias de menor rango o valores de pequeña escala de $\psi_{j,i}(t)$ [15,16]. Las funciones de la wavelet continua con factores discretos de escala y traslación son nombrados wavelets discretas. Finalmente, la señal $f(t)$ puede ser comprimida o expandida en el tiempo ec. (2):

$$f(t) \text{ compresion por un factor } 2^j(s) f_s(t) = \frac{1}{\sqrt{2^j}} f\left(\frac{t}{s}\right) \quad (2)$$

$$\hat{f}(w) \text{ compresion por un factor } \frac{1}{2^j} \hat{f}_{2^j}(w) = \frac{1}{\sqrt{2^j}} 2^j \hat{f}(2^j w) = \sqrt{2^j} \hat{f}(2^j w)$$

El coeficiente de la descomposición de la función en una base ortogonal de wavelets es calculada, aplicando una convolución discreta con h y con g , y realizando el muestreo de la ec. (3) y (4). Ver Figura 3.

$$x_{baja}[k] = \sum_n x[n] * h[2k - l] \quad (3)$$

$$x_{alta}[k] = \sum_n x[n] * g[2k - h] \quad (4)$$

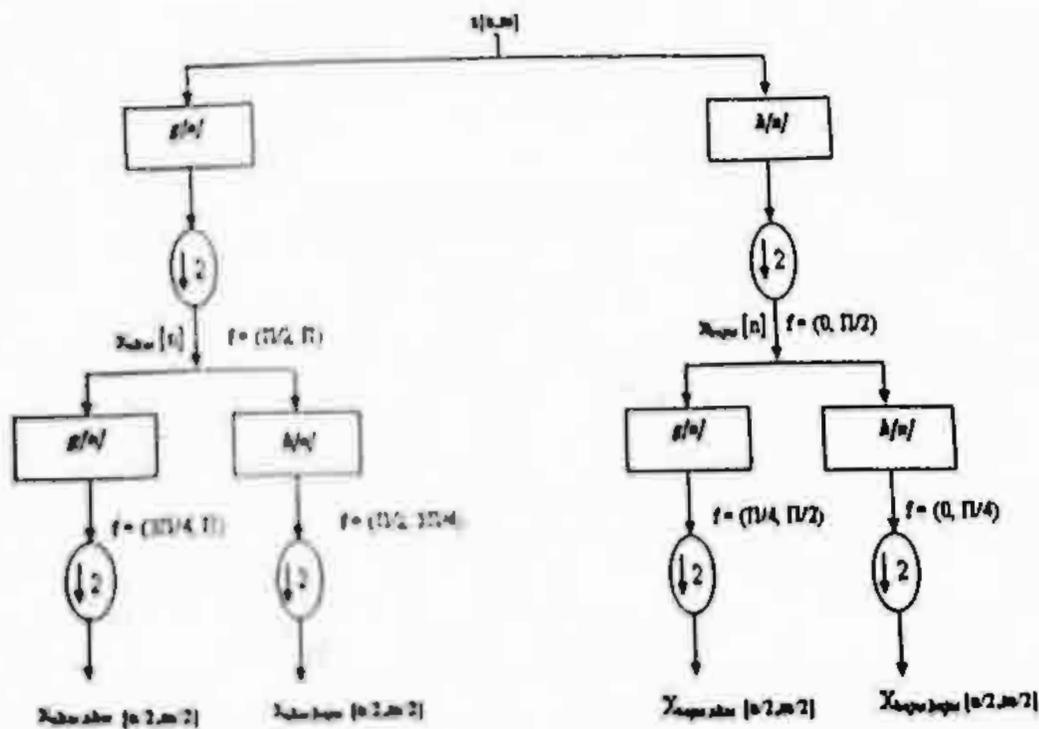


Fig 3. Esquema de la descomposición wavelet.

Entonces la imagen anfritona $x[n,m]$ pasa a través de una serie de filtros de espejo en cuadratura [10, 11, 13]. Para la reconstrucción, una interpolación es realizada, insertando ceros, y expandiendo. La figura 4 describe la descomposición de la wavelet discreta para cualquier modelo de color, esta es interpretada como la descomposición de la matriz R representada por $X_R[n,m]$, para la primera descomposición mostrada en la figura 4 se aplica a esta el primer paso del filtrado pasa bajas a través de los renglones y columnas para obtener la sub matriz (a) de la imagen, en la segunda descomposición es aplicada el filtrado pasa bajas a través de los renglones y en las columnas es aplicado el filtrado pasa altas obteniendo la sub matriz (h) , la tercera descomposición es similar para (h) pero invirtiendo el orden del

filtrado primero filtrado con el pasa altas y posteriormente con el filtro pasa bajas y así obtener la sub matriz (v) y finalmente para la sub matriz (d) el filtrado es aplicado en renglones y columnas con el filtro pasa altas. Ver figura 4.

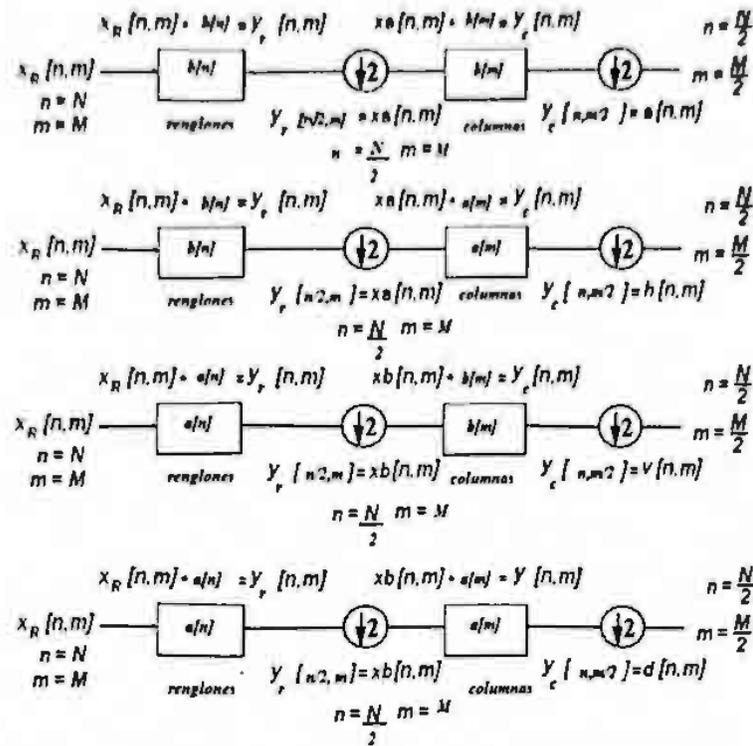


Fig 4. Banco de filtros para codificación en sub bandas donde se representa la descomposición wavelet discreta de una imagen.

La primera sub imagen conocida como detalles es: $a^1 = (a_1, a_2, \dots, a_{m/2})$ para la imagen $x[n,m]$ es obtenida haciendo el promedio de la señal como sigue: El primer valor a_1 es calculado tomando el primer conjunto de valores del vector $x[n,m]: (x_1 + x_2)/2$ y multiplicando este por $\sqrt{2}$, esto es, $a_1 = (x_1 + x_2)/\sqrt{2}$, similarmente $a_2 = (x_3 + x_4)/\sqrt{2}$, etc. Como se muestra a continuación,

$$a_{[n/2, m/2]} = \frac{x_{2m-1} + x_{2m}}{\sqrt{2}} \quad (5)$$

donde n, m es el número total de elementos del vector. La otra sub señal es también conocida como la primera fluctuación o vector de detalles de la imagen $x[n,m]$ y es denotada como: $d^1 = (d_1, d_2, \dots, d_{n/2, m/2})$. Es calculada tomando la diferencia entre el primer par de valores de $x, (x_1 - x_2)/2$ y entonces multiplicado y dividido por $\sqrt{2}$, ec. (7),

$$d_{n/2, m/2} = \left(\frac{x_{2m-1} - x_{2m}}{\sqrt{2}} \right) \quad (6)$$

Donde n, m es el tamaño del vector. [14,15,16]

3 Resultados Experimentales

Muchas de las medidas de distorsión o de calidad utilizadas en las imágenes durante el proceso de información visual pertenecen al grupo de medidas de diferencias de distorsión [6], están son basadas sobre la diferencia entre la imagen modificada y la imagen original. La más común medida de distorsión es la relación señal a ruido pico por sus siglas en ingles conocida como PSNR,

$$\text{PSNR} = 10 \cdot \log \left[\frac{(255)^2}{\text{MSE}} \right], \text{dB} \quad (7)$$

donde $\text{MSE} = \frac{1}{M_1 M_2} \sum_{i=1}^{M_1} \sum_{j=1}^{M_2} \|y(i, j) - x(i, j)\|_{L_2}^2$ es el error medio cuadrático,

donde M_1, M_2 son las dimensiones de la imagen, $y(i, j)$ es el vector 3D de los valores del pixel (i, j) de la estegoimagen, $x(i, j)$ es el correspondiente pixel en la imagen anfitriona original, $\|\cdot\|_{L_1}$, $\|\cdot\|_{L_2}$ L1- y L2- son los vectores normales, respectivamente. La desviación del color normalizado (NCD) es usado para la cuantificación del error del color perceptual,

$$\text{NCD} = \frac{\sum_{i=1}^{M_1} \sum_{j=1}^{M_2} \|\Delta E_{Luv}(i, j)\|_{L_2}}{\sum_{i=1}^{M_1} \sum_{j=1}^{M_2} \|E_{Luv}^*(i, j)\|_{L_2}} \quad (8)$$

Aqui, $\|\Delta E_{Luv}(i, j)\|_{L_2} = \left[(\Delta L^*)^2 + (\Delta u^*)^2 + (\Delta v^*)^2 \right]^{1/2}$ es la norma del error del color; ΔL^* , Δu^* , y Δv^* son las diferencias en los componentes L^* , u^* , y v^* , respectivamente, entre los dos vectores de color que representa a la estegoimagen y la imagen host para cada pixel (i, j) de una imagen, y $\|E_{Luv}^*(i, j)\|_{L_2} = \left[(L^*)^2 + (u^*)^2 + (v^*)^2 \right]^{1/2}$ es la norma o magnitud del pixel de la imagen anfitriona en el espacio $L^* u^* v^*$; El índice de calidad (Q) provee la calidad de la estegoimagen,

$$Q = \frac{4\sigma_{xy}}{(\sigma_x^2 + \sigma_y^2)(x^2 + y^2)} \quad (9)$$

Donde x y y son los valores medios de la imagen anfitriona y de la estegoimagen, respectivamente, σ_x^2 y σ_y^2 son las varianzas de la imagen anfitriona y de la estegoimagen, respectivamente, y $\sigma_{xy} = \frac{1}{N-1} \sum_i (x_i - x)(y_i - y)$. Los resultados

experimentales aplicados a los tres modelos de color son mostrados, en cada resultado se va modificando el factor de escala. La Tabla 1, 2 y 3 muestran los resultados en términos de PSNR, MAE (Error Medio Cuadrático), COI (Correlación), Q, NCD, HC (Capacidad de inserción) y RMS (Valor cuadrático medio) [16,17]. En el caso de diferentes valores de n para el factor de escala usamos imágenes con una dimensión de 320×320 , "Mandrill" como la imagen anfitriona y "Lena" como la imagen a ocultar, de las Tablas 1, 2 y 3, podemos ver que a medida que se incrementa el valor de j los resultados obtenidos van mejorando también. La Figura 5 muestra los resultados obtenidos durante el proceso para la estegoimagen Mandril (Fig. 5a-b-c) y la imagen recuperada Lena (Fig. 5d-e-f) de acuerdo con la Tabla 1. Observamos de esta figura que los mejores resultados son obtenidos cuando $j=9$, donde j representa el número de bits de resolución de la imagen a ocultar. De Fig. 5d, 5e, y 5f podemos ver que cuando el valor propuesto del factor de escala incrementa, entonces la calidad subjetiva de la imagen también incrementa.

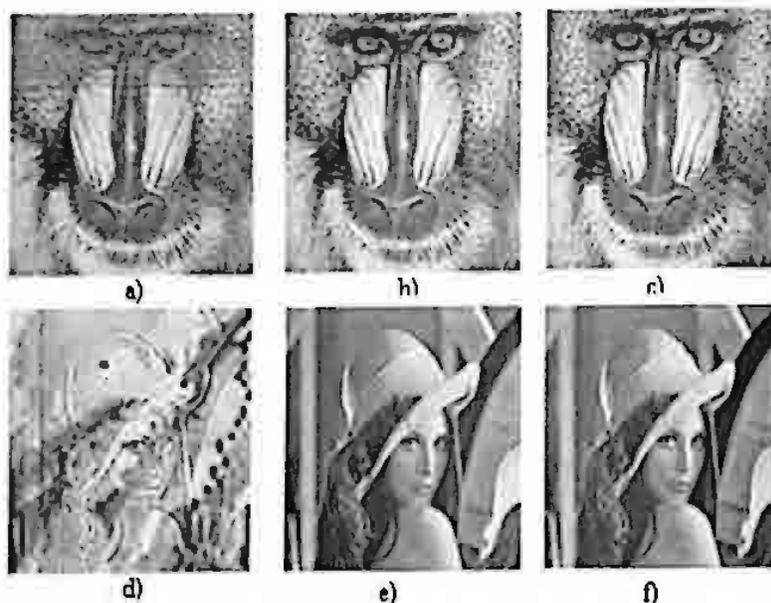


Fig 5. Resultados visuales para diferentes valores del factor de escala j , columna a) y d) con $j=2$, columna b) y e) con $j=5$, y columna c) y f) $j=9$.

Tabla 1. Resultados obtenidos para diferentes valores de j para el modelo de color RGB.

J		PSNR db	COI %	NCD	Q	MAE	HC	RMS
$j=0$	Host image	31.5084	0.7852	0.0020	0.7836	10.4878	73.06Kb	0.4760
	Hide image	16.5327	0.3889	0.4748	0.3586	4.9403		
$j=2$	Host image	31.4999	0.8046	0.0020	0.8040	9.9490	66.31Kb	0.4748
	Hide image	16.9537	0.3889	0.4749	0.3587	4.9471		
$j=5$	Host image	36.1233	0.9781	0.0020	0.9792	3.2086	61.97Kb	0.4614
	Hide image	27.2474	0.9985	0.0020	0.9962	2.7714		
$j=9$	Host image	36.1233	0.9908	8.015e-4	0.9913	2.0309	3.835Kb	0.4610
	Hide image	31.0781	0.9980	0.0020	0.9962	2.7714		

Tabla 2. Resultados obtenidos para diferentes valores de j para el modelo de color HSV.

J		PSNR db	COI %	NCD	Q	MAE	JIC	RMS
$j=0$	Host image	6.2955	0.5217	0.0020	0.0863	113.9768	0.359Kb	0.0020
	Hide image	2.8061	0.0592	0.0020	3.794e-4	177.9792		
$j=2$	Host image	20.1709	0.9015	0.0020	0.9067	5.6167	0.257Kb	0.0017
	Hide image	36.1233	0.9955	0.0020	0.9962	2.7714		
$j=5$	Host image	23.5030	0.9441	0.0019	0.9469	3.4113	0.568Kb	0.0013
	Hide image	36.1233	0.9955	0.0020	0.9962	2.7714		
$j=9$	Host image	38.8068	0.9985	5.970e-4	0.9985	0.8092	0.071Kb	0.0012
	Hide image	36.1233	0.9955	0.0020	0.9962	2.7714		

Tabla 3. Resultados obtenidos para diferentes valores de j para el modelo de color YCbCr.

J		PSNR db	COI %	NCD	Q	MAE	HC	RMS
$j=0$	Host image	19.5146	0.8713	0.0020	0.8751	7.1762	26.34Kb	0.3961
	Hide image	2.7975	0.2008	0.0020	0.0601	178.1406		
$j=2$	Host image	19.7324	0.8782	0.0020	0.8816	7.0325	24.66Kb	0.4694
	Hide image	36.0827	0.9954	0.0020	0.9962	2.7948		
$j=5$	Host image	22.1046	0.9318	0.0015	0.9323	5.5328	12.30Kb	0.4202
	Hide image	36.0827	0.9954	0.0020	0.9962	2.7948		
$j=9$	Host image	28.5553	0.9836	0.0015	0.9843	2.7372	3.33Kb	0.4132
	Hide image	36.0827	0.9954	0.0020	0.9962	2.7948		

La Tabla 4, muestra los resultados obtenidos en el caso de $j=10$ en el factor de escala. Figura 6 presenta los resultados visuales de acuerdo a la Tabla 4. Puede verse en la imagen de error de Lena Figura 6d, la diferencia de valores entre la imagen anfitriona y la imagen recuperada es aproximadamente cero proveyendo que la información oculta sea lo menos imperceptible.

Tabla 4. Resultados obtenidos en el caso de $j=10$ con el factor de escala con el modelo de color RGB.

Host image "Mandril"	Hide image "Lena"
Q=0.9934	Q=0.9962
PSNR=36.1233 dB	PSNR=32.5167 dB
COI=99.34%	COI=99.55%
NCD=6.0486 e-4	NCD=0.0020
MAE=1.7022	MAE=2.7714

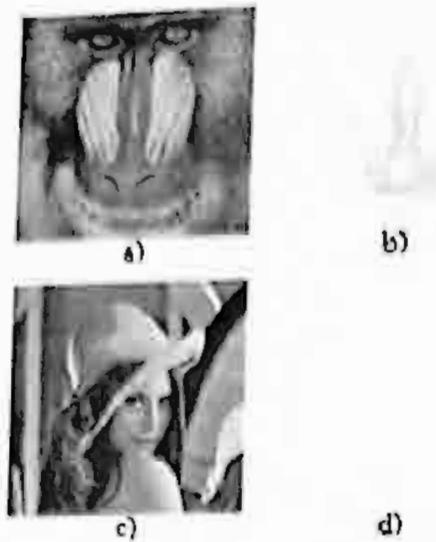


Fig 6. Resultados visuales para $j=10$. a) estego-imagen "Mandrill", b) estego-imagen de error "Mandrill". c) imagen recuperada "Lena". d) imagen de error hide "Lena".

4. Conclusiones

El modelo de color RGB, HSV y YCbCr son alteradas en la contribución de energía en cada sub matriz obtenida en la descomposición wavelet cuando el algoritmo esteganografico se aplica. Se sabe que es un factor clave el valor de $1/\sqrt{2}$ para el ajuste de energía, este valor de ajuste puede ser aplicado a cualquier tipo de modelo de color. Sin embargo, el factor de escalamiento para la conservación de energía arriba mencionado. Aplicando el factor de escala propuesto con una resolución j , se obtiene el valor de $1/\sqrt{2^j}$, este factor de ajuste para la energía en cada sub matriz puede realizar una aproximación de la estegoimagen deformada por la inserción de la información con respecto de la imagen original anfitriona. Se pudo observar en los resultados obtenidos que en cada incremento de j se presentaba una mejor aproximación a la imagen original.

Agradecimientos

Este trabajo es patrocinado por el Instituto Politécnico Nacional de México y CONACyT.

Referencias

1. Chia-Chen, L., Wei-Liang, T., Chin-Chen C.: Multilevel reversible data hiding based on histogram modification of difference. *Pattern Recognition*. 3582-3591 (2008)
2. Chung-Ming, W., Nan-I W., Chwei-Shyong, T., Min-Shiang H.: A high quality steganographic method with pixel-value differencing and modulus function. *Journal of System and Software*. 150-158 (2008)
3. Yuan-Hui, Y., Chin-Chen, C., Yu-Chen, H.: Hiding secret data in images via predictive coding. *Pattern Recognition*. 691-705 (2005)

4. Reddy, A.A., Chatterji, B.N.: A new wavelet based logo-watermarking scheme. *Pattern Recognition Lett.* 1019-1027 (2005)
5. Carvajal-Gómez, B. E., Acevedo M., López-Bonilla, J. L.: Energy conservation in the concealment of information in a video by means of discrete wavelet transform. *JeJEMTA* 4, 27-31 (2008)
6. Kutter, M., Petitcolas, F.A.P.: A fair benchmark for image watermarking systems. *Proc. SPIE Electronic Imaging '99. Security and Watermarking of Multimedia Contents*. 1-14 (1999)
7. Huang, K.Q., Wub, Z.Y., Fung, G.S.K., Chan, H.Y.: Color image denoising with wavelet thresholding based on human visual system model. *Signal Processing: Image communication*. 115-127 (2005)
8. Ginesu, G., Massidda, F., Giusto, D.D.: A multi-factors approach for image quality assessment based on a human visual system model. *Signal Processing: Image communication*. 316-333 (2006)
9. Moon, H.S., You, T., Sohn, M.H., Kim, H.S., Jang, D.S.: Expert system for low frequency adaptive image watermarking: Using psychological experiments on human image perception. *Expert Systems with Applications*. 674-686 (2007)
10. Walker, J.: *A primer on wavelets and their scientific applications*, Chapman & Hall/CRC, London (2003)
11. Vetterli, M., Kočević, J.: *Wavelets and Subband Coding*. Prentice-Hall, New Jersey (1995)
12. Carvajal-Gómez, B.E.: "Técnica de Inserción de Información en Video aprovechando el mismo Ancho de Banda". Master Dissertation Thesis. ESIME-Zacatenco, National Polytechnic Institute (2008)
13. Petrosian, A.A., Meyers, F.G.: *Wavelets in Signal and Image Analysis*. Kluwer Academic Pub., Berlin, vol. 19 (2002)
14. Debnath, L.: *Wavelets and Signal Processing*. Birkhauser, Berlin, (2002)
15. Y. Sheng, "The Transforms and Applications Handbook". Ed. 2da, CRC Press, (2000), ISBN: 9780849385957 .
16. Bogges Albert, Narcovich Francis, "A first course in wavelets with Fourier Analysis". Second edition, Ed. Wiley, ISBN:978-0-470-43117-7.
17. Sos S. Aghaian, David Akopian and Sunil A. D'Souza "Frequency Domain Based Secure Digital Audio Steganography Algorithms," IEEE SP/CAS, 2005 International Workshop on Spectral Methods and Multirate Signal Processing.

Estimación Inicial de Habilidad en Evaluadores Adaptables

Félix Castro¹, Joel Suárez¹ and Raúl Chirinos²

¹ Centro de Investigación en Tecnologías de Información y Sistemas, CITIS, Universidad Autónoma del Estado de Hidalgo, Ciudad Universitaria, Carretera Pachuca-Tulancingo km. 4.5, Hidalgo, México. fcastro@lsi.upc.edu, jsuarez@uaeh.edu.mx

² Universidad Tecnológica de Xicotepec de Juárez, Xicotepec de Juárez, Puebla, México rchirinos7@hotmail.com

Resumen. El presente artículo describe la utilización de la metodología FIR en un experimento realizado con estudiantes de nivel universitario con el objetivo de predecir la habilidad inicial de los estudiantes. En los experimentos se midió la importancia de las variables involucradas en el proceso de estimación de la habilidad. Asimismo, se utiliza una extensión de FIR, denominada LR-FIR, para identificar patrones de comportamiento de los estudiantes, y expresarlos en forma de reglas del tipo IF-THEN. Los resultados obtenidos al aplicar la metodología FIR y el algoritmo LR-FIR aportan valiosa información sobre las variables que tienen mayor importancia en la habilidad del estudiante, así como, la capacidad de tales herramientas en el proceso de estimación inicial de las habilidades de los participantes en un sistema de evaluación adaptable computarizada y su caracterización.

Palabras Clave: Sistema Evaluador Adaptable. Estimación de habilidad. Metodología FIR. LR-FIR.

1 Introducción

La evaluación es un proceso vital en la formación técnica y profesional, por lo que se presentan múltiples desafíos en este campo [1]. Existen modelos de respuesta a la pregunta que garantizan que el proceso de evaluación converja hacia un valor que se supone representa el nivel real de habilidad que un evaluado tiene, sin embargo, esto no siempre es así.

La evaluación adaptable computarizada (EAC) es actualmente una alternativa a otros métodos de evaluación convencionales, y las aplicaciones potenciales de la misma han estado bajo profunda revisión por diferentes grupos de investigación. Ésta permite llevar a cabo evaluaciones enfocadas al individuo (evaluado), en donde la complejidad de las preguntas se adapta a la capacidad de respuesta que tiene el mismo, de tal forma que éstas incrementan en dificultad si la respuesta es correcta, o disminuyen en caso contrario, hasta encontrar su habilidad real [2], con fin de obtener resultados que demuestren de forma más cercana el nivel real del evaluado que la

realiza, tomando en cuenta además que una meta de este tipo de evaluación es emplear el menor número de preguntas para poder estimar el nivel real de un evaluado [3],[4].

Por tanto, como se ilustra en la figura 1, un sistema evaluador adaptable estima de manera recurrente el nivel de habilidad que un evaluado tiene, hasta llegar a la habilidad real, o hasta que se cumpla alguno de los siguientes criterios de parada:

1. Cuando no hay cambio o variación en la habilidad [5].
2. Cuando se establece una longitud fija de reactivos y éstos se terminan [6].
3. Cuando se establece un tiempo límite y éste se agota [7],[8].

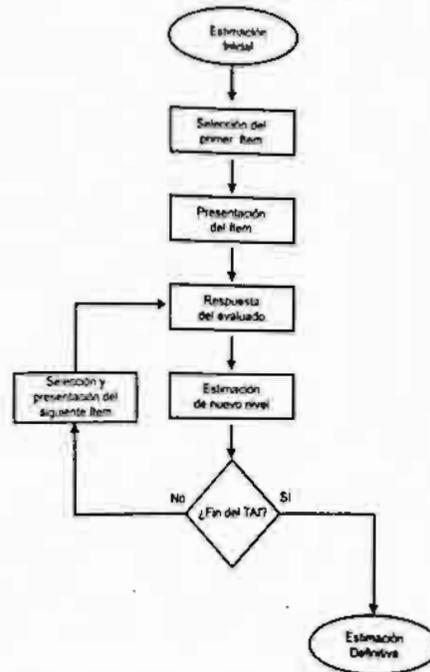


Fig. 1. Diagrama de aplicación de un Evaluador Adaptable Computarizado.

En la sección 2 se presenta un estudio sobre la determinación inicial de habilidad en entornos de evaluación adaptable. Una explicación sobre la metodología FIR se incluye en la sección 3. En la sección 4 se describen los experimentos realizados y en la sección 5 se presentan las conclusiones parciales del presente artículo.

2 Estimación Inicial de Habilidad

La EAC utiliza valores estimados previos, lo que asume la existencia de una semilla o valor estimado de la habilidad buscada [9], en consecuencia es elemental establecer una habilidad inicial que se acerque lo más posible a la habilidad real del sujeto, cumpliendo, de esta forma, con parámetros de justicia, ya que si a un evaluado se le atribuye una estimación inicial de su habilidad muy por debajo o muy por arriba del valor que realmente tiene, seguramente requerirá de un número mayor de reactivos o bien empleará más tiempo para llegar a su nivel real de habilidad, comparado con el tiempo que emplearía si la estimación inicial hubiese sido la correcta [6], ocasionando desmotivación y aburrimiento o por el contrario desconcierto y frustración en el evaluado [9], comportamiento que se observa en la figura 2, donde la habilidad real

del evaluado es de 8.2, en la estimación inicial se le asigna una habilidad de 3.8 (con una diferencia, por tanto de 4.4), por lo cual el evaluado tendrá que contestar preguntas inicialmente muy sencillas para su habilidad, hasta llegar a su habilidad real.

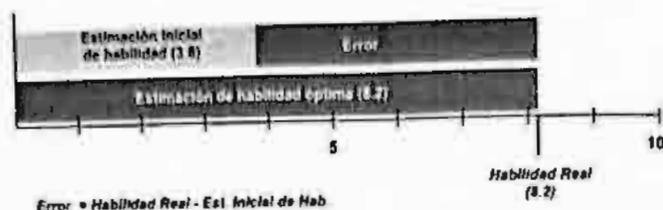


Fig. 2 Importancia de asignar una estimación inicial de habilidad muy próxima a la habilidad real de un evaluado.

Sin embargo, es conveniente considerar que dicha estimación inicial de la habilidad puede depender de múltiples y variadas circunstancias inherentes al evaluado, tales como: trayectoria académica, compromiso, entorno social y familiar, entre otros. Estas circunstancias tienen una incidencia directa en la habilidad del evaluado, por lo cual, mientras el error en la estimación inicial sea menor en relación a la habilidad real, la longitud del proceso de evaluación será en consecuencia menor. En la figura 3 se muestra la relación de variables que existe en una evaluación y que fundamenta lo citado.

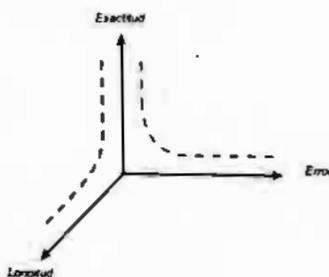


Fig. 3 Relación de variables en una evaluación.

3 Metodología FIR

La alternativa de solución adoptada en la presente investigación es determinar la habilidad inicial de un evaluado, mediante la identificación de un modelo obtenido con la metodología FIR, con fin de determinar las variables, de índole personal, académico y profesional, que tienen mayor influencia sobre la habilidad real de un evaluado, obteniendo además un conjunto de reglas de tipo IF-THEN, mediante el algoritmo LR-FIR, que permiten observar el comportamiento de los evaluados. LR-FIR es una extensión de la metodología FIR.

FIR (Razonamiento Inductivo Difuso, por sus siglas en inglés), es una metodología de modelado y simulación cualitativa que se basa en el análisis del comportamiento de un sistema determinado [10]. Esta metodología ha sido utilizada con éxito en diversos sistemas, logrando predecir con mucha certeza su comportamiento futuro a partir de observaciones pasadas. Para lograr esto, como se ilustra en la figura 4, es

necesario realizar una serie de pasos: Fusificación, Modelado cualitativo, Simulación cualitativa y Defusificación [11].

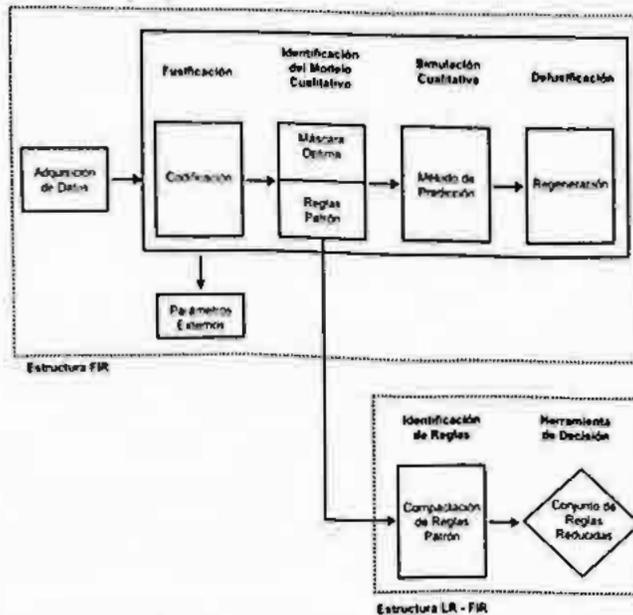


Fig. 4 Fases de la Metodología FIR y Algoritmo LR-FIR.

4 Experimentos Realizados

Con objeto de comprobar la alternativa propuesta, se aplicó la metodología FIR, con datos obtenidos de dos evaluaciones aplicadas a alumnos de dos instituciones de nivel superior: Universidad Tecnológica de Xicotepec de Juárez, UTXJ (157 alumnos de Primer Cuatrimestre de Nivel TSU en Tecnologías de la Información y Comunicación) y Universidad Autónoma del Estado de Hidalgo, UAEH (105 alumnos del primer Semestre de Nivel Licenciatura en Sistemas Computacionales), ambas evaluaciones estuvieron conformadas en dos partes:

1. Aplicación de 30 reactivos del tópico de Lógica Proposicional, con respuestas de tipo: Falso (F) o Verdadero (V).
2. Solicitud de información de cada uno de los evaluados (Personal, académica, familiar).

Se obtuvieron en total 18 características como variables de entrada, y una variable de salida (*Promedio*), dichas variables, así como, una breve descripción, se muestran en la tabla 1.

El experimento inició introduciendo a la aplicación VisualFIR [12] las 19 variables en formato de matlab. Para fines de la investigación, se dividieron los resultados y características de los evaluados en dos grupos: uno con 120 y otro con los 37 restantes (de los datos obtenidos en la UTXJ). El primer grupo fue utilizado para entrenar el sistema y generar el modelo que describiera el comportamiento del mismo, mientras que el segundo grupo de datos fue empleado para validar el modelo obtenido. Con la aplicación de la metodología, se obtuvo el modelo que describe su comportamiento, el cual es definido por la máscara óptima:

[0, 0, 0, 0, 0, 0, 0, -1, 0, 0, 0, 0, 0, 0, 0, 0, -2, -3, 1]

Tabla 1. Características de tipo personal académica y personal que fueron solicitadas a los 157 evaluados previo a la evaluación aplicada.

VARIABLE	Descripción
ACTITUD	Actitud del evaluado hacia el área a evaluar
ASP_FUTURAS	Aspiraciones Futuras de los evaluados
COMPROMISO	Nivel de compromiso en sus estudios
COMPUTADORA	Nivel de manejo de la computadora
EDAD	Edad del evaluado
EST_MADRE	Grado de estudios de la madre
EST_PADRE	Grado de estudios del padre
INGLES	Nivel aproximado de Inglés
INT_PADRES	Interés de los padres
INTERÉS	Interés de los evaluados en el tópico
LUGAR_HNOS	Lugar entre sus hermanos
NO_HERMANOS	No. de hermanos que tiene el evaluado
OCUP_MADRE	Ocupación de la madre
OCUP_PADRE	Ocupación del padre
PROCEDENCIA	Lugar de procedencia de los evaluados
SEXO	Género del evaluado
SIT_SOC	Situación socioeconómica familiar
TIEMPO_LECT	Tiempo de lectura sobre el tópico (semanal)
PROMEDIO	Promedio obtenido de la evaluación

Los valores negativos se refieren a aquellas variables de entrada que tienen relación funcional con la variable de salida, los valores positivos se refieren a las variables de salida, mientras que los valores en cero, representan que no hay relación entre las variables entrada-salida. Concluyendo entonces que las variables de entrada que tienen relación con la variable Promedio son: *Inglés*, *Sit_Soc* y *Tiempo_Lect* (Nivel de Inglés, Situación Socioeconómica familiar y Tiempo de Lectura dedicado a la semana del tópico en cuestión).

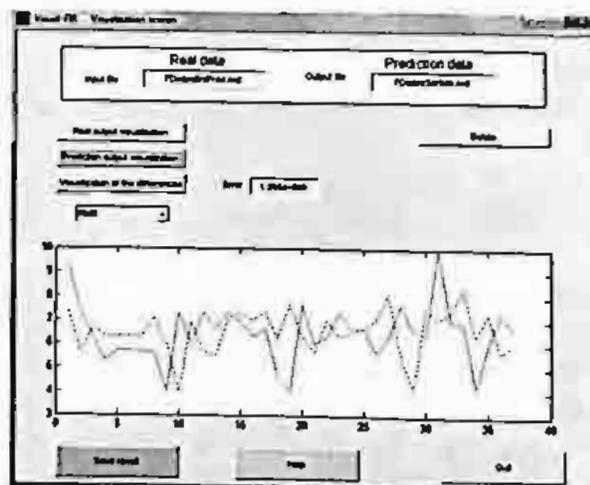


Fig. 5 Gráfica que muestra la comparación entre la habilidad real de los evaluados, la predicción hecha por FIR y el error absoluto.

Una vez modelado el comportamiento del sistema, se procedió a la etapa de estimación de habilidad de los estudiantes (no incluidos en el conjunto de entrenamiento) mediante el motor de predicción de FIR. Con la finalidad de evaluar la calidad de las estimaciones hechas por FIR, éstos se contrastan con la habilidad real de los estudiantes.

Los resultados obtenidos con este experimento se ilustra en la figura 5, donde se observa que el error RMS=1.555, los datos reales se ilustran con una línea continua, mientras que los datos predichos se muestran con una línea punteada. Esta gráfica resalta el poder predictivo de FIR ya que muestra que los datos predichos siguen el comportamiento de los datos reales de los evaluados.

Una vez obtenido el modelo con los datos de la UTXJ, se procedió a validar el mismo, para lo cual se consideraron los datos obtenidos en la evaluación realizada con alumnos de la UAEH como Test. Es importante mencionar que debido a que FIR no predice datos cuyo comportamiento no haya sido observado durante la fase de entrenamiento, se consideraron únicamente para el Test 96 de los 105.

El resultado arrojado con este experimento permite observar que el error aumenta (RMS=2.3446) lo que puede observarse en la Figura 6, donde la línea continua representa el comportamiento con datos reales, mientras que la línea punteada, el comportamiento con los datos obtenidos de la predicción.

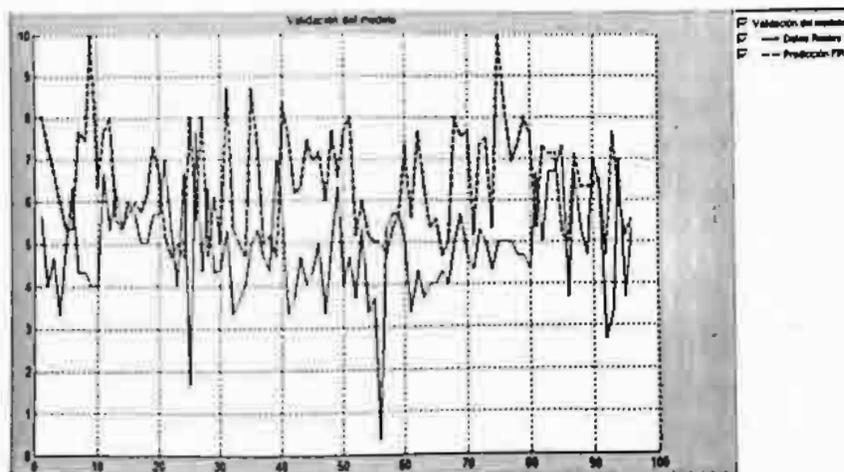


Fig. 6 Predicción obtenida para validar el modelo de los datos de la UTXJ, empleando los datos de la UAEH para Test.

Los motivos que pudieron haber ocasionado esta diferencia se mencionan a continuación:

- Los modelos educativos de las instituciones son distintos:
 - UTXJ: La institución maneja periodos cuatrimestrales, Nivel T.S.U. (Técnico Superior Universitario);
 - UAEH: Institución con periodo semestral, Nivel de Licenciatura;
- El entorno social en que se encuentran los estudiantes es diferente, ya que mientras que los de la UTXJ se encuentran en un entorno rural, los evaluados de la UAEH se encuentran en un entorno urbano.

Se realizaron experimentos con la finalidad de comparar los resultados de predicción obtenidos con la metodología FIR con otros métodos como MLP y ZeroR.

Las gráficas obtenidas se muestran en la figura 7, de la que es posible observar que el poder de predicción de FIR es más que evidente ya que la habilidad real que tuvieron los evaluados sigue el patrón generado por dicha metodología.

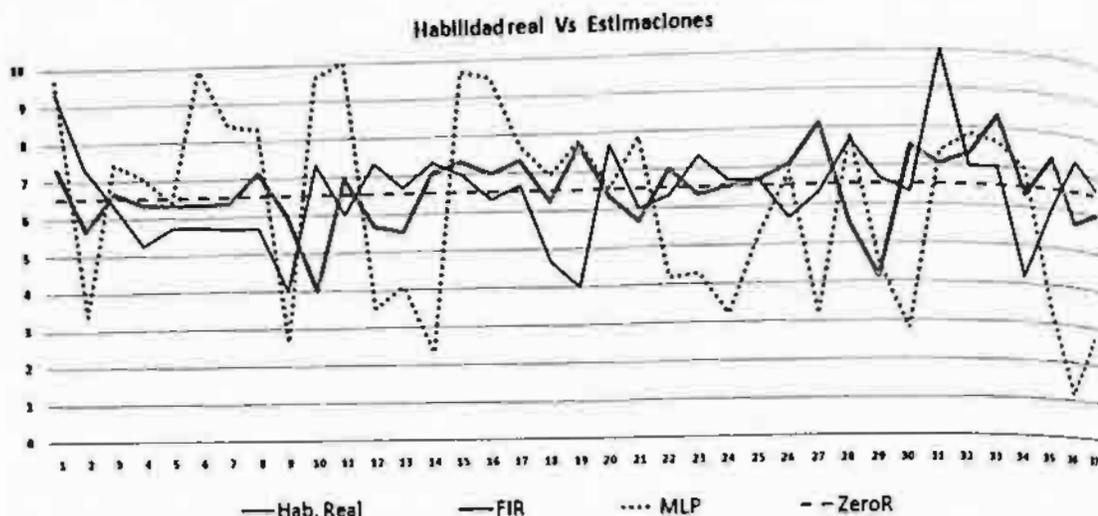


Fig. 7 Gráfica que muestra el comportamiento con diferentes métodos, incluyendo FIR.

Además de predecir el promedio de los evaluados, en este artículo se realizaron experimentos para describir el comportamiento de los evaluados. Para esto, se utilizó el algoritmo LR-FIR, que describe el comportamiento del sistema en análisis mediante reglas del tipo IF-THEN. Las reglas lingüísticas que describen el comportamiento de los evaluados se muestran en la tabla 2. ESP representa la métrica de especificidad y SENS la sensibilidad, ambas obtenidas a partir de la matriz de confusión generada de los experimentos realizados.

Tabla 2. Reglas que describen el comportamiento de aprendizaje de los evaluados que participaron en los experimentos.

REGLAS	ESP.	SENS.
IF Ingles=30%-60% AND Sit_Soc=Media-Baja AND Tiempo_Lect<=1Hr. THEN Promedio= 0 -7	0.87	0.59
IF Ingles=10%-20% AND Sit_Soc=Baja AND Tiempo_Lect<=30min. THEN Promedio=0-7	1	0.18
JOINT QUALITY OF THE ACTUAL CLASS	0.78	0.77
IF Ingles>=30% AND Sit_Soc=Media-Baja AND Tiempo_Lect>=1Hr. THEN Promedio=7-8.5	0.82	0.76
IF Ingles 50%-60% AND Sit_Soc=Baja AND Tiempo_Lect>=2Hrs. THEN Promedio = 7 - 8.5	1	0.24
JOINT QUALITY OF THE ACTUAL CLASS	0.82	1
OTHERWISE Promedio = 8.5 - 10		

5 Conclusiones

Los experimentos realizados utilizando la metodología FIR demuestran que es posible la estimación inicial de los estudiantes mediante la identificación de un modelo que minimiza la complejidad, ya que realiza una reducción de la dimensionalidad al sólo incluir 3 (*Nivel de Inglés, Situación Socioeconómica familiar y Tiempo de Lectura semanal*) de las 18 variables, sin perder poder de predicción.

La predicción hecha con la metodología FIR, utilizando sólo las características mencionadas, permite estimar la habilidad inicial de los evaluados con un porcentaje de error aceptable. Los resultados de predicción obtenidos con FIR se compararon con otros métodos de predicción, obteniendo resultados favorables. Las 3 variables seleccionadas representan características de índole personal, si se utilizaran variables que describan datos académicos históricos o aquellos que reflejen el conocimiento o habilidad de los evaluados, el error de predicción sería menor, sin embargo, esto implica realizar evaluaciones diagnósticas o acciones similares, dificultando el proceso de estimación de habilidad en entornos de evaluación adaptable computarizada, por esta razón, los resultados obtenidos con el enfoque propuesto en este proyecto de investigación se consideran importantes y un valioso primer intento por optimizar el proceso de evaluación en ambientes de evaluación adaptable.

Mediante la utilización del algoritmo LR-FIR se pudo caracterizar el comportamiento de los evaluados por medio de la extracción de reglas lingüísticas que describen los patrones de comportamiento de aprendizaje. Las reglas obtenidas son muy intuitivas, además de concordar con la percepción de los profesores del área de conocimiento evaluada (lógica proposicional).

Agradecimientos. Los autores desean agradecer las facilidades y soporte proporcionados para la realización de este proyecto por parte del Consejo de Ciencia y Tecnología del Estado de Hidalgo (Proyecto FOMIX 2008/97071).

Referencias

- [1] McDonald, A., Boud, D., Francis, J., Gonzi, A.: Nuevas perspectivas sobre la evaluación. Sección para la Educación Técnica y Profesional. UNESCO, París.
- [2] Olea, J., Ponsoda, V.: Tests Adaptativos Informatizados. Universidad Autónoma de Madrid, España.
- [3] Abdullah, S.H., Student Modelling by Adaptive Testing - A knowledge-based approach. A thesis submitted to the University of Kent at Canterbury, in the subject of Computer Science, for the degree of Doctor of Philosophy, June, (2003).
- [4] Aguado, D., Santa-Cruz, C., Dorronsoro, J., Rubio-Franco, V.: Algoritmo mixto mínima entropía-máxima información para la selección de ítems en un test adaptativo informatizado. Universidad Autónoma de Madrid. Revista Psicothema ISSN 0214 - 9915. Vol. 12, Supl. Num. 2, 12-14.
- [5] Rojas, M., Manríquez, G., Gatica, Y., Salcedo-Lagos, P.: Curso de UML multiplataforma adaptativo basado en la teoría de respuesta al ítem. Revista Ingeniería Informática, Edición Número 10, (2004).
- [6] Conejo, R., Millán, E., Pérez, J.L., Trella, M.: Modelado del alumno: un enfoque bayesiano. Inteligencia Artificial, Revista Iberoamericana de Inteligencia Artificial, No. 12, 50-58, (2001).
- [7] Ponsoda-Gil, V., Hontangas, P., Olea, J., Revuelta, J., Abad, F.J., Ximénez, C.: Investigación actual en tests adaptativos informatizados. Universidad de Valencia, Universidad Autónoma de Madrid.
- [8] Fernández-Álvarez, M.: Las nuevas tecnologías en el área de testing: Tests informatizados y Tests adaptativos informatizados (TAIS). International Council for Educational Media -

ICEM. Nuevas Tecnologías y Nuevos Retos Educativos. Departamento de Filologías Inglesa y Alemana, Facultad de Letras. Universidad de Granada.

- [9] López-Cuadrado, J., Pérez, T.A., Armendariz, A.J.: Evaluación mediante Tests: ¿Por qué no usar el ordenador?. Revista Iberoamericana de Educación (ISSN: 1681-5653), (2005).
- [10] Nebot, A.: Qualitative Modeling and Simulation of Biomedical Systems Using Fuzzy Inductive Reasoning. Ph.D. thesis. Dept. Llenguatges i Sistemes Informàtics, Universitat Politècnica de Catalunya, Barcelona, Spain, 1994.
- [11] Castro, F., Nebot, A., Múgica, F.: On the extraction of decision support rules from fuzzy predictive models. Applied Soft Computing, 2011, Vol. 11, issue 4, 3463-3475, (2011).
- [12] Escobet, a., Nebot, A., Cellier, F.E.: Visual-FIR: A Tool for Model Identification and Prediction of Dynamical Complex Systems, Simul. Model. Pract. Th., 16, 76-92, 2008.

Razonamiento Intencional y No-Monotonicidad

José Martín Castro-Manzano¹, Axel Arturo Barceló-Aspeitia¹ y Alejandro Guerra-Hernández²

¹ Instituto de Investigaciones Filosóficas
Universidad Nacional Autónoma de México
Circuito Mario de la Cueva s/n Ciudad Universitaria, México, D.F., México, 04510
jmcmanzano@hotmail.com, abarcelo@minerva.filosoficas.unam.mx

² Departamento de Inteligencia Artificial
Universidad Veracruzana
Sebastián Camacho No. 5, Xalapa, Ver., México, 91000
aguerra@uv.mx

Resumen El razonamiento intencional es un proceso dinámico que implica creencias, intenciones y tiempo. Por ello requiere tanto una semántica temporal como un comportamiento no-monotónico. En este trabajo discutimos la naturaleza no-monotónica del razonamiento intencional y posteriormente proponemos un modelo para representarlo. Así, presentamos la adaptación y extensión de $CTL_{AgentSpeak(L)}$ con un marco no-monotónico. De esta manera sugerimos una lógica que trata con la no-monotonicidad del razonamiento intencional mientras toma cierto control con el tiempo. Y presentamos algunas de sus propiedades metalógicas.

Keywords: Lógica no-monotónica, lógica temporal, lógica BDI.

1. Introducción

El razonamiento intencional es un proceso dinámico que conlleva creencias, intenciones y tiempo. Ha sido representado y modelado por lógicas BDI [18,20,22]; sin embargo, al seguir estos modelos surgen dos problemas: primero, que el razonamiento humano no es y no debería ser monotónico [15], por lo que los modelos lógicos tienen que tener un comportamiento no-monotónico. Y segundo, los estados intencionales deben respetar las normas temporales, y entonces los modelos tienen que ser temporales también. Por tanto, no es difícil notar que el proceso de prueba del razonamiento intencional tiene que tener cierto tipo de control sobre el tiempo así como tomar en cuenta creencias e intenciones para realizar inferencias de forma no-monótona.

En el estado del arte las lógicas derrotables (*defeasible*) han sido principalmente desarrolladas para razonar sobre creencias [16] pero apenas han sido usadas para razonar sobre estructuras temporales [10]; por otro lado, las lógicas BDI han sido usadas para razonar sobre estados intencionales durante el tiempo pero son monotónicas. En este trabajo exponemos la naturaleza no-monotónica del razonamiento intencional y, para resolver el doble problema

que describimos renglones arriba, presentamos la adaptación y extensión de $CTL_{AgentSpeak(L)}$ [12] con un marco no-monotónico. Así pues, proponemos una lógica derrotable temporal que trata con la no-monotonidad de las intenciones mientras toma en cuenta el tiempo. Además mostramos algunas de sus propiedades metalógicas.

El trabajo está organizado de la siguiente manera. En la Sección 2 exponemos la naturaleza no-monotónica del razonamiento intencional. En la Sección 3 mostramos el sistema y algunas de sus propiedades metalógicas. En la Sección 4 discutimos los resultados y mencionamos trabajo futuro.

2. Razonamiento intencional y no-monotonidad

Como Bratman ha argumentado, los planes son también intenciones [4]. Así podemos justificar a las intenciones basadas-en-políticas como estructuras de la forma $te : ctx \leftarrow body$ [2] (véase Cuadro 1). Ahora, consideremos el siguiente ejemplo por mor del argumento: $sobre(X, Y) \leftarrow poner(X, Y)$. Esta intención nos dice que, para que un agente logre $sobre(a, b)$, éste típicamente tiene que poner a en b . Si imaginamos que tal agente está inmerso en un ambiente dinámico, es seguro que el agente intentará poner, típicamente, a sobre b ; sin embargo, un agente *racional* lo hará únicamente mientras sea *posible*.

Por tanto, resulta muy natural decir que algunas intenciones se mantienen típicamente pero no absolutamente. Y entonces también es razonable concluir que las intenciones, y en particular las basadas-en-políticas, permiten una forma de razonamiento intencional derrotable [9]. No obstante, los modelos BDI actuales son monotónicos y las lógicas no-monotónicas apenas son usadas para razonar sobre tiempo [10] o estados intencionales. Así, una lógica temporal derrotable que se las vea con la no-monotonidad de las intenciones mientras proporciona cierto cuidado con estructuras temporales no ha sido desarrollada aún.

Así por ejemplo, la Lógica de Primer Orden estándar es una instancia de razonamiento monotónico atemporal; la Lógica de Defaults [19] es un caso de razonamiento no-monotónico pero atemporal. Por otro lado, las lógicas BDI [18,20,22] son ejemplos de lógicas con elementos temporales pero que son monotónicas. Nuestra propuesta, en cambio, es un caso de razonamiento temporal y no-monotónico.

Los modelos BDI tradicionales formalizan el razonamiento intencional de manera monotónica [5,13,18,20,22], mientras nuestra propuesta lo hace de manera no-monotónica. Esto no sólo es necesario, está también justificado, pues así como los cambios en las creencias han requerido de una teoría de revisión de creencias [1] o una lógica no-monotónica [16], los cambios en las intenciones requieren una teoría de revisión de intenciones o una lógica no-monotónica de intenciones.

3. Marco no-monotónico

A pesar de los enormes avances en esta área, si tomamos en cuenta los fundamentos filosóficos de la agencia BDI [4], no es difícil notar que la mayoría de lógicas BDI falla en considerar las propiedades de las intenciones: propiedades funcionales como la proactividad, admisibilidad e inercia; propiedades descriptivas como la parcialidad, jerarquía y dinamismo; y, por supuesto, propiedades normativas como la consistencia interna, la consistencia fuerte y la consistencia medios-fines. La explicación de estas propiedades se encuentra en [4]. Siguiendo este orden de ideas proponemos el siguiente marco:

Definición 1 (*Marco intencional no-monotónico*) *Un marco intencional no-monotónico es una tupla $\langle B, I, F_B, F_I, \vdash, \vdash\sim, \neg, \sim, \succ \rangle$ donde:*

- B denota la base de creencias.
- I denota el conjunto de intenciones.
- $F_B \subseteq B$ denota las creencias básicas.
- $F_I \subseteq I$ denota las intenciones básicas.
- \vdash y \neg son relaciones de consecuencia fuerte.
- $\vdash\sim$ y \sim son relaciones de consecuencia débil.
- $\succ \subseteq I^2$ t.q. es acíclica.

Con la ayuda de este marco podemos representar la naturaleza no-monotónica del razonamiento intencional. Por el momento asumiremos que la estrategia de compromiso está embebida en la arquitectura del agente, i.e., la inercia de las intenciones está dada por un mecanismo fijo que es de tipo abierto (*single-minded*) [17], pues si no hay compromiso o si el agente tiene un compromiso tipo ciego (*blindly-committed*), no tiene sentido hablar de inercia [11,12], i.e., en reconsiderar intenciones.

Como es usual, B denota las creencias, las cuales son literales. F_B representa las creencias que se consideran básicas y similarmente, F_I denota intenciones básicas. Cada intención $\phi \in I$ es una estructura $te : ctx \leftarrow body$ donde te representa la meta de la intención –y así preservamos proactividad–, ctx un contexto y el resto denota al cuerpo. Cuando ctx o $body$ son vacíos escribimos $te : \top \leftarrow \top$ o simplemente te .

También preservamos la consistencia interna al dejar que el contexto de la intención, $ctx(\phi)$, $ctx(\phi) \in B$ y que te sea la cabeza de la intención. Así, la consistencia fuerte es implicada por la consistencia interna (dado que la consistencia interna significa que $ctx(\phi) \in B$). La coherencia medios-fines es implicada por el criterio de admisibilidad y la jerarquía de las intenciones es representada por la relación de orden que tomamos como acíclica para resolver conflictos entre intenciones. Nuevamente, todas estas propiedades pueden hallarse en [4]. Y con este marco en mente podemos proponer una noción de inferencia donde decimos que ϕ es fuertemente (débilmente) derivable a partir de una secuencia Δ si y sólo si hay una prueba de $\Delta \vdash \phi$ ($\Delta \vdash\sim \phi$). Y también, que ϕ no es fuertemente (débilmente) derivable si y sólo si hay una prueba de $\Delta \neg \phi$ ($\Delta \sim \phi$), donde $\Delta = \langle B, I \rangle$.

4. $NCTL_{AgentSpeak(L)}$

Usamos $CTL_{AgentSpeak(L)}$ [12] como una herramienta lógica para la especificación formal. Por supuesto, inicialmente este acercamiento es similar al de un sistema BDI^{CTL} definido a partir $B^{KD45}D^{KD}I^{KD}$ con los operadores temporales siguiente (\bigcirc), eventualmente (\diamond), siempre (\square), hasta (U), opcional (E), inevitable (A), y así, de acuerdo a CTL^* [6,8]. En esta sección expondremos la sintaxis y semántica de $CTL_{AgentSpeak(L)}$.

4.1. Sintaxis y semántica de $AgentSpeak(L)$

Un agente ag está formado por un conjunto de planes ps y creencias bs (literales instanciadas). Cada plan tiene la forma $te : ctx \leftarrow h$. El contexto ctx de un plan es una literal o una conjunción de ellas. Un cuerpo no-vacío h es una secuencia finita de acciones $A(t_1, \dots, t_n)$, de metas g (*achieve!* o *test?* una fórmula atómica $P(t_1, \dots, t_n)$), o actualizaciones u de creencias (adición + o eliminación -). \top denota los elementos vacíos, sean estos cuerpos, contextos, intenciones. Los eventos disparadores te son actualizaciones de creencias o metas. La sintaxis se muestra en el Cuadro 1.

$ag ::= bs \ ps$	$h ::= h_1; \top \mid \top$
$bs ::= b_1 \dots b_n \ (n \geq 0)$	$h_1 ::= a \mid g \mid u \mid h_1; h_1$
$ps ::= p_1 \dots p_n \ (n \geq 1)$	$at ::= P(t_1, \dots, t_n) \ (n \geq 0)$
$p ::= te : ctx \leftarrow h$	$a ::= A(t_1, \dots, t_n) \ (n \geq 0)$
$te ::= +at \mid -at \mid +g \mid -g$	$g ::= !at \mid ?at$
$ctx ::= ctx_1 \mid \top$	$u ::= +b \mid -b$
$ctx_1 ::= at \mid \neg at \mid ctx_1 \wedge ctx_1$	

Cuadro 1. Sintaxis de $AgentSpeak(L)$ adaptada de [2]

La semántica operacional de $AgentSpeak(L)$ se puede encontrar en [2] y está definida por un sistema de transición entre configuraciones $\langle ag, C, M, T, s \rangle$. Bajo esta semántica una *Corrida* = $\{(\sigma_i, \sigma_j) \mid \Gamma \vdash \sigma_i \rightarrow \sigma_j\}$ donde Γ es un sistema de transición definido por la semántica operacional de $AgentSpeak(L)$ y σ_i, σ_j son configuraciones agente.

4.2. Sintaxis de $BDI^{CTL}_{AS(L)}$

$CTL_{AgentSpeak(L)}$ puede ser visto como una instancia de BDI^{CTL} . Acercamientos similares han sido propuestos para otros lenguajes de programación [7]. La idea es definir la semántica BDI en términos de estructuras de $AgentSpeak(L)$. Así pues, necesitamos un lenguaje capaz de expresar estados temporales e intencionales. Por tanto, requerimos, en primer lugar, una manera de expresar estas características.

Definición 2 (Sintaxis de $BDI_{AS(L)}^{CTL}$) Si ϕ es una fórmula atómica de *Agent-Speak(L)*, entonces $BEL(\phi)$, $DES(\phi)$ e $INT(\phi)$ son fórmulas bien formadas de $BDI_{AS(L)}^{CTL}$.

Para especificar el comportamiento temporal usamos CTL^* .

Definición 3 (Sintaxis temporal de $BDI_{AS(L)}^{CTL}$) Cada fórmula $BDI_{AS(L)}^{CTL}$ es una fórmula de estado s :

- $s ::= \phi | s \wedge s | \neg s$
- $p ::= s | \neg p | p \wedge p | Ep | Ap | \bigcirc p | \diamond p | \square p | p \cup p$

4.3. Semántica de $BDI_{AS(L)}^{CTL}$

Inicialmente la semántica de BEL , DES e INT es adoptada de [3]. Así, usamos la siguiente función:

$$\begin{aligned} \text{agoals}(\top) &= \{\}, \\ \text{agoals}(i[p]) &= \begin{cases} \{at\} \cup \text{agoals}(i) & \text{si } p = +!at : ct \leftarrow h, \\ \text{agoals}(i) & \text{de otro modo} \end{cases} \end{aligned}$$

la cual nos regresa el conjunto de fórmulas atómicas (at) sujetas a una meta tipo *achieve* ($+$!) e $i[p]$ denota la pila de intenciones con p en el tope.

Definición 4 (Semántica de $BDI_{AS(L)}^{CTL}$) Los operadores BEL , DES e INT están definidos en términos de un agente ag y su configuración $\langle ag, C, M, T, s \rangle$:

$$BEL_{\langle ag, C, M, T, s \rangle}(\phi) \equiv \phi \in bs$$

$$INT_{\langle ag, C, M, T, s \rangle}(\phi) \equiv \phi \in \bigcup_{i \in C_I} \text{agoals}(i) \vee \bigcup_{(te, i) \in C_E} \text{agoals}(i)$$

$$DES_{\langle ag, C, M, T, s \rangle}(\phi) \equiv \langle +!\phi, i \rangle \in C_E \vee INT(\phi)$$

donde C_I denota las intenciones actuales y C_E las intenciones suspendidas.

4.4. El sistema $NBDI_{AS(L)}^{CTL}$

El siguiente paso es la propuesta de un sistema denotado por $NBDI$ porque tiene una conducta no-monotónica. Una intención ϕ en $NBDI_{AS(L)}^{CTL}$ es, por supuesto, una estructura, $\langle g : ctx \leftarrow body \rangle$ donde g es la cabeza, ctx es el contexto y $body$ es el cuerpo. Además denotamos una intención ϕ cuya cabeza es g con $\phi[g]$. Y una intención negativa es denotada por $\phi[g^c]$, e.d., la intención ϕ con $\neg g$ como cabeza. La semántica de esta teoría requiere de una estructura de Kripke $K = \langle S, R, V \rangle$ donde S es el conjunto de configuraciones agente, R es una relación de acceso definida a partir del sistema de transición Γ y V es una función de valuación que va de las configuraciones agente a las proposiciones verdaderas en esos estados.

Definición 5 Sea $K = \langle S, \Gamma, V \rangle$, entonces:

- S es un conjunto de configuraciones agente $c = \langle ag, C, M, T, s \rangle$.
- $\Gamma \subseteq S^2$ es una relación total tal que para todo $c \in S$ hay un $c' \in S$ t.q. $(c, c') \in \Gamma$.
- V es una valuación t.q.:
 - $V_{\text{BEL}}(c, \phi) = \text{BEL}_c(\phi)$ donde $c = \langle ag, C, M, T, s \rangle$.
 - $V_{\text{DES}}(c, \phi) = \text{DES}_c(\phi)$ donde $c = \langle ag, C, M, T, s \rangle$.
 - $V_{\text{INT}}(c, \phi) = \text{INT}_c(\phi)$ donde $c = \langle ag, C, M, T, s \rangle$.
- Los caminos son series de configuraciones c_0, \dots, c_n t.q. $\forall i (c_i, c_{i+1}) \in R$. Usamos x^i para indicar el i -ésimo estado del camino x . Entonces:

S1 $K, c \models \text{BEL}(\phi) \Leftrightarrow \phi \in V_{\text{BEL}}(c)$

S2 $K, c \models \text{DES}(\phi) \Leftrightarrow \phi \in V_{\text{DES}}(c)$

S3 $K, c \models \text{INT}(\phi) \Leftrightarrow \phi \in V_{\text{INT}}(c)$

S4 $K, c \models E\phi \Leftrightarrow \exists x = c_1, \dots \in K | K, x \models \phi$

S5 $K, c \models A\phi \Leftrightarrow \forall x = c_1, \dots \in K | K, x \models \phi$

P1 $K, c \models \phi \Leftrightarrow K, x^0 \models \phi$ donde ϕ es una fórmula de estado.

P2 $K, c \models \bigcirc\phi \Leftrightarrow K, x^1 \models \phi$.

P3 $K, c \models \diamond\phi \Leftrightarrow K, x^n \models \phi$ para $n \geq 0$

P4 $K, c \models \square\phi \Leftrightarrow K, x^n \models \phi$ para toda n

P5 $K, c \models \phi \cup \psi \Leftrightarrow \exists k \geq 0$ t.q. $K, x^k \models \psi$ y para toda $j, k, 0 \leq j < k | K, c^j \models \phi$
o $\forall j \geq 0 : K, x^j \models \phi$

Como decíamos en la Sección 3, tenemos cuatro casos de prueba: si la secuencia es $\Delta \vdash \phi$, decimos que ϕ es fuertemente probable; si es $\Delta \nvdash \phi$ decimos que ϕ no es fuertemente probable. Si es $\Delta \vdash \sim \phi$ decimos que ϕ es débilmente probable y si es $\Delta \nvdash \sim \phi$, entonces ϕ no es débilmente probable.

Definición 6 (Prueba) Una prueba de ϕ a partir de Δ es una secuencia finita de creencias e intenciones que satisface:

1. $\Delta \vdash \phi$ syss
 - 1.1. $\square A(\text{INT}(\phi))$ o
 - 1.2. $\square A(\exists \phi[g] \in F_I : \text{BEL}(\text{ctx}(\phi)) \wedge \forall \psi[g'] \in \text{body}(\phi) \vdash \psi[g'])$
2. $\Delta \vdash \sim \phi$ syss
 - 2.1. $\Delta \vdash \phi$ o
 - 2.2. $\Delta \nvdash \sim \phi$ y
 - 2.2.1. $\diamond E(\text{INT}(\phi) \cup \neg \text{BEL}(\text{ctx}(\phi)))$ o
 - 2.2.2. $\diamond E(\exists \phi[g] \in I : \text{BEL}(\text{ctx}(\phi)) \wedge \forall \psi[g'] \in \text{body}(\phi) \vdash \psi[g'])$ y
 - 2.2.2.1. $\forall \gamma[g^c] \in I, \gamma[g^c]$ falla en Δ o
 - 2.2.2.2. $\psi[g'] \succ \gamma[g^c]$
3. $\Delta \nvdash \phi$ syss
 - 3.1. $\diamond E(\text{INT}(\neg \phi))$ y
 - 3.2. $\diamond E(\forall \phi[g] \in F_I : \neg \text{BEL}(\text{ctx}(\phi)) \vee \exists \psi[g'] \in \text{body}(\phi) \nvdash \psi)$
4. $\Delta \nvdash \sim \phi$ syss
 - 4.1. $\Delta \nvdash \phi$ y
 - 4.2. $\Delta \vdash \neg \phi$ o
 - 4.2.1. $\square A \neg(\text{INT}(\phi) \cup \neg \text{BEL}(\text{ctx}(\phi)))$ y
 - 4.2.2. $\square A(\forall \phi[g^c] \in I : \neg \text{BEL}(\text{ctx}(\phi)) \vee \exists \psi[g'] \in \text{body}(\phi) \nvdash \psi[g'])$ o
 - 4.2.2.1. $\exists \gamma[g^c] \in I$ t.q. $\gamma[g^c]$ tiene éxito en Δ y
 - 4.2.2.2. $\psi[g'] \not\succeq \gamma[g^c]$

4.5. Consistencia

Un cuadrado de oposición es una colección de relaciones lógicas mostradas en un diagrama. A continuación sugerimos un cuadrado de oposición para describir algunas propiedades metalógicas del sistema presentado aquí.

Proposición 1 (*Subalternas₁*) Si $\vdash \phi$ entonces $\vdash \sim \phi$.

Corolario 1 (*Subalternas₂*) Si $\sim \vdash \phi$ entonces $\vdash \phi$.

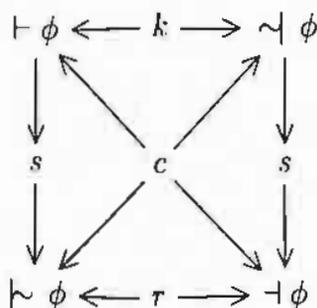
Proposición 2 (*Contradictorias₁*) No existe ϕ t.q. $\vdash \phi$ y $\vdash \neg \phi$.

Corolario 2 (*Contradictorias₂*) No existe ϕ t.q. $\vdash \sim \phi$ y $\sim \vdash \phi$.

Proposición 3 (*Contrarias*) No existe ϕ t.q. $\vdash \phi$ y $\sim \vdash \phi$.

Proposición 4 (*Subcontrarias*) Para toda ϕ , $\vdash \phi$ o $\sim \vdash \phi$.

Uniendo estos resultados podemos construir el siguiente cuadrado de oposición donde *c* denota las contradictorias, *s* las subalternas, *k* las contrarias y *r* las subcontrarias.



Estos resultados son importantes por las siguientes propiedades: la Proposición 1 y el Corolario 1 representan supraclasicidad; la Proposición 2 y el Corolario 2, la consistencia del sistema; mientras las restantes proposiciones especifican la coherencia del cuadro y del sistema en general.

4.6. Corrección

Además de la consistencia, que es una propiedad deseable, presentamos una forma de corrección (*soundness*). La idea es que el sistema es correcto con respecto a su semántica. Por tanto, como es usual, necesitaremos nociones de satisfacción y validez.

Definición 7 (*Satisfacción*) Una fórmula ϕ es verdadera en K si y sólo si ϕ es verdadera en todas las configuraciones σ en K . Es decir, $K \models \phi \Leftrightarrow K, \sigma \models \phi$ para toda $\sigma \in S$.

Definición 8 (Corrida de un agente en un modelo) Dada una configuración inicial β , un sistema de transición Γ y una valuación V , $K_r^\beta = \langle S_r^\beta, R_r^\beta, V \rangle$ denota una corrida de un agente en un modelo.

Definición 9 (Validez) Una fórmula $\phi \in BDI_{AS(L)}^{CTL}$ es verdadera para cualquier corrida agente en Γ $\text{sys} \forall K_r^\beta \models \phi$

Más aún, denotaremos $(\exists K_r^\beta \models \phi \cup \neg \text{BEL}(\text{ctx}(\phi))) \vee \models \phi$ con $\approx \phi$. Podemos observar, además, que $\models \phi \geq \approx \phi$ y $\approx \phi \geq \models \phi$. Ahora, con estas observaciones debemos encontrar una serie de traducciones t.q.:

Proposición 5 Las siguientes relaciones se dan:

$$a) \text{ Si } \vdash \phi \text{ entonces } \models \phi \quad b) \text{ Si } \vdash \phi \text{ entonces } \approx \phi$$

Prueba. Caso base. Tomando Δ_i como una secuencia con $i = 1$, tenemos dos casos base.

Caso a) Si asumimos $\vdash \phi$, tenemos dos subcasos. El primer subcaso está dado por la Definición 6 ítem 1.1. Entonces tenemos $\Box A(\text{INT}(\phi))$. Esto significa, por la Definición 5 ítems P4 y S5 y Definición 4, que para todos los caminos y todos los estados $\phi \in C_I \vee C_E$. Podemos representar esta expresión por medio de una traducción en términos de corridas. Dado que los caminos y los estados son secuencias de configuraciones agente, tenemos que $\forall K_r^\beta \models \phi$, lo cual implica $\models \phi$. El segundo subcaso está dado por la Definición 6 ítem 1.2, que en términos de corridas significa que para todas las corridas $\exists \phi[g] \in F_I : \text{BEL}(\text{ctx}(\phi)) \wedge \forall \psi[g'] \in \text{body}(\phi) \vdash \psi[g']$. Dado que Δ_1 es de un solo paso, $\text{body}(\phi) = \top$ y para toda corrida $\text{BEL}(\text{ctx}(\phi))$, $\text{ctx}(\phi) \in F_B$. Entonces $\forall K_r^\beta \models \phi$ que, igual que en el caso anterior, implica $\models \phi$.

Caso b) Supongamos que $\vdash \phi$. Entonces tenemos dos subcasos. El primero está dado por la Definición 6 ítem 2.1. Así, tenemos que $\vdash \phi$ que, como renglones arriba, ya implica $\models \phi$. Por otro lado, por el ítem 2.2, tenemos $\neg \neg \phi$ y dos alternativas. La primera alternativa, ítem 2.2.1, es $\Diamond E(\text{INT}(\phi) \cup \neg \text{BEL}(\text{ctx}(\phi)))$. Por tanto, podemos reducir esta expresión por medio de la Definición 5 ítems P3 y S4, para una traducción en términos de corridas: $\exists K_r^\beta \models \phi \cup \neg \text{BEL}(\text{ctx}(\phi))$, que implica $\approx \phi$. La segunda alternativa viene del ítem 2.2.2, $\Diamond E(\exists \phi[g] \in I : \text{BEL}(\text{ctx}(\phi)) \wedge \forall \psi[g'] \in \text{body}(\phi) \vdash \psi[g'])$ que en términos de corridas significa que para alguna corrida $\exists \phi[g] \in I : \text{BEL}(\text{ctx}(\phi)) \wedge \forall \psi[g'] \in \text{body}(\phi) \vdash \psi[g']$, pero Δ_1 es un solo paso, y por tanto $\text{body}(\phi) = \top$. Entonces, hay una corrida en la cual $\exists \phi[g] \in I : \text{BEL}(\text{ctx}(\phi))$, i.e., $(\exists K_r^\beta \models (\phi \cup \neg \text{BEL}(\text{ctx}(\phi))))$ al usar el caso débil de la Definición 6 P5. Por tanto, por adición, $(\exists K_r^\beta \models (\phi \cup \neg \text{BEL}(\text{ctx}(\phi)))) \vee \models \phi$, y por tanto, $\approx \phi$.

Caso inductivo. Caso a) Asumamos que para $n \leq k$, si $\Delta_n \vdash \phi$ entonces $\Delta \models \phi$. Y supongamos Δ_{n+1} . Más aún, supongamos $\Delta_n \vdash \phi$, entonces tenemos dos alternativas. Siendo la primera, por Definición 6 ítem 1.1, que tenemos una intención ϕ t.q. $\text{ctx}(\phi) = \text{body}(\phi) = \top$. Dado que $\text{body}(\phi)$ es vacío, trivialmente se da en n , y por la hipótesis de inducción, $\text{body}(\phi) \subseteq \Delta_{n+1}$, y por tanto $\models \phi$.

En segundo lugar, por la Definición 6 ítem 1.2, para todas las corridas $\exists\phi[g] \in I : \text{BEL}(\text{ctx}(\phi)) \wedge \forall\psi[g'] \in \text{body}(\phi) \vdash \psi[g']$. Entonces, para todas las corridas n , $\forall\psi[g'] \in \text{body}(\phi) \vdash \psi[g']$, y por la hipótesis de inducción, $\text{body}(\phi) \subseteq \Delta_{n+1}$, i.e., $\Delta \vdash \psi[g']$. Por tanto, $\models \phi$.

Caso b) Supongamos que para $n \leq k$, si $\Delta_n \vdash \phi$ entonces $\Delta \approx \phi$. Y supongamos que Δ_{n+1} . Asumamos que $\Delta_n \vdash \phi$. Tenemos dos alternativas. La primera está dada por la Definición 6 ítem 2.1, i.e., $\vdash \phi$, que de hecho implica $\models \phi$. La segunda alternativa está dada por el ítem 2.2, $\Delta \vdash \neg\phi$ y dos subcasos: $\Diamond E(\text{INT}(\phi) \cup \neg\text{BEL}(\text{ctx}(\phi)))$ o bien $\Diamond E(\exists\phi[g] \in I : \text{BEL}(\text{ctx}(\phi)) \wedge \forall\psi[g'] \in \text{body}(\phi) \vdash \psi[g'])$. Si consideramos el primer subcaso hay corridas n que cumplen con la definición de $\approx \phi$. En el caso restante tenemos que $\forall\psi[g'] \in \text{body}(\phi) \vdash \psi[g']$, dado que $\text{body}(\phi) \subseteq \Delta_n$, por la hipótesis de inducción $\Delta \vdash \psi[g']$, y por tanto, $\Delta_{n+1} \vdash \phi$, i.e., $\approx \phi$. ■

Más aún, podemos encontrar traducciones para los fragmentos restantes:

Proposición 6 *Las siguientes relaciones se dan:*

- a) Si $\vdash \phi$ entonces $\models \phi$ b) Si $\sim \vdash \phi$ entonces $\approx \phi$

5. Conclusión

Expusimos una lógica derrotable y temporal para tratar la no-monotonía del razonamiento intencional. Pudimos hacer esto al extender un marco no-monotónico con una semántica temporal. Entonces mostramos algunas de las propiedades del sistema, en particular, observamos que preserva supraclasicidad, consistencia y corrección.

Finalmente, como parte de nuestro trabajo actual podemos mencionar que aún estamos refinando las definiciones y buscamos una relación entre la noción de revisión y no-monotonía. Además, podemos prever aplicaciones prácticas en el campo de sistemas agentes, ya que nuestra propuesta está íntimamente relacionada con AgentSpeak(L) y una implementación puede seguirse orgánicamente.

Agradecimientos. Los autores agradecen a los revisores anónimos por sus acertados comentarios y precisas correcciones. El primer autor está apoyado por la beca CONACyT 214783.

Referencias

1. Alchourrón, C. E., Gärdenfors, P., Makinson, D.: On the logic of theory change: Partial meet contraction and revision functions. *Journal of Symbolic Logic* (1985) 50 510–530
2. Bordini, R.H., Wooldridge, M., Hübner, J.F.: *Programming Multi-Agent Systems in AgentSpeak using Jason* (Wiley Series in Agent Technology). John Wiley & Sons (2007)

3. Bordini, R.H., Moreira, Á.F.: Proving BDI Properties of Agent-Oriented Programming Languages. *Annals of Mathematics and Artificial Intelligence* (2004) 42 197–226
4. Bratman, M.E.: *Intention, Plans, and Practical Reason*. Cambridge University Press (1999)
5. Cohen, P., Levesque, H.: Intention is choice with commitment. *Artificial Intelligence* (1990) 42(3), 213–261
6. Clarke, E.M. Jr., Grumberg, O., Peled, D.A.: *Model Checking*. MIT Press (1999).
7. Dastani, M., van Riemsdijk, M.B., Meyer, J.C.: A grounded specification language for agent programs. *Proceedings of the 6th international joint conference on Autonomous agents and multiagent systems '07 AAMAS '07* (2007) 1–8
8. Emerson, A.: Temporal and modal logic. *Handbook of Theoretical Computer Science*, Elsevier Science Publishers (1995) 995–1072
9. Governatori, G., Padmanabhan, V. and Sattar, A.: A Defeasible Logic of Policy-based Intentions. *Proceedings of the 15th Australian Joint Conference on Artificial Intelligence: Advances in Artificial Intelligence*. LNAI-2557, Springer Verlag (2002)
10. Governatori, G., Terenziani, P.: Temporal Extensions to Defeasible Logic. *Proceedings of the 20th Australian joint conference on Advances in artificial intelligence* (2007) 476–485
11. Guerra-Hernández, A., Castro-Manzano, J.M., El-Fallah-Seghrouchni, A.: Toward an AgentSpeak(L) Theory of Commitment and Intentional Learning. *Proceedings of the 7th Mexican International Conference on Artificial Intelligence: Advances in Artificial Intelligence MICAI 2008*, LNCS, vol. 5317, 848–858, Springer-Verlag, Berlin Heidelberg, (2008)
12. Guerra-Hernández, A., Castro-Manzauo, J.M., El-Fallah-Seghrouchni, A.: CTL AgentSpeak(L): a Specification Language for Agent Programs. *J. Algorithms* (2009) 64(1) 31–40
13. Konolige, K., Pollack, M. E.: A representationalist theory of intentions. *Proceedings of International Joint Conference on Artificial Intelligence (IJCAI-93)* (1993) 390–395
14. K. M. Sim: Epistemic Logic and Logical Omniscience: A Survey. *International Journal of Intelligent Systems*, vol. 12, 57–81, John Wiley and Sons, Inc (1997).
15. Nute, D.: Defeasible logic. *INAP 2001*, LNAI 2543M, Springer-Verlag, (2003) 151–169
16. Prakken, H., Vreeswijk, G.: Logics for defeasible argumentation. In D. Gabbay and F. Guenther (eds.), *Handbook of Philosophical Logic*, second edition, Vol 4, Kluwer Academic Publishers (2002) 219–318
17. Rao, A.S., Georgeff, M.P.: Modelling Rational Agents within a BDI Architecture. In: Huhns, M.N., Singh, M.P., (eds.) *Readings in Agents*, Morgan Kaufmann (1998) 42–55
18. Rao, A.S.: AgentSpeak(L): BDI agents speak out in a logical computable language. In: de Velde, W.V., Perram, J.W. (eds.) *MAAMAW*. LNCS, vol. 1038, Springer, Heidelberg (1996) 42–55
19. Reiter, R.: A logic for default reasoning. *Artificial Intelligence*, (1980) 13 81–132
20. Singh, M.P., Rao, A.S., Georgeff, M.P.: Formal Methods in DAI: Logic-Based Representation and Reasoning. In: *Multiagent Systems: A Modern Approach to Distributed Artificial Intelligence*, MIT Press, Cambridge (1999) 331–376
21. van der Hoek, W., Jamroga, W., Wooldridge, M.: Towards a theory of intention revision. *Synthese* (2007) 155(2) 265–290
22. Wooldridge, M.: *Reasoning about Rational Agents*. MIT Press (2000).

Caracterización de la Maduración Ósea mediante Conjuntos Aproximados y Conjuntos Difusos.

José Alejandro Cid Medina¹, Fernando López Irrarangori¹

¹Facultad de Mecánica y Eléctrica, Posgrado de la Ingeniería en Sistemas,
Universidad Autónoma de Nuevo León, México
{alex,femy}@yalma.fime.uanl.mx

Resumen. En este trabajo se presenta el problema de la estimación del potencial de crecimiento de niños y adolescentes con edades entre 9 y 16 años a partir de la estimación de la maduración ósea empleando cefalometrías laterales que abarcan las vértebras cervicales 2da a las 4ta. Se aplica la teoría de conjuntos aproximados (Rough Sets) para establecer reglas que caractericen las etapas de maduración ósea establecidas por Lamparski, así como etapas intermedias introducidas por Prado y López en un trabajo reciente. Posteriormente se emplean estas reglas para realizar una clasificación más fina empleando conjuntos difusos (Fuzzy Sets) donde establece si un paciente está al inicio de una etapa, establecido en la etapa o saliendo de la etapa incorporando al modelo el carácter continuo del proceso de crecimiento

1 Introducción

En los tratamientos de ortodoncia, ortopédicos, cirugía maxilofacial, endocrinológicos, y por lo general en tratamientos pediátricos el momento ideal para iniciarlos o modificarlos es cuando el paciente está cerca de su estado potencial de crecimiento ya que se estimula aún más la respuesta biológica [1] para lograr resultados óptimos que son afectados por el crecimiento [1],[2]. Mientras que el proceso de crecimiento del ser humano está correlacionado con la maduración ósea, no lo está con la edad cronológica [3], según Hernández [4] "... el conocimiento de este dato (maduración ósea) es fundamental para realizar una estimación de la talla final, para valorar las posibilidades terapéuticas de un niño con patología del crecimiento, y para controlar los efectos del tratamiento. Incluso en otras especialidades distintas de la pediatría y endocrinología pediátrica, como la traumatología y ortodoncia, el conocimiento de la maduración ósea es también de gran utilidad para decidir el momento de iniciar algunos tratamientos...". Esto también aplica para el potencial de crecimiento, mismo que se relaciona estrechamente con la madurez esquelética [5], ambos son influidos por factores tales como: alimentación, calidad de vida, zona geográfica y etnicidad de la muestra.

El proceso de crecimiento es continuo y no uniforme, con periodos de aceleración y desaceleración[5], en [1] se plantea que "el inicio, intensidad y duración del peak puberal de crecimiento facial presentan variaciones considerables entre individuos" esto aplica también para el crecimiento de huesos en general, por ello cualquier clasificación no es más que una aproximación, que será tan fina como etapas se identifiquen, para algunos tratamientos basta con considerar periodos (etapas) de entre 3 y 6 meses, por ejemplo ciertos tratamientos ortodóncicos, sin embargo otros requieren de mayor precisión, como cirugías maxilofaciales u ortopédicas.

La estimación de la madurez ósea es una tarea difícil, varios investigadores han utilizado diferentes indicadores de madurez que se consideran poco confiables como: la edad cronológica, desarrollo dental, peso, estatura, cambios de voz, menarquía, características sexuales secundarias [1] maduración sexual. Los estudios efectuados tomando información de cefalogramas craneofaciales [6] pueden considerarse como buenos indicadores de madurez.

Otros indicadores como la raza y el sexo, se estudiaron ampliamente en otras investigaciones, y ninguno de los autores informó importancia de estos parámetros [7].

Durante años el indicador de madurez ósea considerado el mejor y más confiable es el que se evalúa con las radiografías mano-muñeca [5]. Este indicador se utiliza para tratamientos ortopédicos funcionales de mandíbula, ortodoncia y en tratamientos pediátricos en general [6].

Este método presenta algunos inconvenientes entre los que se discuten los siguientes:

- Se usa más de una radiografía.
- Se expone al individuo a radiaciones adicionales
- Las radiografías se deben tomar en la misma posición

- En algunas radiografías se transponen los huesos lo que dificulta su estudio [6]
- La mayoría de estas evaluaciones no son automáticas por ello los resultados observados son subjetivos [6].

Zaror [8] menciona que un buen indicador o método para estimar la maduración ósea "1. Debe de ser eficaz al determinar el peak de crecimiento mandibular (aplica también al peak de los huesos en general), 2.- Debe evitar exposiciones adicionales a rayos X. 3.- Debe tener fácil aplicación e interpretación, 4.- Debe tener consistencia en la interpretación de los datos (inter-examinadores). 5.- Debe tener una utilidad diagnóstica al anticipar la ocurrencia del peak". Al respecto en [6] se afirma que hasta el momento no existe un método que sea un estándar para la estimación ósea, ni un método que la describa en términos simples.

En el presente trabajo se presenta una metodología de apoyo a la decisión para la clasificación de la maduración ósea, empleando radiografías laterales de las vértebras cervicales de un paciente con edad entre los 9 y 16 años, suficientemente fina como para que sirva en un espectro amplio de aplicaciones médicas y arroje estimaciones suficientemente precisas del potencial de crecimiento mediante la caracterización más simple de la maduración ósea. La metodología emplea los conjuntos aproximados (rough sets [9]) para generar reglas de clasificación en base a la forma, concavidad inferior y crecimiento de las vértebras 2, 3 y 4. Estas reglas son a su vez empleadas para establecer los conjuntos difusos (fuzzy sets) que caracterizarán finalmente cada etapa de maduración ósea en tres niveles: inicio, desarrollo y salida. Las medidas son tomadas del análisis de la radiografía lateral de las vértebras; las reglas son utilizadas para la caracterización. El problema es formulado como un problema de clasificación multicriterio. Se presenta la aplicación de la metodología en un estudio realizado en niños entre 10 y 16 años de edad con oclusión normal, es decir que sean niños sanos.

Los autores establecieron como hipótesis de partida que la clasificación más fina a través de la caracterización basada en las radiografías de las vértebras cervicales con el uso de conjuntos aproximados y conjuntos difusos aportarán resultados que ayuden a identificar la etapa y grado de transición de la madurez ósea en periodos de 1 a 3 meses.

La contribución científica de este trabajo se establece en el desarrollo de una metodología de estimación del potencial de crecimiento a partir del Análisis Multicriterio de la Decisión que permite:

Estimación "más fina" de las etapas de la maduración ósea. Al respecto, San Roman [5] menciona "el crecimiento es continuo, por lo que cualquier clasificación solo es una aproximación a la madurez ósea".

Tanto la metodología como el método desarrollado resultan viables de implementar computacionalmente y los resultados son fácilmente interpretables por los especialistas.

Ampliando un poco más sobre la contribución científica, con anterioridad a este trabajo los resultados de estimación de la maduración ósea empleando vértebras cervicales han tenido una aceptación limitada por parte de los especialistas debido a lo complicado de los modelos empleados que exigen un conocimiento matemático de los especialistas para poder interpretar correctamente los resultados. Por otra parte la generalización de los resultados no es posible sin volver a realizar todo el proceso para lo que se requiere la intervención del analista de la decisión. La metodología presentada en este trabajo no presenta esos inconvenientes, se basa en supuestos que pueden ser comprendidos sin dificultad por los especialistas y los resultados se presentan en formas de reglas de tipo "si... entonces" lo que facilita su interpretación.

2 Trabajos previos

En la búsqueda de un indicador de la maduración ósea que fuera tan exacto como lo es la evaluación que se hace con radiografías de la mano- muñeca, recientemente Lamparski [10], y otros autores determinaron una nueva forma de evaluar la madurez del esqueleto con vértebras cervicales como referencia [5].

Los nuevos métodos están basados en radiografías laterales cefalométricas, que son usadas de manera rutinaria en diagnósticos ortodóncicos [3] en las que se estudian las vértebras cervicales, un gran número de estos nuevos métodos está basado en el método de Lamparski. Su evaluación es subjetiva [3].

La maduración ósea está relacionada con el crecimiento y forma de las vértebras [6], la estimación de la madurez ósea es una tarea complicada [6]. Varios investigadores han realizado una serie de trabajos que demuestran científicamente que la estimación de la maduración ósea basada en las radiografías de las vértebras cervicales dan resultados tan precisos como los que se obtienen con radiografías de mano/muñeca, y a diferencia de este último solo se necesita una radiografía. [1]. Debido a esto al individuo se le deja de exponer a radiaciones adicionales y solo se estudian algunos huesos Lamparski [10] y Baccetti et al. [11], estudian de la segunda a la sexta, mientras que Hassel y Farman [12] estudian de la segunda a la cuarta vértebra respectivamente) simplificando el estudio, mismo procedimiento que siguen López et al. [6].

La mayoría estos estudios concluyen que evaluar la madurez ósea trabajando con las vértebras cervicales es más sencillo, más económico, y evita exposiciones adicionales a radiaciones. Esta manera de estimar la madurez ósea ha demostrado ser más fiable que los realizados con radiografías de mano-muñeca.

Townsend y Grave sugieren que la evaluación de la maduración vertebral se puede aplicar de manera general a la práctica de ortodoncia [7]

Lamparski [10] establece un patrón donde estudia la morfología de los cuerpos vertebrales. Toma en cuenta las siguientes características: concavidad del borde inferior, altura y forma de las vértebras [3] para estimar la maduración ósea. Examina de la segunda a la sexta vértebra ahí es donde se presentan los cambios de madurez que sirven para estimar la madurez ósea. Estandarizo 6 estados en los cuales se producen los cambios morfológicos de las vértebras cervicales. Comprobó estadísticamente que usar esta forma de estimación de madurez ósea es tan válida y confiable como la que se realiza con radiografías de mano y muñeca, teniendo como ventaja el uso de una sola radiografía para este estudio. Los indicadores de madurez son los mismo para hombres y mujeres. La diferencia radica en que los hombres tardan más en madurar que las mujeres [7]. [6]

Tomando como punto de partida el trabajo de Lamparski [10], Hassel y Farman [12] realizaron su propia investigación y concluyeron que basta con estudiar en las radiografías cefalométricas la segunda (C2), tercera (C3) y cuarta vértebra (C4) para estimar la maduración ósea se evaluó este indicador correlacionándolo con el realizado por Fishman que estima la maduración ósea utilizando radiografías de mano-muñeca, ambas radiografías fueron tomadas el mismo día. La aportación de este estudio clasifico los estadios de maduración en 6 etapas: 1. Iniciación, 2. Aceleración, 3. Transición, 4. Desaceleración, 5. Maduración, 6. Finalización. Cada una tiene características diferentes. Características relacionadas a la forma de la vértebra que indican madurez y si hay un potencial de crecimiento.

Según San Román [5], al considerar el estudio de la concavidad inferior, la altura y la morfología de las vértebras para estimar la madurez ósea, queda demostrado que el mejor indicador es la concavidad inferior de las vértebras.

Los estudios realizados tienen un único objetivo, evaluar la edad ósea de un paciente y establecer si el crecimiento ha sucedido, está sucediendo o sucederá [7]. Quintana [1] dice "...el beneficio de conocer la etapa de madurez ósea al permite adoptar medidas respecto a las terapias ortopédicas que se indiquen en pacientes con anomalías esqueléticas".

Los métodos creados hasta la actualidad investigan como estimar la maduración ósea, se avanza y se mejora en este campo a través de los años recientes, aun así falta una metodología que sea un estándar que permita obtener resultados más exactos y precisos. Por ejemplo se puede establecer que un estado de madurez de un individuo puede estar en la etapa 3 de maduración ósea, pero no sabemos si se acaba de ingresar o está por terminar e ingresar a otra. Este dato resulta de interés a los especialistas en la toma de decisiones del comienzo o modificación de algún tratamiento específico que esté relacionado con el crecimiento.

Recientemente López y Prado [6] realizaron un estudio donde caracterizan la maduración ósea empleando conjuntos aproximados. Ellos identifican 9 criterios que están basadas en: el ángulo de maduración, la forma y el crecimiento de las vértebras 2da, 3ra y 4ta. También ellos introducen etapas de transición entre cada pareja de etapas consecutivas de Lamparski (por ejemplo se introduce la etapa de Iniciación -> Aceleración que se interpreta como la etapa de transición entre iniciación y aceleración, para identificar pacientes que tienen características de ambas etapas), con el objetivo de capturar el carácter continuo del proceso de crecimiento óseo. Sin embargo, para algunas especialidades médicas esta clasificación resulta muy grosera ya que cada etapa puede tener una duración que va de los 3 a los 8 meses.

3 Conceptos básicos sobre conjuntos aproximados y conjuntos difusos.

3.1 Conjuntos aproximados

La teoría de los conjuntos aproximados (rough sets en inglés) esta centrada en la idea de la clasificación, donde por clasificación se entiende la habilidad de un agente (humano, robot, etc.) o grupo de agentes para discernir ciertos fenómenos, procesos, objetos, etc. de una realidad concreta. Luego, conocimiento en este contexto es entendido como una "familia de clasificaciones" por diferentes patrones de un dominio de interés. Esta familia de clasificaciones representa hechos explícitos de una realidad concreta, conjuntamente con la capacidad de derivar o descubrir hechos implícitos de conocimiento explícito [9].

Esta teoría es fácil de entender y tiene, entre otras, las ventajas que a continuación se expresan: provee algoritmos eficientes para encontrar patrones ocultos en los datos, encuentra conjuntos reducidos de datos (reducción de datos), evalúa la significancia de los datos, genera conjuntos mínimos de reglas de decisión a partir de los datos, ofrece interpretaciones simples de los resultados, puede emplearse para análisis cuantitativos y cualitativos de datos, identifica relaciones que no pueden encontrarse empleando métodos estadísticos.

Los datos analizados con la teoría de los conjuntos aproximados son estructurados en un conjunto de objetos (observaciones, juicios, estados, etc.) descritos por un conjunto de atributos multivalentes. El conjunto de objetos puede representarse y sus atributos pueden representarse mediante una tabla de decisión.

Sea $S = \langle U, R \rangle$ una tabla de decisión, donde U es un conjunto de objetos y $R = (C \cup D)$ es un conjunto de atributos. C y D representan conjuntos de atributos de condición y decisión respectivamente. Desde un punto de vista matemático una

clasificación en S no es más que una relación de equivalencia definida en U , por ello en lo que resta se empleará el término relación de equivalencia en lugar de clasificación. Los conjuntos minimales de relaciones de equivalencia que aseguran la misma calidad en la aproximación que el conjunto completo son denominados reductos. La intersección de todos los reductos es denominada núcleo, la eliminación de cualquier elemento del núcleo conlleva a la pérdida de calidad en la clasificación.

Los elementos previamente descritos constituyen la base de los algoritmos que son utilizados para reducir las filas y las columnas de una tabla de decisión sin afectar la fuerza de clasificación de sistema. Estos algoritmos facilitan el obtener algoritmos de decisión asociados a la tabla. Para profundizar más sobre este tema se puede consultar [9], [13]. Algunas aplicaciones recientes sobre conjuntos aproximados pueden encontrarse en [14], [15].

3.2 Conjuntos difusos

La teoría de los subconjuntos difusos o borrosos (palabras intercambiables en este contexto) fue desarrollada por [16] con el fin de representar matemáticamente la imprecisión intrínseca de ciertas categorías de objetos. Los subconjuntos difusos (o partes borrosas de un conjunto) fueron inventados para modelar la representación humana de los conocimientos (por ejemplo para medir nuestra ignorancia o una imprecisión objetiva) y mejorar así los sistemas de decisión, de ayuda a la decisión, y de inteligencia artificial.

Un **conjunto difuso**, es un conjunto que puede contener elementos de forma parcial. Es decir que la propiedad $x \in U$ puede ser cierta con un grado de verdad.

Se mide esta posibilidad de pertenecer (o pertenencia) con un número $\mu_A(x)$ entre 0 y 1, llamado **grado de pertenencia** de X a A . Si es 0, $x \notin A$, si es 1, entonces $x \in U$ totalmente, y si $0 < \mu_A \leq 1$, $x \in A$ de una manera parcial.

Un subconjunto A de B se caracteriza, por tanto, por esta función de pertenencia μ_A , de B en $[0,1]$. Es preciso fijar el conjunto B para definir la función μ_A que a su vez define A . Por eso se habla de subconjunto difuso y no de conjunto difuso.

Nótese que μ_A es una proposición en el contexto de la lógica difusa, y no de la lógica usual binaria, que sólo admite dos valores: cierto o falso.

La **lógica difusa**, según Zadeh [17], es una metodología para el análisis de datos cuando la información disponible es demasiado imprecisa para justificar el uso de números, y existe una tolerancia a la imprecisión que puede ser explotada para alcanzar tratabilidad, robustez, bajo costo de solución y mejor representación de la realidad.

Para profundizar más en este tema se puede revisar [16], y [18], para una la aplicación de la lógica difusa en la clasificación.

4 Metodología propuesta

A continuación se describe la metodología propuesta para la caracterización del potencial de crecimiento de pacientes con maloclusión normal y edades entre 10 y 16 años.

Paso #1 En primer lugar, a partir de medidas longitudinales tomadas de cefalometrías laterales como se describe en [19] se conforma una tabla de decisión en la que se introducen los atributos de condición y decisión propuestos en [6] y cuyas filas representan a pacientes. La intersección de una fila con una columna representa entonces el valor de una propiedad o atributo de una vértebra cervical derivada de una radiografía de un paciente.

Los atributos propuestos en [6] se asocian a las vértebras 2da, 3ra y 4ta reflejando características de forma y tamaño de las mismas:

1. Concavidad del borde inferior de cada vértebra (2da, 3ra y 4ta) que los expertos identifican como: plano, ligeramente cóncavo, cóncavo y cóncavo pronunciado.
2. Aumento de la altura del borde posterior de las vértebras 3ra y 4ta, respecto a un promedio de la población estudiada con edades entre 9 y 10 años. Los expertos estimaron el aumento en: no significativo, significativo y muy significativo.
3. Aumento de la altura del borde anterior de las vértebras 3ra y 4ta, respecto a un promedio de la población estudiada con edades entre 9 y 10 años. Los expertos estimaron el aumento en: no significativo, significativo y muy significativo.
4. Aumento del ancho en las vértebras 2da, 3ra y 4ta. Aquí también los expertos coincidieron en medir el aumento en: no significativo, significativo y muy significativo.

5. Finalmente se consideró la forma de las vértebras 3ra y 4ta. Misma que los expertos consideraron como: cuña pronunciada, cuña moderada, cuadrado, rectángulo más ancho que alto, rectángulo más alto que ancho.

Paso#2 El siguiente paso de la metodología consiste en aplicar la metodología de los conjuntos aproximados [9] a la tabla de decisión obtenida en el paso anterior para generar reglas que caractericen las etapas definidas por Lamparski y etapas de transición definidas por [6] en términos simples, luego como resultado de la aplicación de los pasos anteriores cada paciente es clasificado en alguna de las etapas siguientes: Iniciación, Iniciación->Aceleración, Aceleración,

Aceleración ->Transición, Transición, Transición->Desaceleración, Desaceleración, Desaceleración->Madurez, Madurez, Madurez->Terminación, Terminación.

Sin embargo, esta clasificación no brinda información "fina" sobre en qué momento de la etapa se encuentra el paciente, por ejemplo si se encuentra iniciando la etapa, o ya se encuentra en un periodo en el cual claramente se reconocen las características de la etapa o si está saliendo de la etapa. Precisamente el obtener una clasificación más fina es la motivación de los pasos subsiguientes de la metodología. De esta manera se reduce el error en la identificación de la etapa, se realiza una clasificación más realista al reconocer que el proceso de maduración es continuo en el modelo.

Paso#3 El paso siguiente consiste en definir conjuntos difusos para cada categoría de cada atributo presente en la tabla de decisión conformada en el paso#1 (ver figuras 5.1 a 5.5 para ejemplos). Se procede de la forma siguiente: para cada atributo se define un conjunto difuso que representa cada categoría del atributo, por ejemplo para los atributos que representan el ángulo de maduración para las vértebras 2da, 3ra y 4ta se definen para cada una conjuntos difusos que representan las categorías: plano, ligeramente cóncavo, cóncavo, cóncavo pronunciado.

Para construir los conjuntos difusos se realiza un análisis inverso identificando las medidas longitudinales de cada paciente asociado a una etapa de maduración ósea (resultado de los dos primeros pasos), luego se descartan los outliers y se calculan la media y la varianza de las medidas longitudinales de cada categoría de cada atributo de cada vertebra de cada paciente para cada clase. Inicialmente se construyen las funciones de pertenencia de cada categoría tomando el valor medio como el máximo valor de pertenencia (1) y la desviación estándar como el mínimo (0). Se utilizan funciones triangulares y tipo rampas (izquierda o derecha) para representar las funciones de pertenencia. Luego esos valores se validan y ajustan con la opinión de expertos. Como resultado de este paso se obtienen conjuntos difusos que representan todas las categorías de todos los atributos considerados en los dos primeros pasos.

Paso#4 En el paso final se definen tres niveles para cada clase: inicio, meseta, salida. Cada nivel se representa por un número difuso que resulta de la integración de los números difusos definidos previamente para cada categoría en cada criterio. Para la definición de los conjuntos difusos que representan los niveles de cada categoría puede emplearse un panel de expertos que proporcionen los datos en un proceso diseñado rigurosamente para garantizar un alto nivel de confianza.

Al final se obtiene una clasificación para cada paciente que no sólo indica la etapa de maduración ósea en que se encuentra el paciente, sino también a que nivel de esa etapa se encuentra, si está iniciando la etapa o esta en una fase de consolidación de la etapa o se encuentra saliendo de la misma. Lo que reduce la incertidumbre en la caracterización de la maduración ósea.

5 Experimentación

Para validar la metodología se realizó un experimento con los datos de un conjunto de 80 niñas con oclusión normal y edades entre 10 y 16 años, tomado de [19], de las cuales se tenían datos longitudinales de las vértebras cervicales 2da, 3ra y 4ta, tales como: ancho, altura del borde anterior, altura del borde posterior y ángulo de maduración ósea (concavidad del borde inferior). Se incluye la comparación con la clasificación dada por López y Prado [6], no se compara con otros métodos de clasificación ya que esto se realizó en ese trabajo y se muestra que su metodología aporta mejores valores en cuanto a cobertura y calidad.

A estas niñas se le aplicó la metodología según se describe en López y Prado [6] de lo que resultaron 24 reglas que arrojaron la clasificación que se muestra en la Tabla 5.1, en la Tabla 5.2 se muestra la cobertura y otros parámetros de calidad de la clasificación realizada.

Tabla 5.1 Clasificación conjuntos aproximados, clase a iniciación, b iniciación->aceleración, c: aceleración, d: aceleración->transición, e: transición, f: transición->desaceleración, g: desaceleración, h: desaceleración->madurez

a	b	c	d	e	f	g	h	Clase
10	0	0	0	0	0	0	0	a
1	29	0	2	0	0	0	0	b
0	0	0	0	0	0	0	0	c
0	1	0	18	0	0	0	0	d
0	0	0	0	0	0	0	0	e
0	0	0	2	0	14	0	0	f
0	0	0	0	0	1	0	0	g
0	0	0	0	0	0	0	2	h

Tabla 5.2 Cobertura de conjuntos aproximados

Reglas de decisión	Correcta clasificación	Cobertura
28	1	0,938
24	1	0,875
20	1	0,8
25	0,788	1

Para definir los conjuntos difusos y obtener las reglas de clasificación finales se empleó el paquete InFuzzy [20], desarrollado por la Universidad de Santa Cruz do Sul, en su versión 1.1. Este es un software muy versátil, práctico y muy completo en cuanto a opciones para modelar conjuntos difusos e inferir reglas difusas para la clasificación.

Realizando un análisis de las medidas longitudinales correspondientes cada clase de pacientes establecida por la aplicación de los pasos #1 y #2 de la metodología se establecieron las funciones de pertenencia correspondientes a los conjuntos difusos establecidos para cada categoría de cada atributo (según se describe en el paso #3 de la metodología), se eligieron funciones triangulares por su simplicidad. En las figuras 5.1, 5.2, 5.3, 5.4 y 5.5 se muestran las funciones de pertenencia de cada conjunto difuso.

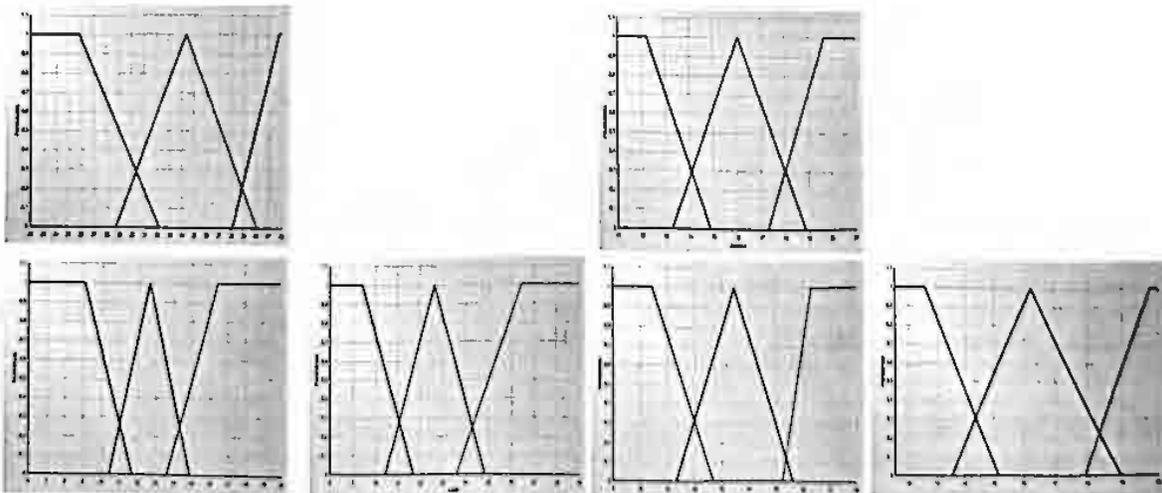


Fig. 5.1 Atributo "Abp" de las vértebras 2,3 y 4. Número difuso de rampa izquierda(No Significativo), Triangular(Significativo) y Rampa Derecha(Muy Significativo).

Fig. 5.2 Atributo "Ancho" de las vértebras 2,3 y 4. Número difuso de rampa izquierda (No Significativo), Triangular(Significativo) y Rampa Derecha(Muy Significativo).

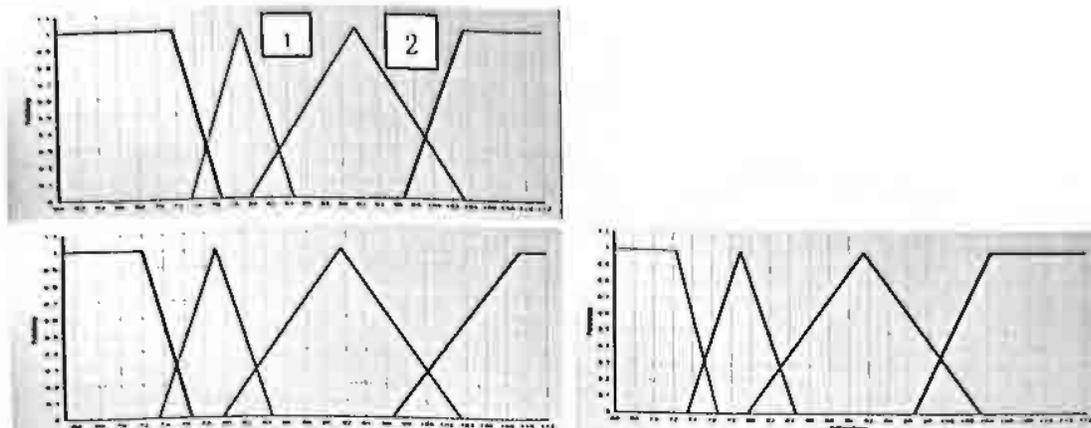


Fig. 5.3 Atributo "AngMe" de las vértebras 2,3 y 4. Número difuso de rampa izquierda(Cóncavo pronunciado), Triangular 1(Cóncavo), Triangular 2 (Ligeramente Cóncavo) y Rampa Derecha(Plano).

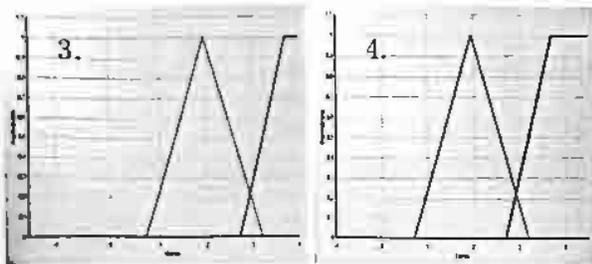


Fig. 5.4 Atributo "Forma3_1 y Forma4_1" de las vértebras 3 y 4. Triangular (Cuña pronunciada) y Rampa Derecha(Cuña moderada).

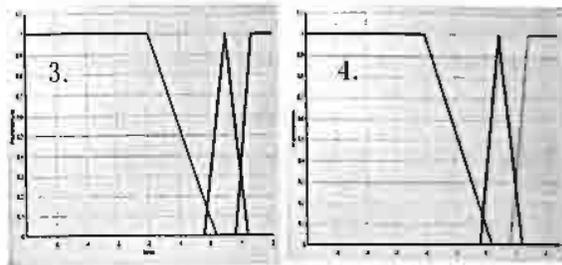


Fig. 5.5 Atributo "Forma3_2 y Forma4_2" de las vértebras 3 y 4. Rapa por izquierda(Más ancho que alto), Triangular (Cuadrada) y Rampa Derecha(Más alto que ancho).

Luego para cada paciente se calcularon los valores de pertenencia a cada categoría de cada atributo y se aplicó un operador de consolidación para determinar la categoría a la que se asociaría cada paciente a cada atributo, en este caso se dejó al software elegir de forma automática el operador de entre un conjunto muy completo (Centro de gravedad, Primeros de los máximos, centro de los máximos, median de los máximos y altura).

Finalmente se obtuvo la clasificación que se resume en la tabla 5.3

Tabla 5.3. Clasificación obtenida por el sistema difuso

1	Iniciación	2
2	Sale_Iniciacion y Entra_Iniciacion/Aceleracion	9
3	Sale_Iniciacion/Aceleracion Y entra a Aceleracion	39
4	Sale_Aceleracion Entra_AceracionTransicion	5
5	Sale_AcelelacionTransicion y Entra_Transicion	20
6	Sale_Transicion y Entra_Transicion/Desaceracion	2
7	Sale_Transicion/Desaceleracion y Entra_Desaceleracion	3
8	Sale_Desacelelacion y Entra_Desaceleracion/Madurez	0
9	Estado de Madurez	0
	Total Fuzzy Sets	80

Analizando en detalle los resultados de la clasificación obtenidos por la aplicación de la metodología de López&Prado [6] y la metodología presentada en este trabajo (Rough/Fuzzy) se observó una coincidencia razonable en 70 de los 80 pacientes incluidos en el estudio.

Para identificar la coincidencia se tomó el criterio de que ambas clasificaciones ubicaban al paciente en niveles diferentes de una misma etapa o niveles adyacentes de dos etapas consecutivas (por ejemplo: la clasificación por conjuntos aproximados ubica a un paciente en la etapa iniciación, mientras que la clasificación por conjuntos difusos del mismo paciente lo ubica en la etapa inicial de la transición a la etapa de aceleración con un nivel bajo de pertenencia).

Las discrepancias entre las clasificaciones se muestran en la tabla 5.4 a y b.

Tabla 5.5 Muestra las medidas longitudinales (a) y lingüísticas (b) de los pacientes en los cuales ambas clasificaciones presentan discrepancias.

a)

No.	Edad	Sexo	2ABP	2ANCHO	3ABA	3ANCHO	3ABP	4ABA	2AngMC	3AngMC	4AngMC	4ANCHO	4ABP
82	13	2	29	13	13	15	14.5	11	96	87	80	14	14
106	14	2	36	14.5	13	17	15	12.5	71	86	89	17	16
125	15	2	32	14	13	15.5	14.5	11	72	83	88	14.5	14.5
105	14	2	36	14.5	12.5	15	15.5	12.5	79	77	74	15	15
129	15	2	33	13.5	14.5	16	17	15.5	75	76	74	15	16
122	15	2	32	11.5	13	13	12	11	90	72	70	13	13
130	15	2	39	16	15	17	15.5	14	67	80	74	16.5	16
141	16	2	33	12	15.5	15.5	14.5	14	68	80	70	16.5	13
149	16	2	28	13	13	18.5	14.5	12.5	68	70	77	18	16

a) continuación

No.	Edad	Sexo	forma3_1	forma4_1	forma3_2	forma4_2	Clasificación Metodología Rough/Fuzzy	Clasificación Lopez&Prado[12]
82	13	2	1.5	3	-1.5	-1	Sale_Aceleracion Entra_AceracionTransicion	Iniciacion/Aceleracion
106	14	2	2	3.5	-3	-2	Sale_Aceleracion Entra_AceracionTransicion	Iniciacion/Aceleracion
125	15	2	1.5	3.5	-2	-1	Sale_Aceleracion Entra_AceracionTransicion	Iniciacion/Aceleracion
105	14	2	3	2.5	-0.5	-1	Sale_Transicion Entra_TransicionDesaceleracion	Aceleracion/Transicion
129	15	2	2.5	0.5	0	0	Sale_Transicion Entra_TransicionDesaceleracion	Aceleracion/Transicion
122	15	2	-1	2	-2	-1	Sale_AcelelacionTransicion Entra_Transicion	Desaceleracion
130	15	2	0.5	2	-2.5	-1.5	Sale_AcelelacionTransicion Entra_Transicion	Desaceleracion
141	16	2	-1	-1	-2	-4.5	Sale_AcelelacionTransicion Entra_Transicion	Desaceleracion
149	16	2	1.5	3.5	-5	-3	Sale_AcelelacionTransicion Entra_Transicion	Transicion/Desaceleracion

b)

No. Paciente	2da Vertebra	3ra Vertebra	4ta Vertebra	Clasificación Rough/Fuzzy	Clasificación López&Prado [12]
82	Iniciación/Aceleración	Iniciación/Aceleración	Transición/Desaceleración	Salir_Aceleración_Entrar_Aceleración/Transición	Iniciación/Aceleración
106	Desaceleración	Iniciación/Aceleración	Iniciación/Aceleración	Salir_Aceleración_Entrar_Aceleración/Transición	Iniciación/Aceleración
125	Desaceleración	Iniciación/Aceleración	Iniciación/Aceleración	Salir_Aceleración_Entrar_Aceleración/Transición	Iniciación/Aceleración
105	Aceleración/Transición	Aceleración/Transición	Desaceleración/Madurez	Salir_Transición_Entrar_Transición/Desaceleración	Aceleración/Transición
129	Aceleración/Transición	Aceleración/Transición	Madurez	Salir_Transición_Entrar_Transición/Desaceleración	Aceleración/Transición
122	Iniciación/Aceleración	Desaceleración	Desaceleración/Madurez	Salir_Aceleración/Transición_Entrar_Transición	Desaceleración
130	Desaceleración	Aceleración/Transición	Desaceleración/Madurez	Salir_Aceleración/Transición_Entrar_Transición	Desaceleración
141	Desaceleración	Aceleración/Transición	Desaceleración/Madurez	Salir_Aceleración/Transición_Entrar_Transición	Desaceleración
149	Desaceleración	Transición/Desaceleración	Transición	Salir_Aceleración/Transición_Entrar_Transición	Transición/Desaceleración

Como se observa en todos los casos donde existen discrepancias no se evidencia un crecimiento normal, pues hay una diferencia muy grande en las etapas de maduración de las vértebras, por ejemplo en el paciente 82, primera fila Tabla 5.4. a y b). la diferencia entre las vértebras 2da, 3ra con la cuarta es demasiado grande y aun así se cataloga a la niña como que está en la etapa de transición de la iniciación a la aceleración. Este es un caso muy raro, pues se ha encontrado que hay una correlación fuerte entre las etapas de dentición y la maduración ósea (aunque es mucho más complicado establecer la maduración ósea por la etapa de dentición como afirma [19]). Probablemente se hayan cometido errores en las mediciones, o simplemente hay que realizar un estudio de mayor envergadura para verificar las condiciones de esos pacientes. Sin embargo, a pesar de que la clasificación no es correcta al emplear los conjuntos aproximados si es consistente, mientras que para el caso de la clasificación con los conjuntos difusos es inconsistente (ver por ejemplo el caso 149, aunque en este caso se justifica por la naturaleza del caso).

Para los lectores interesados contactar a los autores para solicitar todos los datos del experimento.

6 Conclusiones

La metodología desarrollada permite realizar una clasificación de las etapas de maduración más fina que las desarrolladas hasta la fecha, lo que reduce la incertidumbre en la duración de la etapa y la estimación del crecimiento potencial de los pacientes. No requiere un esfuerzo mucho mayor por parte del especialista al solicitar pocos datos además de los solicitados en la metodología desarrollada por López y Prado, que es uno de los trabajos más completos realizados para estimar la etapa de maduración ósea a partir de medidas longitudinales.

En el caso de estudio se evidenció una alta congruencia con la clasificación dada en López y Prado, salvo algunos casos que presentan inconsistencias en los datos con respecto a la hipótesis de que son pacientes con oclusión normal.

La metodología arrojó inconsistencia en la detección de una clase, ubicando a 3 pacientes en la misma cuando tienen diferencias substanciales en los atributos de condición, este hecho muestra que probablemente no se aplicable bajo condiciones de alta entropía en los datos. Por lo que se recomienda chequear exhaustivamente los datos antes de iniciar las pruebas, así como ampliar la base de datos de pacientes para estudios futuros.

7 Bibliografía

- 1 Quintana, R.Z. and Bravo, H.P. and Zaror, Q.R. and Paniagua, B.H., Determinación de la Maduración Esquelética según el Método de Evaluación Cervicovertebral y su Relación con la Oportunidad de Tratamiento con Clase II Dentoesquelética, *J. Odontostomat.*, 2(1)27, 31, (2008)
- 2 Hessa A. Al Khal, Ricky W. K. Wong and A. Bakr M. Rabie, Elimination of hand-wrist radiographs for maturity assessment in children needing orthodontic therapy, *Skeletal Radiology* 2008 Volume 37, Number 3, 195-200 (2008)
- 3 Caldas M., Ambrosano G. M. B., Haiter-Neto F., Use of cervical vertebral dimensions for assessment of children growth, *Journal of Applied Oral Science*, 15(2) 144-147(2007)
- 4 Hernández, M., Maduración ósea y predicción de talla, *Rev. Pediatr.*; 32: 265 - 272 (1991)
- 5 San Román, P. and Palma, J.C. and Oteo, M.D. and Nevado, E. Skeletal maturation determined by cervical vertebrae development, *European Journal of Orthodontics* 24, 303-311 (2002)
- 6 López F., Prado L. M., and Garza R., Multi-criteria classification applied to characterize skeletal maturation in male and female from 9 to 16 years with normal occlusion Technical Report PISIS-2010-03, Graduate Program in Systems Engineering, UANL, San Nicolás de los Garza, México, 13 May 2010
- 7 Polizin S., Skeletal and vertebral age - a literature review, <http://www.scribd.com/doc/17583100/Skeletal-and-Vertebral-Age-A-Literature-Review>
- 8 Zaror QR & Paniagua BH Skeletal maturation determination by cervical vertebral assessment method and its relationship with dentoskeletal Class II treatment opportunity, *Int. J. Odontostomat*, 2008, 2(1),27-31.
- 9 Pawlak Z. Rough sets: theoretical aspects of reasoning about data, Springer, Kluwer Academic Publisher, (1991)
- 10 Lamparski DG & Nanda SK. Skeletal age assessment utilizing cervical vertebrae. *The british library*, 2002, 39:171-184.
- 11 Baccetti T., Franchi L., and McNamara J. A., An improved version of the cervical vertebral maturation (cvm) method for the assessment of mandibular growth. *Angle Orthodontist*, 72(4):316-323, (2002)
- 12 Hassel B. and Farman A. G., Skeletal maturation evaluation using cervical vertebrae. *American Journal of Orthodontics and Dentofacial Orthopedics*, 107(1):58-66. (1995).
- 13 Greco S., Matarazzo B., Slowinski R. Rough sets theory for multicriteria decision analysis, *European Journal of Operational Research*, 129:1-47 (2001)
- 14 Greco S., Dembczyński K., and Słowiński R. Rough set approach to multiple criteria classification with imprecise evaluations and assignments, *European Journal of Operational Research*, 198(2): 626-636 (2009)
- 15 Li Y., Liao X., Zhao W., A rough set approach to knowledge Discovery in analyzing competitive advantages of firms, *Annals of Operation Research* 168: 205-223 (2009)
- 16 Lotfi A. Z. "Conjuntos difusos y los sistemas". En: Fox J, editor. *teoría del sistema*. Brooklyn, NY: Universidad Politécnica Press, 1965: 29-39 (1965)
- 17 Lotfi A. Zadeh, Life Fellow, IEEE, Fuzzy Logic = Computing with Words, *IEEE Transactions On Fuzzy Systems*, Vol. 4, No. 2, May 1996
- 18 P. Melin et al. (Eds.): *Anal. and Des. of Intel. Sys. using SC Tech.*, ASC 41, pp. 53-62, 2007 [18]
- 19 Calvo D., *Edad ósea en las vértebras cervicales en población de 9 a 16 años con oclusión normal*. Master's thesis, Facultad de Ciencias Médicas, Matanzas, Cuba, (2006).
- 20 Posselt, E. L.; Frozza, R.; Molz, R. F.. *Software Infuzzy 2011*. Programa de Mestrado em Sistemas e Processos Industriais PPGSPI, UNISC, 2011. Disponível em: <http://www.unisc.br/ppgspl>

Lenguaje de Definición, Manipulación y Minería de datos

Cristina Díaz Jiménez y Oscar Herrera Alcántara

Departamento de Sistemas, Universidad Autónoma Metropolitana Azcapotzalco
Av. San Pablo 180, Col. Reynosa Tamaulipas, 02200, México, D.F.

cdiazj88@gmail.com
oha@correo.azc.uam.mx
<http://ia.azc.uam.mx>

Resumen Este artículo presenta avances en el desarrollo de una extensión al lenguaje estructurado de consultas (SQL) que agrega funcionalidades de algoritmos de aprendizaje supervisado y no supervisado, útiles en tareas de minería de datos. Se incluye un módulo de procesamiento de datos con redes neuronales de perceptrones en multicapa (MLP) y otro con mapas autoorganizados de Kohonen (SOM).

Palabras clave. SQL, Minería de Datos, Aprendizaje, Redes Neuronales, Patrones de diseño de software.

1. Introducción

El lenguaje estructurado de consultas (SQL por sus siglas en inglés: Structured Query Language) está formado por dos sublenguajes: el Lenguaje de Definición de Datos (LDD) y el Lenguaje de Manipulación de Datos (LMD)[1]. El LDD permite definir las estructuras de datos en una base de datos (tablas, vistas) en tanto que el LMD implementa operadores que actúan sobre los datos de acuerdo con el Álgebra Relacional.

Una Base de Datos Relacional se define como un conjunto de relaciones normalizadas cuya cardinalidad varía con el tiempo, en donde cada relación R consta de un encabezado y un cuerpo. El encabezado define un conjunto de G pares (atributo, dominio) en donde el dominio es la categoría que identifica al conjunto de valores atómicos que se pueden almacenar en una base de datos en forma de campo. El cuerpo define un conjunto de C tuplas (instancias) formadas de G pares (atributo, valor) en donde los atributos toman valores del dominio respectivo. Es común referirse a una relación como tabla, a la cardinalidad de una relación como número de renglones (C) de la tabla, al grado como número de atributos (G), a una tupla como renglón, y valor atómico como campo de una tabla. Una de las ventajas más importantes de SQL es que al ser una implementación del patrón intérprete[2] permite definir una cadena de texto acorde con la gramática misma de SQL, y por lo tanto es posible ejecutarla con independencia física de los datos (no importa el hardware, sistema operativo o tecnología de

programación empleada para almacenar y manipular los datos). El SQL inmerso se incluye en los lenguajes de programación de propósito general tales como Java[3] o C++, a fin de agregar funcionalidades para conectarse a una base de datos, definir tipos, y manipular los datos. El SQL interactivo actúa como un intérprete de comandos en donde cada sentencia SQL se interpreta, ejecuta y muestra los resultados en una consola, pero no tiene variables ni lógica propios de un lenguaje de programación.

Los manejadores de bases de datos (SGBD) suelen incluir un módulo con SQL interactivo. El SQL inmerso suele incluirse como bibliotecas para varios lenguajes de programación. El SQL interactivo es frecuentemente usado en tareas de carga de datos, respaldos y consultas emergentes. El SQL inmerso ha permitido desarrollar aplicaciones en diferentes lenguajes de programación y arquitecturas de desarrollo de software (Cliente/Servidor, Web, etc.) y ha sido factor fundamental en el desarrollo de sistemas de información en donde se ha evolucionado de aplicaciones de escritorio con acceso a archivos dependientes de la tecnología y de las prácticas de cada programador a aplicaciones con manejadores de bases de datos bajo SQL estándar, en donde un mayor número de usuarios, con niveles que van de principiante a avanzado, pueden almacenar, recuperar y manipular datos e inclusive intercambiarlos con otros sistemas de información[4].

Con el creciente uso de sistemas de información, la cantidad de datos almacenada se ha incrementado considerablemente y el tipo de información requerida por los usuarios es cada vez más sofisticada, especialmente para la toma de decisiones que representan la principal ventaja estratégica de las empresas. Es necesario aplicar técnicas de reducción de dimensión (grado de las relaciones), de reducción del número de renglones (cardinalidad de las relaciones) sin perder la representatividad de los datos (muestreo), y otras técnicas relacionadas con la minería de datos, que consiste en identificar patrones no triviales inmersos en los datos que son útiles para una empresa. La minería de datos brinda por tanto las bases para desarrollar sistemas de toma de decisiones, en donde más que hablar de información se hace referencia al conocimiento extraído del conjunto de datos usando métodos de aprendizaje automático. Estos conceptos, limitados en los sistemas de información, son la piedra angular de las aplicaciones de inteligencia de negocios (BI por sus siglas en inglés: Business Intelligence)[5].

La idea fundamental en este artículo es: "Dado que la minería de datos requiere definir estructuras y manipular datos es posible aprovechar las ventajas del SQL de un DBMS, que al extenderlo con módulos de algoritmos de aprendizaje automático puede brindar un ambiente interactivo o inmerso que facilite el desarrollo de aplicaciones de inteligencia de negocios en diferentes arquitecturas". Para lograr lo anterior se propone construir un lenguaje de definición, manipulación y de minería de datos: SQL Minero.

El artículo está organizado en secciones, en la Sección 2 se revisa la estructura de SQL Minero. En la Sección 3 se revisa la extensión de SQL con algoritmos de aprendizaje automático. En la Sección 4 se reportan resultados experimentales, y finalmente en la Sección 5 se presentan conclusiones y trabajos futuros.

2. SQL Minero y Aprendizaje automático

SQL Minero implementa el patrón compositor que, en términos del paradigma orientado a objetos envuelve la funcionalidad otro objeto -en este caso el objeto SQL- para agregarle funcionalidades de minería de datos. Actualmente sólo se le han agregado dos sentencias para procesamiento de tuplas: una para clasificación y otra para agrupamiento. La primera sentencia es:

“Select * From Table Clustering with MLP ParameterList”

que emplea una red de perceptrones en multicapa (MLP[6], MultiLayer Perceptron) entrenada con el algoritmo de retropropagación y, la segunda es:

“Select * From Table Clustering with SOM ParameterList”

que usa el algoritmo de mapas autoorganizados de Kohonen (SOM[7], Self Organizing Map).

En SQL Minero, las redes neuronales MLP implementan el tipo de aprendizaje supervisado con datos almacenados en una tabla con los datos de entrenamiento. En esa tabla se incluye una columna con las clases a las que pertenecen los tuplas (objetos) y mediante un proceso iterativo el objeto supervisor ajusta las conexiones de la red neuronal de acuerdo al error en cada iteración hasta lograr que para cada tupla se obtenga el indicador de la clase deseada.

De manera análoga, las redes neuronales SOM implementan el tipo de aprendizaje no supervisado que agrupan tuplas (objetos) en clases inicialmente desconocidas. El ajuste se hace mediante un proceso iterativo que mapea las tuplas a un mapa topológico donde hay neuronas ubicadas en coordenadas cartesianas. Las neuronas actúan como vectores centroides representativas de un grupo de tuplas. La distancia euclidiana entre las neuronas permite etiquetarlas para identificar las tuplas en clases.

Se define a una sentencia de SQL Minero con la siguiente gramática[8]:

SQLMinero \rightarrow SQL | SQLMinero

SQLMinero \rightarrow Select *Atributos* From *Table Clustering with Algorithm ParameterList*

Algorithm \rightarrow MLP | SOM

en donde:

- SQL puede ser una sentencia de SQL estándar (patrón compositor, se usa el objeto original SQL y se agrega funcionalidad).
- Se usan negritas para indicar las variables de las reglas de producción, y cursivas para los parámetros provistos por el usuario.
- ParameterList es la lista de parámetros requeridas por el software WEKA[9][5] (ver Sección 3).

Ciertamente éste no es el primer lenguaje que incluye funcionalidades de minería de datos, véase por ejemplo Oracle Data Mining (ODM) [10], DB2 Data Warehouse Edition Data Mining (DB2DWEDM)[11], entre otros. Sin embargo, éstos programas además de ser caros, son honerosos computacionalmente, complejos en su uso, requerir de otros productos comerciales, reservado para pocos

usuarios, distribuidos como parte de otros productos, y principalmente no proveen los códigos fuente que permitan incluir nuevos algoritmos de aprendizaje automático. Con SQL Minero se pretende construir una herramienta de fácil instalación, configuración y uso, que promueva la enseñanza, desarrollo y uso de algoritmos de aprendizaje automático en la explotación inteligente de bases de datos relacionales.

Para que la gramática anterior tenga sentido, es necesario cumplir con la propiedad de cerradura[12], los operadores relacionales básicos de restricción, selección y multiplicación lo hacen, y por tanto cualquier operación sobre relaciones devuelve una relación, esto es: $R \leftarrow R1 \text{ operador } R2$.

De esta manera, para visualizar los resultados de la clasificación de los datos almacenados en una tabla, se agrega otra columna con los resultados de la red neuronal MLP o SOM, y puesto que a una tabla a la que se le agrega una columna genera otra tabla se mantiene la propiedad de cerradura. Lo anterior es cierto para la clasificación con MLP o el agrupamiento con SOM. En el caso de la clasificación la clase de cada tupla se conoce, y la columna adicional tendrá la clasificación provista por la red MLP. En el caso del agrupamiento no se conoce inicialmente la clase de cada tupla, y la columna adicional tendrá la clasificación provista por la red SOM.

Otra parte de la información útil en el proceso de aprendizaje automático es la matriz de confusiones, que permite ilustrar en una tabla el número de tuplas mal clasificadas, y las correctamente clasificadas. Una clasificación perfecta se reporta en celdas en diagonal que suman el 100% de las muestras, y ceros en las celdas restantes.

ParameterList permite ajustar la calidad del aprendizaje, que podrá ejecutarse en modo por lotes en SQL Minero interactivo, o mediante un programa a través de SQL Minero inmerso. Puesto que la matriz de confusiones es una tabla conviene almacenarla como metadato de las consultas de SQL Minero, en futuras versiones se espera desplegarlo mediante una instrucción como:

```
Select MatrizDeConfusion From Table  
Clustering with Algorithm ParameterList
```

La matriz de confusión es generada por el software Weka.

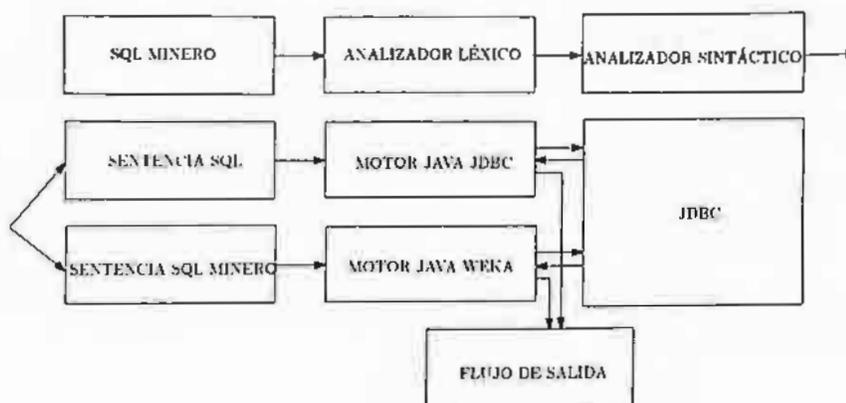
3. SQL Minero y WEKA

WEKA[9] es una colección de algoritmos de aprendizaje automático para minería de datos. Los algoritmos pueden aplicarse directamente a un conjunto de datos o usarse inmerso en código Java. WEKA contiene herramientas de preprocesamiento de datos, clasificación, regresión, agrupamiento, extracción de reglas de asociación y visualización.

Weka es un proyecto de código abierto que se distribuye bajo licencia GNU General Public License[13], motivo por el se ha elegido para implementar SQL Minero.

SQL Minero está implementado en Java y usa las bibliotecas de WEKA para clasificación y agrupamiento, en particular se usa la clase *weka.classifiers.functions.MultilayerPerceptron*, en tanto que para SOM se usa la clase *weka.classifiers.neural.lqu.Som* disponible como plug-in para Weka [14]. La arquitectura de SQL minero interactivo se ilustra en la figura 3 en donde se puede observar un módulo de análisis léxico que recibe una sentencia de SQL Minero, identifica los tokens y envía los *tokens* al analizador sintáctico a fin de identificar la regla de producción en la gramática y proceder a su ejecución. Si la sentencia es SQL tradicional se delega al motor del DBMS mediante el conector JDBC, se captura el flujo de datos con el resultado y se envía a la pantalla. En caso de ser una sentencia de SQL Minero se identifica el tipo de algoritmo a usar (MLP o SOM), se construye un archivo temporal en formato ARFF (Attribute-Relation File Format) requerido por Weka, se ejecuta el algoritmo de clasificación o agrupamiento, y el flujo de datos con el resultado se envía a la pantalla.

Figura 1. Arquitectura de SQL Minero.



El formato ARFF[15] permite definir conjuntos de datos para entrenamiento y prueba. En el proceso de entrenamiento se usa el 90% de los datos originales (ajustar las conexiones de la red neuronal MLP) y el restante 10% de los datos se usan para el proceso de prueba (revisar la calidad del aprendizaje). En SQL Minero interactivo cuando se ejecuta un MLP aparece una interfaz gráfica de usuario de Weka que muestra la arquitectura de la red neuronal, inicia los procesos de entrenamiento y prueba, y después Weka devuelve los valores de los pesos sinápticos.

Una vez realizado el entrenamiento con MLP o SOM, es posible usar esas redes para clasificar nuevos objetos (tuplas, renglones) que sean provistos en otra tabla (se desean conocer sus clases). A este proceso le llamamos predicción o prospección. De esta manera se espera extender SQL Minero para que en una sola instrucción anidada realice prospección. Por ejemplo, una primera tabla Tminero

que tenga los datos de entrenamiento-prueba, y otra tabla TmineroSinClases que tenga nuevos objetos que se desean clasificar. La instrucción podría lucir como:

```
Select Atributos,Clase From TmineroSinClases predict with
Select Atributos From Tminero clustering with Algorithh Parameters
```

La predicción será calculada en base al modelo neuronal construido con la segunda parte de la sentencia.

4. Experimentos

Se trabajó con datos de vinos del repositorio[16] almacenados en la base de datos dbminero en el manejador MySQL corriendo sobre GNU/Linux Debian. Los datos corresponden resultados del análisis químico de vinos que crecieron en la misma región de Italia, pero en tres cultivos. El análisis determinó las cantidades de 13 características en cada uno de los tres tipos de vino. El número de instancias es 178 de las cuales 58 son de clase 1, 71 son de clase 2, y 48 son de clase 3. Cada instancia se almacenó como renglón en la tabla tminero de MySQL[17]. Se trabajaron con los 178 renglones pero aquí se reportan sólo tres, a fin de ahorrar espacio y con fines ilustrativos. El siguiente script permite cargar los datos:

```
Insert into tminero (Clase,f1,f2,...,f13) VALUES
(1,14.23,1.71,2.43,15.6,127,2.8,3.06,0.28,2.29,5.64,1.04,3.92,1065),
(2,12.37,0.94,1.36,10.6,88,1.98,0.57,0.28,0.42,1.95,1.05,1.82,520),
(3,12.86,1.35,2.32,18,122,1.51,1.25,0.21,0.94,4.1,0.76,1.29,630);
```

El contenido parcial de la tabla TMinero se muestra en la Tabla 1.

Cuadro 1. Datos de tres clases de vinos.

Clase	f1	f2	f3	f4	f5	f6	f7	f8	f9	f10	f11	f12	f13
1	14.23	1.71	2.43	15.6	127	2.8	3.06	.28	2.29	5.64	1.04	3.92	1065
2	12.37	.94	1.36	10.6	88	1.98	.57	.28	.42	1.95	1.05	1.82	520
3	12.86	1.35	2.32	18	122	1.51	1.25	.21	.94	4.1	.76	1.29	630

El formato ARFF para Weka requiere conocer la siguiente información:

- El número de clases. Se puede calcular con SQL mediante:
SELECT DISTINCT COUNT(clase) FROM tminero;
- El número de características. Está dado por el número de columnas menos uno¹ en tminero. Se puede obtener de los metadatos mediante:

¹ La primera columna es el número de clase, para entrenar al MLP se usa pero no para el SOM.

```
SELECT COUNT(*) FROM INFORMATION_SCHEMA.COLUMNS
WHERE TABLE_SCHEMA=bdminero AND TABLE_NAME=tminero;
```

A fin de obtener mejores resultados de clasificación o agrupamiento conviene aplicar una normalización que equivale a llevar los valores de cada columna (dominio) al intervalo numérico [0, 1]. Esto puede hacerse mediante la fórmula:

$$NuevoValor = \frac{ValorActual - ValorMinimo}{ValorMaximo - ValorMinimo} \quad (1)$$

En el caso del aprendizaje no supervisado con SOM la columna clase no es requerida en el entrenamiento, así que antes de crear el archivo ARFF se hace una proyección que elimina la columna clase de la tabla tminero. En SQL esto se logra mediante:

```
Select f1,f2,f3,f4,f5,f6,f7,f8,f9,f10,f11,f12,f13 from Tminero;
```

Para fines comparativos los resultados de MLP y SOM devuelven las columnas de la tabla original más una columna adicional con la clase calculada con el MLP/SOM para los datos de entrenamiento. Aunque el SOM no requiere la columna clase, ésta se incluye con fines comparativos con el resultado de la proyección.

El archivo vinos.arff tiene el formato siguiente:

```
@RELATION vinos
@ATTRIBUTE f1 NUMERIC
@ATTRIBUTE f2 NUMERIC
@ATTRIBUTE f3 NUMERIC
...
@ATTRIBUTE f13 NUMERIC
@ATTRIBUTE class {1,2,3}
@DATA
14.23,1.71,2.43,15.6,127,2.8,3.06,.28,2.29,5.64,1.04,3.92,1065,1
12.37,.94,1.36,10.6,88,1.98,.57,.28,.42,1.95,1.05,1.82,520,2
12.86,1.35,2.32,18,122,1.51,1.25,.21,.94,4.1,.76,1.29,630,3
```

Cuando los datos son nominales (como los días de la semana) éstos se pueden convertir a numéricos calculando el número de clases o valores diferentes, y asignando un valor numérico a cada clase en el intervalo [0,1]. Actualmente sólo se tiene el SQL Minero interactivo, y se han probado las siguientes instrucciones:

```
SQL Minero>Select * from tminero clustering with MLP -H 10;
1|14.23|1.71|2.43|15.6|127|2.8|3.06|.28|2.29|5.64|1.04|3.92|1065|1
2|12.37|.94|1.36|10.6|88|1.98|.57|.28|.42|1.95|1.05|1.82|520|2
3|12.86|1.35|2.32|18|122|1.51|1.25|.21|.94|4.1|.76|1.29|630|3
```

ParameterList es en este caso -H 10, que define el número de neuronas en la primera capa oculta del MLP. En este momento no se ha ajustado la arquitectura del MLP para proveer un aprendizaje satisfactorio, y ha quedado como trabajo pendiente.

Con la finalidad de comparar la clase original (clase) con la provista por el MLP (clasemlp) se puede cambiar el orden de las columnas a través de la consola de SQL Minero interactivo:

```
SQL Minero>Select f1,f2,f3,f4,f5,f6,f7,f8,f9,
                f10,f11,f12,f13,clase,clasemlp
                from tminero clustering with MLP -H 10;
14.23|1.71|2.43|15.6|127|2.8|3.06|.28|2.29|5.64|1.04|3.92|1065|1|1
12.37|.94|1.36|10.6|88|1.98|.57|.28|.42|1.95|1.05|1.82|520|2|2
12.86|1.35|2.32|18|122|1.51|1.25|.21|.94|4.1|.76|1.29|630|3|3
```

Las dos últimas columnas verifican una correcta clasificación de estos tres vinos (clase 1, clase 2 y clase 3). Se probó el algoritmo SOM con un mapa de 3 neuronas, haciendo que las dimensiones del mapa fueran 3 neuronas de altura, y 1 sola de amplitud (H. height=3; W, width=1) mediante la instrucción:

```
SQL Minero>Select f1,f2,f3,f4,f5,f6,f7,f8,f9,
                f10,f11,f12,f13,clase,clasesom
                from tminero clustering with SOM -H 3 -W 1;
14.23|1.71|2.43|15.6|127|2.8|3.06|.28|2.29|5.64|1.04|3.92|1065|1|1
12.37|.94|1.36|10.6|88|1.98|.57|.28|.42|1.95|1.05|1.82|520|2|2
12.86|1.35|2.32|18|122|1.51|1.25|.21|.94|4.1|.76|1.29|630|3|
```

Que al igual que el MLP, las dos últimas columnas verifican una correcta clasificación de estos tres vinos (clase 1, clase 2 y clase 3). Los demás valores de los parámetros para MLP o SOM están inicializados con valores por defecto en Weka.

También se probó la siguiente consulta simple de SQL para verificar que se podía invocar SQL estándar:

```
SQL Minero> select * from tminero;
+-----+-----+-----+-----+ ... +-----+
| clase | f1    | f2    | f3    | ... | f13   |
+-----+-----+-----+-----+ ... +-----+
|      1 | 14.23 | 1.71  | 2.43  | ... | 1065  |
|      2 | 12.37 | 0.94  | 1.36  | ... | 520   |
|      3 | 12.86 | 1.35  | 2.32  | ... | 630   |
+-----+-----+-----+-----+ ... +-----+
3 rows in set (0.00 sec)
```

Esta sentencia se ejecutó satisfactoriamente.

5. Conclusiones y trabajo futuro

Es posible extender al lenguaje SQL estándar a través del patrón compositor para incluir funcionalidades de minería de datos, y obtener un lenguaje como el aquí propuesto llamado SQL Minero. Se pueden usar los algoritmos implementados en Weka y otros plug-ins en combinación con un manejador como

MySQL para implementarlo. La documentación y códigos fuente disponibles de Weka facilitaron la implementación de SQL Minero. Aunque por el momento únicamente se ha incluido un par de algoritmos basados en redes neuronales, MLP y SOM, es claro que se pueden incluir un mayor número de algoritmos de aprendizaje automático útiles en el proceso de minería de datos. Al extender el SQL estándar se promueve el desarrollo de aplicaciones de inteligencia de negocios que son una clara evolución de los sistemas de información en donde sólo se recuperaba información, y que ahora con minería de datos es posible extraer conocimiento intrínseco en los datos útil en la toma de decisiones que proveen una ventaja estratégica en las empresas. Como trabajo futuro se espera continuar con este proyecto para construir el SQL Minero inmerso en Java, lo cual facilitará la construcción de aplicaciones de inteligencia de negocios con ambientes gráficos y en arquitectura Web. Se contempla también incluir algoritmos tales como Máquinas de Soporte Vectorial en clasificación y regresión, y árboles de decisión para construir sistemas basados en reglas. y además se ha considerado la posibilidad de modificar el código fuente de algún manejador de código abierto para extender el SQL estándar a SQL Minero.

Referencias

1. Date, C.J.: *An Introduction to Database Systems*. 7 sub edn. Addison Wesley Longman (January 2000)
2. Gamma, E., Helm, R., Johnson, R., Vlissides, J.: *Design Patterns: Elements of Reusable Object-Oriented Software*. 1 edn. Addison-Wesley Professional (November 1994)
3. Java: <http://www.oracle.com/technetwork/java/index.html>. (September 2011)
4. Britos, P., et. al. et. al.: *Minería de datos basada en sistemas inteligentes*. 1 edn. Nueva Librería, Buenos Aires (2005)
5. Witten, I.H., Frank, E., Hall, M.A.: *Data Mining: Practical Machine Learning Tools and Techniques, Third Edition*. 3 edn. Morgan Kaufmann (January 2011)
6. Haykin, S.: *Neural Networks: A Comprehensive Foundation*. 2 edn. Prentice Hall (July 1998)
7. Kohonen, T.: *Self-Organizing Maps*. 3rd edn. Springer (December 2000)
8. Aho, A.V., Lam, M.S., Sethi, R., Ullman, J.D.: *Compilers: Principles, Techniques, and Tools*. 2 edn. Prentice Hall (September 2006)
9. Hall, M.A., Frank, E., Holmes, G., Pfahringer, B., Reutemann, P., Witten, I.H.: *The weka data mining software: An update*. SIGKDD Explorations 11(1) (2009)
10. Oracle: *Data mining*. <http://www.oracle.com/technetwork/database/enterprise-edition/odm-techniques-algorithms-097163.html> (September 2011)
11. IBM: *Db2 data warehouse edition data mining*. <http://www-01.ibm.com/software/data/iminer/> (September 2011)
12. Hopcroft, J.E., Motwani, R., Ullman, J.D.: *Introduction to Automata Theory, Languages, and Computation*. 3 edn. Addison Wesley (July 2006)
13. License, G.G.P.: <http://www.gnu.org/copyleft/gpl.html>
14. Algos: <http://weka.classalgos.sourceforge.net/>. (September 2011)
15. ARFF: <http://www.cs.waikato.ac.nz/ml/weka/arff.html>. (September 2011)
16. UCI: <http://archive.ics.uci.edu/ml/machine-learning-databases/wine/wine.names>. (September 2011)
17. MySQL: <http://www.mysql.com/>. (September 2011)

Clasificación Translingüe de Documentos usando Similitudes del Conjunto Objetivo¹

Escobar-Acevedo, Adelina^{1,2}, Montes-y-Gómez, Manuel²,
Villaseñor-Pineda, Luis², Guzmán-Cabrera Rafael³

¹Universidad Tecnológica del Valle de Toluca.

²Instituto Nacional de Astrofísica, Óptica y Electrónica
{aescobar, mmontesg, villasen}@inaoep.mx

³Universidad de Guanajuato, DICIS
guzmanc@ugto.mx

Resumen— La Clasificación Translingüe de Documentos (CTD) pretende aprovechar recursos existentes en un idioma para clasificar documentos en otro idioma. Adicionales a los efectos de la traducción, la CTD también enfrenta la discrepancia cultural entre idiomas que implica que documentos pertenecientes a la misma clase traten tópicos diferentes. Para determinar el impacto de la discrepancia cultural, se realiza un análisis de vocabulario y un análisis basado en gráficas de similitud en los conjuntos de entrenamiento en tres idiomas. Dado que en la CTD el conjunto de entrenamiento y el conjunto objetivo provienen de distribuciones diferentes, se aplica un método para aprovechar las similitudes existentes entre documentos del conjunto objetivo a fin de mejorar la exactitud de la clasificación.

1 Introducción

Actualmente se genera alrededor del mundo gran cantidad de información. Debido a la imposibilidad humana de manejar enormes cantidades de textos, es necesario automatizar procesos que manipulen y organicen tales volúmenes de documentos [1]. Existen distintas líneas de investigación como los sistemas de recuperación de información (Information Retrieval) [2], búsqueda de respuestas (Question Answering) [3]; generación automática de resúmenes (Text Summarization) [4], y clasificación de textos (Text Categorization) [5]; entre otros.

La clasificación automática de textos se define como la tarea de asignar documentos a clases predefinidas [5]. La clasificación automática de textos es una tarea supervisada, lo que significa que no sólo se conocen previamente las categorías sino que debe contarse con un conjunto de entrenamiento. La generación de los conjuntos de entrenamiento es un proceso costoso ya que usualmente es una tarea manual efectuada por un experto. Bajo el entorno multilingüe, la clasificación automática de textos multilingües plantea el problema de clasificar documentos escritos en diferentes idiomas bajo las mismas clases.

¹ El presente trabajo se realizó gracias al apoyo recibido por el PROMEP/103.5/11/4403

Algunas compañías e instituciones necesitan buscar y organizar eficientemente repositorios multilingües de documentos. El manejo de colecciones multilingües se convierte en un problema debido a que el proceso de etiquetado se multiplica para cada idioma que se desee clasificar. Ante ello, la CTD representa una solución viable ya que se define como la tarea de asignar clases a documentos escritos en un idioma objetivo usando recursos de un idioma fuente [6] y promete reducir el esfuerzo humano a proveer el conjunto de entrenamiento en solo un idioma [7].

La CTD presenta dos características particulares que la convierten en una tarea muy compleja. Por una parte la distorsión inevitable introducida por el traductor, y por otra, la discrepancia cultural entre idiomas. Recordemos que el idioma es el medio de expresión de un grupo cultural y socialmente homogéneo. Por ejemplo, si se deseara hacer una clasificación de notas periodísticas donde una clase sea deportes, es muy probable que los deportes en un idioma varíen considerablemente respecto a otro. Términos comunes en un grupo pueden ser escasos o inexistentes en otro; es de esperarse que los periódicos mexicanos aborden más sobre fútbol soccer que los americanos y a su vez deportes como el rugby, la pelota vasca y el cricket sean más nombrados en países europeos.

El presente trabajo está organizado de la siguiente manera. La sección 2 presenta el estado del arte mencionando los principales trabajos relacionados. En la sección 3 se describe el corpus de trabajo y hace un análisis relacionado a la solidez de los conjuntos de entrenamiento y prueba. La sección 4 muestra el método de refinamiento. En la sección 5 se presentan los resultados obtenidos; por último, la sección 6 presenta las conclusiones e ideas futuras.

2 Trabajo Relacionado

Existen trabajos que proponen realizar la clasificación multilingüe mediante la creación de clasificadores monolingües [8], [9], [10]. Bajo este enfoque, cada conjunto de prueba debe ser enviado a su respectivo clasificador, por lo que se construye un clasificador por cada idioma que se desee incluir. Otra propuesta consiste en realizar la clasificación por entrenamiento polilingüe [8], [10], [11], la cual consiste en crear un único clasificador para todos los idiomas con el objetivo de recibir documentos sin especificar de qué idioma se trata; cada idioma agregado de forma posterior implica la actualización del clasificador. Tanto las soluciones que proponen crear clasificadores monolingües como las soluciones por entrenamiento polilingüe, suponen la existencia de conjuntos etiquetados en todos los idiomas requeridos.

La CTD asume que el corpus etiquetado está disponible en un único idioma y debe ser usado para clasificar documentos de otros idiomas. Para franquear la barrera del idioma se han usado recursos como términos comunes [8], diccionarios [6], [12], tesauros [6], [13], y traducción automática [7], [8], [10]. El uso de traductores automáticos presenta la ventaja de estar disponibles para varios idiomas. A diferencia de los métodos que se centran principalmente en el problema de la traducción, en este artículo se propone abordar directamente la diferencia cultural.

La mayoría de los métodos calcula la probabilidad del documento de pertenecer a una clase utilizando únicamente los atributos del conjunto de entrenamiento, con ello,

Clasificación Translingüe de Documentos usando Similitudes del Conjunto Objetivo

Escobar-Acevedo, Adelina^{1,2}, Montes-y-Gómez, Manuel²,
Villaseñor-Pineda, Luis², Guzmán-Cabrera Rafael³

¹Universidad Tecnológica del Valle de Toluca.

²Instituto Nacional de Astrofísica, Óptica y Electrónica
{aescobar, mmontesg, villasen}@inaoep.mx

³Universidad de Guanajuato, DICIS
guzmanc@ugto.mx

Resumen— La Clasificación Translingüe de Documentos (CTD) pretende aprovechar recursos existentes en un idioma para clasificar documentos en otro idioma. Adicionales a los efectos de la traducción, la CTD también enfrenta la discrepancia cultural entre idiomas que implica que documentos pertenecientes a la misma clase traten tópicos diferentes. Para determinar el impacto de la discrepancia cultural, se realiza un análisis de vocabulario y un análisis basado en gráficas de similitud en los conjuntos de entrenamiento en tres idiomas. Dado que en la CTD el conjunto de entrenamiento y el conjunto objetivo provienen de distribuciones diferentes, se aplica un método para aprovechar las similitudes existentes entre documentos del conjunto objetivo a fin de mejorar la exactitud de la clasificación.

1 Introducción

Actualmente se genera alrededor del mundo gran cantidad de información. Debido a la imposibilidad humana de manejar enormes cantidades de textos, es necesario automatizar procesos que manipulen y organicen tales volúmenes de documentos [1]. Existen distintas líneas de investigación como los sistemas de recuperación de información (Information Retrieval) [2], búsqueda de respuestas (Question Answering) [3]; generación automática de resúmenes (Text Summarization) [4], y clasificación de textos (Text Categorization) [5]; entre otros.

La clasificación automática de textos se define como la tarea de asignar documentos a clases predefinidas [5]. La clasificación automática de textos es una tarea supervisada, lo que significa que no sólo se conocen previamente las categorías sino que debe contarse con un conjunto de entrenamiento. La generación de los conjuntos de entrenamiento es un proceso costoso ya que usualmente es una tarea manual efectuada por un experto. Bajo el entorno multilingüe, la clasificación automática de textos multilingües plantea el problema de clasificar documentos escritos en diferentes idiomas bajo las mismas clases.

¹ El presente trabajo se realizó gracias al apoyo recibido por el PROMEP/103.5/11/4403

Algunas compañías e instituciones necesitan buscar y organizar eficientemente repositorios multilingües de documentos. El manejo de colecciones multilingües se convierte en un problema debido a que el proceso de etiquetado se multiplica para cada idioma que se desee clasificar. Ante ello, la CTD representa una solución viable ya que se define como la tarea de asignar clases a documentos escritos en un idioma objetivo usando recursos de un idioma fuente [6] y promete reducir el esfuerzo humano a proveer el conjunto de entrenamiento en solo un idioma [7].

La CTD presenta dos características particulares que la convierten en una tarea muy compleja. Por una parte la distorsión inevitable introducida por el traductor, y por otra, la discrepancia cultural entre idiomas. Recordemos que el idioma es el medio de expresión de un grupo cultural y socialmente homogéneo. Por ejemplo, si se deseara hacer una clasificación de notas periodísticas donde una clase sea deportes, es muy probable que los deportes en un idioma varíen considerablemente respecto a otro. Términos comunes en un grupo pueden ser escasos o inexistentes en otro; es de esperarse que los periódicos mexicanos aborden más sobre fútbol soccer que los americanos y a su vez deportes como el rugby, la pelota vasca y el cricket sean más nombrados en países europeos.

El presente trabajo está organizado de la siguiente manera. La sección 2 presenta el estado del arte mencionando los principales trabajos relacionados. En la sección 3 se describe el corpus de trabajo y hace un análisis relacionado a la solidez de los conjuntos de entrenamiento y prueba. La sección 4 muestra el método de refinamiento. En la sección 5 se presentan los resultados obtenidos; por último, la sección 6 presenta las conclusiones e ideas futuras.

2 Trabajo Relacionado

Existen trabajos que proponen realizar la clasificación multilingüe mediante la creación de clasificadores monolingües [8], [9], [10]. Bajo este enfoque, cada conjunto de prueba debe ser enviado a su respectivo clasificador, por lo que se construye un clasificador por cada idioma que se desee incluir. Otra propuesta consiste en realizar la clasificación por entrenamiento polilingüe [8], [10], [11], la cual consiste en crear un único clasificador para todos los idiomas con el objetivo de recibir documentos sin especificar de qué idioma se trata; cada idioma agregado de forma posterior implica la actualización del clasificador. Tanto las soluciones que proponen crear clasificadores monolingües como las soluciones por entrenamiento polilingüe, suponen la existencia de conjuntos etiquetados en todos los idiomas requeridos.

La CTD asume que el corpus etiquetado está disponible en un único idioma y debe ser usado para clasificar documentos de otros idiomas. Para franquear la barrera del idioma se han usado recursos como términos comunes [8], diccionarios [6], [12], tesauros [6], [13], y traducción automática [7], [8], [10]. El uso de traductores automáticos presenta la ventaja de estar disponibles para varios idiomas. A diferencia de los métodos que se centran principalmente en el problema de la traducción, en este artículo se propone abordar directamente la diferencia cultural.

La mayoría de los métodos calcula la probabilidad del documento de pertenecer a una clase utilizando únicamente los atributos del conjunto de entrenamiento, con ello,

el resto de las propiedades del conjunto objetivo se ignoran. En contraste, el método de refinamiento incluye una segunda fase en la que cada documento es representado con atributos propios del conjunto objetivo y considera el voto de sus documentos similares.

3 Análisis del Corpus de Trabajo

Con el fin de observar los efectos de la discrepancia cultural en la CTD, se realizaron dos análisis sobre el corpus de trabajo. El primero consiste en las gráficas de similitud que permiten observar la solidez de las clases en cada conjunto de documentos y comparar los traslapes entre clases por idioma. El segundo análisis consiste en identificar el vocabulario común entre conjuntos de entrenamiento y prueba, ya que será éste quien determine la exactitud de la clasificación inicial.

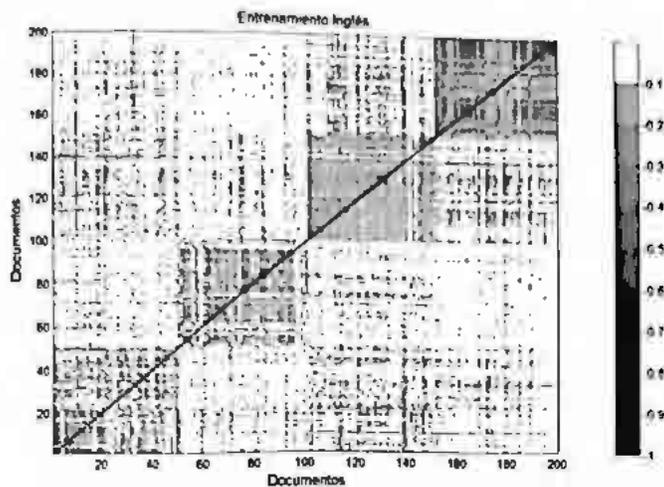
Se utilizó un subconjunto balanceado del corpus Reuters RCV-1 [14], considerando 3 idiomas: inglés, español y francés, con noticias de cuatro clases diferentes correspondientes a 'Policía', 'Desastres', 'Política' y 'Deportes'. Para cada idioma se utilizaron 200 noticias para entrenamiento y 120 como prueba, 50 y 30 por clase respectivamente. Para el pre-procesamiento de los documentos, se utilizaron listas de palabras vacías en los idiomas correspondientes. Finalmente, se utilizó el traductor automático Wordlingo para traducir los conjuntos de entrenamiento del idioma fuente al idioma objetivo.

3.1 Análisis por Gráficas de Similitud

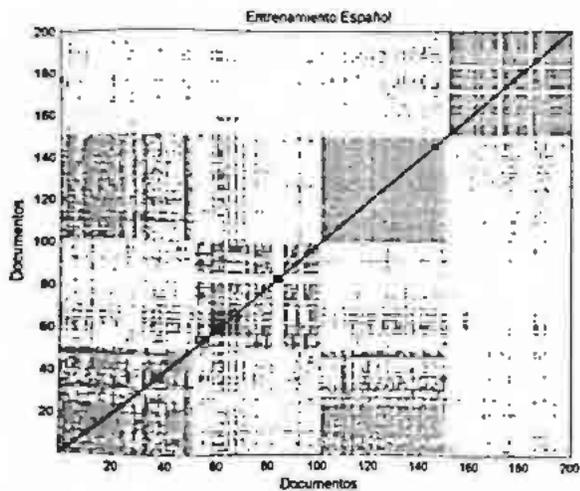
Dada la cantidad de documentos, es complicado para un humano hacer un análisis global del conjunto. Por ello se realizan gráficas de similitud que permiten ver qué tan bien están representadas las clases por sus vocabularios identificando traslapes y disimilitudes. Las gráficas se obtienen al representar vectorialmente cada documento mediante su vocabulario con pesado booleano, y compararlo con el resto de la colección mediante una medida de similitud como el Coeficiente de Dice.

En la Fig. 1, la escala de grises determina el grado de similitud de los documentos. Se muestra que la gráfica del conjunto en inglés (a) tiene clases definidas que no se confunden con otras clases. Cada documento perteneciente a una clase se parece a las de su misma clase y es diferente a los de otras clases en cuanto al vocabulario que utiliza. Particularmente la primera clase resulta problemática ya que es poco sólida, contiene varios documentos que no son similares entre sí o cuyo grado de similitud es bajo. El conjunto de español (b) presenta traslapes en la clase 1 y 3 pero la clase 4 es sólida. El conjunto de francés (c) muestra vocabulario similar en todas sus clases aun cuando las similitudes son bajas, entre 0.2 y 0.3, en este conjunto, la clase 3 es la más marcada por sus grados de similitud mayores.

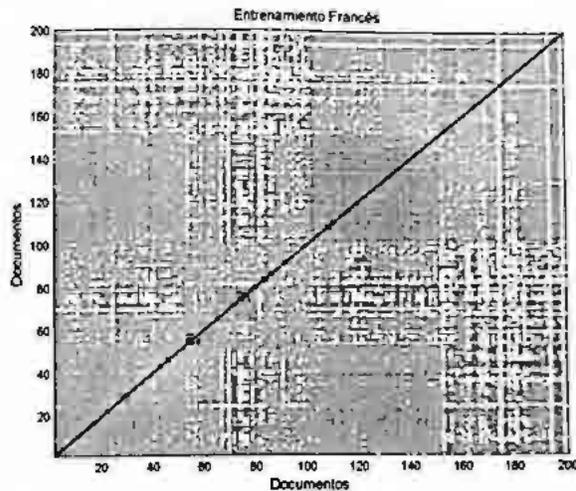
Los traslapes de clases indican que existe vocabulario utilizado comúnmente entre ellas y el clasificador puede confundirlas. Las clases poco sólidas al contrario, contienen documentos cuya representación es diferente al resto de los documentos de su clase [15]. Esta representación suele aportar escasa información al clasificador para asignar clases y a su vez al método propuesto por similitudes.



(a)



(b)



(c)

Figura 1. Gráficas de similitud de los conjuntos de entrenamiento (a) inglés, (b) español y (c) francés

3.2 Análisis por Vocabulario

Como menciona [16] “En general aprender es más confiable cuando los ejemplos siguen una distribución similar a los ejemplos de prueba futuros”. Aunque se espera que las distribuciones sean similares y compartan características, el conjunto de entrenamiento y prueba no son tomados de la misma distribución en la CTD. Incluso, en los conjuntos de prueba no existen muchas de las palabras que contienen los documentos de entrenamiento [8]. Para determinar el porcentaje de términos comunes se realizó una comparación simple.

Sea $T_c = \{t_{c1}, \dots, t_{cm}\}$ el conjunto de términos llamados vocabulario de entrenamiento, obtenidos de la traducción del conjunto de entrenamiento del idioma fuente al objetivo y sea $T_p = \{t_{p1}, \dots, t_{pm}\}$ el conjunto de términos llamados vocabulario de prueba en su idioma original; entonces el conjunto $T_{com} = T_c \cap T_p$ es el vocabulario común.

Tabla 1: Tamaño de Vocabulario para la CTD

Fuente	Objetivo	T_c	T_p	T_{com}
Inglés	Inglés	10,892	7,658	5,452
Español	Español	12,295	8,051	5,182
Francés	Francés	14,072	9,258	6,000
Inglés	Español	13453	8051	3640
Inglés	Francés	12426	9258	4131
Español	Inglés	9012	7658	3351
Español	Francés	10666	9258	3793
Francés	Inglés	11338	7658	3700
Francés	Español	14684	8051	3920

De la Tabla 1, se obtiene que el vocabulario común es aproximadamente 50% del vocabulario de entrenamiento en el caso monolingüe y de aproximadamente 31% en el caso translingüe. Esta caída en el vocabulario común afecta la exactitud de la CTD.

4 Método de Refinamiento de Clases

Bajo la premisa de que el conjunto de prueba es una muestra de la distribución que nos interesa captar, se espera que los documentos pertenecientes a la misma clase sean similares entre sí. Así, se utiliza un método de refinamiento de la clasificación [17], en el cual, adicional a la clasificación asignada se recibe la votación de los documentos con mayor similitud, ver Fig. 2.

El método se describe en el siguiente algoritmo:

1. Se construye un clasificador C_1 utilizando el conjunto de entrenamiento D_E en el idioma fuente L_1 .
2. Se clasifica cada documento $d_j \in D_p$ en el idioma objetivo L_2 a fin de asignar una clasificación inicial, la cual se representa como c_j^0 donde el superíndice indica el número de iteración en la que fue asignada la clase.
3. Se representa al conjunto objetivo D_p con su propio vocabulario $T_p = \{t_{p1}, \dots, t_{pm}\}$ y se enlistan los k vecinos más cercanos d_{m1}, \dots, d_{mk} de cada documento.

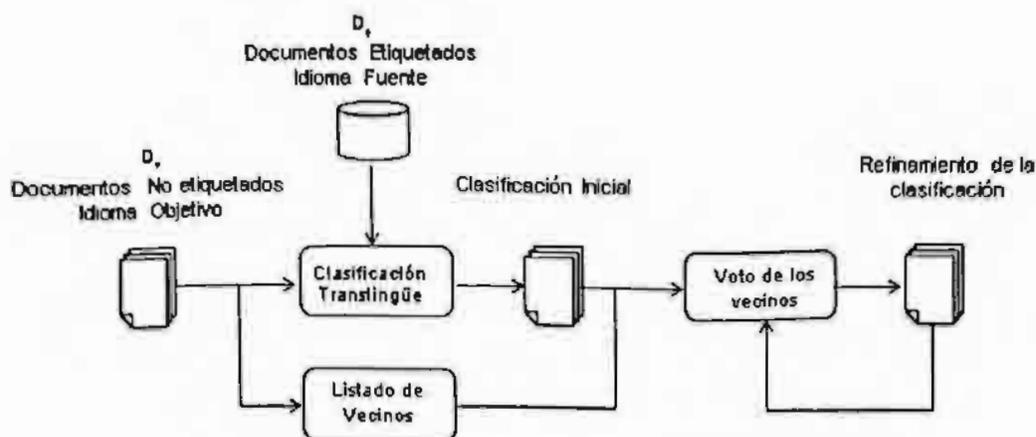


Figura 2. Método propuesto de refinamiento de CTD.

4. Sea c_j^n la clase asignada por el clasificador al documento d_j en la iteración n y $c_{n1}^n \dots c_{nk}^n$ las clases de sus k vecinos. Entonces $c_j^{n+1} = c_{nm}^n$ si $c_{n1}^n = c_{n2}^n = \dots = c_{nk}^n$, re-clasificando al documento con la clase de sus vecinos y $c_j^{n+1} = c_j^n$ si no se cumple que $c_{n1}^n = c_{n2}^n = \dots = c_{nk}^n$.
5. Se actualizan las etiquetas de cada documento.
6. El paso 4 y 5 se repiten iterativamente hasta que no existen cambios, es decir $\forall (d_j \in D_p): c_j^{n+1} = c_j^n$ o hasta un número fijo de iteraciones.

5 Resultados

La clasificación se realiza utilizando como método de aprendizaje Naïve Bayes y como medida de evaluación la exactitud, que indica el porcentaje de documentos que fueron asignados correctamente por el clasificador. Dado que se considera un consenso entre vecinos para refinar la clasificación, el método considera un mínimo de 3 vecinos. La Tabla 2 muestra los resultados de referencia monolingües. La Tabla 3 proporciona los resultados de la CTD sin intervención del método y al aplicarlo considerando el voto de sus 3, 4 y 5 documentos más similares.

Tabla 2. Resultados de Referencia Monolingües

Fuente	Objetivo	Exactitud
Inglés	Inglés	0.916
Español	Español	0.916
Francés	Francés	0.933

Tabla 3. Resultados del Método de Similitud

Fuente	Objetivo	Exactitud			
		Inicial	3	4	5
Inglés	Español	0.716	0.725	0.733	0.725
Inglés	Francés	0.758	0.775	0.766	0.766
Español	Inglés	0.816	0.900	0.900	0.883
Español	Francés	0.808	0.833	0.816	0.825
Francés	Inglés	0.858	0.958	0.925	0.925
Francés	Español	0.833	0.841	0.841	0.841

La comparación entre la exactitud de la clasificación monolingüe simple mostrada en la Tabla 2 y la exactitud de la clasificación translingüe simple mostrada en la tercera columna de la Tabla 3 hace evidente que existe una caída de exactitud al realizar CTD. La aplicación del método de refinamiento por similitudes reporta siempre un incremento en la exactitud entre el 1 y el 11.65% respecto a la clasificación inicial, llegando a ser comparables incluso con el caso monolingüe. Cabe mencionar que el número de iteraciones promedio se encuentra entre 3 y 4 antes de la convergencia del método.

6 Conclusiones

Los análisis realizados a los corpus nos permiten apreciar la dificultad y problemática de la CTD. Las gráficas de similitud permiten anticipar el desempeño que tendrá un sistema de clasificación automática de documentos en función de la separación de clases y traslapes que se observan. Esta información puede ser útil al momento de seleccionar los atributos que serán utilizados como entrenamiento para llevar a cabo la tarea. Particularmente los efectos de la discrepancia cultural tienen efectos relevantes en la clasificación debido a la reducción del vocabulario común con el que se representan los conjuntos de prueba. Sin embargo, aprovechar la propia distribución del conjunto objetivo es benéfico para la clasificación final, ya que provee información adicional para apoyar la decisión del clasificador facilitando la correcta asignación de clases. En general, aplicar el método invariablemente incrementó la exactitud inicial. Los mejores resultados llegan a ser comparables con la clasificación monolingüe. En un trabajo futuro se probarán métodos con incorporación de ejemplos con el fin de hacer una adaptación paulatina del clasificador al idioma objetivo.

Referencias

- [1] Galicia Haro S. N. y Gelbukh A. Investigaciones en análisis sintáctico para el español. Instituto Politécnico Nacional, México 2007.
- [2] Bolshakov J. and Gelbukh A. Computational Linguistics: models, resources, applications. México: Fondo de Cultura Económica, 2004.

- [3] Aceves Pérez R. M., Villaseñor-Pineda L., and Montes-y-Gómez M. "Using N-gram Models to Combine Query Translations in Cross-Language Question Answering". *International Conference on Intelligent Text Processing and Computational Linguistics CICLing-2007* (Springer) 4394 (2007): 485-493.
- [4] Villatoro-Tello E., Villaseñor-Pineda L., and Montes-y-Gómez M. "Using Word Sequences for Text Summarization". *LNCS (LNAI) 4188* (2006): 297.
- [5] Sebastiani F. "Machine Learning in Automated Text Categorization". *ACM computing Surveys* 34 (March 2002): 1-47.
- [6] Gliozzo A. and Strapparava C. "Exploiting Comparable Corpora and Biligual Dictionaries for Cross-Language Text Categorization". *Proceedings of the 21st International Conference on Computational Linguistics and 44th Annual Meeting of the ACL*. Sydney: Association for Computational Linguistics, 2006. 553-560.
- [7] Rigutini L., Maggini M., and Bing L. "An EM based training algorithm for Cross-Language Text Categorization". *Proceedings Web Intelligence Conference*. Compiègne, France: IEEE, 2005.
- [8] Bel N., Koster C., and Villegas M. "Cross-Lingual Text Categorization". *7th European Conference on Digital Libraries, ECDL*. Trondheim Norway, 2003. 126-139.
- [9] García Adeva J.J., Calvo R.A., and López de Ipiña D. "Multilingual Approaches to Text Categorisation". *The European Journal for the Informatics Professional* 6, no. 3 (2005): 43-51.
- [10] Jalam R. *Apprentissage automatique et catégorisation de textes multilingues*. Lyon: PhD Tesis, Université Lumière Lyon 2, 2003.
- [11] Wei C.P, Shi H., and Yang C.C. "Feature Reinforcement Approach to Polylingual Text Categorization". *10th International Conference on Asian Digital Libraries, ICADL 2007, Lecture Notes in Computer Science*, Springer, 2007. 99-108.
- [12] Olsson J.S., Oard D.W, and Hajic J. "Cross-Language Text Classification". *SIGIR*. Salvador, Brazil: ACM, 2005. 15-19.
- [13] De Melo G. and Siersdorfer S. "Multilingual Text Classification using Ontologies". *29th European Conference on IR Research, ECIR 2007, Lecture Notes in Computer Science*. Springer, 2007.
- [14] Lewis D.D., Yang Y., Rose T.G., and Li F. "RCV1: A New Benchmark Collection". *Journal of Machine Learning Research* 5 (2004): 361-397.
- [15] Manning C.D. and Schütze H. *Foundations of Statistical Natural Language*. The MIT Press, 2001.
- [16] Mitchell T.M. *Machine Learning*. McGraw-Hill, 1997.
- [17] Escobar-Acevedo A., Montes-y-Gómez M., and Villaseñor-Pineda L., "Using Nearest Neighbor Information to Improve Cross-Language Text Classification", *MICAI 2009*. Guanajuato, Mexico. *Lecture Notes in Artificial Intelligence* 5845, Springer.

SmartFUT:

Desarrollo de un simulador de futbol con aprendizaje por refuerzo y un clasificador de jugadores con SVM y redes neuronales

Oscar Franco Romero

Bárbara Verónica Buendía Hernández

Instituto Tecnológico de Monterrey Campus Estado de México
Atizapán de Zaragoza, Estado de México

Abstract

El aprendizaje por refuerzo es una rama del Aprendizaje Automático que se ha utilizado en una gran cantidad de aplicaciones, sobre todo para modelar fenómenos que pueden describirse mediante *cadena de Markov*, en donde la acción de un agente o individuo depende únicamente de la situación en la que se encuentre y no del camino que se ha realizado previamente.

En este artículo se presenta el desarrollo de un simulador de futbol denominado *SmartFUT* en el cual se hará uso del aprendizaje por refuerzo en los jugadores de ambos equipos, así como también los resultados de la simulación. Asimismo se presentará el trabajo y los resultados obtenidos de un clasificador de jugadores en dos clases distintas: su calidad como jugador y su desempeño en un partido. Para clasificar dichos jugadores se hará uso de las técnicas de *Máquinas de Soporte Vectorial (SVM)* y *Redes Neuronales*.

1. Introducción

El aprendizaje por refuerzo es una rama del Aprendizaje Automático inspirada en la filosofía conductista, cuya línea esencial de pensamiento es el estudio de las reacciones de un individuo dependientes del entorno en el que se encuentre [10]. Los experimentos de Iván Pávlov y su conocida ley del reflejo condicionado han sido fuente principal de influencia para

dicha corriente filosófica [9].

Este tipo de aprendizaje tiene como objetivo principal el uso de la idea de recompensa-castigo en donde un agente o individuo aprende cierta función, la cual le permitirá tomar una decisión sobre qué acción realizar en un instante en el tiempo, a partir de la información que este percibe de su entorno o del estado en que se encuentre. De ahí que reciba el nombre de *Aprendizaje por Refuerzo* [4].

Una condición importante para poder aplicar el aprendizaje por refuerzo es que el fenómeno se pueda modelar mediante *cadena de Markov*, ya que la acción a escoger por el individuo en una determinada situación depende únicamente de esta situación o estado y no del camino que se ha realizado para llegar a ella [6]. Asimismo, los elementos esenciales del aprendizaje por refuerzo son [8]:

1. Un conjunto de estados del entorno S .
2. Un conjunto de acciones A .
3. *Reglas de transición* entre los diferentes estados y acciones.
4. La *función de evaluación* que contiene la información a obtener del entorno a partir de cada acción posible.
5. La *función de recompensa-castigo* que define las ganancias obtenidas en base al objetivo a alcanzar.

En este artículo se presentará el desarrollo de un simulador de fútbol denominado *SmartFUT* en el cual se hará uso del aprendizaje por refuerzo, en donde cada jugador de dos equipos distintos actúa como un individuo que percibe información de su entorno, determina la acción que mayor ganancia le genera y así aprende en base a una función definida de recompensa-castigo.

Así también se mostrará el trabajo realizado de un clasificador de jugadores, con base en atributos definidos que estos poseen, en dos clases o vertientes: su calidad como jugador y su desempeño en un partido en particular. Para clasificar dichos jugadores se hará uso de las técnicas de *Máquinas de Soporte Vectorial (SVM)* y *Redes Neuronales*.

El simulador de fútbol, así como el generador de jugadores para el clasificador, fueron realizados en el lenguaje de programación *Java* en el entorno de desarrollo de *NetBeans*. Las técnicas de clasificación antes mencionadas fueron implementadas en el software *Matlab*.

En la sección 2 se mencionarán algunos trabajos relacionados con el aprendizaje por refuerzo. En la sección 3 se explicará a detalle la definición del problema y el funcionamiento tanto del simulador como del clasificador realizados. En la sección 4 se mostrarán los resultados obtenidos y en la sección 5 se darán a conocer las conclusiones y el trabajo futuro.

2. Trabajos relacionados

En la revista *Journal of Quantitative Analysis in Sports: Vol. 5: Iss. 1, Article 7* publicada en 2009 se examinó el uso de redes neuronales como una herramienta para predecir el éxito de los equipos de basketball en la NBA (National Basketball Association). Se recolectaron estadísticas de 620 juegos de la NBA y fueron utilizados para entrenar una variedad de redes neuronales como las de regresión generalizada. Los resultados obtenidos de las redes neuronales son comparados con predicciones hechas por numerosos expertos en el campo del basketball. Las mejores redes son capaces de corregir la predicción del equipo ganador 74.33% del tiempo (en promedio).

Otra aplicación es la selección de equipos para com-

petencias deportivas internacionales, ya que esto requiere predecir el desempeño de los atletas. Se ha explorado en el uso de redes neuronales para clasificar jugadores y seleccionar jugadores específicos para una competencia. Un ejemplo es el *cricket*, el cual es un juego muy reconocido en la Gran Bretaña así como en otros países de Europa. En este caso se utilizan redes neuronales para predecir el desempeño de los jugadores de *cricket* en el futuro basado en su desempeño en el pasado.

Se han clasificado a los jugadores de *cricket* en tres categorías: ejecutante bueno, regular y que tiende al fracaso. Se recolectaron datos del desempeño de los jugadores de 1985 a 2006-2007. El modelo de red neuronal fue entrenada progresivamente y probada usando cuatro conjuntos de datos. De aquí se obtuvieron los mejores desempeños y con esta base se formó el equipo que iría a la Copa Mundial del 2007.

En la Universidad Nacional de Irlanda, se desarrolló una tesis llamada *The application of Reinforcement Learning to the Simple Soccer Game*. En dicha tesis se pretende mostrar que el desempeño del jugador siempre es relativo al oponente así que mostraron que se puede medir la capacidad de un equipo de fútbol soccer a través de la diferencia en sus tasas de puntuación, esto en relación a su oponente. Estas tasas se calculan en base a la anotación y concesión de goles, considerándolas como una habilidad combinada llamada *ganar* [2].

También se explica que el fútbol es una área de investigación de la inteligencia artificial y que incluso existen concursos como *RoboCup*, el cual simula juegos de soccer y que se utiliza a menudo como punto de referencia para la simulación multi-agente y la investigación. Se describe tal cual como una herramienta de investigación y educación para sistemas multi-agentes y de inteligencia artificial.

De la aplicación anterior se desprende el proyecto llamado *RoboCup*. En esta aplicación se describe cómo un equipo llamado 'Los guardianes' tratan de mantener el control de la pelota el mayor tiempo posible a pesar del esfuerzo del equipo de 'Los tomadores'. 'Los guardianes' aprenden individualmente cuándo mantener la pelota y cuándo pasar a un compañero de su equipo. El aprendizaje se realiza por refuerzo utilizando políticas de premio y castigo

principalmente.

También se utiliza para la toma de decisiones y para la implementación de eventos estocásticos. Está orientado para la toma de decisiones rápidas en lugar de esperar a largos procesos de deliberación. *RoboCup* ha sido utilizado como la base de éxito en competencias internacionales y para retos de investigación.

3. Definición y desarrollo de los escenarios

3.1. SmartFUT: Simulador de fútbol

El simulador de fútbol *SmartFUT*, como ya se mencionó previamente, fue realizado en *Java* en el entorno de desarrollo de *NetBeans*. En primera instancia, se definieron las condiciones iniciales y el escenario completo sobre el cual trabajará el simulador, que son básicamente la alineación de los equipos (es decir, la cantidad de jugadores por cada zona del campo), la estructura de los jugadores y algunas restricciones o limitaciones a tomar en cuenta.

La alineación o formación para ambos equipos será invariablemente la 4-4-2, es decir, 4 defensas, 4 mediocampistas y 2 delanteros, más el portero. Cada jugador poseerá tres habilidades o atributos que se definen así:

- *Recuperación*: Habilidad para recuperar el balón cuando este se encuentra en posesión del rival.
- *Pases*: Habilidad para poder trasladar el balón exitosamente hacia un compañero del mismo equipo en cualquier zona del campo.
- *Disparo*: Habilidad para disparar a la portería rival y concretar un gol a favor del equipo.

En el caso del portero y los defensas, la habilidad de *Recuperación* será la que mayor nivel presenten, es decir, su habilidad predominante. Para el caso de los jugadores de mediocampo la habilidad predominante será la de *Pases* y para los delanteros será la de *Disparo*. Estos valores se inicializan de manera aleatoria al inicio del juego en base a estas orientaciones descritas.

Algunas de las restricciones impuestas en el simulador es el caso de la primer jugada, en donde se entrega de forma aleatoria a uno de los cuatro mediocampistas que se encuentran más cerca del círculo central. También cuando el balón llegue a estar en posesión del portero (cuando este haya detenido el disparo a gol de un jugador rival), su única acción posterior a realizar será el pase a uno de sus defensas.

Teniendo definidos tanto el escenario como las condiciones iniciales bajo las cuales trabajará *SmartFUT*, retomamos los conceptos básicos del aprendizaje por refuerzo, así como los elementos con los que debe contar, de tal suerte que se definieron los siguientes conjuntos de estados (*S*) y acciones (*A*) por los cuales se conforma el escenario del juego.

▪ Estados

- S_0 : Zona de defensa, equipo local.
- S_1 : Zona de mediocampo, equipo local.
- S_2 : Zona de delantera, equipo local.
- S_3 : Gol, equipo local.
- S_4 : Portería, equipo local.
- S_5 : Zona de defensa, equipo visitante.
- S_6 : Zona de mediocampo, equipo visitante.
- S_7 : Zona de delantera, equipo visitante.
- S_8 : Gol, equipo visitante.
- S_9 : Portería, equipo visitante.

▪ Acciones

- A_1 : Pase a un jugador de la defensa.
- A_2 : Pase a un jugador del mediocampo.
- A_3 : Pase a un jugador de la delantera.
- A_4 : Disparo a portería.

Al iniciar el juego, y una vez se ha efectuado la primer jugada, el jugador que posee el balón tiene que tomar una decisión sobre cuál de las cuatro acciones existentes realizar. Para determinar tal decisión, el jugador debe obtener información de su entorno a través de la *función de evaluación*. Esta función se compone de tres elementos: un valor correspondiente a la *Matriz de Ganancias* o de *Conveniencia*, un valor

correspondiente a la *Matriz de Proximidad* y el nivel de la habilidad o atributo del mismo jugador.

La *Matriz de Ganancias* o de *Conveniencia* es una matriz que ilustra con valores de 0 a 1 la ganancia que representa dicha acción dependiendo de la posición a la que pertenece el jugador que posee el balón.

	Defensa (G ₁)	Medio (G ₂)	Delantero (G ₃)	Gol (G ₄)
Defensa	0.2	0.5	0.8	1
Medio	0.1	0.45	0.85	1
Delantero	0.05	0.4	0.7	1

Por otro lado, la *Matriz de Proximidad* ilustra de igual forma con valores de 0 a 1 la cercanía que tiene el jugador con respecto a cualquier zona del campo.

	Defensa (P ₁)	Medio (P ₂)	Delantero (P ₃)	Gol (P ₄)
Defensa	1	0.8	0.3	0.05
Medio	0.8	1	0.6	0.25
Delantero	0.3	0.6	1	0.8

El nivel de la habilidad a tomar en cuenta depende de la acción en sí, ya que para el cálculo del valor de las acciones A₁, A₂ y A₃ se tomará la habilidad de *Pases*, mientras que para la acción A₄ se tomará la habilidad de *Disparo*. Es así que la *función de evaluación* queda determinada de la siguiente manera:

$$V_n = G_n * P_n * H(\text{Pases}) \rightarrow \forall n \in N = [1, 3] \quad (1)$$

$$V_n = G_n * P_n * H(\text{Disparo}) \rightarrow n = 4 \quad (2)$$

Una vez que el jugador ha obtenido los valores de cada una de las acciones posibles, elige aquél que le representa la mayor ganancia y la ejecuta. Acto seguido, *SmartFUT* pone en juego la acción elegida por el jugador, es decir, una vez el balón se pone en juego, se va a determinar si la acción elegida termina en *éxito* o *fracaso*. Estos son los dos eventos posibles a ocurrir en esta fase del juego.

El *Evento de éxito* es la acción misma determinada por el jugador. Por otro lado, el *Evento de fracaso* es el evento que determina la zona del campo del equipo rival al cual se dirigirá el balón una vez que la acción

elegida por el jugador no pudo llevarse a cabo. Esta zona del campo del equipo rival se determina dependiendo de la zona del campo a la cual pertenece el jugador que eligió previamente la acción. Estos eventos se muestran en las siguientes tablas:

Defensa	Evento éxito	Evento fracaso
A ₁	Pase a defensa	Delantero rival
A ₂	Pase a medio	Delantero rival
A ₃	Pase a delantero	Medio rival
A ₄	Gol	Portero rival

Medio	Evento éxito	Evento fracaso
A ₁	Pase a defensa	Delantero rival
A ₂	Pase a medio	Medio rival
A ₃	Pase a delantero	Medio rival
A ₄	Gol	Portero rival

Delantero	Evento éxito	Evento fracaso
A ₁	Pase a defensa	Medio rival
A ₂	Pase a medio	Medio rival
A ₃	Pase a delantero	Defensa rival
A ₄	Gol	Portero rival

En este punto del juego, se calculan las probabilidades de ambos eventos con base en las siguientes fórmulas, donde JO es el jugador origen, ZD es la zona destino del equipo al que pertenece el jugador origen, ZR es la zona del campo rival y PR es el portero rival:

Cuando el *Evento de éxito* es alguna de las acciones A₁, A₂ o A₃:

$$P_1 = H(\text{Pases})JO * H(\text{Recuperacion})ZD \quad (3)$$

$$P_2 = H(\text{Recuperacion})ZR \quad (4)$$

Cuando el *Evento de éxito* es la acción A₄:

$$P_1 = H(\text{Disparo})JO \quad (5)$$

$$P_2 = H(\text{Recuperacion})PR \quad (6)$$

Al término de estos cálculos, se realiza una especie de *ruleta* en donde se distribuyen dichas probabilidades y mediante un número aleatorio se determina cuál de los dos eventos ocurrirá.

Por último, se pone en marcha el motor del *Aprendizaje por refuerzo* que es precisamente la *función de recompensa-castigo* para todos los jugadores involucrados según el evento ocurrido. Los valores de dicha función para los jugadores involucrados se muestran a continuación, donde O es el jugador origen, R es el jugador del equipo rival, POR es el portero, DEF es un jugador de defensa, MED es un jugador de medio-campo y ATK es un jugador de la delantera:

Si el *Evento de éxito* es el elegido:

	POR O	DEF O	MED O	ATK O
A ₁	-	1%	0.8%	0.4%
A ₂	-	2%	1%	0.8%
A ₃	-	5%	2%	1%
A ₄	-	30%	15%	5%

	POR R	DEF R	MED R	ATK R
A ₁	-	-	-1%	-0.5%
A ₂	-	-	-1%	-0.5%
A ₃	-	-1%	-0.5%	-
A ₄	-1%	-	-	-

Si el *Evento de fracaso* es el elegido:

	POR O	DEF O	MED O	ATK O
A ₁	-	-1%	-0.8%	-0.4%
A ₂	-	-2%	-1%	-0.8%
A ₃	-	-5%	-2%	-1%
A ₄	-	-2%	-5%	-10%

	POR R	DEF R	MED R	ATK R
A ₁	-	-	1%	0.5%
A ₂	-	-	1%	0.5%
A ₃	-	1%	0.5%	-
A ₄	5%	-	-	-

Los valores presentados anteriormente van orientados hacia una habilidad en específico, dependiendo del evento y la acción ocurrida. Por ejemplo, si la acción elegida por un jugador fue hacer un pase, su habilidad en juego es la de *Pases*, si fue la de disparar a la portería entonces será la de *Disparo*, y si el rival es quien tiene el balón, su habilidad en juego será la de *Recuperación*. Es así que con este método los jugadores intentarán mejorar sus propias habilidades en beneficio del equipo y así buscar meter la mayor cantidad de goles posible.

3.2. Clasificador de jugadores

El clasificador de jugadores, como se mencionó en la introducción, fue realizado en el lenguaje *Java* en el entorno de desarrollo de *NetBeans*.

Este clasificador crea una cantidad específica de jugadores con valores de atributos inicializados de la misma manera que en el simulador, con la diferencia de que se le agrega un valor más por cada habilidad o atributo que define el *desempeño* del jugador después de un partido. Esta medida de *desempeño* se obtiene restando el valor inicial y el valor final de una habilidad en particular.

Es así que, con base en los atributos que cada jugador posee (*Recuperación*, *Pases* y *Disparo*), se clasificará a los jugadores en dos clases o vertientes: su *habilidad* individual y su *desempeño* en un partido. Para realizar esta clasificación se hará uso de las técnicas de *Máquinas de Soporte Vectorial (SVM)* y *Redes Neuronales* con ayuda del software *Matlab*. En la técnica de SVM se especificó el uso de 2 *Kernels*, y en el caso de la red neuronal se definió la cantidad de 100 épocas para entrenar dicha red.

4. Resultados

4.1. SmartFUT: Simulador de fútbol

En la interfaz diseñada se aprecian el promedio de las habilidades tanto por equipo como por zona del campo, permitiendo apreciar de manera visual qué tan buen o mal equipo es cada uno. Asimismo, hay una sección de la interfaz en donde se despliega el nivel de las habilidades del jugador que posee el balón en ese momento. Finalmente, también se despliega la información sobre la decisión tomada por el jugador y el evento ocurrido tras su elección.

Se pusieron en marcha 10 partidos de fútbol, donde en ambos equipos se inicializaron sus jugadores de la manera descrita en la sección anterior. Tras el término de estos partidos, se pudieron apreciar las siguientes observaciones:

1. El equipo que en un inicio tenía en promedio un mejor nivel de *Pases*, terminó ganando en un 70% de las ocasiones.

2. El equipo que en un inicio tenía en promedio un mejor nivel de *Pases* en sus mediocampistas, logró acertar un pase a otro jugador de su mismo equipo en un 65 % de las ocasiones.
3. El equipo que en un inicio tenía en promedio un mejor nivel de *Disparo* en sus delanteros, logró anotar en un 80 % aproximadamente todas las ocasiones de gol que tuvieron al momento de elegir como acción más valiosa el disparo a la portería.
4. El equipo que en un inicio tenía al mejor portero, terminó ganando en un 50 % de las ocasiones.
5. El equipo que en un inicio tenía en promedio un mejor nivel de *Recuperación*, logró ganar el partido en un 50 % de las ocasiones.
6. El equipo que al final de un partido tenía en promedio un mejor nivel de *Pases*, terminó ganando en un 80 % de las ocasiones.
7. El equipo que al final de un partido tenía en promedio un mejor portero, terminó ganando en un 70 % de las ocasiones.
8. La acción de disparar a portería fue tomada en un 100 % de las ocasiones por los delanteros.

Es en base a estas observaciones que podemos determinar que el aprendizaje que reciben los jugadores durante todo el partido, resulta cada vez más crucial conforme va avanzando, ya que mientras más recompensa reciban, mayor será la probabilidad de que el evento sea favorable, determinando así en gran medida el resultado del partido.

También se aprecia claramente la gran importancia que tiene la habilidad de *Pases*, debido a que esta es la que determina en gran medida si la acción elegida tendrá una muy buena probabilidad de ocurrir. La zona tanto del mediocampo como la del portero, son las más importantes dada la influencia que tuvieron en los partidos simulados para ayudar a ganar a su respectivo equipo, a pesar de que fueron los delanteros quienes realizaron todos los disparos a portería.

4.2. Clasificador de jugadores

En el clasificador de jugadores se crearon cuatro bases de datos con la información de 100, 250, 350 y 500 jugadores respectivamente. Los resultados para calificar a los jugadores como 'buenos' o 'malos', mediante las técnicas mencionadas previamente, se muestran a continuación:

Calidad del jugador:

Instancias	% error SVM	% error RN
100	0 %	34.01 %
250	0 %	28.00 %
350	0 %	35.43 %
500	0 %	35.00 %

Desempeño del jugador:

Instancias	% error SVM	% error RN
100	0 %	48.00 %
250	0 %	23.60 %
350	0 %	17.43 %
500	0 %	13.00 %

En los resultados obtenidos se puede apreciar claramente la precisión y eficacia que tiene la SVM con respecto a la red neuronal, ya que en todos los casos el porcentaje de error de aprendizaje fue de un 0 % en el primero, algo que nunca sucedió con el otro.

Cabe mencionar que en el caso de la red neuronal, en cuanto a la clasificación de los jugadores en base a su desempeño en un partido, mientras mayor fue la cantidad de instancias a clasificar, fue mejorando su porcentaje de error en el aprendizaje. Sin embargo, la SVM resultó ser la mejor herramienta por mucho para clasificar a los jugadores en ambas categorías.

5. Conclusiones y trabajo futuro

Modelar un partido de fútbol con todos los factores que por la naturaleza de este deporte implica es sumamente complicado, pero el haber realizado este modelo sumamente simple, basado en el método de *Aprendizaje por refuerzo*, dio como resultado una simulación que puede describir de forma muy básica pero bastante aceptable y visible la forma en cómo

un jugador elige la mejor acción en determinado momento del partido y en una zona del campo en particular, y también del aprendizaje que estos obtienen dependiendo de si tienen éxito o no sus decisiones tomadas, viéndose así reflejado en el desempeño final que obtuvieron en un partido.

En la parte del clasificador, fue indudable la superioridad en cuanto a precisión y eficacia de la técnica de SVM con respecto a la red neuronal, siendo de un 0% en el porcentaje de errores cometidos en la clasificación en las dos categorías definidas.

En cuanto a trabajo futuro, se pueden mejorar tanto la *función de evaluación* como la *función de recompensa-castigo*, dándole un peso específico a cada una de las variables que forman parte de cada uno, así como darle una mayor variedad de eventos posibles a ocurrir, para hacerlo más realista.

Asimismo se puede hacer una fusión con el clasificador desarrollado para así permitir una predicción de cuál equipo puede llevarse el triunfo, en base a las habilidades y al posible desempeño que pudiera tener en el partido. El aprendizaje por refuerzo es un método muy prometedor que se ajusta aceptablemente en el desarrollo de simuladores para este deporte en particular.

Referencias

- [1] José Carlos Cortizo and Luis Ignacio Díaz. Aprendizaje por refuerzo. *Madrimasid: Un lugar para la ciencia y la tecnología*, 2008.
- [2] Eamon Costello. The application of reinforcement learning to the simple soccer game, 2007.
- [3] N. Cristianini and J. Shawe-Taylor. *An introduction to Support Vector Machines: and other kernel-based learning methods*. Cambridge University Press, 2000.
- [4] Carlos Diuk. Aprendizaje por refuerzos. Technical report, Department of Psychology, Princeton Neuroscience Institute.
- [5] Leslie Kaelbling, Michael Littman, and Andrew Moore. Reinforcement learning: A survey. *Journal of Artificial Intelligence Research*, 1996.
- [6] José M. Sempere. Aprendizaje por refuerzo. Technical report, Departamento de Sistemas Informáticos y Computación, Universidad Politécnica de Valencia.
- [7] Richard Sutton. Learning to predict by the method of temporal differences. *Machine Learning (Springer)*, 1988.
- [8] Richard Sutton and Andrew Barto. *Reinforcement Learning: An Introduction*. MIT Press, 1998.
- [9] Wikipedia. Iván Pávlov. <http://en.wikipedia.org/wiki/Ivan-Pavlov>.
- [10] Wikipedia. Reinforcement Learning. <http://en.wikipedia.org/wiki/Reinforcement-learning>.
- [11] Zhao Zhenning and Xu Yongmao. Introduction to fuzzy theory and neural networks and their application. *Beijing: Tsinghua University Press*, 1996.

Reconocimiento de Actividades Mediante el Acelerómetro de un Celular

Enrique García, Ramón Brena,

ITESM, Campus Monterrey, Av. Eugenio Garza Sada 2501 Sur, Monterrey, N.L., México
{a00927248, ramon.brena}@itesm.mx

Resumen. Este trabajo presenta la realización de un clasificador de actividades físicas de un usuario utilizando el acelerómetro de tres ejes de un teléfono celular. Las actividades a clasificar fueron: caminar, correr, subir escaleras, bajar escaleras y permanecer en reposo. El método de aprendizaje automático utilizado para la clasificación fue el de los *k-vecinos más cercanos*. A pesar de que este método es muy simple, se alcanzó una precisión del 89.33% en el total de las clasificaciones. De igual forma también se comparó la precisión de la clasificación de actividades con los algoritmos *C4.5* y *Naive Bayes*.

Palabras clave: detección de actividades físicas, *k-vecinos más cercanos*, acelerómetro, teléfono celular

1 Introducción

Reconocer las actividades que está realizando una persona [1] es una parte importante de la inteligencia ambiental [2]. Saber el estado de una persona puede proporcionar información muy útil a otros sistemas para saber cómo actuar en determinada situación. Por ejemplo, para el cuidado de la salud se puede utilizar la detección de caídas para alertar al personal médico en caso de accidentes; en seguridad, se pueden detectar comportamientos anómalos en masa que pueden ser indicador de que hay una situación peligrosa.

Hay muchas fuentes posibles de información sobre la actividad de un usuario, tales como la hora del día, los registros históricos, la detección de sus movimientos, etc. Muchos de los métodos de reconocimiento de actividades utilizan diversos tipos de sensores, ya sea un solo sensor o colecciones de sensores en interacción. En particular, en este trabajo estamos utilizando sensores de movimiento mediante aceleración, o acelerómetros [3], los que pueden ser sensores dedicados o ser parte de otro dispositivo, como un teléfono celular ("smartphone").

Dado que el uso de teléfonos celulares se ha ido incrementando y las capacidades de estos (velocidad, memoria, número de sensores, etc.) también han ido en aumento, es posible utilizarlos para obtener información del usuario así como también de su entorno. En este trabajo se busca aprovechar la capacidad de los teléfonos celulares para hacer la detección de actividades y de este modo liberar al usuario de la necesidad de portar más sensores y del cableado que ello implica.

La estructura de este escrito es como sigue: después de esta introducción, en la sección 2 discutimos otros trabajos relacionados o similares, en la sección 3 describimos el método utilizado, tras el cual presentamos los experimentos en la sección 4 y finalmente cerramos con una conclusión.

2 Trabajos Relacionados

Brezmes, Gorricho y Cotrina [4] presentan una implementación de reconocimiento de actividades mediante acelerómetros de un teléfono celular usando el método de los *k-vecinos más cercanos* para las actividades de caminar, subir/bajar escaleras, sentarse, pararse y caídas, que es desde luego parecido al presente trabajo. Sin embargo, la descripción de su método experimental es tan poco detallada, que resulta imposible replicar sus experimentos sin hacer grandes suposiciones, lo que mina grandemente el valor de sus resultados.

Karantonis, *et al.* [5] presentan la implementación de un clasificador de actividades en tiempo real capaz de calcular el gasto de energía metabólica usando una unidad de sensor. Tian, *et al.* [6] utilizan la información del GPS para limitar el conjunto de actividades registradas por el acelerómetro y enviar automáticamente mensajes con el estado de la persona a sitios de micro-blog como twitter. Seon-Woo y Mase [7] presentan un método basado en *odometría* para determinar la ubicación relativa en interiores de una persona utilizando diferentes sensores como acelerómetro, compás digital y sensor de velocidad angular. Una parte importante de éste último es la detección del estado *caminando* ya que encontraron que la principal fuente de error en la localización era una mala clasificación de las actividades.

Mannini y Sabatini [8] utilizaron la información de 5 acelerómetros biaxiales localizados en la cadera, muñeca, brazo, tobillo y en el muslo para la clasificación de 7 actividades que incluyen: sentado, parado, acostado, caminar, subir escaleras, correr y andar en bicicleta. Ellos reportan precisiones de entre 93% y 98.5% utilizando diferentes clasificadores. Nuestro trabajo difiere en que nosotros nos concentramos en el uso de un solo acelerómetro que forma parte de un teléfono celular ya que como se menciona en la sección 1 esto libera al usuario de la necesidad de tener instrumentos adheridos en su cuerpo.

3 Método

En esta sección se describe el método de recolección de datos, la extracción de atributos, el entrenamiento y finalmente la clasificación.

3.1 Recolección de Datos

Las actividades a clasificar fueron 5: caminar, correr, subir escaleras, bajar escaleras y permanecer en reposo. Se hicieron sesiones de entrenamiento que constaron de un minuto para cada actividad, tiempo durante el cual se recolectan los datos del

acelerómetro por medio de una aplicación corriendo localmente en el teléfono celular. Cada minuto de muestreo se tomaron 2400 lecturas, poniendo a funcionar la aplicación. Se escribió una aplicación en Android 2.2 [9] para la recolección y etiquetado de los datos de entrenamiento; en la Figura 1 se muestra la interfaz de dicha aplicación.

Para las actividades que involucran escaleras se tomaron 10 muestras de 6 segundos cada una en lugar de una sola de 1 minuto (es difícil encontrar escaleras tan largas como para estar subiendo durante un minuto sin detenerse) para completar una sesión de entrenamiento.



Fig. 1. Aplicación para obtener los datos de entrenamiento.

3.2 Extracción de Atributos y Entrenamiento

Se utilizó un aprendizaje basado en instancias [10] usando el método de los *k-vecinos más cercanos* (KNN, por sus siglas en inglés). Cada instancia consta de la información recolectada durante 2 segundos (80 lecturas de cada eje). Se decidió hacerlo de 2 segundos por que en un segundo todavía no hay información suficiente como para saber de qué actividad se trata. A las 80 lecturas tomadas durante 2 segundos se les extrajeron 9 atributos en total. Estos atributos comprenden la media y desviación estándar para cada eje (x, y, z), y la correlación para cada par de ejes (xy, xz, yz). Se decidió utilizar estos atributos ya que Ravi, *et al.* [11] concluyen que con ellos se pueden obtener buenos resultados. Kwapisz, Weiss, y Moore [12] utilizan otros atributos como la diferencia promedio absoluta, la aceleración promedio resultante, el tiempo en milisegundos entre las cimas de las curvas sinusoidales y la distribución encajonada. Durante un minuto de entrenamiento se generan 30 instancias por cada actividad. Para el caso de los *k-vecinos más cercanos* una vez que se generan estas instancias se dice que el algoritmo ya está entrenado. La Figura 2 muestra la gráfica para cada una de las actividades durante tres segundos así como sus respectivos valores tomados directamente del acelerómetro para cada eje. Como se puede ver, los valores para el eje y generalmente son mayores. Esto es porque cuando el teléfono celular está en posición vertical y en reposo, el acelerómetro marca valores aproximados a 9.81.

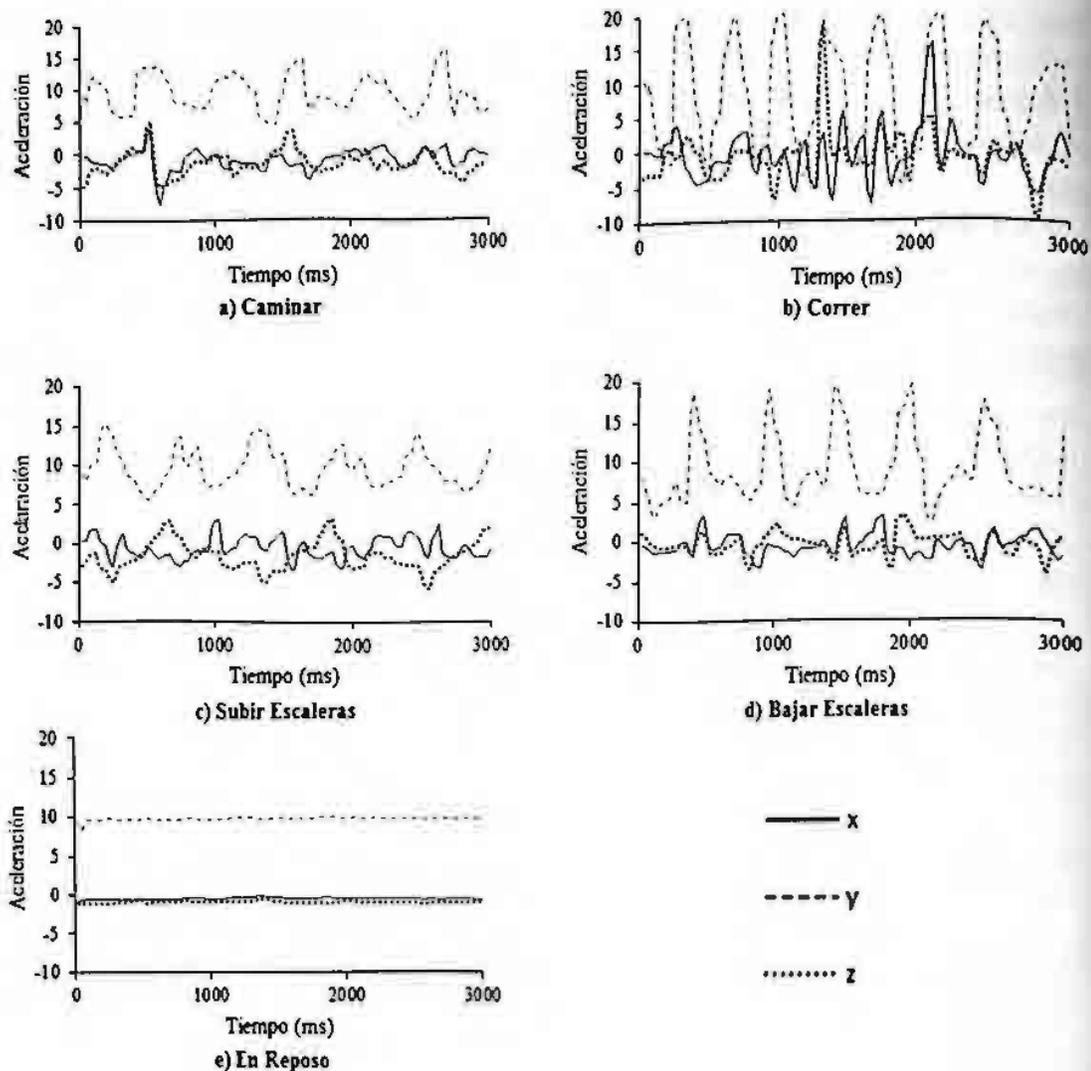


Fig. 2. Gráficas de aceleración para las cinco actividades.

3.3 Clasificación

Para este trabajo se decidió hacer la implementación del prototipo en el teléfono celular con el algoritmo de los *k-vecinos más cercanos* ya que al ser de tipo *lazy* (no requiere de hacer cálculos antes de la clasificación) permite agregar nuevos datos de entrenamiento en tiempo de ejecución sin la necesidad de volver a realizar cálculos costosos. También hicimos una comparación con los algoritmos C4.5 y Naive Bayes, que presentamos adelante. El algoritmo C4.5 genera un árbol de decisión que es el que se utiliza para hacer la clasificación y fue desarrollado por Quinlan [13]. Naive Bayes está basado en el teorema de Bayes de probabilidad condicional y asume independencia entre los atributos. A pesar de esta suposición de independencia en la práctica genera muy buenos resultados [14].

El método de los *k*-vecinos más cercanos calcula la distancia entre la instancia a clasificar y cada una de las instancias del conjunto de datos de entrenamiento y toma las *k* instancias más cercanas. Para calcular la distancia hay varias opciones entre las que están la distancia euclidiana y la distancia Manhattan. Para este experimento la distancia euclidiana descrita en la ecuación 1 arrojó mejores resultados:

$$\sqrt{\sum_{i=1}^k (b_i - c_i)^2} \quad (1)$$

Donde *k* es el número de atributos, *b* es el valor del atributo *i* de la instancia a clasificar y *c* es el valor del atributo *i* de la instancia dentro del conjunto de datos de entrenamiento. En este trabajo se utilizó *k*=3 determinado en base a pruebas. Sin embargo el valor de *k* puede ser determinado utilizando técnicas de optimización [15].

Para la clasificación en tiempo real se cargan en memoria los datos de entrenamiento y cada segundo se crea una instancia con la información del último y penúltimo segundo, esto es, cada segundo se hace la detección de la actividad usando información de los dos segundos anteriores.

4 Experimentos

En esta sección se describen los experimentos realizados y después se presentan los resultados obtenidos de la clasificación de actividades.

4.1 Descripción de los Experimentos

Para este experimento se utilizó un teléfono LG Optimus Me con sistema operativo Android 2.2 [9]. Este teléfono tiene un acelerómetro de 3 ejes modelo KR3DM/H de STMicroelectronics con un rango máximo de 19.6000 m/s². El sensor regresa los valores para la aceleración en cada uno de los ejes (*x*, *y*, *z*) y un timestamp en nanosegundos. El muestreo se hizo a una frecuencia de 40Hz (una lectura cada 25 milisegundos). Este sensor hace muestreos de hasta 90Hz aproximadamente. Se decidió utilizar una frecuencia de 40Hz para ahorrar recursos ya que en el futuro uno de los objetivos es utilizar la información de otros sensores como gps, compás, bluetooth, redes inalámbricas, etc. para generar una hipótesis más precisa de la actividad que se esté realizando. El teléfono se ubicó en el cinturón del pantalón entre el ombligo y el lado derecho de la cadera.

El primer experimento consistió en un entrenamiento de 3 minutos para cada actividad a excepción de *caminar*, que fueron 4 minutos. Esto equivale a 480 instancias de entrenamiento. El conjunto de instancias de prueba (*test-set*) consistió de un minuto para cada actividad que equivale a 150 instancias en total.

El segundo experimento consistió en agregar un minuto más de entrenamiento al conjunto de datos de entrenamiento del primer experimento, esto es, 4 minutos para cada actividad a excepción de *caminar*, que fueron 5 minutos. Esto equivale a 630

instancias de entrenamiento. Los datos de prueba al igual que para el primer experimento fueron de un minuto.

4.2 Resultados

La precisión total (porcentaje del total de instancias de prueba de todas las actividades clasificadas correctamente) para el primer experimento con el algoritmo de los *k-vecinos más cercanos* fue de 82.66%, la Tabla 1 muestra la matriz de confusión. En esta tabla se puede ver, por ejemplo, que de las 30 instancias correspondientes a caminar, 20 fueron clasificadas correctamente y 10 fueron clasificadas como de subir escaleras. La diagonal muestra el número de instancias clasificadas correctamente. La precisión total con los algoritmos *C4.5* y *Naive Bayes* fue de 86.66% y 90% respectivamente. Las Tablas 2 y 3 muestran su respectiva matriz de confusión.

La mayor fuente de error fue entre la clasificación de bajar escaleras y caminar. Utilizando el algoritmo de los *k-vecinos más cercanos*, 11 instancias de tipo *bajar escaleras* fueron clasificadas como si fueran de tipo *caminar* y 4 como si fueran de tipo *subir escaleras*.

Para el segundo experimento (agregando un minuto más de entrenamiento) la precisión total con el algoritmo de los *k-vecinos más cercanos* fue de 89.33%, la Tabla 4 muestra la matriz de confusión. Como se puede ver en la Tabla 4 con el minuto adicional de entrenamiento se redujo el error en las clasificaciones de bajar escaleras. La precisión total para el segundo experimento con los algoritmos *C4.5* y *Naive Bayes* fue de 87.33% y 93.33% respectivamente. Las Tablas 5 y 6 muestran su respectiva matriz de confusión. La Tabla 7 muestra los porcentajes de precisión para cada actividad utilizando cada uno de los 3 algoritmos en los dos experimentos.

Tabla 1. Matriz de Confusión del Primer Experimento con *k-vecinos más cercanos*.

Actividad	Clasificada Como				
	Caminar	Correr	Subir Escaleras	Bajar Escaleras	Reposo
Caminar	20	0	10	0	0
Correr	0	30	0	0	0
Subir Escaleras	1	0	29	0	0
Bajar Escaleras	11	0	4	15	0
En Reposo	0	0	0	0	30

Tabla 2. Matriz de Confusión del Primer Experimento con *C4.5*.

Actividad	Clasificada Como				
	Caminar	Correr	Subir Escaleras	Bajar Escaleras	Reposo
Caminar	21	0	7	2	0
Correr	0	30	0	0	0
Subir Escaleras	4	0	26	0	0
Bajar Escaleras	5	0	2	23	0
En Reposo	0	0	0	0	30

Tabla 3. Matriz de Confusión del Primer Experimento con *Naive Bayes*.

Actividad	Clasificada Como				
	Caminar	Correr	Subir Escaleras	Bajar Escaleras	Reposo
Caminar	25	0	5	0	0
Correr	0	30	0	0	0
Subir Escaleras	0	0	30	0	0
Bajar Escaleras	9	0	1	20	0
En Reposo	0	0	0	0	30

Tabla 4. Matriz de Confusión del Segundo Experimento con *k-vecinos más cercanos*.

Actividad	Clasificada Como				
	Caminar	Correr	Subir Escaleras	Bajar Escaleras	Reposo
Caminar	20	0	8	2	0
Correr	0	30	0	0	0
Subir Escaleras	1	0	29	0	0
Bajar Escaleras	3	0	2	25	0
En Reposo	0	0	0	0	30

Tabla 5. Matriz de Confusión del Segundo Experimento con C4.5.

Actividad	Clasificada Como				
	Caminar	Correr	Subir Escaleras	Bajar Escaleras	Reposo
Caminar	21	0	3	6	0
Correr	0	30	0	0	0
Subir Escaleras	3	0	26	1	0
Bajar Escaleras	4	0	2	24	0
En Reposo	0	0	0	0	30

Tabla 6. Matriz de Confusión del Segundo Experimento con *Naive Bayes*.

Actividad	Clasificada Como				
	Caminar	Correr	Subir Escaleras	Bajar Escaleras	Reposo
Caminar	23	0	3	4	0
Correr	0	30	0	0	0
Subir Escaleras	0	0	30	0	0
Bajar Escaleras	2	0	1	27	0
En Reposo	0	0	0	0	30

Tabla 7. Porcentaje de precisión para cada actividad en los dos experimentos. KNN: k-vecinos más cercanos. C4.5: Algoritmo C4.5. NB: Naive Bayes.

Actividad/Precisión	Primer Experimento			Segundo Experimento		
	KNN	C4.5	NB	KNN	C4.5	NB
Caminar	66.66%	70%	83.33%	66.66%	70%	76.66%
Correr	100%	100%	100%	100%	100%	100%
Subir Escaleras	96.66%	86.66%	100%	96.66%	86.66%	100%
Bajar Escaleras	50%	76.66%	66.66%	83.33%	80%	90%
En Reposo	100%	100%	100%	100%	100%	100%
Precisión Total	82.66%	86.66%	90%	89.33%	87.33%	93.33%

4.3 Discusión

Un aspecto que se observa claramente en los experimentos presentados es cómo con un minuto más de entrenamiento la precisión total aumentó considerablemente la

precisión, sobre todo en aquellos casos en que el primer experimento daba resultados deficientes. En efecto, KNN aumentó hasta 66% (15% en promedio), mientras que en C4.5 mejoró hasta 4% (1% en promedio), y en NB hasta un 35%, aunque en el caso de la actividad de caminar disminuyó en un 8%, posiblemente relacionado con el mejoramiento en otras actividades (el promedio de mejora es 6%). Vemos, pues que fue KNN el que se benefició más con el minuto adicional de entrenamiento.

5 Conclusiones y trabajo futuro

En este trabajo se vio cómo un algoritmo simple como el de los *k-vecinos más cercanos*, aplicado a secuencias de datos de acelerómetros de un teléfono celular, puede dar buenos resultados en la tarea de reconocimiento de actividades físicas de usuarios. Para la siguiente etapa de nuestra investigación se tiene planeada la incorporación de otros sensores que traen los teléfonos celulares para mejorar la precisión y una vez mejorado el grado de precisión se utilizará la detección de actividades en el contexto de seguridad.

Referencias

1. Kim, E., Helal, S., Cook, D.: Human Activity Recognition and Pattern Discovery. IEEE Pervasive Computing, vol. 9(1) pp. 48--53 (2010)
2. Cook, D.J., Augusto, J.C., Jakkula, V.R.: Ambient intelligence: Technologies, applications, and opportunities. Pervasive and Mobile Computing. vol. 5(4) pp. 277--298 (2009)
3. Dimension Engineering, LLC. A beginner's guide to accelerometers, <http://www.dimensionengineering.com/accelerometers.htm>
4. Brezmes, T., Gorricho, J.L., Cotrina, J.: Activity Recognition from Accelerometer Data on a Mobile Phone. In: Proceedings of the IWANN '09, pp. 796--799. Springer, (2009).
5. Karantonis, D.M., Narayanan, M.R., Mathie, M., Lovell, N.H., Celler, B.G.: Implementation of a real-time human movement classifier using a triaxial accelerometer for ambulatory monitoring. IEEE Transactions on Information Technology in Biomedicine vol. 10 pp. 156--167 (2006)
6. Tian, H., Lei, P., Xingjuan, L., Shusong, X.: Wearable activity recognition for automatic microblog updates. In: IEEE/ASME International Conference on Advanced Intelligent Mechatronics, pp. 1720--1723 (2009)
7. Seon-Woo, L., Mase, K.: Activity and Location Recognition Using Wearable Sensors. IEEE Pervasive Computing, vol. 1(3) pp. 24--32 (2002)
8. Mannini, A., Sabatini, A.M.: Machine learning methods for classifying human physical activity from onbody accelerometers. In: Sensors, vol. 10(2) pp. 1154--1175 (2010)
9. Android 2.2 Platform, <http://developer.android.com/sdk/android-2.2-highlights.html>
10. Aha, D., Kibler, D., Albert, M.K.: Instance-Based Learning Algorithms. Machine Learning, vol. 6 pp. 37--66 (1991)
11. Ravi, N., Dandekar, N., Mysore, P., Littman M.L.: Activity Recognition from Accelerometer Data. In: AAI, pp. 1541--1546 (2005)

12. Kwnpisz, J. R., Weiss, G.M., Moore, S.A.: Activity recognition using cell phone accelerometers. In: Proceedings of the Fourth International Workshop on Knowledge Discovery from Sensor Data, pp. 10--18 (2010)
13. Quinlan, J.R.: Improved use of continuous attributes in C4.5. *Journal of Artificial Intelligence Research*, vol. 4 pp.77--90 (1996)
14. Witten, I.H., Frank, E.: *Data Mining: Practical Machine Learning Tools and Techniques*, Second Edition. Morgan Kaufmann, Elsevier, San Francisco, CA (2005)
15. Segaran, T.: *Programming Collective Intelligence*. O'Reilly Media, Inc. Sebastopol, CA (2007)

Interoperabilidad de bases de datos heterogéneas soportada con mecanismos semánticos.

David González-Marrón^{1,2}, Miguel González-Mendoza², Neil Hernández-Gress²

¹ Instituto Tecnológico de Pachuca, Carretera México-Pachuca Km 81.5, Pachuca, Hidalgo, México
dgonzalez@itpachuca.edu.mx

² Instituto Tecnológico y de Estudios Superiores de Monterrey, Carretera Lago de Guadalupe Km 2.5, Atizapán de Zaragoza, Edo. de México, México
{ mgonza, ngress }@itesm.mx

Salvador A. Gutiérrez-Ayala³, Hugo Vázquez-Paleo³

³ Instituto Tecnológico Superior de Apatzingán, Carretera Apatzingán Aquililla Km 3.5, Apatzingán Michoacán A.P. 49, México
{salvador.a.gutierrez.a, hugo.vazquez.paleo}@gmail.com

Keywords: Bases de Datos Heterogéneas, Web semántica, Ontologías

Abstract: En este trabajo, se propone un mecanismo de extracción de información de bases de datos relacionales y su almacenamiento en un formato semántico dentro de un modelo relacional. Se realiza una comparación entre el modelo de almacenamiento propuesto, con métodos de almacenamiento semánticos ya existentes. Se indica el mecanismo propuesto para generación de ontologías que permitan integrar diferentes bases de datos en una única relación y mejorar el proceso de explotación de datos

I Introducción.

La web semántica tiene como su principal propósito, la compartición de información, entre usuarios, para encontrar y combinar información fácilmente, uno de los principales requerimientos a cumplir es que ésta actividad sea realizada de manera automática sin intervención humana [1].

El alcance de esta investigación está dirigida a trabajar con información de bases de datos al interior de las empresas, considerando que dentro de éstas, existen diversidad de sistemas que automatizan la recolección de datos que se generan en cada transacción, sin embargo es prudente mencionar que estas bases de datos generadas aunque abordan frecuentemente la misma problemática, no son totalmente compatibles entre sí, lo que ocasiona que personal del área de Tecnologías de información, gaste mucho de su tiempo y de sus recursos organizando datos a fin de generar información relevante para la toma de decisiones [8].

Lo que se pretende realizar en este proyecto, es utilizar la propuesta generada por la web semántica para alinear automáticamente los datos de bases de datos relacionales existentes al interior de las empresas y proporcionar un método alternativo de explotación de información

Para lograr esto, se propone el uso de una metodología que requiere la utilización de diferentes procesos como son : a) Utilización de ingeniería inversa para extraer la información de las bases de datos y almacenarla en archivos semánticos; b) Almacenamiento de la información extraída en un manejador de base de datos relacional con formato semántico que sea fácil de consultar y que tenga como premisa disminuir el espacio de almacenamiento; c) Utilización de ontologías, para la alineación de términos que considere los mecanismos de agregación, especialización y sinonimia; d) Finalmente un mecanismo que permita realizar consultas fácilmente a la base de datos semántica generada.

Dentro del artículo, se menciona la experiencia adquirida en el proceso de conversión de bases de datos relacionales en información semántica, el mecanismo utilizado para guardar la información semántica obtenida en bases de datos semánticas relacionales, utilizando para esto rutinas desarrolladas por productores de software semántico, así como rutinas de almacenamiento especialmente desarrolladas.

En la sección 2, se aborda el almacenamiento de la información semántica en bases de datos relacionales, ya que el formato semántico tiende a ser muy extenso, principalmente por su simplicidad y generalidad para ser aplicados a diferentes bases de datos.

En la sección 3 se especifica la forma en que las ontologías son utilizadas para interconectar bases de datos.

En la sección 4 se muestra el diseño propuesto para lograr hacer un lenguaje sencillo que permita explotar la información almacenada.

En la sección 5 se mencionan los desarrollos planeados para contar con una metodología integrada para integrar bases de datos heterogéneas en un dominio común de conocimiento.

2 Comparación de almacenamiento con otros sistemas existentes.

La realización de almacenamiento semántico, comparado con el relacional, resulta ser mucho más extenso, ya que el modelo relacional está basado en principios de teoría de conjuntos y optimización de espacio, para lo cual fue generado un proceso denominado normalización, el cual busca la transformación de los datos a estructuras que cumplan con los requerimientos de organización impuestos por las formas normales, las cuales buscan descomponer la información a datos con mejores características de almacenamiento sin pérdida de información, las cuales van desde la primera forma normal (1NF), segunda forma normal (2NF), tercer forma Normal (3NF), forma normal BoyCe and Codd (BCNF), cuarta forma normal(4NF) y la quinta forma normal (5NF), los datos almacenados una vez normalizados son accedados con operadores existentes basados en los operadores relacionales de selección, proyección, restricción y unión [4]. en la siguiente gráfica se muestra el universo de relaciones existentes en el modelo relacional.

Universo de Relaciones (normalizadas y no normalizadas)



Figura 1: Relaciones existentes de acuerdo al modelo relacional

El modelo semántico a su vez tiene un propósito diferente, éste busca almacenar información, no basado en una teoría de conjuntos, sino que se ocupa del significado contenido en las palabras y oraciones, de manera similar a como es el lenguaje humano, aquí se busca describir que acción realiza un sujeto hacia un objeto, un ejemplo de estos lenguajes son los lenguajes de tipo SVO (Sujeto-Verbo- Objeto) que en base a su tipología lingüística, junto con los lenguajes SOV (Sujeto-Objeto-Verbo), conforman más del 75% de los lenguajes del mundo [12].

El modelado semántico resume la complejidad lógica representativa de las Bases de Datos, permitiendo que se incremente el uso de parte de los usuarios, los cuales no necesitan ser conscientes de la estructura lógica de la base de datos.

Las Bases de Datos Semánticas tienen la propiedad de abstracción de datos y la de separación de componentes lógicos y conceptuales, permitiendo representar la realidad de una manera más natural. Sin embargo debido a la similitud que tienen al lenguaje natural los datos de las Bases de Datos Semánticas, se pueden presentar algunas ambigüedades, como por ejemplo mayor complejidad lógica y problemas al interpretar las consultas [10].

En este modelo se almacenan un conjunto de oraciones, las cuales estarán interconectadas cuando el objeto al que se le realiza una acción, es en otra oración un sujeto, siendo frecuente encontrar que muchas oraciones se encuentran relacionadas mediante esta asociación (objeto-sujeto). Este modelo tiene como característica un gran número de asociaciones (objeto-sujeto). El propósito de este trabajo es el de crear una infraestructura que permita realizar consultas de tipo extensivas, entendiéndose como extensivas a aquellas involucran información de la base de datos, en contraparte a las consultas relacionadas con la estructura de las ontologías. Se muestra en este trabajo la conformación de una base de datos plana, la cual es el resultado de la aplicación de ingeniería inversa a una base de datos relacional, proceso que se puede repetir de manera iterativa a diferentes bases de datos relacionadas.

Actualmente el modelo propuesto está conformado por una tabla universal conocida como tabla vertical, el cual es utilizado por Jena y TAP [13], aunque existen diferentes propuestas para almacenar bases de datos semánticas como lo es la base de datos horizontal utilizada por Agrawal et al. [14], el modelo híbrido compuesto por clases horizontales y por tablas de propiedad propuesto por Z. Pan y J Heflin [15], así como otros métodos propuestos que buscan mejorar el tiempo de realización de consultas considerando una organización de datos diferente.

Por la naturaleza de integración de diversas bases de datos, la conceptualización y conformación de éstas en una BD plana, resulta la de más fácil utilización para nuestro propósito de integración, aunque el proceso de explotación se complica, ya que el proceso de búsqueda de la solución involucra una búsqueda exhaustiva en la entidad de almacenamiento, generándose un proceso de tipo recursivo.

Se ha considerado que el tener una sola BD universal, permite por su simplicidad entender la estructura de organización de los datos, permitiendo aplicar posteriormente diferentes técnicas que nos permitan mejorar el tiempo de respuesta de las consultas y la disminución de espacio de almacenamiento. Diferentes iniciativas existen para disminuir el espacio de almacenamiento, como el propuesto por Byrnes [16], quien busca disminuir el número de tripletas con una optimización de la información almacenada, retirando información con poco significado semántico, otro trabajo de interés es el realizado por Levandoski y Mokbel [17], que proponen reestructurar las tablas de almacenamiento para reducir el número de joins en el plan de consulta y mantener el almacenamiento de datos nulos debajo de un límite razonable.

A continuación se describe el proceso de generación de un modelo relacional a semántico. Haciendo uso de un diseño relacional simple, conformado por tres entidades:

Department: Entidad que almacena información de los departamentos existentes en una empresa.

Employee: Entidad para almacenar información de empleados que laboran dentro de la empresa.

Job: Relación que permite integrar información acerca de los trabajos llevados a cabo por los empleados, así como departamento donde realizan sus actividades.

Los componentes de cada entidad y relación se muestran a continuación, se señalizan las llaves primarias usando letras en negritas, y las llaves foráneas utilizando un óvalo.

Employee (**emp_id**, emp_name, emp_birthdate, enrollment_date, emp_sex, emp_marital_status)

Department (**dept_id**, dept_description)

Job (**emp_id**, **dept_id**, job_description, job_salary)

Un ejemplo de datos almacenados en cada tabla utilizados para el ejemplo se hace a continuación.

emp_id	emp_name	emp_birth_date	enrollment_date	emp_sex	emp_marital_status
1	Juan	2009-11-11	2008-11-11	M	C
2	David	2000-09-14	2010-11-14	M	S
3	Armando	1995-10-13	2001-08-06	M	S

Dept_id	Dept_description
1	departamento1
2	departamento2
3	departamento3

emp_id	dept_id	Job_description	job_salary
1	1	director	56,234.00
2	1	Jefe de depto	13,245.00
3	3	Cantador	9,245.00

Sujeto	Predicado	Objeto
employee/1	emp_id	1
employee/1	emp_name	Juan
employee/1	emp_birth_date	2009-11-11
employee/1	enrollment_date	2008-11-11
employee/1	emp_sex	M
employee/1	emp_marital_status	C
...
...
...
job/3/3	job_salary	9,245.00

Almacenamiento de información en formato relacional

Almacenamiento de Información en forma semántica utilizando tripletas

Como puede ser visto el número de registros (tuplas) a ser generados por el modelo semántico, basado en un diseño relacional, puede ser descrito con la siguiente fórmula

$$T = \sum_{i=1}^n N_i \times M_i$$

Donde:

- T = total de registros o tuplas generadas
- N_i = número de columnas de la tabla,
- M_i = número de renglones de la tabla,
- n = número de tablas de la base de datos

En el ejemplo mostrado el número de tuplas generadas es de $T = 36$.

Puede ser visto que se incrementa de manera aritmética el volumen de almacenamiento en almacenamiento con tripletas, además de que los tipos de datos utilizados son todos de tipo string.

A continuación se hace una descripción de la propuesta de almacenamiento desarrollada, la cual se basa en 2 grandes fases:

Fase1: Proceso de extracción de información de las bases de datos relacionales que se desean almacenar en un formato semántico, para esto se hace uso de software basado en ingeniería inversa, el cual produce archivos de salida en diferentes formatos de tipo RDF (*Resource Description Framework*), este proceso puede ser observado en la figura2.



Figura2: Generación de archivos semánticos usando fuentes relacionales

El software utilizado para extraer la información de las bases de datos y su conversión a tripletas en formato RDF, es el D2R Server [5], el cual permite igualmente la consulta de información utilizando el lenguaje de consultas SPARQL, este software ha sido desarrollado por la Universidad Libre (*Freie*) de Berlín, la cual colabora en los desarrollos propuestos por la W3C.

Fase2: Proceso de almacenamiento de información generada en la fase 1 en archivos RDF a un formato semántico propuesto basado en tripletas, el diseño seleccionado integra toda la información existente en el archivo RDF proveniente de una base de datos, el cual puede contener información de una o más entidades de la base de datos, procediéndose a insertar los datos en una sola entidad (base de datos plana) con un diseño de tripletas, el cual está formado por una estructura compuesta de 3 atributos (Sujeto, Predicado y Objeto). Como lo que se pretende es la consolidación de información de todas las bases de datos que se quieren relacionar mediante este proceso, se procede a hacer esto, tantas veces como bases de datos existan, las cuales deben pertenecer a un dominio de conocimiento. Esta fase puede ser visualizada en la figura siguiente:

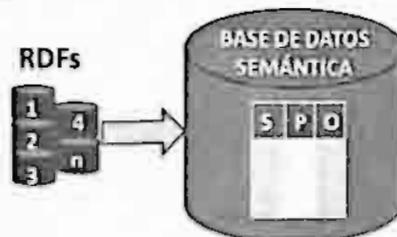


Figura 3: Proceso de Generación de base de datos semántica

El almacenamiento de datos requiere una estructura que permita mejorar la realización de consultas, así como disminución de espacio de almacenamiento. Esta es una de las razones por las que se seleccionó el almacenamiento de la información generada en archivos RDF en una base de datos relacional, en lugar de utilizar los repositorios de archivos RDF utilizados tradicionalmente, otra razón es el buen desempeño que la tecnología relacional tiene para permitir la explotación de información semántica, como es reportado en los trabajos de Bizer y Cyganyac [5], además de la familiarización que tienen los usuarios con la tecnología relacional, al ser la BD semántica originada de datos provenientes del ambiente relacional.

Para definir la estrategia de almacenamiento en el modelo relacional, se ha buscado cumplir con los siguientes requerimientos (flexibilidad, disminución de número de tuplas y disminución de espacio en tuplas):

- 1) La flexibilidad es requerida, debido a que la información proviene de diversas bases de datos, por lo que es necesario contar con un esquema simple y suficientemente flexible que permita almacenar todos los datos de manera transparente a su origen, para esto se ha propuesto un diseño centralizado de datos en una sola base de datos plana, que almacene la mayor parte de la información intensiva en una estructura conformada por tres atributos (sujeto, predicado, objeto), siendo el tipo de datos seleccionado para cada atributo de tipo string. Es importante mencionar que cada sujeto es único y su repetición dentro de una entidad indicaría una relación de datos existente con otra tupla, se ha trabajado con la generación automática de ontologías extensivas, en vez de la generación automática de mapeos dirigidos por semántica de dominio [11], debido a que es más general, permitiendo una adaptación a cualquier dominio.
- 2) Actualmente se ha optado por almacenar la información del nivel extensivo en el esquema de tripletas en una sola entidad, ya que el volumen generado a este nivel es el predominante, la búsqueda de eficiencia en el almacenamiento todavía no se aborda, pero se ha considerado hacerlo cuando el proceso de explotación de datos semánticos se tenga perfectamente definido, y dependerá del proceso de almacenamiento del nivel intencional, donde se especifica el nivel conceptual de los elementos que componen las bases de datos, el cual actualmente se encuentra en proceso de diseño.

3) Una disminución de almacenamiento que es factible hacer, es la de reducir el tamaño de las tuplas generadas, considerando que la estrategia de solución propuesta es para operar dentro de la intranet. Realizándose los siguientes cambios:

- a) Extracción de referencias existentes en las URIs (Uniform Resource Identifier), el cual se utiliza para identificar un nombre o un recurso dentro de la internet. Considerando que el URI está compuesto por 2 partes: a) El URL (Uniform Resource Locator) el cual sirve para indicar dónde encontrar el recurso dentro del internet y b) El URN (Uniform Resource Name) el cual permite identificar de manera única un elemento dentro del internet.

La extracción de información de la URI que es innecesaria tal como: Protocolo, dominio o IP para acceder al servidor y la ruta en el servidor para acceder al recurso en internet. Esto permite disminuir el tamaño de almacenamiento de la base de datos semántica.

Para la realización de esta actividad, se utilizó entre otras, la librería para inserción de archivos RDF en una base de dato relacional denominada RDFAPI-PHP [6], un desarrollo de la W3C que permite entre otras cosas, almacenar archivos RDF en bases de datos relacionales, posteriormente, se procedió a desarrollar rutinas de filtrado que extraen la información innecesaria mencionada anteriormente, lo que llevó a una disminución del almacenamiento de más de un 50 %. Con respecto a métodos de almacenamiento ya existentes, a continuación se muestra una tabla que muestra el proceso de filtrado.

TABLE 1. EJEMPLIFICACIÓN DE FILTRADO EN UNA TUPLA

<i>Elemento</i>	<i>Original</i>	<i>Filtrado</i>
Sujeto	salida02.rdf#employee/1	salida02#employee/1
Predicado	Resource: vocab:employee_emp_id	emp_id
Objeto	Literal: 1(rdf:datatype="http://www.w3.org/2001/XMLSchema#int")	1

- b) Utilización de tipos de datos string flexibles dentro de la BD, que contengan un tamaño máximo de almacenamiento que nos permita almacenar la información existentes en las BD, pero que al mismo tiempo si son menores se ajusten al tamaño del dato proveniente de la BD, un ejemplo de estos tipos de datos es el tipo "varchar" el cual es aplicable a diversas bases de datos, como lo son: Mysql, Postgres, Oracle ó SQLServer.

3 Uso de ontologías para lograr interoperabilidad

Las ontologías tienen como propósito facilitar la comunicación y el intercambio de información entre diferentes sistemas, pudiendo ser usadas para varios propósitos. Como pueden ser: a) para realizar razonamiento inductivo; b) para realizar clasificación; y c) para uso en técnicas de resolución de problemas entre otros. Por otro lado la interoperabilidad es la capacidad que tiene un producto o un sistema, cuyas interfaces son totalmente conocidas, para funcionar con otros productos o sistemas existentes o futuros y eso sin restricción de acceso o de implementación

Cuando toda la información se encuentra almacenada en una entidad conformada por tripletas, es requerido contar con un mecanismo que permita una asociación lógica de los datos almacenados, debido a que la simplicidad de la estructura de almacenamiento, produce una gran cantidad de registros que se encuentran almacenados en una sola entidad, sin embargo este almacenamiento no tiene una estructura lógica que permita enlazar los datos con un sentido semántico, para lo cual se requiere una capa que permita organizar los datos y así poder enlazar adecuadamente cada una de las tripletas almacenadas.

Una ejemplificación del problema, se ilustra a continuación, cuando diferentes bases de datos contienen información relacionada, sin embargo no son totalmente compatibles debido a un problema de impedancia, el cual se debe a que algunos atributos son diferentes o existen conflictos de tipos de datos entre ellos.

Se muestra un ejemplo hipotético basado en el dominio de la epidemiología, en el cual diferentes bases de datos contienen algunas diferencias que son origen de ciertos conflictos.

La entidad analizada se denomina DetalleS y contiene cuatro atributos en cada una de las bases de datos

DetalleS (r1M, r1F, r5M, r5F)	Base de datos 1
DetalleS (r1M, r1F, r6M, r6F)	Base de datos 2
DetalleS (r1M, r1F, r10M, r10F)	Base de datos 3

En la entidad DetalleS, se presentan diferentes versiones que pueden tener significados diferentes.

En la primer base de datos el atributo "r1M" se utiliza para reportar un cierto tipo de epidemia entre uno y cuatro años para el sexo masculino. Para la segunda base de datos el mismo atributo "r1M" se utiliza para reportar número de epidemias, pero ahora entre uno y cinco años para el sexo masculino. En la tercera base de datos el mismo atributo se utiliza para reportar número de epidemias entre uno y nueve años para el sexo masculino.

Como puede ser visto en este caso hay una dependencia de datos entre el primer atributo de la tabla, con el tercero, lo que hace que el primer elemento contenga información semántica diferente en cada una de las tablas de la BD. A fin de evitar ambigüedad en el significado del primer atributo para cada tabla, es necesario realizar una distinción entre cada uno de los atributos de la tabla DetalleS para cada una de las bases de datos.

El URI hace ésta distinción ya que está diseñado para operar en internet, sin embargo contiene información innecesaria si el almacenamiento se realiza para ser operado en una intranet. En este caso el URI contiene información de la base de datos origen y de la tabla origen, lo que evita ambigüedad de los atributos con mismo nombre que se encuentran en diferentes bases de datos.

La ontología en su parte intencional, debe ser capaz de describir conceptualmente la información almacenada en la parte extensiva, además de sus reglas de comportamiento.

El proceso de creación de una ontología en el área intencional, debe ser realizado por expertos en el área, siendo el intercambio de información con otros expertos una tarea fundamental para lograr una adecuada estandarización, en cambio en el área extensiva de datos puede ser automatizada con herramientas ya existentes, como se ha podido constatar con las experiencias con el software utilizado (D2R Server y RDFAPI-PHP), existiendo en el mercado un buen número de productos equivalentes que pueden automatizar el proceso de consolidación de la información en una sola base de datos semántica, sin embargo las adecuaciones para hacerla operar eficientemente es lo que se ha realizado hasta ahora dentro de este trabajo (filtrado de información semántica no relevante en una intranet).

La creación de ésta parte intencional, puede ser realizada auxiliándose de software existente como Protegé [7], a fin de facilitar la edición de la ontología. Así mismo el lenguaje de OWL (Ontology Web Language) se ha considerado como el estándar más robusto para la creación de la ontología en ésta parte [8]. Se ha seleccionado un modelo de ontología tesauros, debido a que permite la expresión de relaciones sinonímicas y jerárquicas, lo cual se ha considerado suficiente para alinear los atributos de las bases de datos heterogéneas.

4 Simplificación de Consultas.

El procesamiento de consultas involucra la utilización de datos almacenados en la BD semántica, así como el uso de información almacenada en la parte intencional, a fin de responder a las consultas solicitadas, esto puede ser visto en la figura 4 siguiente:

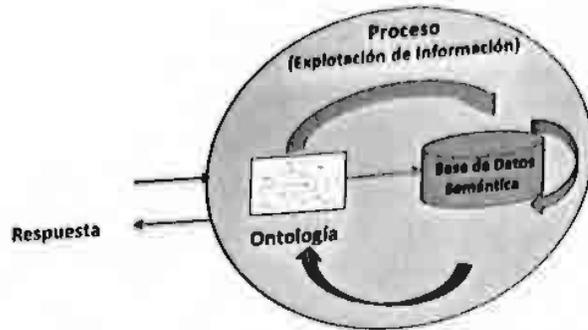


Figura 4. Proceso de Explotación de información en BD semántica

A continuación se muestra un ejemplo que ilustra el proceso de simplificación de consultas usando ontologías. Se considera que la consulta que se desea realizar es listar el nombre de las personas con influenza, la consulta deseada podría formularse como "Selecciona personas con influenza". La consulta resulta transparente al usuario final, debido a que no se especifican ni las bases de datos ni las entidades involucradas, esta transparencia es posible gracias a la ontología en su parte intencional, la cual se utiliza para resolver la consulta, generando los queries equivalentes para dar respuesta a la petición, así como integrando las respuestas parciales de los queries parciales proporcionan, a fin de integrar una respuesta para el usuario.

El ejemplo siguiente muestra tres diferentes bases de datos involucradas, las cuales tienen ciertas diferencias, por lo cual es requerida la generación de tres diferentes consultas parciales para proporcionar la respuesta solicitada.

TABLE II. CAMPOS DIFERENTES INVOLUCRADOS PARA RESOLVER LA CONSULTA

Término	Base de datos1.	Base de datos2.	Base de datos3.
Influenza	campo15=1	epidem=6	dis="Inf"

En la tabla anterior puede ser visto que el término "Influenza" tiene diferentes acepciones en cada una de las bases de datos: En la "base de datos1" cuando el campo15 tiene un valor de 1, en la "base de datos 2" cuando el campo epidem tiene un valor de 6 y en la "base de datos 3" cuando el campo dis tiene el valor de *Inf*. Como cada base de datos tiene campos diferentes, se requiere la producción de los siguientes queries en SQL.

Select person when field15=1

Select person when epidem=6

Select person when dis="INF"

Una vez ejecutadas las 3 consultas (queries), se integran los resultados obtenidos y se proporciona una sola respuesta al usuario, en el ejemplo mostrado las 3 consultas generadas pueden ser integradas con la utilización del conector OR, pero no siempre es posible realizar una acción de integración de la consulta, el ejemplo equivalente en este caso sería ;

Select person when field15=1 OR epidem=6 OR dis="INF"

La información requerida para producir la respuesta, se obtiene gracias a la información almacenada en la parte extensiva de la BD semántica, junto con la integración de la lógica de explotación de los datos que se encuentran en

la parte intencional, actividades adicionales como filtraje de datos repetidos o identificación de respuestas con grado de incertidumbre deberán ser realizadas antes de mostrar la respuesta obtenida al usuario final.

5 Conclusiones.

Los avances obtenidos hasta ahora, han permitido integrar diferentes bases de datos en una sola base de datos semántica en un modelo relacional. Para la realización de este proceso de integración, se ha utilizado hasta ahora software semántico desarrollado por la W3C, el cual se ha podido constatar que es bastante confiable, dos fases para el proceso de integración han sido utilizados, el proceso de generación automática de archivos RDF desde bases de datos relacionales con el producto D2R Server, y la integración de información semántica que es innecesaria en intranets, lo que ha permitido obtener mejoras en el almacenamiento y el resultado ha sido una reducción de espacio de almacenamiento en la BD semántica integrada. El proceso de generación de la ontología intencional está en proceso de desarrollo, pero se ha seleccionado el lenguaje de ontologías denominado OWL como el estándar de creación, por ser el más completo para almacenar el sentido semántico de los datos, un ejemplo de esta aplicación, ha sido la utilización de la propiedad owl:sameAs, que nos permite crear sinónimos para ciertos conceptos que son comunes en el caso de base de datos heterogéneas. Es importante mencionar que muchas otras propiedades del OWL serán utilizadas para producir una ontología tipo tesoro, la cual se ha considerado suficientemente completa para la integración de BD heterogéneas.

6. Trabajos futuros

Se continuará con el proceso de familiarización de un dominio de conocimiento en particular, a fin de demostrar la eficacia de la metodología propuesta, se validará si es conveniente reorganizar la estructura de la base de datos semántica planteada para mejorar la eficiencia de las consultas. Se enfocará el trabajo a la generación de la ontología en su parte intencional, a fin de comprobar su eficacia para enlazar adecuadamente los datos semánticos y permitir una explotación simple de los datos. Se desarrollará un lenguaje de consultas que permita validar el comportamiento de la BD semántica, que permita manejar grados de incertidumbre al momento de realizar consultas, que proporcione las respuestas ordenadas en base al grado de certidumbre de éstas. Se plantea igualmente una validación con otras metodologías existentes a fin de contrastar los resultados obtenidos.

References

- [1] OMG, "Ontology Definition Metamodel", Adopted Specification 2006-10-11, <http://www.omg.org/docs/ptc/06-10-11.pdf> (2007-03-20).
- [2] B Kashyap V, "Design and creation of ontologies for environmental information retrieval", 12th Workshop on Knowledge Acquisition, Modeling and Management (KAW'99), Banff, Canada, October 1999, <http://seml.ucalgary.ca/lsi/kaw/KAW99/papers/Kashyap1/kashyap.pdf> (2007-03-15)
- [3] Christian Bizer and Andy Seaborne, "D2rq: Treating non-rdf databases as virtual rdf graphs", 3rd International Semantic Web Conference (ISWC2004), 2004 <http://www.wiwiss.fu-berlin.de/suhl/bizer/pub/Bizer-D2RQ-ISWC2004.pdf>
- [4] Date C.J., "An introduction to DataBase Systems", Further Normalization I: 1NF, 2NF, 3NF, BCNF, Normal Forms, Sixth Edition, Addison Wesley 1995
- [5] Bizer C., Cyganiak, R., "D2RQ — Lessons Learned", Position paper for the W3C Workshop on RDF Access to Relational Databases, Cambridge, USA, 25-26 October 2007
- [6] RDFAPI Home, Documentation, 2004 <http://www4.wiwiss.fu-berlin.de/bizer/rdfapi/tests.html>
- [7] Protégé, <http://protege.stanford.edu>, Consulted: May 18th 2010, Stanford Center for Biomedical Informatics Research at the Stanford University School of Medicine.
- [8] OWL Web Ontology Language Overview, W3C Recommendation 10 February 2004

- [9] IDC Worldwide Data Integration and Access Software 2008-2012 Forecast Doc No 211636, Apr. 2008
- [10] Davidson, James y Jerrold Kaplan, "Natural Language Access to Data Bases: Interpreting Update Requests". Computer Science Department, Stanford University, California. ACM Press, April 1983
- [11] S Sahoo, O Bodenreider, J Rutter, K Skinner, A Sheth, "An ontology-driven semantic mashup of gene and biological pathway information: Application to the domain of nicotine dependence". *Journal of Biomedical Informatics*, Volume 41 Issue 5, October, 2008.
- [12] Crystal, David (1997) "The Cambridge Encyclopedia of Language" (2nd edition ed.) Cambridge: Cambridge University Press ISBN 0-521-55967-7
- [13] D. Beckett and J. Grant, Mapping Semantic Web Data with RDBMSes. 2001.
http://www.w3.org/2001/sw/Europe/reports/scalable_rdbms_mapping_report
- [14] R. Agrawal, A. Somani, and Y. Xu. Storage and Querying of E-Commerce Data. In Proc. of VLDB 2001.
- [15] Z. Pan and J. Heflin, DLDB: Extending Relational Databases to Support Semantic Web Queries, Technical Report: LU-CSE-04-006, Department of Compute Science, Lehigh University, 2006.
- [16] Kate Byrne. Relational Databases to RDF translation in the cultural Heritage Domain. Intenet May 2008. Based on chapter of PhD thesis.
- [17] L. Levandoski, M. Mokbel, RDF Data-Centric Storage, IEEE International Conference on Web Services, 2009

Preguntas de Definición para Generación Automática de Exámenes a partir de un Texto en Español

Alejandro González Zeta¹, Ledeneva Yulia¹, René García Hernández¹,
Rafael Cruz Reyes¹, Citlalih Gutiérrez Estrada² and Sergio Díaz Zagal²

¹Universidad Autónoma del Estado de México
Unidad Académica Profesional Tianguistenco,
Paraje el Tejocote San Pedro Tlantizapan, 52600, Estado de México
alejandrozeta@hotmail.com, yledeneva@yahoo.com, renearnulfo@hotmail.com

²Laboratorio de Reconocimiento de Patrones, Instituto Tecnológico de Toluca
Av. Tecnológico s/n, Ex Rancho La Virgen, Metepec, 52140, México
rcruzrey@gmail.com, citlalihg@yahoo.com.mx, sergio_diaz@yahoo.fr

Resumen. La necesidad del apoyo al docente y alumno ha motivado el desarrollo de las tecnologías de información en el aula. Se ha logrado un avance significativo en el proceso de enseñanza-aprendizaje a través de los métodos y las herramientas ad-hoc. En este artículo, se propone un método para generar automáticamente los exámenes detectando la información más importante de las colecciones de textos grandes en español a partir de las descripciones multipalabra. El método propuesto se aplicó para generar las preguntas de definición detectados a partir de las Secuencias Frecuentes Maximales (SFMs). Se presenta la evaluación de las preguntas generadas realizada por los expertos.

Palabras clave: Tecnologías en Educación. Búsqueda de Respuestas. Preguntas de Definición. Secuencias Frecuentes Maximales. Descripciones Multipalabra.

1 Introducción

Las computadoras son herramientas que se utilizan en infinidad de áreas porque permiten realizar tareas repetitivas en menor tiempo que una persona. Para que una computadora realice una tarea, debe tener bien delimitada el área de trabajo y las variables que utilizarán para realizar la actividad para la cual fue programada. Si alguna de estas dos cambia, muy probablemente no se obtengan los resultados deseados por parte de la computadora. Los métodos computacionales actuales permitan a una computadora distinguir dentro de un texto las necesidades dadas por un usuario. Uno de estos ejemplos son los buscadores de internet, donde el usuario introduce una consulta de unas o varias palabras, posteriormente la computadora muestra una serie de documentos asumiendo que es lo que interesa al usuario. Otro ejemplo, son los generadores de resúmenes [1 - 5], donde el objetivo es que la

computadora extraiga de un texto – palabras, oraciones o párrafos más importantes generando un resumen.

El área del Procesamiento Automático de Textos (PAT) se encarga de desarrollar los modelos de lenguaje que pueden ser implementados en una computadora para realizar las tareas como extracción de información, traducción, automática diccionarios electrónicos, desambiguación del sentido de las palabras, recuperación de información, generación de textos, etc. [6 - 12].

En los últimos años, se han aumentado las tecnologías de información en el proceso enseñanza-aprendizaje, de este modo mejorando el apoyo para el docente y comprensión de los contenidos de un área de conocimientos apoyándose de los avances de PAT. Uno de los instrumentos más utilizados por el docente, para realizar una prueba en la que se mide el nivel de conocimientos, aptitudes, habilidades o unas capacidades físicas determinadas, es un examen. Se usa como herramienta para determinar la idoneidad de alguien para la realización de una actividad o el aprovechamiento de unos estudios.

En este artículo, se propone crear un método que facilitará al docente generar los exámenes para las colecciones de documentos grandes. Se aborda el problema de generar los exámenes solamente para las partes más importantes de las lecturas propuestas por el docente utilizando descripciones multipalabra [1]. Se experimenta con las preguntas de definición que se generan a partir de las Secuencias Frecuentes Maximales [13 - 15].

En la sección 2, se presenta el estado de arte, en la sección 4 se detalla la extracción de pares definición-concepto. El método propuesto se describe en la sección 4. Finalmente, se presenta la experimentación y trabajo futuro.

2 Estado del Arte

Una de las aplicaciones del área de Procesamiento Automático de Textos (PAT) son los sistemas de Recuperación de Información (RI) consisten en recuperar todos los documentos que son relevantes para una consulta de usuario desplegando los resultados en el orden de relevancia [16 -19, 20]. Por ejemplo, los motores de búsqueda en internet [10 - 11]. El motor de búsqueda más utilizado es *Google*, donde el usuario coloca una o varias palabras que se llaman palabras de consulta, posteriormente como resultado se obtiene una serie de documentos que contienen las palabras de la consulta introducidas por el usuario.

El modelo espacio vectorial es el más utilizado en la actualidad en los sistemas RI. Este modelo entiende que los documentos pueden expresarse en función de unos vectores que representan la frecuencia de aparición de los términos en los documentos. Los términos que forman esa matriz serían términos no vacíos, es decir, dotados de algún significado a la hora de recuperar información [9].

2.1 Generación de exámenes

La mayoría de los sistemas elaboran los exámenes a través de un banco de reactivos (o ejemplos) que se encuentran en línea y se generan por los expertos de manera manual. Los exámenes se generan en forma aleatoria, ponderada y automática de cualquier tema de un curso o del curso completo. El docente selecciona los números de reactivos que conformarán el examen y de esta manera se compone el examen generado. Otro de los métodos genera los exámenes a partir de un corpus o base de datos que captura el docente a mano que contiene las preguntas generadas con antelación [21].

La herramienta "*Exámenes*" [22] para generación de exámenes nos permite confeccionar, almacenar y ordenar infinidad de tipos de exámenes diferentes, una herramienta especialmente diseñada para centros de enseñanza tanto oficiales como particulares. Para crear los exámenes es necesario empezar una base de datos que estará clasificada en temas y preguntas, además de poder organizarla en niveles y algunas otras opciones que perfilarán aún más las creaciones del docente. Una vez creados temas con sus preguntas y respectivas respuestas podremos generar los exámenes, diferenciando la generación manual (selección realizada por los docentes) y la generación automática (la computadora elegirá las preguntas de manera aleatoria).

La herramienta "*Hot Potatoes*" [23] es un conjunto de seis herramientas que pueden ayudar a educadores, profesores, y escritores a publicar material educativo en la Web. Las pequeñas herramientas de Hot Potatoes te permiten crear multi-elección interactiva, respuesta corta, rellenar en el espacio, crucigramas, y ejercicios de desordenación de una frase usando HTML y JavaScript sin necesidad de tener ningún conocimiento ni de HTML ni de JavaScript. JBC te permite crear exámenes "tipo test" donde cada pregunta puede tener hasta cinco respuestas, y donde cada número de ellas pueden ser correctas o no. Cada estudiante recibe un porcentaje de acierto después de cada pregunta acertada. JQuiz te facilita el diseño de preguntas con la posibilidad de que el estudiante rellene con palabras o frases como respuesta. JCloze crea ejercicios para rellenar en el espacio. Hasta 100 respuestas correctas pueden especificarse para cada espacio en blanco, además la posibilidad de incluir una pequeña pista para cada espacio. JCross diseña crucigramas que se pueden rellenar on-line. Se puede usar parrillas de hasta 20x20 letras. JMix te permite crear ejercicios de ordenación de frases. Puedes especificar hasta 100 respuestas correctas distintas. JMatch crea ejercicios de emparejamiento u ordenación. Una lista de objetos fijos aparecen a la izquierda (pueden ser imágenes o texto), y una lista de objetos desordenados a la derecha. Hot Potatoes soporta el uso de acentos y te facilita la opción de acceder al código de las páginas para cualquier modificación.

En este tipo de métodos y herramienta del estado de arte no se aplica el tratamiento inteligente de textos, tampoco se generan los exámenes a partir del material bibliográfico del curso, sino a partir de preguntas generadas por el docente.

2.2 Tipos de Pregunta

Derivada de los sistemas de RI, se encuentran los sistemas de Búsqueda de Respuesta (BR) que es una tarea compleja que combina las técnicas de la PAT, RI y aprendizaje automático [12 - 13]. El objetivo principal de los sistemas BR es localizar la respuesta correcta a una pregunta escrita en lenguaje natural en una colección de documentos que generalmente no está estructurada siendo internet la principal base de datos de búsqueda. Los sistemas de BR funcionan como un motor de búsqueda, donde la entrada es una pregunta en lenguaje natural y la salida es la respuesta a la pregunta (no una lista de documentos completos como los sistemas de IR) [9 - 11].

Este trabajo se encuentra dentro del área de BR. Existen varios tipos de preguntas, pero en este trabajo se presentan los siguientes tipos de preguntas siendo los más frecuentes en los exámenes, entre las que se encuentran:

1. **Factuales:** Son preguntas basadas en los hechos, pidiendo el nombre de una persona, un lugar, el grado de algo, etc.
2. **Factuales con restricción temporal:** Son preguntas de Tipo factual pero restringidas temporalmente.
3. **Definición:** Son preguntas como "¿Qué / Quién es X?", es decir, preguntas pidiendo la función / trabajo / información importante acerca de alguien, o hacer preguntas para la misión / nombre completo / importante información sobre una organización [13]. Las preguntas de definición se dividen en dos tipos:
 - a. Organización: donde se espera como respuesta la equivalencia de siglas.
 - b. Persona: donde se espera como respuesta el cargo o rol de una persona.

En este artículo, se aplica el método propuesta a generar un examen de forma automática conformado por las preguntas de definición.

2.2 Secuencias Frecuentes Maximales

Las Secuencias Frecuentes Maximales (SFMs) pueden ser de gran utilidad, puesto que podían representar las partes más importantes de los textos. Dada una colección de textos, el hecho de que algunas secuencias se repitan en varios de ellos, nos señala la importancia de la información que contienen dichas secuencias. También hay que destacar que la aplicabilidad de la extracción de SFMs es muy amplia tomando en cuenta que la técnica es independiente del dominio, de la temática y del lenguaje en el que estén escritos los textos [14, 15].

La SFM es una secuencia de palabras que debe aparecer en un número dado de ejemplos (por ejemplo, documentos, oraciones, etc.) y además, no debe estar contenida en otra secuencia de palabras. Asumiendo que D es un conjunto de textos (por texto nos referimos a un documento completo o incluso a una sola oración), donde cada texto consiste de una secuencia de palabras.

Definición 1. Una secuencia $p = a_1...a_k$ es una subsecuencia de una secuencia q si todos los elementos a_i , $1 \leq i \leq k$, ocurren en q y además ocurren en el mismo orden que en p . Si una secuencia p es una subsecuencia de una secuencia q , entonces se dice que p ocurre en q .

Definición 2. Una secuencia p es frecuente en D si p es una subsecuencia de al menos β textos de D , donde β es un umbral de frecuencia predefinido.

Definición 3. Una secuencia p es una secuencia frecuente maximal en D si no existe alguna otra secuencia p' en D tal que p es una subsecuencia de p' y p' es frecuente en D .

En este artículo, se propone un método para la generación automática de exámenes a partir de preguntas de definición de un texto en español considerando solamente las partes más importantes usando el método de descripciones multipalabra [1]. Para determinar cuál de las preguntas son las más importantes se aplica el método de las SFMs [15, 16] para identificar las partes de texto que son las más importantes para después generar la pregunta.

3 Extracción de Definición-Concepto

Dentro del estado del arte se encuentran trabajos cuyos objetivos están enfocados en responder preguntas de definición. Una pregunta de definición es aquella que tiene como respuesta una frase corta o conjunto de frases cortas que describen al concepto por el que se pregunta.

Responder este tipo de preguntas de manera automática no es una tarea fácil, ya que implica una búsqueda exhaustiva en grandes cantidades de información. Además, hay que tomar en cuenta que existen diferentes formas en las que una definición puede ser introducida en un texto, por lo tanto tratar de extraerla significa un gran esfuerzo [13].

El método propuesto indica que la etapa de pre-procesamiento el documento de normaliza, posteriormente pasa a la etapa de extracción definición-concepto que está inspirado en el modelo de espacio vectorial [16]. El texto se guarda en la entrada del método en un vector para posteriormente compararlo con una regla que se describe a continuación:

<texto frontera1>< CONCEPTO> <texto intermedio>< DESCRIPCIÓN ><texto frontera2>

<texto frontera1> Un, una, los, la, el, vacío

< CONCEPTO> Palabra o palabras que se van a definir

<texto intermedio> es, un, significa, se refiere

< DESCRIPCIÓN > Texto entre texto intermedio y el texto frontera

<texto frontera2> Se considera un punto (.)

El método funciona de la siguiente forma:

1. El texto es guardado dentro de un vector y cada palabra es analizada hasta encontrar palabras que coincidan con el texto intermedio (es, un, significa, se refiere).
2. Cuando se encuentra el texto intermedio, el método compara las 2 palabras que se encuentran a un lado hasta encontrar el texto frontera 1.
3. Si es encontrado el texto frontera 1, también es encontrado el concepto.



Figura 1. Vector con: a) texto intermedio, b) concepto y c) texto frontera 1.

4. El siguiente paso es encontrar del texto intermedio hacia la parte derecha del vector hasta encontrar un punto, y a esto se considera como la descripción.

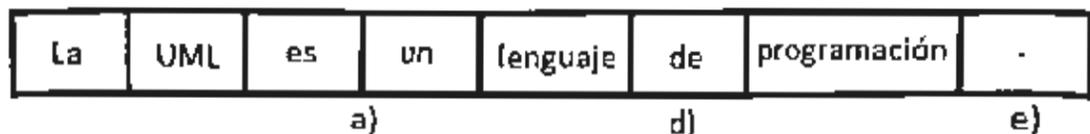


Figura 2. Vector con: a) texto intermedio, d) definición y e) texto frontera 2.

4 Método Propuesto

A continuación, se describen las etapas del método propuesto:

Pre-procesamiento: el método recibe como entrada un texto en forma digital que es pre-procesado para que sea reconocido por el módulo dos.

Extracción de definición-concepto: este módulo encuentra los posibles conceptos y su definición en un texto. Se han generado reglas que encuentren una secuencia de palabras específicas que sugieren que es un concepto y su definición, así se asegura que la respuesta está dentro del texto. Una vez que el método haya encontrado los conceptos y sus definiciones los guarda en una base de datos.

DBQA: Una vez almacenados en la base de datos el concepto y su definición, se genera otro par ordenado que es la pregunta y su respuesta y que es también almacenada en la base de datos.

Preguntas importantes: en esta etapa se generan las preguntas más importantes. Para realizar esto se utiliza el método Secuencias Frecuentes Maximales [14 - 15] para determinar los conceptos que se suponen más importantes del texto y se genere su respectiva pregunta.

Evaluación: la evaluación de los resultados obtenido fue realizada por los expertos.

En la figura 1, se muestra el método propuesto que contiene 5 etapas que permitan la generación automática de los exámenes a partir de las preguntas de definición de un texto en español.

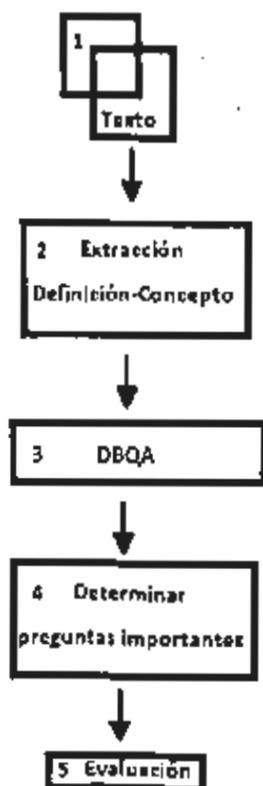


Figura 3. Esquema general del método propuesto compuesto por 5 etapas.

Cabe mencionar, en la segunda etapa, se encuentran todos los conceptos y sus respectivas definiciones (que son las respuestas). Los pares ordenados (concepto-definición) se guardan en una base de datos, para que después en la etapa tres DBQA se generen las preguntas de definición tomando los conceptos guardados previamente en la etapa dos.

5 Experimentación y trabajo futuro

El método propuesto fue implementando en el lenguaje de programación Java considerando varias ventajas, por ejemplo, trasladar a diferentes Sistemas Operativos sin necesidad de alterar el código.

Se realizaron las pruebas con 30 documentos en el área de la computación seleccionados aleatoriamente. El método se probó con el siguiente tipo de preguntas de definición:

<texto fronteral>< CONCEPTO> <texto intermedio>< DESCRIPCIÓN ><texto frontera2>

La evaluación de los resultados fue llevada a cabo por expertos que fueron los encargados de comparar los resultados obtenidos por el método y los obtenidos por ellos, coincidiendo las definiciones de ambos.

Como trabajo futuro, se probarán más tipos de preguntas de definición. En el estudio realizado, se ha encontrado más de 32 formas de encontrar una definición en un texto, por lo tanto se considerarán para diversificar exámenes generados. Asimismo, se probarán otros tipos de preguntas. Se utilizarán las colecciones de documentos más grandes. Una de las dificultades más importantes que se detectó es la naturaleza no-estructural de los textos utilizados. Se propondrá un método que pueda generar sus propias reglas para identificar cuando es una definición tomando como base el análisis de las descripciones multipalabra previamente almacenadas o creadas por el método propuesto.

Referencias

1. Yulia Ledeneva, Alexander Gelbukh, René García Hernández. Terms Derived from Frequent Sequences for Extractive Text Summarization. LNCS 4919, Springer-Verlag, ISSN 0302-9743, 2008.
2. Yulia Ledeneva, René García Hernández, Alexander Gelbukh. Multi-document Summarization using Maximal Frequent Sequences. Research in Computer Science, vol. 47, ISSN 1870-4069, 2010.
3. Yulia Ledeneva, Alexander Gelbukh, René García Hernández. Keeping Maximal Frequent Sequences Facilitates Extractive Summarization. Research in Computing Science, vol. 34, ISSN 1870-4069, 2008.
4. Ledeneva, Y.: Automatic Language-Independent Detection of Multiword Descriptions for Text Summarization, National Polytechnic Institute, PhD. Thesis, Mexico, 2009.
5. Esaú Villatoro-Tello, Luis Villaseñor-Pineda, Manuel Montes-y-Gómez, and David Pinto-Avendaño. Multi-Document Summarization Based on Locally Relevant Sentences. Mexican International Conference on Artificial Intelligence MICAI 2009. IEEE Computer Society, 2009.
6. Sidorov, G. Lemmatization in automatized system for compilation of personal style dictionaries of literary writers. In: "Word of Dostoyevsky", Russian Academy of Sciences, 1996.
7. Gelbukh, A. and Sidorov, G. Approach to construction of automatic morphological analysis systems for inflective languages with little effort. Lecture Notes in Computer Science, N 2588, Springer-Verlag, 2003.
8. Yulia Ledeneva, Grigori Sidorov: Recent Advances in Computational Linguistics. Informatica (Slovenia), 2010.
9. Bolshakov I., Gelbukh A. Computational Linguistics: Models, Resources, Applications. IPN-UNAM-FCE, ISBN 970-36-0147-2, 2004.
10. M. C. Denicia-Carral, M. Montes-y-Gómez, L. Villaseñor-Pineda, R. García Hernández. A Text Mining Approach for Definition Question Answering. 5th International Conference on Natural Language Processing, FinTal 2006. Turku, Finland, August 2006. Lecture Notes in Artificial Intelligence 4139, Springer 2006.
11. R. M. Aceves-Pérez, L. Villaseñor-Pineda, M. Montes-y-Gómez Using N-gram Models to Combine Query Translations in Cross-Language Question Answering. International Conference on Intelligent Text Processing and Computational Linguistics CICLing-2006. Mexico City, Mexico, February 2006. Lecture Notes in Computer Science 3878, Springer, 2006.
12. Mihalcea R. Random Walks on Text Structures. CICLing 2006, LNCS, vol. 3878, Springer-Verlag, 2006.

13. M. C. Denicia Carral. Respondiendo a Preguntas de Definición mediante el Descubrimiento de Patrones Léxicos. Tesis de maestría, 2007.
14. R. A. García-Hernández, J. F. Martínez-Trinidad, J. A. Carrasco-Ochoa. A Fast Algorithm to Find All the Maximal Frequent Sequences in a Text, 9th Iberoamerican Congress on Pattern Recognition (CIARP), LNCS vol. 3287, Springer-Verlag 2004.
15. R. A. García-Hernández, J. F. Martínez-Trinidad, J. A. Carrasco-Ochoa. New Algorithm for Fast Discovery of Maximal Sequential Patterns in a Document Collection, CILing, LNCS, vol. 2945, Springer-Verlag 2006.
16. Baeza-Yates R. Modern Information Retrieval. Addison Wesley Longman Publishing Co. Inc., 1999.
17. P. Ingwersen. Information Retrieval Interaction, (Ed. Taylor Graham, Los Ángeles CA), (2002).
18. Manning C., P. Raghavan, H. Schütze. An Introduction to Information Retrieval. Cambridge University Press, 2007.
19. J. L. Vicedo. La Búsqueda de Respuestas: Estado Actual y Perspectivas de Futuro. Inteligencia Artificial, Revista Iberoamericana de Inteligencia Artificial, 2004.
20. A. Singhal. Modern Information Retrieval: A Brief Overview. In IEEE Data Engineering Bulletin, 2001.
21. Luis Martínez, Francisco Mata, Pedro J. Sánchez. "Sigebaja": Generación y Evaluación Automática de Exámenes, Sistema con Autenticación, Privacidad e Integridad. 2011
22. Herramienta de Softonic. "Exámenes" - Generador de exámenes de forma automática y manual". <http://exámenes.softonic.com/> Versión 1.20
23. Herramienta de Softonic. "Hot Potatoes" - Seis utilidades para creación de exámenes y ejercicios en la red. <http://hot-potatoes.softonic.com/descargar> Versión 6.3.0.4

Cálculo de FS-Testores Difusos Utilizando Algoritmos Genéticos

Maria Esther Guevara Cruz¹, Saturnino Job Morales Escobar², Ivonne Rodríguez Pérez², José Francisco Martínez Trinidad¹

¹Universidad Tecnológica Fidel Velázquez.
ttguevara@yahoo.com.mx

²Centro Universitario UAEM Valle de México.
sjob_d@yahoo.com, irodriguezperez@yahoo.com

³Instituto Nacional de Astrofísica, Óptica y Electrónica.
fmartine@inaoep.mx

Resumen. El problema de selección de variables en el enfoque Lógico Combinatorio del Reconocimiento de Patrones, se aborda desde dos variantes, para reducir la dimensión del espacio de representación o para determinar aquellas que son más importantes en la aparición de algún evento. En ambas, se ha utilizado el concepto de testor o extensiones del mismo, por ejemplo, los *FS-testores difusos*. Sin embargo, la complejidad computacional para su cálculo es exponencial. En este trabajo se presenta una propuesta para encontrar *FS-testores típicos difusos* utilizando Algoritmos Genéticos.

Palabras clave. Reconocimiento de Patrones, Testores típicos, Clasificación Supervisada, Selección de Variables, Algoritmos Genéticos.

1 Introducción

Actualmente, se tiene la capacidad de obtener datos con alta calidad y en grandes cantidades. Este hecho da la posibilidad de utilizar herramientas de análisis e interpretación de información para apoyar la toma de decisiones críticas basadas en la formación de grupos, clasificación de objetos o selección de variables. Sin embargo, el desarrollo de esas herramientas depende fuertemente de los modelos para el procesamiento de la información.

Un problema importante que se presenta en el proceso de modelación es la representación de los objetos que intervienen en el problema a tratar. En el enfoque Lógico Combinatorio del Reconocimiento de Patrones (RP) [1, 2, 3], los objetos se describen mediante un conjunto de variables que da lugar a un espacio de representación formado por el producto cartesiano de los dominios de dichas variables. Así, cada objeto se representa por una n -ada donde se pueden mezclar variables cuya naturaleza puede ser numérica o no numérica. En problemas de selección de variables, se

consideran dos objetivos: uno reducir la dimensión del espacio donde se ubican los objetos o encontrar variables con mayor importancia. Aunque en la mayoría de las investigaciones sobre el problema de selección de variables solamente se consideran variables cuantitativas, se encuentran reportes con variables mezcladas [9]. Este problema de selección de variables está fuertemente relacionado con el problema de clasificación supervisada. En este contexto, la selección de variables se ha tratado utilizando el concepto de Testor [4]. Inicialmente, un testor fue definido como el subconjunto de variables que permite hacer la diferenciación completa de objetos de clases diferentes, posteriormente, con base en las necesidades de modelación de cada problema, se han introducido extensiones de este concepto. Estas extensiones permiten resolver problemas para los cuales los anteriores no son adecuados. Algunas extensiones son: testores de *Goldman's*, ε -testores, k -testores, y el que se define en [1], el *FS-testor*.

En la selección de variables utilizando el concepto de *FS-testor*, la descripción de los objetos se puede hacer con valores cuantitativos, cualitativos o simbólicos y la pertenencia de un objeto a una clase se da en términos de pertenece o no a una clase, así, a cada objeto se le puede asociar un *l-uplo* de pertenencia a las clases, indicando con cero no pertenece y 1 si pertenece.

Aún en problemas donde la cantidad de variables no es muy grande, se debe verificar la definición del *FS-testor* en todos los posibles subconjuntos de variables, y por lo tanto la complejidad computacional para calcularlos depende exponencialmente de la cantidad de variables usadas. Esto provoca que el concepto de *FS-testor* sea muchas veces inaplicable en problemas prácticos. El presente trabajo plantea el uso de algoritmos genéticos para el cálculo de *FS-testores difusos* como una alternativa que calcula sólo un subconjunto del espacio total de soluciones.

Este trabajo está organizado de la siguiente manera. En la sección 2 se presentan conceptos básicos y la notación utilizada. En la sección 3, se plantean los conceptos de *FS-testor* y de la familia difusa de los *FS-testores* y por último, en la sección 4, se expone la propuesta del cálculo de *FS-testores* utilizando un Algoritmo Genético.

2 Conceptos Básicos.

Con el objetivo de presentar el concepto de *FS-testor*, daremos algunas definiciones y la notación utilizada en el presente documento, como se define en [2, 4].

Sea U un universo de objetos estructurados en l clases K_1, \dots, K_l . Los objetos están descritos en términos del conjunto de variables $R = \{x_1, \dots, x_n\}$, para cada elemento $O_i \in U$ la descripción está dada por el n -uplo $I(O_i) = (x_1(O_i), \dots, x_n(O_i))$. Sea M_p el conjunto de valores admisibles de la variable x_p , $p = 1, \dots, n$, es decir, $x_p(O_i) \in M_p, i = 1, \dots, m$. Para cada objeto O_i de U hay una l -upla de pertenencia $\alpha(O_i) = (\alpha_1(O_i), \dots, \alpha_l(O_i))$, en donde

$\alpha_j(O_i) \in \{0,1\}$, 1 indica que el objeto pertenece a la clase $K_j, j=1, \dots, l$. Para efectos del cálculo, esta información se almacena en una estructura denominada matriz de aprendizaje (MA). La figura 1 muestra de manera general el contenido de dicha matriz de aprendizaje.

	x_1	\dots	x_n	α_1	\dots	α_l
O_1	$x_1(O_1)$	\dots	$x_n(O_1)$	$\alpha_1(O_1)$	\dots	$\alpha_l(O_1)$
\vdots	\vdots	\vdots	\vdots	\vdots	\vdots	\vdots
O_m	$x_1(O_m)$	\dots	$x_n(O_m)$	$\alpha_1(O_m)$	\dots	$\alpha_l(O_m)$

Fig. 1. Matriz de Aprendizaje MA

Definición 2.1. Un criterio de comparación para la variable $x_i \in R$ es una función $C_i : M_i \times M_i \rightarrow V_i$, en donde $\forall i, V_i$ es un conjunto totalmente ordenado. Esta función da el grado de similitud entre un par de valores admisibles para x_i . C_i puede ser booleano, k-valente o real.

Definición 2.2. Sea $T = \{x_{p1} | \mu_T(x_{p1}), \dots, x_{ps} | \mu_T(x_{ps})\}$ un conjunto difuso de R . Una subdescripción o descripción parcial de O_j en términos de los variables de T es la s-tupla

$$I_T(O_j) = \left(\left(x_{p1}(O_j), \tau(x_{p1}) \right), \dots, \left(x_{ps}(O_j), \tau(x_{ps}) \right) \right).$$

Definición 2.3. Sea $M_{i_1} \times \dots \times M_{i_s}$ para cualquier $T = \{x_{i_1}, \dots, x_{i_s}\} \subseteq R$, siendo R el conjunto de variables en términos de los que son descritos los objetos, definimos una función $\beta : (M_{i_1} \times \dots \times M_{i_s})^2 \rightarrow V$, donde V es un conjunto totalmente ordenado y β es una función de semejanza (parcial) con denominaciones análogas a C_i en dependencia de V , la cual cumple $\beta_T(O_i, O_j) \geq \max_{O' \in M} \{\beta_T(O, O')\}$, siendo M el conjunto de una muestra de objetos del universo. Cuando $s = n$, diremos que β es una función de semejanza total. Con el propósito de simplificar la notación, usaremos $I(O_i)$ o simplemente O_i para denotar al objeto y a su descripción.

También, consideraremos la imagen de la función β en $V = [0,1]$, MA como se definió en la Figura 1, y la función v para comparar l -uplos de pertenencia: $v' : V^l \times V^l \rightarrow V$.

3 FS-testores

El concepto de *FS-testor*, se basa en la idea de que las descripciones de objetos que pertenecen a una clase deben ser más semejantes que aquellas descripciones de objetos que no pertenecen a la misma clase.

Para el concepto de *FS-testor* se considera lo expuesto en [1, 4, 6].

Sea MA como en la Figura 1 y R el conjunto de variables por el cual se describen los objetos. Para MA se ha definido una función ν de comparación de l -uplos de pertenencia a las clases:

$$\nu: [0,1]^l \times [0,1]^l \rightarrow V'$$

y D' es el subconjunto de V' donde dos l -uplos de pertenencia se consideran semejantes y β una función de semejanza.

Definición 3.1 (Propiedad diferenciante). $T \subseteq R$ (T subconjunto difuso de R) es un conjunto *FS-diferenciante de variables* con respecto a ν , D' y β de MA si:

$$\forall O_i, O_j \in MA \left[\nu(\alpha(O_i), \alpha(O_j)) \notin D' \right] \Rightarrow \left[\beta_T(O_i, O_j) \leq \beta_R(O_i, O_j) \right]$$

Si analizamos las propiedades informativas de un subconjunto de variables, es conveniente no sólo tener en cuenta su capacidad de diferenciar objetos de clases diferentes, sino también de mantener un buen parecido entre los objetos de una misma clase.

Definición 3.2 (Propiedad caracterizante). $T \subseteq R$ es un conjunto *FS-caracterizante de variables* con respecto a ν , D' y β de MA si y sólo si

$$\forall O_i, O_j \in MA \left[\nu(\alpha(O_i), \alpha(O_j)) \in D' \right] \Rightarrow \left[\beta_R(O_i, O_j) \leq \beta_T(O_i, O_j) \right]$$

La siguiente definición de testor es más restrictiva con respecto a las otras extensiones del concepto de testor. Para que un subconjunto de variables la pueda cumplir, tanto su capacidad diferenciante como caracterizante no deben ser peores que la del conjunto de referencia.

Definición 3.3 (FS-Testor). $T \subseteq R$ es un *FS-testor* con respecto a ν , D' y β de MA si y sólo si es a la vez un conjunto *FS-diferenciante* y *FS-caracterizante* de MA con respecto a los mismos parámetros.

El Conjunto Difuso de los FS-testores

La definición 3.3 de un subconjunto T de R es más restrictiva que las definiciones anteriores de testor por lo que en muchos casos resulta difícil de cumplir (puede darse

el caso en que para un problema particular no encontremos tal subconjunto de R , lo que significaría resolver el problema considerando todo el espacio de representación con sus correspondientes costos computacionales). Es en estos casos cuando se podría experimentar aunque sea con un subconjunto de R que se acerque más al cumplimiento de la definición 3.3. Esto da lugar a definir el concepto de la familia difusa de los *FS-testores* de la siguiente manera: todos los subconjuntos que cumplan la definición 3.3 pertenecen con grado 1, el resto de los subconjuntos pertenecen con un grado en el intervalo abierto $[0,1)$. El grado de pertenencia de un subconjunto será mayor en la medida en que se acerque más al cumplimiento de la propiedad expresada en la definición 3.3.

Para definir la familia de *FS-testores*, consideremos lo siguiente: sea $T \subseteq R$ y los conjuntos:

$$OD = \{(O_i, O_j) \in MA \mid v(\alpha(O_i), \alpha(O_j)) \notin D'\}$$

el conjunto de pares de objetos con l -uplos de pertenencia diferentes y $C(OD)$ el complemento del conjunto OD es decir,

$$C(OD) = \{(O_i, O_j) \in MA \mid v(\alpha(O_i), \alpha(O_j)) \in D'\}$$

el conjunto de pares de objetos con l -uplos de pertenencia semejantes.

Sean $S^R(T)$ y $D^R(T)$ los siguientes conjuntos:

$$S^R(T) = \{(O_i, O_j) \in OD, \beta_T(O_i, O_j) \succ \beta_R(O_i, O_j)\},$$

es decir, los pares de objetos con l -ulos de pertenencia diferentes y que no cumplen la propiedad diferenciante.

$$D^R(T) = \{(O_i, O_j) \in C(OD), \beta_T(O_i, O_j) \prec \beta_R(O_i, O_j)\},$$

es decir, los pares de objetos con l -uplos de pertenencia semejantes y que no cumplen la propiedad caracterizante.

Definición 3.4. Denominaremos *familia difusa de los FS-testores* de MA con respecto v, D', R y β , al conjunto difuso

$$\zeta = \{T_p \mid \mu_\zeta(T_p): T_p \subseteq R\}$$

donde

$$\mu_\zeta(T_p) = 1 - \frac{|S^R(T_p) \cup D^R(T_p)|}{|OD \cup C(OD)|}.$$

El grado de pertenencia de un subconjunto a la familia difusa de los *FS-testores* de una MA, evalúa la medida en que este subconjunto cumple la definición 3.3 desde un punto de vista "cuantitativo". Sin embargo, esta magnitud no tiene en cuenta un aspecto "cualitativo" importante, en qué medida se diferencian los valores de semejanza de un par de objetos en el subconjunto analizado y el conjunto de referencia.

A continuación se define el grado de pertenencia tomando en cuenta los dos aspectos abordados anteriormente para los casos en que la función de semejanza tenga imagen en el intervalo $[0,1]$:

$$\mu_{\zeta}(T_p) = \delta_1 \eta_1(T_p) + \delta_2 \eta_2(T_p) + \delta_3 \eta_3(T_p)$$

donde

$$\eta_1(T_p) = 1 - \frac{|S^R(T_p) \cup D^R(T_p)|}{|OD \cup C(OD)|}$$

$$\eta_2(T_p) = \begin{cases} 1 & \text{si } S^R(T_p) = \emptyset \\ 1 - \frac{1}{|S^R(T_p)|} \sum_{(O_i, O_j) \in S^R(T_p)} [\beta_{T_p}(O_i, O_j) - \beta_R(O_i, O_j)] & \text{si } S^R(T_p) \neq \emptyset \end{cases}$$

$$\eta_3(T_p) = \begin{cases} 1 & \text{si } D^R(T_p) = \emptyset \\ 1 - \frac{1}{|D^R(T_p)|} \sum_{(O_i, O_j) \in D^R(T_p)} [\beta_R(O_i, O_j) - \beta_{T_p}(O_i, O_j)] & \text{si } D^R(T_p) \neq \emptyset \end{cases}$$

y δ_1 , δ_2 y δ_3 son coeficientes de ponderación tales que $0 \leq \delta_i \leq 1$, $i = 1, 2, 3$ y $\delta_1 + \delta_2 + \delta_3 = 1$

El concepto de *FS-testor* es útil en la medida que sea posible determinar cuándo dos l -uplos de pertenencia son semejantes y cuándo no.

El valor $\mu_{\zeta}(T_p)$ nos dice en que medida un subconjunto de variables T_p es un *FS-testor*, es decir, que tan buenas son las descripciones de los objetos en términos de estas variables para diferenciar entre objetos de clases con l -uplos de pertenencia no semejantes, y por otro lado, hacer más semejantes a los objetos que están en la misma clase.

4 Algoritmo Genético

Una herramienta que en la actualidad ha apoyado a los problemas de optimización son los llamados algoritmos genéticos (AG), los cuales, establecen una analogía entre el conjunto de soluciones de un problema, llamado fenotipo, y el conjunto de individuos de una población natural, codificando la información de cada solución en una cadena, generalmente binaria, llamada cromosoma. Los símbolos que forman la cadena son llamados genes. Cuando la representación de los cromosomas se hace con cadenas de dígitos binarios se le conoce como genotipo.

Los cromosomas evolucionan a través de iteraciones, llamadas *generaciones*. En cada generación, los cromosomas son evaluados usando alguna *medida de aptitud*. Las siguientes generaciones (nuevos cromosomas), llamada *descendencia*, se forman utilizando dos operadores:

- a) Combinando dos cromosomas de la generación actual usando un operador de *cruza*.
- b) Modificando un cromosoma usando un operador de *mutación*.

El funcionamiento de un algoritmo genético básico es el siguiente:

Primero, se genera aleatoriamente la población inicial, que está constituida por un conjunto de cromosomas, que representan las posibles soluciones del problema. En caso de no hacerlo aleatoriamente, es importante garantizar que dentro de la población inicial, se tenga la diversidad estructural de estas soluciones para tener una representación de la mayor parte de la población posible o al menos evitar la convergencia prematura.

A cada uno de los cromosomas de esta población se aplicará la función de aptitud para saber qué tan "buena" es la solución que se está codificando.

Después de saber la aptitud de cada cromosoma se procede a elegir los cromosomas que serán cruzados en la siguiente generación.

Los cromosomas con mejor aptitud tienen mayor probabilidad de ser seleccionados. Existen dos métodos más comunes de selección: El de la ruleta, y el Torneo.

El cruzamiento es el principal operador genético, representa la reproducción sexual, opera sobre dos cromosomas a la vez para generar dos descendientes donde se combinan las características de ambos cromosomas padres.

Los operadores de cruzamiento más utilizados en la representación binaria de los cromosomas son: De punto, de Dos Puntos y Uniforme.

El desempeño de un algoritmo genético depende, pues, de estos dos operadores.

El AG se deberá detener cuando se alcance la solución óptima, pero ésta generalmente se desconoce, por lo que se deben utilizar otros criterios de detención. Normalmente se usan dos criterios: correr el AG un número máximo de iteraciones (generaciones) o detenerlo cuando no haya cambios en la población.

Como se mencionó, el problema de selección de variables se puede ver como un problema de optimización, ya que queremos encontrar, bajo alguna heurística, el subconjunto de variables que permitan hacer mejor la diferenciación y las semejanzas de objetos de clases diferentes y de la misma clase respectivamente.

Se tienen algunas propuestas que utilizan a los algoritmos genéticos para la búsqueda de los Testores y Testores Típicos: Algoritmo Genético para Calcular Testores

Típicos de Costo Mínimo [5], Algoritmo Genético para Calcular FS-Testores Difusos [6], Algoritmo Genético para el Cálculo de Φ -testores Difusos [7]. El algoritmo que a continuación presentamos es una propuesta que aparece en [8], y que toma como base el trabajo de [6]. El algoritmo es una alternativa nueva para el cálculo de los *FS-testores difusos*.

El algoritmo propuesto, retoma los conceptos de *FS-testor difuso* expuesto en [1], donde para que un subconjunto de atributos $T \subseteq R$ se considere un *FS-testor* de *MA* con respecto a v, D', β y R , debe cumplir las propiedades diferenciante y caracterizante, así como también los conceptos relacionados a la familia difusa de *FS-testores* y el grado de pertenencia a la misma.

La propuesta para el cálculo de los *FS-testores difusos*, incorpora el análisis de las capacidades informativas de los subconjuntos de variables que pertenecen a la familia de los *FS-testores*, ya que se cuantifica cuán cerca o no está un subconjunto de cumplir la definición de *FS-testor*.

El algoritmo genético, trabaja directamente con la *MA* y la función de semejanza que un problema imponga.

El subconjunto de n variables T con los que se pueden describir a los objetos de *MA*, se consideran como un individuo, mismo que tiene una representación en forma de un n -uplo binario, cero representa la ausencia de la variable, y 1 su presencia.

Los parámetros con los que trabaja el algoritmo son: tamaño de la población inicial (m)-generada aleatoriamente-, número de iteraciones, y el número de *FS-testores* difusos que se desea calcular (w).

La función de aptitud y los operadores utilizados se describen a continuación:

Función de Aptitud. Como función de aptitud se toma

$$\mu_{\zeta}(T_p) = \delta_1 \eta_1(T_p) + \delta_2 \eta_2(T_p) + \delta_3 \eta_3(T_p) .$$

Cruza. El operador de cruza aplicado en este algoritmo es de un punto, para esto se elige el gen que ocupa la posición correspondiente al punto medio de la cadena, llamemos a este punto p , cuando el valor de n es un entero impar, se toma al gen que se encuentra en una posición posterior al punto medio.

Para aplicar el operador de cruza, la población es evaluada y ordenada de manera descendente de acuerdo con la función de aptitud. A continuación se toma a dos individuos de la población, al más apto y al menos apto, como padres para generar dos descendientes.

El proceso se repite de la misma forma con el resto de la población, tomando al individuo más apto y al menos apto que sigan a los tomados en el paso anterior. Si el tamaño de la población es impar, el individuo menos apto no es considerado para la cruza, tomando así para la primera cruza al que le sigue ascendentemente.

Mutación. El operador de mutación utilizado es de un bit, es decir, se toma un gen del individuo aleatoriamente para invertir su valor.

Selección. Los mejores individuos, los que obtuvieron la aptitud más alta, pasarán a la siguiente generación.

El pseudocódigo que expresa el algoritmo se muestra a continuación:

Algoritmo genético para calcular FS-testores difusos (agfst).
 ENTRADA: MA , número de individuos de la población original ($tamPob$), número de iteraciones ($numIter$), número de mejores individuos a obtener (w).

SALIDA: El conjunto W con los w individuos que son *FS-testores difusos*, o que se acercan más a serlo.

1. Lee($MA, tamPob, numIter, w$)
 $W \leftarrow \emptyset$
 2. generaPoblación($tamPob$)
 3. quitaRepetidosCeros($poblaciónActual, tamPob$)
 4. evaluaPoblación($poblaciónActual$)
 5. ordenaPoblación($poblaciónActual$)
 6. Para $i=1$ a $i=numIter$
 - a. cruza($poblaciónActual$)
 - b. muta($poblaciónActual$)
 - c. quitaRepetidosCeros($nuevaPoblación, tamPobN$)
 - d. evaluaPoblación($nuevaPoblación$)
 - e. ordenaPoblación($nuevaPoblación$)
 - f. obtenFSt($nuevaPoblación, w$)
 - g. actualizaPoblación($nuevaPoblación$)
 7. Escribe (W)
- Fin.

En cada iteración, con el número de individuos (m'') se puede presentar alguno de los siguientes casos:

$m'' \geq w$, en éste, se seleccionan de la población a los w mejores individuos.

Si $m'' < w$, entonces los m'' individuos que en ese momento existan como población se agregan a los w mejores. Esto genera un conjunto de $m'' + w$ individuos, mismos que son ordenados de acuerdo con su valor de aptitud. De este conjunto ordenado se selecciona a los primeros w individuos.

5 Conclusiones

El concepto de testor desarrollado bajo el enfoque Lógico Combinatorio del Reconocimiento de Patrones, ha permitido ajustarlo a las condiciones de problemas reales a partir de la modelación matemática del mismo. El algoritmo genético propuesto proporciona una alternativa al problema del cálculo de *FS-testores difusos* y así la utilización de este concepto en la solución a problemas de selección de variables.

El desarrollo del algoritmo genético para el cálculo de los *FS-testores difusos* responde a la necesidad de contar con herramientas computacionales que apoyen al proceso de clasificación y en forma general al análisis de datos de forma efectiva y eficiente, sobre todo en aquellos problemas en los que las descripciones de los objetos

se dan en términos cuantitativos y cualitativos y para los cuales los modelos cuantitativos no se ajustan tan fácilmente.

Sin embargo, hay que hacer mención que el éxito de la presente propuesta depende fuertemente de la modelación de las funciones utilizadas para comparar tanto las variables (criterios de comparación) como de la función que compara los objetos (función de semejanza). En relación a esta última, debemos destacar que se puede utilizar la misma función de semejanza para el proceso de clasificación como para la selección de variables.

Con este algoritmo se hicieron pruebas que se reportan en [8], sin embargo, se considera necesario realizar nuevos experimentos que muestren su desempeño utilizando otras bases de datos e incluso se pruebe modificando los operadores del AG y se utilicen otras funciones de semejanza.

Referencias.

1. E. Alba.: Nuevas extensiones del concepto de testor para diferentes tipos de funciones de semejanza. Tesis Doctoral ICIMAF, Cuba (1997).
2. J. Ruiz Shulcloper, A. Guzmán Arenas, J. F. Martínez Trinidad.: Enfoque Lógico Combinatorio al reconocimiento de Patrones. Selección de Variables y Clasificación Supervisada, IPN, México (1999).
3. J. F. Martínez-Trinidad, A. Guzmán-Arenas.: The logical combinatorial approach to pattern recognition an overview through selected works. In: Pattern Recognition 34(4), pp 741-751 (2001).
4. M. Lazo-Cortes, J. Ruiz-Shulcloper, E. Alba-Cabrera.: An overview of the evolution of the concept of testor. In: Pattern Recognition, 34(4), 753-762, (2001).
5. Sánchez, G.,Lazo, M., Fuentes, O.:Algoritmo Genético para calcular testores de costo mínimo. In: SIARP'99, pp. 207-213, La Habana, Cuba, (1999).
6. Martínez,F., Sánchez, G., Rugeiro, B. :GeneticAlgorithm to Compute FS-Testors. In: WESEAS Transaction on Systems, Issue 2, Vol. 1, pp. 267-272 (2002).
7. Santos, G., José, A.: Algoritmo Genético para el cálculo de los ϕ -Testores Difusos, Tesis para obtener el grado de Maestro en Ciencias Computacionales, Instituto Nacional de Astrofísica Óptica y Electrónica, México (2003).
8. Guevara, M. Esther.: Algoritmo Genético para la selección de variables y cálculo de su importancia informacional, usando el concepto de FS-Testore Difuso. Tesis para obtener el grado de Maestro en Ciencias de la Computación, Benemérita Universidad Autónoma de Puebla, México (2004).
9. Wenyin Tang , K.Z. Mao.: Feature selection algorithm for mixed data with both nominal and continuous features. In: Pattern Recognition Letters, Volume 28, Issue 5, pp. 563-571 (2007).

Optimización de Parámetros de una Línea de Estampado Usando Redes Neuronales y Estrategias Evolutivas

Susana Gutiérrez, Giovanni Lizárraga, Marco Jimenez.

Corporación Mexicana de Investigación en Materiales S.A. de C.V, Ciencia y Tecnología
790, 25290, México.

{sgutierrez@comimsa.com, glizarraga@comimsa.com, marcojimenez@comimsa.com}

Resumen. Actualmente, en la industria manufacturera tener procesos eficientes y eficaces es muy importante debido a que proporcionan estabilidad permitiendo la calidad en los productos. La optimización de parámetros esenciales en los sistemas de alimentación de prensas permite que estas trabajen de una mejor forma, incrementando el tiempo de vida de los dispositivos involucrados como máquinas y troqueles. En este artículo se usan redes neuronales artificiales para modelar un sistema de troquelado, para posteriormente realizar la optimización de parámetros por medio de estrategias evolutivas, con el objetivo de lograr la mejor sincronización de los equipos en la producción de piezas del sistema de estampado estudiado.

Palabras Clave: Redes neuronales artificiales, Sistema de estampado, Alimentación de prensas, Estrategias evolutivas.

1 Introducción

El estampado de metales ayuda a la formación de hojas o láminas permitiendo obtener productos esenciales para la vida diaria como es el caso de escritorios, archiveros, carrocerías de autos, fuselajes de aviones, latas de bebidas, joyería, entre otros [1]. En el estampado se utilizan prensas para que por medio de la presión se pueda formar la pieza requerida. En el proceso de estampado y conformado de metales se sigue una serie de pasos que permiten obtener productos de forma específica, este proceso inicia con la lámina ya sea en rollo o placa, para posteriormente realizar diferentes operaciones por ejemplo: el ranurado, corte, serrado o enderezado, después se procede al formado de la pieza mediante troqueles (punzones, dados) ya sean éstos compuestos, progresivos o de transferencia. En la operación de troquelado se pueden realizar las siguientes operaciones: punzonado, corte de piezas, estampado, doblado, formado de rodillos, embutido, entre otros. Finalmente se procede a la limpieza de la pieza y la pintura en caso de ser necesario. Debido a la complejidad del proceso de troquelado se requiere un estricto control en cada uno de sus pasos.

La industria manufacturera depende de que sus procesos de estampado y conformado sean eficaces para mantenerse en el mercado en forma competitiva, con

este fin las empresas utilizan maquinaria automatizada como: porta-rollos, enderezadores, empujadores, lubricadores, pero esto no es suficiente debido a que estas máquinas independientes unas de otras.

Al realizar la automatización de un proceso, como por ejemplo la alimentación de un troquel, es frecuente que aparezcan varios parámetros que requieren ser sincronizados y optimizados para que el sistema automatizado opere de manera óptima. En muchas ocasiones, la sincronía y optimización de los parámetros se hace a prueba y error y no se sigue la metodología científica. El objetivo es la optimización de la alimentación a una prensa utilizada en el proceso de troquelado, mediante la sincronización de cada una de las máquinas que intervienen en la alimentación a la prensa. Con la finalidad de lograr el objetivo del trabajo se utilizó una red neuronal y el método de estrategias evolutivas, considerando como variables principales: la velocidad de alimentación, la presión de lubricación, el nivel de lubricación y el avance de la lámina a la prensa.

En la literatura se pueden encontrar varios temas utilizados para la solución de problemas en los sistemas de estampado como el tomado por Fadón et al. [2], en el que se muestra el proceso de diseño de un troquel basados en la animación por medio del modelado sólido, en el cual se diseñaron las piezas que componen el ensamblaje y posteriormente se realizó una simulación lo que les permitió obtener una animación del funcionamiento del troquel. Considerando que el estampado de metal en blanco depende de la habilidad del operador A.Y.C. Neea and V.C. Venkatesha [3] desarrolló un algoritmo heurístico para optimizar el material, teniendo en cuenta factores como el mínimo desperdicio de material, corrección de la anchura de la lámina y su orientación, proporcionando información sobre la alineación y precisión del centro de la prensa. Yanagida y Azushima [4] desarrollaron un estudio con el fin de medir el coeficiente de fricción del estampado en caliente utilizando una simulación numérica mediante elemento finito, con el fin de disminuir la carga y el desgaste en el proceso de estampado evaluando en el sistema de estampado las condiciones de lubricado y los lubricantes que se usan en forja caliente. En el estudio para la optimización de parámetros en diferentes áreas se encontró la investigación de Jorge Hurtado [5], la cual estudia los parámetros en el diseño de estructuras complejas utilizando estructuras metálicas, con el fin de simplificar el trabajo se utilizan redes neuronales artificiales (RNA) para representar el comportamiento del sistema, para posteriormente combinarlo con algoritmos evolutivos (AE) como método de optimización global de las funciones considerando condiciones de probabilidad. Para el estudio de desgaste Marek Balazinski [6], utiliza 3 métodos de inteligencia artificial para estimar el desgaste de la herramienta en un torno realizando una comparación entre los métodos llamados "convencionales" (redes neuronales y lógica difusa) con la propuesta del autor. Esta propuesta consiste en usar una combinación de red neuronal de retro-propagación y el método de lógica difusa para determinar el desgaste de la herramienta por medio de la fuerza de corte.

En los artículos mencionados, se puede observar que no se ha considerado la alimentación de prensas como un punto importante en la formación de productos, a pesar de que éste se encuentra antes de la fabricación y que contribuye en evitar el desgaste de los herramientales y el desperdicio de material en el proceso. Por lo que se decidió probar un sistema que hace uso de redes neuronales y estrategias evolutivas para ajustar los parámetros de un sistema de alimentación de un troquel.

Debido a que este es un trabajo de aplicación, nos enfocamos en usar técnicas básicas y bien establecidas de inteligencia artificial, con la idea de partir de las técnicas más sencillas y dependiendo de los resultados, continuar con técnicas más avanzadas. Por ejemplo, una red "feedforward" con una capa oculta y estrategias evolutivas, son dos herramientas clásicas en inteligencia artificial que han sido usadas con éxito con anterioridad y cuya implementación es sencilla. Por lo que se consideraron estas técnicas como herramientas iniciales para resolver el problema.

El resto del artículo se organiza como sigue: en la Sección 2, se da una explicación de las redes neuronales artificiales. En la Sección 3, se explica en qué consiste el método de optimización a través de estrategias evolutivas. En la Sección 4, se muestra un caso de aplicación de estas metodologías de inteligencia artificial para ajustar parámetros. Finalmente, en la Sección 5, se dan las conclusiones.

2 Redes Neuronales Artificiales

2.1 Conceptos Generales

El estudio de redes neuronales artificiales es parte de la inteligencia artificial. Las redes neuronales artificiales son sistemas inspirados en el cerebro humano, que se utilizan para aproximar funciones o a la clasificación de patrones, entre otros usos. Los estudios en RNA empezaron con el trabajo de McCulloch y Pitts [7] y fue seguido por Rosenblatt [8], Widrow y Hoff [9], Minsky y Papert [10], y muchos otros.

Un ejemplo de una RNA se puede ver en la Figura 1, donde podemos ver algunos de los elementos de una RNA.

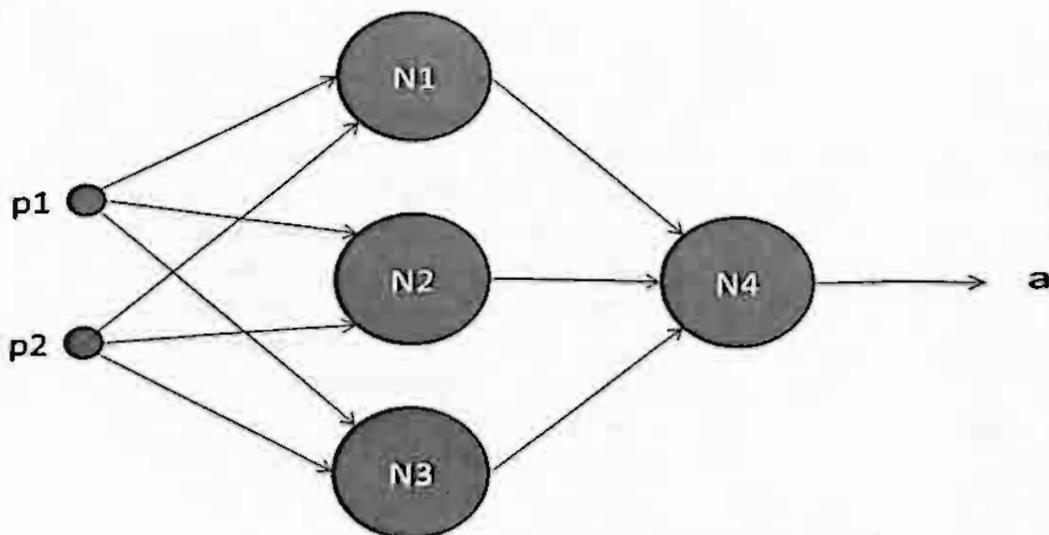


Figura 1. Ejemplo de una red neuronal artificial

En este ejemplo tenemos dos valores de entrada representada por P1 y P2, pero una RNA puede tener cualquier número finito de entradas. Las neuronas en esta red están representadas por los círculos N1, N2, N3 y N4. Las neuronas N1, N2, N3 forman la primera capa de la red. La primera capa recibe y procesa las entradas de la red, y transmite una salida a la segunda capa. La segunda capa consiste en N4. En este ejemplo, la segunda capa recibe los resultados de la primera capa y devuelve la salida de la red (a). Esta red representa una función que recibe dos valores (P1 y P2) y devuelve un valor (a), las RNA puede tener varias capas. La última capa se conoce como capa de salida y devuelve la respuesta de la red a los patrones de entrada. Las capas entre la capa de salida y las entradas se llaman capas ocultas. Las capas ocultas hacen la mayor parte del trabajo cuando se aproxima una función. Cada capa de una RNA, incluyendo la capa de salida, puede tener un número arbitrario de neuronas. Cada neurona de una capa transmite una señal a cada neurona de la capa siguiente. Esta topología se inspira en la forma en que las neuronas están interconectadas en el cerebro.

La topología general de una neurona en una RNA se muestra en la Figura 2.

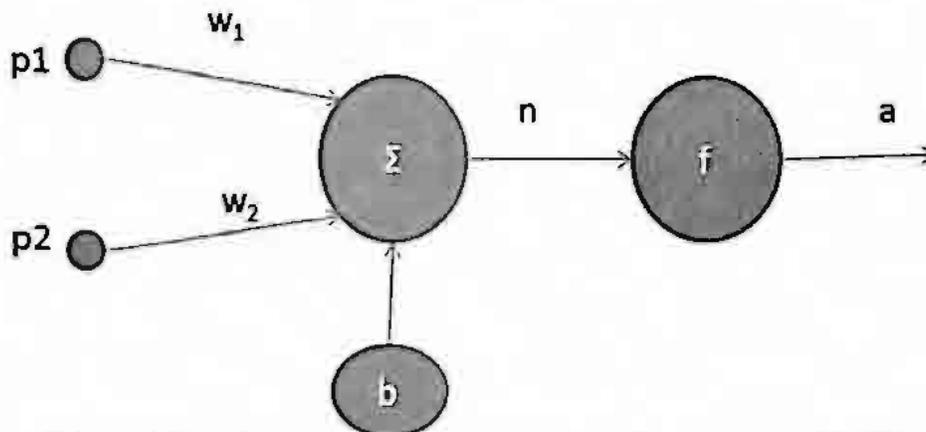


Figura 2. Topología general de una neurona en una red neuronal artificial.

En esta figura se puede observar que una neurona recibe dos entradas p1 y p2. Las entradas pueden ser los valores de entrada a la red, o las salidas de las capas anteriores. Cada entrada se multiplica por un peso w_{ij} , donde i es el número de la neurona que recibe la entrada y j es el número de la neurona que envía la señal. Estas multiplicaciones se suman junto con otro peso b , también llamado sesgo. El resultado de esta suma se llama n , el cual es modificado por una función f , la cual se conoce como función de transferencia. Finalmente, la evaluación de la función de transferencia es la salida "a" de la neurona.

Se ha demostrado que una RNA con dos capas de neuronas puede aproximar cualquier función en cualquier grado, siempre y cuando se tengan suficientes neuronas en la capa oculta. Para este fin, la capa oculta debe tener una función de transferencia monótona creciente y acotada, y la capa de salida debe utilizar una función de transferencia lineal[11]. Sin embargo, es necesario estimar el número de neuronas en la capa oculta. Los datos obtenidos de las muestras deben semejarse a

una función de cualquier grado por lo que es posible utilizar redes neuronales artificiales para estimar una población mayor de datos.

2.2 Entrenamiento de la Red Neuronal Artificial.

La predicción de la red neuronal artificial está basada en el ajuste de pesos, para así aproximar la función deseada. Una ventaja de las RNAs es su facilidad de "aprender" los valores para los pesos siguiendo las reglas de aprendizaje. Una de las reglas es el procedimiento de calibración de los parámetros de la red hasta que se tiene una aproximación deseada.

Para las redes multicapa el procedimiento de entrenamiento es el algoritmo de retro-propagación [12]. Para el entrenamiento de la red es necesario un conjunto inicial, se necesita un conjunto de entrada $\{p_1, p_2, \dots, p_n\}$ para los valores de entrada (donde p_i puede ser un vector) y el conjunto correspondiente a $\{t_1, t_2, \dots, t_n\}$ considerado como los valores de salida. El entrenamiento consiste en tener el menor error entre las salidas de la red y las salidas esperadas t_i .

3 Estrategias Evolutivas

Las estrategias evolutivas (EE) son algoritmos de optimización que buscan el valor mínimo o máximo de una función, llamada función objetivo [13]. Las EE son muy útiles para problemas no lineales y forman parte de los diversos paradigmas de la computación evolutiva al igual que los algoritmos genéticos [14], se inspiran en la teoría de la evolución para buscar soluciones a problemas complejos. En las EE varios conceptos presentes en la teoría de la evolución como son: mutaciones, cruza entre individuos, selección natural, supervivencia del más apto, etc. son adaptados para su uso en un programa de computadora.

Las estrategias evolutivas trabajan con poblaciones de posibles soluciones a un problema, también llamadas soluciones candidato. Estas soluciones candidato son evaluadas de acuerdo a una función de aptitud, que generalmente es la función objetivo del problema de optimización a resolver. Una vez evaluadas, se seleccionan de forma determinista a las mejores soluciones candidato según su aptitud. Esta selección de soluciones se usará para generar una nueva población de individuos. El principio básico en que se basan las estrategias evolutivas es que los mejores padres de una población generarán hijos que tendrán mejor aptitud que sus padres.

Un punto importante en computación evolutiva, es como representar a las soluciones candidato para un problema. En estrategias evolutivas, todas las soluciones se representan como un vector de números reales. Este vector contiene los parámetros que definen el estado de un sistema. Por ejemplo, la distancia horizontal a la que viaja una bala de cañón, está en función del ángulo de disparo α , y la resistencia del viento β , de manera que un elemento de la población puede ser representado con un vector $x = [x_1, x_2]$, donde x_1 es un valor dado para α y x_2 es un valor dado para β . Dados α y β se puede evaluar la distancia horizontal máxima a la que viaja la bala de cañón usando una fórmula de tiro parabólico. Esta distancia será la aptitud de la solución candidato x .

Otro punto importante es como recombinar y variar a los elementos de una población para generar nuevas poblaciones. En EE, el elemento más importante para generar una nueva población es la mutación. En la mutación, los elementos de la población son alterados de forma aleatoria, sumando a cada componente x_i de una solución candidato, una cantidad que se elige siguiendo una distribución normal. La cruce entre elementos de la población es considerada como un elemento secundario en la estrategia evolutiva, y se realiza eligiendo un punto intermedio entre dos padres (cruza normal), o más de dos padres (cruza panmítica). Los padres son tomados de forma aleatoria de entre los mejores individuos de la población.

Una vez generada una nueva población, esta se evalúa y se procede a seleccionar a los padres de la siguiente población. El proceso de selección en las EE es determinista, si se van a elegir a μ futuros padres de una población de λ hijos ($\lambda > \mu$), simplemente se ordenan los μ hijos de mejor a peor de acuerdo a la función de aptitud y se eligen los primeros λ . Existen varias versiones de las estrategias evolutivas dependiendo de la cantidad de padres que generan la siguiente población y si un padre puede pasar a la siguiente generación. En la versión EE(1+1), un padre genera un único hijo y se elige al mejor de los dos como padre de la siguiente generación. En la versión EE(1, λ), un padre genera λ hijos y se elige al mejor de los hijos como padre de la siguiente generación. En la versión EE($\mu + \lambda$), μ padres generan λ hijos y se elige a los mejores μ elementos de la combinación de padres e hijos. También está la versión EE($\mu; \lambda$), que es parecida a la versión EE($\mu + \lambda$), con la diferencia de que los μ padres de la siguiente generación se eligen exclusivamente de los λ hijos.

El algoritmo general de una estrategia evolutiva es como sigue:

- 1: Generar de forma aleatoria una población inicial P_λ de λ elementos.
- 2: Para $i = 1$ hasta G (G es el máximo número de iteraciones o generaciones)
 - 3: Evaluar la aptitud de P_λ .
 - 4: Seleccionar de P_λ los mejores μ elementos (P_μ).
 - 5: Mutar y recombinar los elementos de P_μ , hasta crear una nueva población P_λ de λ elementos.
- 6: Reportar a la mejor solución encontrada.

4 Caso de Estudio

Este proyecto se desarrolló en una empresa que produce partes para calentadores de gas. El proyecto se implementó en el área de estampado, la cual cuenta con una línea automatizada de alimentación a prensa para la producción de subensambles de producto final.

Inicialmente se realizó el estudio del área para identificar las variables del proceso y los medios de optimización, al final del estudio se consideraron cuatro variables críticas en el proceso: la velocidad de alimentación a la prensa, la presión de lubricación, el nivel de lubricación y el avance de la lámina. Se tomó como respuesta

de salida, el exceso de material en la pieza, es decir, la función objetivo fue minimizar el exceso de material en la pieza. La variable de salida es importante, porque el exceso de material en las piezas generadas es una fuente de problemas una vez que se ensambla el producto final por lo que es importante mantener este exceso en los límites de tolerancia.

Una vez identificadas las variables principales y la respuesta de salida, se procedió a modelar la relación de las variables de entrada y salida a través de una red neuronal, pues no es posible crear una función analítica que modele este sistema. Una vez construida y entrenada la red neuronal, procede a encontrar los parámetros óptimos a través de una estrategia evolutiva. Una de las ventajas de las estrategias evolutivas, es que no requieren el cálculo del gradiente de la función objetivo a optimizar y trabajan muy bien con variables continuas, que en este caso es una ventaja, pues no es posible calcular la derivada de la función y el problema contiene variables continuas.

Los detalles de la modelación se dan en la sección siguiente.

4.1 Entrenamiento y Validación de la Red Neuronal Artificial

En esta sección se presenta el procedimiento de entrenamiento, validación y resultados del entrenamiento de la red neuronal artificial. Para obtener muestras para entrenar la RNA, se realizaron 35 experimentos, en los cuales se variaron los valores de las variables críticas y se midió la respuesta como el exceso de material promedio de todas las piezas producidas. No se pudieron realizar más experimentos debido a limitaciones en la disponibilidad del equipo, ya que este se encuentra en producción y al costo de los experimentos, pues en cada experimento se gastaba materia prima.

El conjunto de entradas y salidas fue normalizado en un rango de -1 a 1 para facilitar el procedimiento de entrenamiento. El conjunto de entrenamiento fue dividido en tres grupos el 60% de los datos tomó como conjunto de entrenamiento, el 20% se tomó como conjunto de validación durante el entrenamiento y el 20% se tomó como conjunto de prueba de la aproximación final, el entrenamiento usa el algoritmo de Levenberg-Marquardt [15]. La razón para dividir las muestras en los tres grupos antes mencionados, es para evitar que la red neuronal se sobreentrene, es decir, que sea muy efectiva al simular los valores de los datos con los que fue entrenada, pero sea incapaz de reproducir apropiadamente nuevos datos. Para esto se usó el método de detención temprana. Este método consiste en entrenar la RNA exclusivamente con el conjunto de entrenamiento y en cada iteración, medir el error de predicción respecto al conjunto de validación. Conforme la red es entrenada, su error respecto tanto al conjunto de entrenamiento como de validación se reduce, pero llega un momento en que el error respecto al conjunto de validación empieza a incrementarse mientras que el error respecto al conjunto de entrenamiento sigue bajando. Es en este momento cuando se considera que la red se está sobreentrenando y se detiene el entrenamiento. El conjunto de prueba se usa como una verificación final de la capacidad de predicción de la RNA.

Con respecto a la topología usada la red neuronal artificial tienen cuatro entradas siendo las variables críticas independientes y una salida correspondiente al exceso de material en las piezas, así se tiene una capa de salida con una neurona. Se utilizó

como función de transferencia una tangente-sigmoide en la capa oculta y una lineal en la capa de salida.

Para determinar el número de neuronas a usarse en la capa oculta, se corrieron varios experimentos en el entrenamiento de la red usando 3, 4, 5, 6, 7 y 8 neuronas en la capa oculta, para posteriormente calcular el error entre la salida de la red y las salidas de la muestra y elegir el que daba el menor error. El número de neuronas con el mínimo error es cinco, con un error de 0.0032.

Como se mencionó anteriormente, la muestra inicial se divide en tres partes: el conjunto de entrenamiento, con el cual se entrena la RNA; el conjunto de validación, que se usa durante el entrenamiento para evitar que la red se sobreentrene; y el conjunto de prueba, que se utiliza para verificar la generalidad de la red. Una vez que la red es entrenada, se usó un análisis de regresión entre la salida de la red y la salida esperada de la muestra, obteniendo como resultados de la validación que el conjunto de entrenamiento tiene un ajuste de 91%, el de validación de 80 %, el de prueba 81.5% y el de la combinación de los tres conjuntos es de 81.6%. Considerando que un 80% de ajuste es el mínimo necesario para considerar un ajuste es confiable, podemos considerar que la red neuronal aproxima aceptablemente al sistema.

4.2 Optimización a Través de Estrategias Evolutivas

Una vez creado el modelo, se procedió buscar el óptimo de este usando una estrategia evolutiva. Se programó una $EE(\mu; \lambda)$, con $\lambda = 100$ y $\mu = 400$. El algoritmo se corrió por 1000 generaciones. La función objetivo a optimizar fue la RNA ajustada. Los límites de búsqueda fueron: velocidad de alimentación [10 - 30] pulgadas por minuto; presión de lubricación [15 - 35] psi; nivel de lubricante [1 - 4]; el avance [1.5 - 3.5] pulgadas.

Después de correr la estrategia evolutiva en 10 ocasiones, se tomó el mejor valor generado que consistió en la siguiente combinación de parámetros: velocidad de alimentación = 24 pulgadas por minuto; presión de lubricación = 29 psi; nivel de lubricante = 1; avance = 1.5 pulgadas.

5 Conclusiones

Después de aplicar técnicas de inteligencia artificial, se pudo identificar un conjunto de parámetros optimizados, que minimizan el exceso de material en las piezas producidas. Esto ha permitido reducir el re-trabajo de piezas defectuosas, aumentando la productividad. Los parámetros obtenidos ya se usan en planta, y han facilitado en gran medida la operación de la alimentación de la prensa. Pues se evita la necesidad de ajustar por ensayo y error los parámetros de la máquina.

Las redes neuronales resultaron una herramienta muy versátil para su uso en este trabajo. Igualmente, las estrategias evolutivas fueron implementadas usando código libre de internet, por lo que su uso también fue extremadamente sencillo y rápido.

Podemos concluir que las herramientas de inteligencia artificial pueden implementarse de forma rápida y efectiva para resolver problemas industriales en un tiempo corto y con poca inversión, lo que representa una gran ventaja competitiva.

Como trabajo futuro, se planea explorar otras arquitecturas de redes neuronales, como por ejemplo redes neuronales con funciones de base radial[16]. Esto con el fin de encontrar un mejor ajuste de los datos.

6 Agradecimientos:

Agradecemos de forma especial al CONACYT por su apoyo para la realización de este proyecto a través del Programa de Estímulos a la Innovación 2010, en la modalidad INNOVAPYME.

7 Referencias

1. Serope Kalpakjian y Steven R. Schmid. *Manufactura, ingeniería y tecnología*. Pearson education. Cuarta Edición (2002). 393-402.
2. Fandón, S, Hoyos, C, Fernández, D. Diseño de troqueles de estampación mediante modelado sólido. Recuperado el 2 de febrero de 2009
3. A.Y.C. Neea and V.C. Venkatesha. A Heuristic Algorithm for Optimum Layout of Metal Stamping Blanks (2008)
4. A. Yanagida and A. Azushima. Evaluation of coefficients of friction in hot stamping by hot flat drawing test. (2009).
5. Hurtado Jorge y Álvarez Diego. Optimización basada en confiabilidad por medio de redes neuronales y algoritmos evolutivos, *Revista internacional de métodos numéricos para cálculo y Diseño en ingeniería* (2002).
6. Marek Balizinski, Ernest Czogala, Krzysztof Jemielniak, Jacek Leski. Tool condition monitoring using artificial intelligence methods. *Engineering Applications of Artificial Intelligence* (2002).
7. McCulloch, W. and Pitts, W. A logical calculus of the ideas immanent in nervous activity. In *Bulletin of Mathematical Biophysics*. 5: 115-133. (1943).
8. Rosenblatt, F. The perceptron: A probabilistic model for information storage and organization in the brain. In *Psychological Review*. 65: 386-408. (1958).
9. Widrow, B. and Hoff, M.E. Adaptive switching circuits. In *1960 IRE WESCON Convention Record*. IRE Part 4: 96-104. (1960).
10. Minsky, M. and Papert, S. *Perceptrons*. MIT Press. Cambridge, MA. (1969).
11. Hornik, K.M., Stinchcombe, M. and White, H. Multilayer feedforward networks are universal approximators. *Neural Networks*, 2: 359-366. (1989).
12. Werbos, P.J. Beyond regression: New tools for prediction and analysis in the behavioral sciences. *PHD Thesis*, Harvard University, Cambridge, MA. (1974).

13. Hans-Paul Schewefel, *Evolution and Optimum Seeking: The Sixth Generation*, 1993, John Wiley & Sons, Inc., New York, Usa.
14. David E. Goldberg, *Genetic Algorithms in Search, Optimization, and Machine learning*, 1989, Addison Wesley, Boston MA.
15. Hagan, M.T., Demuth, H.B. and Beale, M. *Neural Network Design*. PWS Publishing Co., Boston, Ma. (1997).
16. D.S. Broomhead and D. Lowe, *Multivariable functional interpolation and adaptive networks*. *Complex Syst.* (1988).

Aplicación de Redes Neuronales Artificiales para la predicción de datos de Saturación de Agua de Yacimientos Petroleros

D. A. Hernández-Ramírez¹, M. Romero-Salcedo², A. K. Hernández V.³, G. Galeano Arroyo⁴, H. López⁵
{Programa de Investigación y Posgrado¹, Programa de Investigación en Matemáticas Aplicadas y Computación²} Instituto Mexicano del Petróleo, Tecnológico de Estudios Superiores de Ecatepec^{3,4}, Universidad Autónoma del Estado de México⁵, México, D.F., México
{dbernand,mromeros}@imp.mx,ingalbahdz@yahoo.com.mx, ggaleanoarroyo@yahoo.com.mx, hlopeznicolas@hotmail.com

Abstract— Se propuso el desarrollo de modelos no lineales basados en Redes Neuronales Artificiales estimar datos de la Saturación de Agua en yacimientos petroleros, considerada como una propiedad fundamental, pero de gran dificultad a determinar a partir de datos históricos. Se retomaron datos de mediciones obtenidas a través de registros geofísicos de pozos. Se diseñaron e implementaron diversas topologías de Redes Neuronales y se analizó el desempeño (análisis de la confiabilidad de la predicción) de cada caso de estudio en particular.

Palabras Clave: *Redes Neuronales Artificiales, Predicción, Saturación de Agua, Simulación y Entrenamiento, Registros Geofísicos de Pozos.*

I. INTRODUCCIÓN

En la Industria del Petróleo existe un gran número de actividades que se realizan tanto en laboratorio como en campo, muchas de ellas se enfocan a optimizar la recuperación de hidrocarburos. La Ingeniería Petrolera se encarga de estudiar los yacimientos, los cuales son formaciones geológicas que, a través de millones de años, han almacenado distintos recursos [1]. Las principales reservas que se encuentran son: aceite (petróleo en sus diversos productos primarios), gas y agua. Los estudios que se realizan en la caracterización de yacimientos tienen el objetivo de conocer las propiedades del complejo. El obtener parámetros como saturación de aceite, saturación de agua, temperatura, porosidad, permeabilidad y los niveles de concentración de fluidos es de gran importancia para análisis posteriores. Las técnicas tradicionales emplean métodos y modelos matemáticos para poder determinar algunas propiedades, por ejemplo podemos mencionar a la regresión por mínimos cuadrados, que ha sido muy utilizada para predecir variables en casos donde la información es abundante [2,3], pero estas técnicas dejan de ser viables cuando se manejan datos con incertidumbre.

La Inteligencia Artificial (IA) está conformada por diversas tecnologías que han permitido desarrollar soluciones más confiables y eficaces en la Industria Petrolera. Las áreas de neurocomputación, cómputo suave y computación evolutiva conforman tecnologías que ayudan a realizar estudios de caracterización en complejos en los que es prácticamente imposible realizar investigaciones por métodos convencionales. Estas tecnologías intentan imitar y emular el funcionamiento

del cerebro humano, brindando capacidades de descubrimiento, generalización, abstracción y asociación. La tecnología que ha sido más utilizada y conocida por sus ventajas, en la Ingeniería Petrolera, por el tipo y el estado de la información que se maneja, son las Redes Neuronales Artificiales (RNA's).

Las RNA's son un sistema computacional conexionista que está constituido de un conjunto de elementos simples, altamente interconectados, los cuales procesan información por medio de un estado dinámico de respuesta a estímulos externos. Son un paradigma de aprendizaje y procesamiento automático inspirado en la forma en que funcionan los sistemas neuronales biológicos. Una red neuronal puede manipular información que presenta limitantes para ser procesada mediante modelos. El análisis de la información por medio de esta tecnología, y que estructurada de manera correcta representa información vital, arroja resultados importantes que pueden ser significativos para la caracterización de yacimientos.

Debido a la importancia de la determinación de la propiedad Saturación de Agua (Sw), como parte de los elementos para la adecuada recuperación de hidrocarburos, el grupo de investigación se propuso desarrollar modelos no lineales basados en RNA's con el fin de proveer nuevas herramientas que permitan brindar alternativas complementarias viables para apoyar la estimación de datos de Sw en un yacimiento. Para ello, se retomó el estudio [4] que emplea datos de registros geofísicos de pozos para estimar valores de Sw, que son obtenidos y analizados de manera tradicional. En este trabajo presentamos una comparativa de los resultados estimados en [4] con los predichos por nuestras topologías de RNA's.

II. ANTECEDENTES

Las RNA's han ganado interés en la Ingeniería Petrolera, ya que han mostrado la habilidad para resolver problemas con relaciones no lineales entre variables. En efecto, las RNA's tienen la capacidad de resolver este tipo de relaciones y manejar datos altamente correlacionados, haciéndolas muy apropiadas para su aplicación en problemas complejos y dinámicos. Algunos escenarios pueden ser difíciles de resolver con métodos analíticos o numéricos, debido a la cantidad de datos que se tienen, que muchas veces son insuficientes o con una gran incertidumbre, a las simplificaciones y supuestos

que se realizan a los modelos matemáticos. En general, el tipo de problemas que deben abordarse con RNA's son aquellos que demandan un alto consumo de tiempo de procesamiento, que no necesariamente relacionan linealmente las variables, o que simplemente son imposibles de resolver a través de métodos tradicionales. De esta manera, las RNA's deben usarse en aquellos casos en los que la modelación matemática no es una opción práctica. Una red neuronal es también apropiada para simular la funcionalidad y comportamiento de un sistema y para extraer patrones, estructuras y relaciones entre los datos (ej. análisis e interrelación compleja de datos). En la década de los 90's creció el uso de las RNA's para estudios de tratamiento de información de yacimientos [5-7].

Las RNA's se han usado para modelar, estimar, predecir o determinar propiedades físicas tales como: porosidad y permeabilidad [8-10]; saturación de aceite [8,11], correlación y parámetros de registros de pozos [8, 11-13]. Si bien es claro que el uso de las RNA's puede ser omitido cuando una solución convencional efectiva existe, hay tareas para las cuales esta tecnología una solución única. En particular, se trata de aquellas tareas en donde hay datos con incertidumbre o tienen ruido, donde no está disponible el conocimiento explícito del problema a resolver, o cuando existe no linealidad entre los datos de entrada y salida. Hemos observado que para cada problema a resolver se requiere de un conocimiento adecuado sobre las teorías y prácticas fundamentales de las RNA's con el fin de lograr resultados aceptables. Para un entrenamiento exitoso, la red neuronal debe ser expuesta a datos representativos y suficientes con el fin de que se pueda adquirir conocimiento para predecir con exactitud nuevas situaciones.

Para el adecuado tratamiento de los datos usando RNA's, una metodología [10] propone la adquisición y preparación de los datos, un posterior desarrollo de la red en base a la mayor cantidad de variables disponibles y la descripción de los datos de entrada y salida una vez que concluye el aprendizaje. En otras metodologías estudiadas [6, 8, 14-19], se generalizan pasos, a saber: la construcción de la topología de las RNA's, selección del método de entrenamiento, la clasificación de las variables de entrada, inicialización de los pesos, selección de la función de activación, normalización de los datos y por último el inicio del proceso de aprendizaje, como procesos fundamentales en la aplicación de la tecnología de RNA's.

III METODOLOGIA EMPLEADA

En la actualidad existen simuladores de RNA's que ayudan a la construcción de topologías para su aplicación en diversas áreas. Muchos de ellos son comerciales y se encuentran bajo la licencia de corporativos e instituciones. Algunos investigadores dentro del campo de la industria del petróleo utilizan el software GRNN [6, 14, 15]. Otros han creado herramientas especializadas y que han sido utilizadas con un enfoque especial en el

ámbito petrolero como GERENE [16]. STATISTICA-Neural Networks software, de Statsoft, es un producto de los más completos y proporciona facilidad de construcción. STATISTICA ha sido utilizado para el tratamiento de datos de hidrocarburos [17-19]. De la misma manera también se puede señalar en uso del simulador NS-ANN [10] y MATLAB [8] para estimación de parámetros y predicción de datos. En esta investigación se decidió emplear MATLAB como el software para llevar a cabo el diseño de las RNA's, puesto que presenta una mayor facilidad al tratamiento de los datos, el tiempo de procesamiento de la información y ejecución de las RNA's y una interface de trabajo más cómoda para resolver este tipo de problemas.

Se emplearon varias topologías de RNA's para determinar cuál era la óptima en la predicción de datos. Se analizó el comportamiento de cada topología, junto con el comportamiento global y de esta manera se retuvo el modelo que contara con el mejor desempeño y se comparó con los resultados reportados en la literatura [4] donde el autor emplea datos de tres pozos referenciados como A, B y C, para determinar las propiedades petrofísicas de cada uno, con datos obtenidos mediante registros geofísicos de pozos.

Los registros geofísicos de pozos, consisten en una serie mediciones, obtenidas por una sonda con varios sensores que se introduce en una perforación de barreno para determinar las curvas de cada parámetro que se desea conocer. Con esta técnica se obtiene a diferentes profundidades los parámetros físicos de la formación. Su propósito es determinar las características físicas de las rocas, de los fluidos que la saturan y de las propiedades de la construcción del pozo. Con estos datos se determina la litología, su resistividad real, la densidad volumétrica, su geometría, porosidad y permeabilidad para poder definir los intervalos donde se encuentran las capas productoras. A partir de los registros geofísicos los autores extrajeron 15 propiedades del medio, con una serie de 36 series de datos para el pozo A, 33 series para el pozo B y 31 para el pozo C. Con dichos datos, emplearon la ecuación de Archie para determinar el valor de S_w .

Por nuestro lado, nos dimos a la tarea de extraer solamente las propiedades que se consideran representativas al cálculo de S_w , a partir del conjunto de las 15 propiedades obtenidas por los registros geofísicos, para que estas nos permitieran determinar el valor de S_w para cada profundidad medida. Dichas propiedades fueron la Resistividad (RT) y la porosidad (NPHI). Un ejemplo del conjunto de datos normalizados, así como las variables empleadas, se muestran en la tabla I. Se seleccionaron los pozos B y C como los patrones de entrenamiento y el pozo A fue el pozo que se usó para validación y predicción de datos. Cada modelo no lineal de red neuronal fue implementado a través de un conjunto de dos nodos en la capa de entrada, correspondientes a los valores que influyeron en el cálculo de la S_w (dos propiedades); una capa oculta en dos niveles, cada nivel con un número de nodos variables

y con un solo nodo en la capa de salida, representando la predicción del valor del dato de Sw

IV. APLICACION DE LA METODOLOGIA

De manera inicial se depuraron los datos para obtener la mayor cantidad de información posible sobre el problema a estudiar. La depuración simplifica la información a utilizar para que ésta no tenga errores o los conjuntos de datos no estén incompletos. Además, se determina el rango de valores de los registros, para llevar a cabo una normalización de 0 a 1. Para la construcción de las topologías, usamos redes de cuatro capas (entrada, dos ocultas, salida) para apoyar en la clasificación de parámetros. En [6, 8, 14-19], se encontró que existe un promedio o rango de nodos ocultos: 28 nodos repartidos en las capas ocultas. Sin embargo, también se conoce que un número grande de nodos en la capa oculta, cuando los conjuntos de datos para entrenar la red son relativamente pequeños, como en nuestro caso que era un conjunto de 36 series de datos, puede llegar a sobreentrenar la red neuronal.

Lo anterior permite proponer las siguientes topologías: 2_(5-5)_1, 2_(10-10)_1, 2_(14-14)_1 y 2_(28-28)_1, que indican que en la primera capa se usan dos variables como valores de entrada, las dos capas ocultas (intermedias) presentan un número paralelo de neuronas. 5 y 10 son propuestas como alternativas de no sobre-entrenamiento, 14, al doble, son las 28 neuronas de referencia, y 28 al doble, expande las posibilidades de análisis. Además, se puntualizan otras propiedades de la RNA como la conexión entre las capas y el proceso de aprendizaje (cálculo del error por retro-propagación hacia atrás, backpropagation). Partiendo de estas características, elaboramos diversos patrones de entrenamiento para la RNA, es decir, el conjunto de datos seleccionados para entrenamiento, validación, y predicción de las redes. Posteriormente, entrenamos las redes con las topologías propuestas, al analizar el entrenamiento de la RNA y enfocarse en el proceso de aprendizaje de la misma, se comprueba si la topología de la red es la adecuada, monitoreando la gráfica del error. En este punto se valida si el tamaño de red, el número de capas y neuronas y se asocian adecuadamente los datos del problema. Cuando se ha realizado el entrenamiento y aprendizaje de la red, los datos son sometidos a evaluación con lo cual se obtienen los valores de predicción. Dichos datos se encuentran normalizados también en un rango de 0 a 1. Se realiza una comparación con los datos normalizados originales, con el fin de dar a conocer la capacidad del modelo de RNA creado que sea utilizado. Finalmente se desnormalizaron los datos para hacer una comparación con los datos originales. Para calcular el error de la predicción de cada red neuronal, empleamos el cálculo del error cuadrático medio (RMSE), también usado por varios autores [11-16] asimismo se calcula el coeficiente de correlación entre el dato considerado como valor real, y el valor predicho por cada una de las topologías de RNA's. El análisis más extenso se puede apreciar en las figuras 1-8, donde se

muestra la desnormalización de los datos, así como los coeficientes de correlación

V. RESULTADOS Y DISCUSIÓN

Todas las pruebas experimentales realizadas con las diferentes topologías de RNA's, observaron una amplia relación entre las dos variables indicadas: Porosidad y Resistividad. Se utilizó el método de aprendizaje de tipo supervisado y una función de activación tipo sigmoidea para la capa oculta y la capa de salida. Por último, analizamos los resultados obtenidos de las predicciones corroborando con los datos reales, mediante el cálculo del error. El comparativo de los RMSE y los coeficientes de correlación en la predicción se aprecian en la tabla II. El modelo no lineal de red neuronal que mejores resultados arrojó fue el de topología: 2_(14-14)_1. Se obtuvo un error cuadrático medio de 0.0464, con un coeficiente de correlación entre los datos predichos y los reales de 0.7157.

TABLA I. EJEMPLOS DE DATOS NORMALIZADOS CON LOS CUALES SE CREAN LOS PATRONES DE ENTRENAMIENTO.

Pozo A			
	Resistividad (RT)	Porosidad (NPHI)	Saturación (Sw)
1	0	1	0.225595696
2	0.08477187	0.613756614	0.112477581
3	0.050459447	0.812698413	0.154880861
4	0.118340681	0.695238095	0.118242378
5	0.094438838	0.541798942	1
6	0.02560153	0.492063492	1
7	0.211982791	0.679365079	0.90174225
8	0.012110267	0.84021164	0.010632847
9	0.301853721	0.771428571	0.732897771
10	0.336484836	0.876190476	0.089162183
11	0.240080735	0.477248677	0.840251089
12	0.014447336	0.644444444	1
13	0.037286875	0.698412698	1
14	0.128220109	0.532275132	1
15	0.702395496	0.182010582	0
16	1	0.071957672	0.298744555

TABLA II. RESUMEN DE RESULTADOS DE PREDICCIÓN DE LAS TOPOLOGÍAS PROPUESTAS.

Topología	Error Cuadrático Medio	Coficiente de Correlación
2_5-5_1	0.056434421	0.607739985
2_10-10_1	0.060269729	0.313086292
2_14-14_1	0.046456802	0.715727689
2_28-28_1	0.050010639	0.623825635

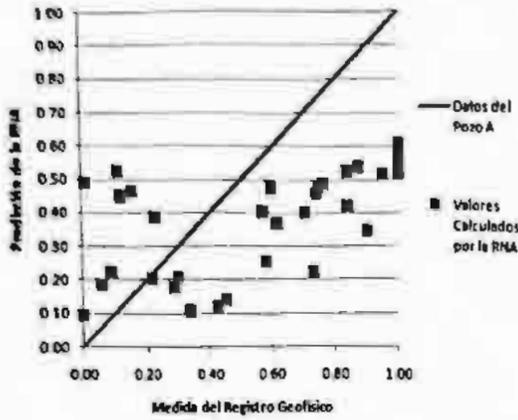


Figura 1 Correlación de la predicción de la Red (Topología) 2_(5-5)_1

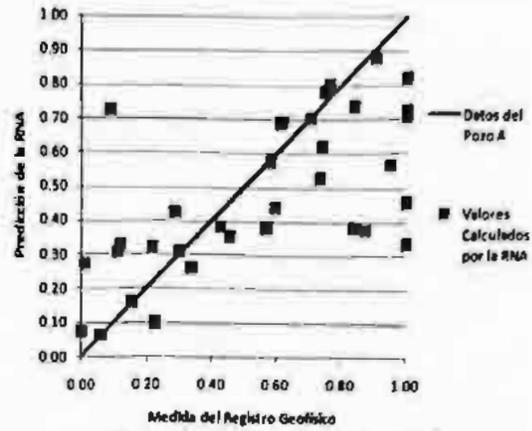


Figura 5 Correlación de la predicción de la Red (Topología) 2_(14-14)_1

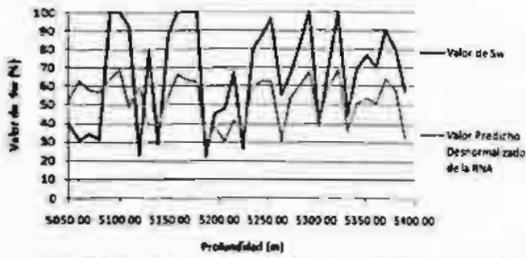


Figura 2 Comparativo de los datos reales contra los predichos por la Red (Topología) 2_(5-5)_1 (valores desnormalizados)

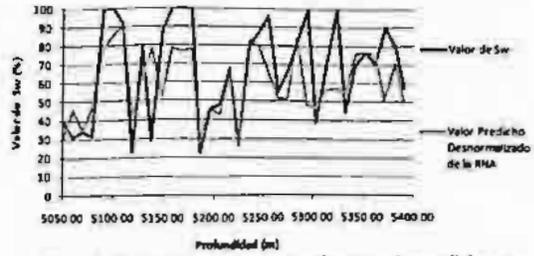


Figura 6 Comparativo de los datos reales contra los predichos por la Red (Topología) 2_(14-14)_1 (valores desnormalizados)

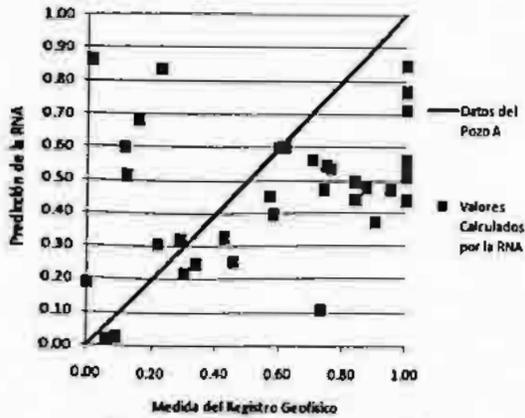


Figura 3 Correlación de la predicción de la Red (Topología) 2_(10-10)_1

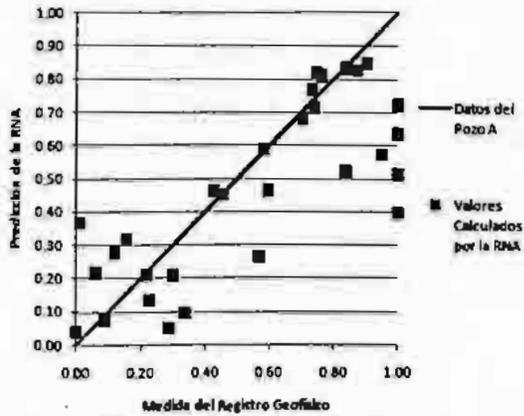


Figura 7 Correlación de la predicción de la Red (Topología) 2_(28-28)_1

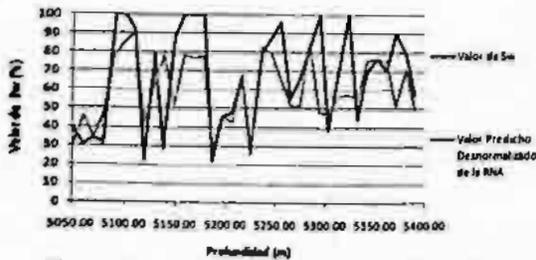


Figura 4 Comparativo de los datos reales contra los predichos por la Red (Topología) 2_(10-10)_1 (valores desnormalizados)

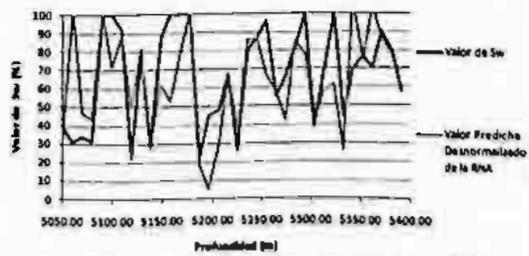


Figura 8 Comparativo de los datos reales contra los predichos por la Red (Topología) 2_(28-28)_1 (valores desnormalizados)

CONCLUSIONES Y TRABAJO FUTURO

Con este trabajo pudimos encontrar un modelo de Red Neuronal óptimo que antes que nada muestra la representatividad de las variables empleadas, para poder encontrar el valor de la variable de estudio. Logramos encontrar un modelo que responde de manera adecuada a nuestras necesidades y que además de todo, cumple con el hecho de poder reproducir de manera certera y completa, los valores que presentan en la referencia de datos, modelo a partir del cual, podemos encontrar valores presentes o futuros de diferentes conjuntos de datos de registros geofísicos, aplicables a más problemas similares. Además de ello, comprobamos que el usar 28 nodos en las capas ocultas (14-14) efectivamente arroja mejores predicciones que otras topologías, comprobando también lo citado en las referencias. Con ello, la incertidumbre del tratamiento previo de datos y la resolución del problema del cálculo de la Sw queda resuelto. Como las predicciones que se encontraron se aplicaron a datos que también al autor le generaron incertidumbre, se entiende que se puede aplicar la propuesta de red, a la resolución de problemas donde los datos con los que se cuenten tengan interferencias, estén incompletos o procedan de registros de baja calidad. Con ello, la exactitud de los valores de los datos es asegurada, dado que una de las ventajas de las RNA's es que no necesitan de la linealidad de las variables.

AGRADECIMIENTOS

Agradecemos a todos los miembros del equipo por todo el trabajo, la ayuda, y las discusiones constructivas en todos los aspectos de este trabajo. En especial agradecemos los revisores anónimos por sus útiles comentarios que ayudaron a mejorar la calidad de este artículo.

REFERENCIAS

- [1] Schlumberger "Las Presiones de las Operaciones de Perforación y Producción", Schlumberger, Oilfield Review, Vol. 17, No. 3, pp. 26-47, Invierno, 2005.
- [2] Wendt, W.A., Sakurai S., Nelson, P.H. "Permeability predictions from well logs using multiple regression". In: Lake, L. W., Carrol, H.B. Jr. (Eds.), Reservoir Characterization. Academic Press, Orlando, pp. 122-181. 1986
- [3] Block, S. "Empirical prediction of porosity and permeability in sandstones". AAPG Bulletin 75 (7), 1145-1160. July, 1991.
- [4] Gorian, J. J. "Cálculo de saturación de agua Sw y exponente de cementación m empleando el registro EPT". Tesis Instituto Politécnico Nacional, Escuela Superior de Ingeniería y Arquitectura, México, 1991.
- [5] Juniardi & Ershaghi, 1993] Juniardi, I. J., Ershaghi, I., "Complexities of Using Neural Networks In Well Test Analysis of Faulted Reservoir", 1993 SPE Western Regional Meeting SPE 26106, pp. 26-28, Marzo, 1993.
- [6] Mohaghegh, S., A. R., Ameri, S.W. Virginia, "A Methodological Approach For Reservoir Heterogeneity Characterization Using Artificial Neural Networks", Society of Petroleum Engineers, Inc. SPE 28394, 1994.
- [7] Wong, P.M., Jian, F.X., Taggart, I.J., "A Critical Comparison of Neural Networks and Discriminant Analysis in Lithofacies, Porosity and Permeability Predictions", Journal of Petroleum Geology 18, pp. 191-206, 1995.
- [8] Bhatt, A., Helle H.B., "Committee Neural Networks for Porosity and Permeability Prediction from Well Logs", Geophysical Prospecting pp. 645-660, Noviembre, 2001.
- [9] Fischetti, A.I., Andrade, A., "Porosity Images from Well Logs", Journal Elsevier, Journal of Petroleum Science and Engineering, pp. 149-158, 2002.
- [10] Salehi F., Azizi R., Salehi A., "Oil Field Development Using Neural Network Model", Huang, D-S., Heutte, L., Looy M. (Eds.) ICIC 2007, CCIS 2, pp. 29-38, 2007.
- [11] Alzate, G., Branch, J.W., Suárez, O.M., Vega, C.A., "Correlación de registros entre pozos con redes neuronales artificiales utilizando múltiples curvas o variables registradas", Boletín de Ciencias de la Tierra ISSN 0120, Número 19, Noviembre, 2006.
- [12] Fischetti, A.I., Andrade, A., "Porosity Images from Well Logs", Journal Elsevier, Journal of Petroleum Science and Engineering, pp. 149-158, 2002.
- [13] Yue, W., Tao, G., "A New Type of Neural Network For Reservoir Identification Using Geophysical Well Logs", Mathematical Geology, Vol. 37, No. 3, Abril, 2005.
- [14] Mohaghegh, S., Arefi, R., Ameri, S., "Petroleum Reservoir Characterization with the Aid of Artificial Neural Networks", Elsevier, Journal of Petroleum Science and Engineering, pp. 261-274, 1996.
- [15] Camargo, P., Maschio, C., Schiozer, D.J., "Use of Neuro-Simulation Techniques as Proxies to Reservoir Simulator Application in Production History Matching", Journal Elsevier, Journal of Petroleum Science and Engineering, pp. 273-280, 2006.
- [16] Pérez M. F., Rozo M. A., Jiménez R. U., Calvete F. E., Camilo Z. C., "Aplicación de las Redes Neuronales al estudio de Yacimientos de Petróleo", Escuela de Ingeniería de Petróleo Universidad Industrial de Santander ECOPELROL, pp. 1-15.
- [17] Alrumah, M., "Neural Networks Predict Well Inflow Performance", pp. 1-75, Diciembre, 2003.
- [18] Petra L., Trapiche A., Naidés C., "Optimización en la estimación de la Permeabilidad utilizando Redes Neuronales.", pp. 1-17.
- [19] Gentry M. D., "Applications of artificial neural networks in the identification of flow units, happy Spraberry Field, Garza County, Texas", A Thesis Submitted to the Office of Graduate Studies of Texas A&M University, Master of Science, pp. 1-174, Diciembre, 2003.

Aplicación de Algoritmos Genéticos a la Compresión de Imágenes con Evolets

Oscar Herrera y Roman Mora

Departamento de Sistemas, Universidad Autónoma Metropolitana Azcapotzalco
Av. San Pablo 180, Col. Reynosa Tamaulipas, 02200, México, D.F.
oha@correo.azc.uam.mx
ing.romannora@gmail.com
http://ia.azc.uam.mx

Resumen En este trabajo se presenta un método de optimización que combina varios algoritmos genéticos a fin de mejorar su convergencia con una función objetivo que maximiza la compresión de imágenes en escala de grises usando filtros digitales paramétricos basados en wavelets (Evolets).

Palabras clave. Wavelets, parametrización, optimización, compresión de imágenes, algoritmo genético.

1. Introducción

Las wavelets son funciones $\psi_{a,b}(x) = a^{-\frac{1}{2}}\psi(ax - b)$ con $a \neq 0$ asociadas a funciones de escalamiento. Las funciones de escalamiento $\varphi(x)$ dan lugar a un Análisis Multiresolución (AMR) [1]. Uno de los resultados fundamentales del análisis de imágenes con wavelets es que se pueden usar un par de filtros digitales para implementar la transformada wavelet discreta (TWD) de acuerdo con la descomposición de un AMR. Se han explorado varios criterios de diseño de wavelets, algunos basados en criterios de suavidad, derivabilidad o de soporte compacto, y otros enfocados a aplicaciones específicas. En este trabajo la idea fundamental es calcular wavelets personalizadas que maximicen la compresión de imágenes en el menor tiempo posible con la ayuda de algoritmos evolutivos, de ahí el nombre de *Evolets*. Uno de los trabajos fundamentales en el diseño de wavelets fue publicado por Daubechies [2] cuyas funciones wavelet tienen soporte compacto y maximizan el número de momentos de desvanecimiento. Varios métodos de diseño de wavelets parten de la ecuación de escalamiento:

$$\varphi(x) = \sum_{k=0}^N h_k \varphi(2x - k) \quad (1)$$

en donde $N \in \mathbb{N}$, $h_k \in \mathbb{R}$, y $h_k = 0$ para $k < 0$ y $k > N$. La secuencia h_k determina la respuesta al impulso de un filtro digital pasabajas y g_k la de un filtro digital pasaaltas [1]. Estos filtros son ortogonales si cumplen la condición

$g_k = (-1)^k h_{1-k}$, para $k = 0, 1, \dots, N - 1$. Los filtros ortogonales son útiles en compresión ya que permiten descomponer imágenes en bandas de frecuencia con mínima redundancia [3]. El problema de diseñar un filtro digital basado en wavelets consiste en determinar la secuencia h_k . En el caso de compresión se busca transformar los valores de los píxeles a otro espacio en donde la entropía sea mínima y por lo tanto se pueda alcanzar la mayor compresión de acuerdo con el Teorema de Shannon [4]. A partir de la ecuación de escalamiento es posible obtener ecuaciones paramétricas que generan familias de wavelets con condiciones preestablecidas, como la ortogonalidad. La parametrización de wavelets no es única, se puede revisar por ejemplo [5][6] y [7]. En este trabajo se ha optado por usar las descritas en [8]. Las ecuaciones de filtros de longitud dos no tienen parámetros, y a mayor longitud de los filtros el número de parámetros se incrementa y, consecuentemente, es más difícil determinar el mejor filtro para una aplicación específica. En procesamiento de imágenes suelen usarse filtros "estándar" como los así llamados *filtros Daubechies*, o *Coiflets* [9], entre otros. Sin embargo, se ha reportado que esos filtros no proveen los mejores resultados para todas las aplicaciones, en particular en cuanto a compresión de imágenes se refiere, como puede constatarse en [10] y [11].

El desafío principal en el diseño de filtros personalizados es el tiempo de ejecución requerido para calcularlos, esto se debe en gran parte a que no es posible aplicar métodos de gradiente por el tipo de función a optimizar. Motivados por el desafío anterior, en este artículo, se propone un método de optimización que reúne varios algoritmos genéticos en un entorno de convivencia e intercambio de información; en analogía a lo que hacemos las personas organizadas en poblaciones que compartimos información a través del proceso de transculturación.

El artículo está organizado en secciones, en la Sección 2 se revisa la parametrización de filtros de longitud 4, 6, 8 y 10. En la Sección 3 se presenta el diseño de filtros digitales basados en wavelets como un problema de optimización. En la Sección 4 se describen algunas propiedades de las parametrizaciones que facilita la integración de varios algoritmos genéticos. En la Sección 5 se reportan resultados experimentales, y finalmente en la Sección 6 se presentan las conclusiones y trabajos futuros.

2. Filtros paramétricos basados en wavelets

La parametrización de filtros basados en wavelets permite reducir el espacio de búsqueda de secuencias que cumplan con las condiciones de ortogonalidad y reconstrucción perfecta. A continuación se revisan las parametrizaciones de filtros de longitud 4, 6, 8 y 10. Se parte del polinomio trigonométrico $H(z) = \sum_k h_k z^k$ que representa la transformada z de la secuencia h_k . Las condiciones dadas por $\sum_{k \in \mathbb{Z}} h_k = 2$ y $\sum_{k \in \mathbb{Z}} h_k h_{k+2l} = 2\delta_{0l}$ ($l \in \mathbb{Z}$)¹ son necesarias para obtener soluciones únicas ortogonales, que en el dominio de la transformada z equivalen a $|H(z)|^2 + |H(-z)|^2 = 1$. La condición $H(z = 1) = 1$ en el dominio

¹ δ_{0l} es la función discreta Kronecker delta.

de la transformada z , es equivalente a $\sum_{k \in \mathbb{Z}} h_k = 2$ en el dominio temporal. Las condiciones $H(z=1) = 1$ y $|H(z)|^2 + |H(-z)|^2 = 1$ implican que $H(z=-1) = 0$. Al realizar manipulaciones algebraicas con las condiciones anteriores es posible obtener las siguientes parametrizaciones para los filtros de longitud 4, 6, 8, y 10 (ver [8]). Para el filtro de longitud cuatro, representado por $H_4(z) = a_0 + b_0z + a_1z^2 + b_1z^3$, las parametrizaciones son:

$$\begin{aligned} H_4[0] &= \frac{1}{4} + \frac{1}{2\sqrt{2}} \cos \alpha_4, & H_4[1] &= \frac{1}{4} + \frac{1}{2\sqrt{2}} \sin \alpha_4 \\ H_4[2] &= \frac{1}{4} - \frac{1}{2\sqrt{2}} \cos \alpha_4, & H_4[3] &= \frac{1}{4} - \frac{1}{2\sqrt{2}} \sin \alpha_4 \end{aligned} \quad (2)$$

donde $\alpha_4 \in [0, 2\pi)$. Las soluciones paramétricas para el filtro de longitud seis $H_6(z) = a_0 + b_0z + a_1z^2 + b_1z^3 + a_2z^4 + b_2z^5$ son:

$$\begin{aligned} H_6[0] &= \frac{1}{8} + \frac{1}{4\sqrt{2}} \cos \alpha_6 + \frac{p}{2} \cos \beta_6, & H_6[1] &= \frac{1}{8} + \frac{1}{4\sqrt{2}} \sin \alpha_6 + \frac{p}{2} \sin \beta_6 \\ H_6[2] &= \frac{1}{4} - \frac{1}{2\sqrt{2}} \cos \alpha_6, & H_6[3] &= \frac{1}{4} - \frac{1}{2\sqrt{2}} \sin \alpha_6 \\ H_6[4] &= \frac{1}{8} + \frac{1}{4\sqrt{2}} \cos \alpha_6 - \frac{p}{2} \cos \beta_6, & H_6[5] &= \frac{1}{8} + \frac{1}{4\sqrt{2}} \sin \alpha_6 - \frac{p}{2} \sin \beta_6 \end{aligned} \quad (3)$$

donde $p = \frac{1}{2} \sqrt{1 + \sin(\alpha + \frac{\pi}{4})}$, $\alpha_6, \beta_6 \in [0, 2\pi)$. Para el filtro de longitud ocho $H_8(z) = a_0 + b_0z + a_1z^2 + b_1z^3 + a_2z^4 + b_2z^5 + a_3z^6 + b_3z^7$ las parametrizaciones son:

$$\begin{aligned} H_8[0] &= \frac{1}{8} + \frac{1}{4\sqrt{2}} \cos \alpha_8 + \frac{1}{2\sqrt{2}} \cos \beta_8 \cos \gamma_8, & H_8[1] &= \frac{1}{8} + \frac{1}{4\sqrt{2}} \sin \alpha_8 + \frac{1}{2\sqrt{2}} \sin \beta_8 \cos \theta_8 \\ H_8[2] &= \frac{1}{8} - \frac{1}{4\sqrt{2}} \cos \alpha_8 + \frac{1}{2\sqrt{2}} \cos \beta_8 \sin \gamma_8, & H_8[3] &= \frac{1}{8} - \frac{1}{4\sqrt{2}} \sin \alpha_8 + \frac{1}{2\sqrt{2}} \sin \beta_8 \sin \theta_8 \\ H_8[4] &= \frac{1}{8} + \frac{1}{4\sqrt{2}} \cos \alpha_8 - \frac{1}{2\sqrt{2}} \cos \beta_8 \cos \gamma_8, & H_8[5] &= \frac{1}{8} + \frac{1}{4\sqrt{2}} \sin \alpha_8 - \frac{1}{2\sqrt{2}} \sin \beta_8 \cos \theta_8 \\ H_8[6] &= \frac{1}{8} - \frac{1}{4\sqrt{2}} \cos \alpha_8 - \frac{1}{2\sqrt{2}} \cos \beta_8 \sin \gamma_8, & H_8[7] &= \frac{1}{8} - \frac{1}{4\sqrt{2}} \sin \alpha_8 - \frac{1}{2\sqrt{2}} \sin \beta_8 \sin \theta_8 \end{aligned} \quad (4)$$

sujeto a:

$$\begin{aligned} &\sqrt{2} \cos \theta_8 \sin \beta_8 - 2 \cos \theta_8 \sin \alpha_8 \sin \beta_8 + \sqrt{2} \cos \beta_8 (\cos \gamma_8 - \sin \gamma_8) \\ &- 4 \cos^2 \beta_8 \cos \gamma_8 \sin \gamma_8 - 2 \cos \alpha_8 \cos \beta_8 (\cos \gamma_8 + \sin \gamma_8) \\ &- \sqrt{2} \sin \beta_8 \sin \theta_8 - 2 \sin \alpha_8 \sin \beta_8 \sin \theta_8 - 4 \cos \theta_8 \sin^2 \beta_8 \sin \theta_8 = 0 \end{aligned} \quad (5)$$

con $\alpha_8, \beta_8, \gamma_8$ y $\theta_8 \in [0, 2\pi)$. Y las soluciones paramétricas para los filtros de longitud diez son:

$$\begin{aligned} H_{10}[0] &= \frac{1}{16} + \frac{1}{8\sqrt{2}} \cos \alpha_{10} + \frac{1}{4\sqrt{2}} \cos \beta_{10} \cos \gamma_{10} + \frac{r}{2} \cos \delta_{10} \\ H_{10}[1] &= \frac{1}{16} + \frac{1}{8\sqrt{2}} \sin \alpha_{10} + \frac{1}{4\sqrt{2}} \sin \beta_{10} \cos \theta_{10} + \frac{r}{2} \sin \delta_{10} \\ H_{10}[2] &= \frac{1}{8} - \frac{1}{4\sqrt{2}} \cos \alpha_{10} + \frac{1}{2\sqrt{2}} \cos \beta_{10} \sin \gamma_{10} \\ H_{10}[3] &= \frac{1}{8} - \frac{1}{4\sqrt{2}} \sin \alpha_{10} + \frac{1}{2\sqrt{2}} \sin \beta_{10} \sin \theta_{10} \\ H_{10}[4] &= \frac{1}{8} + \frac{1}{4\sqrt{2}} \cos \alpha_{10} - \frac{1}{2\sqrt{2}} \cos \beta_{10} \cos \gamma_{10} \\ H_{10}[5] &= \frac{1}{8} + \frac{1}{4\sqrt{2}} \sin \alpha_{10} - \frac{1}{2\sqrt{2}} \sin \beta_{10} \cos \theta_{10} \\ H_{10}[6] &= \frac{1}{8} - \frac{1}{4\sqrt{2}} \cos \alpha_{10} - \frac{1}{2\sqrt{2}} \cos \beta_{10} \sin \gamma_{10} \\ H_{10}[7] &= \frac{1}{8} - \frac{1}{4\sqrt{2}} \sin \alpha_{10} - \frac{1}{2\sqrt{2}} \sin \beta_{10} \sin \theta_{10} \\ H_{10}[8] &= \frac{1}{16} + \frac{1}{8\sqrt{2}} \cos \alpha_{10} + \frac{1}{4\sqrt{2}} \cos \beta_{10} \cos \gamma_{10} - \frac{r}{2} \cos \delta_{10} \\ H_{10}[9] &= \frac{1}{16} + \frac{1}{8\sqrt{2}} \sin \alpha_{10} + \frac{1}{4\sqrt{2}} \sin \beta_{10} \cos \theta_{10} - \frac{r}{2} \sin \delta_{10} \end{aligned} \quad (6)$$

donde $r = \sqrt{\frac{1}{2} - H10[2]^2 - H10[4]^2 - H10[6]^2 - H10[3]^2 - H10[5]^2 - H10[7]^2}$, sujeto a la restricción:

$$\begin{aligned} & \cos\beta_{10}(\cos\gamma_{10}(\sqrt{2} - 2\cos\alpha_{10}) - 8\sqrt{2}r\cos\delta_{10}\sin\gamma_{10}) + \\ & \sin\beta_{10}(\cos\theta_{10}(\sqrt{2} - 2\sin\alpha_{10}) - 8\sqrt{2}r\sin\delta_{10}\sin\theta_{10}) = 0 \end{aligned} \quad (7)$$

con $\alpha_{10}, \beta_{10}, \gamma_{10}, \theta_{10}$, y $\delta_{10} \in [0, 2\pi)$.

3. Compresión de imágenes

En imágenes en escala de grises² con amplitud w y altura h , los valores de cada pixel están en el intervalo $[0, 255]$ y al aplicarle la TWD con h_k y $g_k \in \mathbb{R}$ se obtienen coeficientes de transformación de punto flotante positivos y negativos. El filtro empleado determina el número de coeficientes wavelet que concentran la energía. Al minimizar este número³ la entropía también se minimiza y provee una mayor compresión. De acuerdo con el AMR es posible calcular la TWD en dos dimensiones aplicando primero el paso de transformación en sentido horizontal, seguido de una transformación en sentido vertical [1]. Los filtros basados en wavelets pueden diseñarse para conservar la energía al aplicar la TWD. La energía de una imagen (en el dominio temporal) con $N = wh$ pixeles f_i para $i = 0, 1, \dots, N$ se define como $E = \sum_{i=0}^{N-1} f_i^2$. Si los N coeficientes de transformación wavelet w_i conservan la energía entonces $E = \sum_{i=0}^{N-1} w_i^2$. La energía acumulada con q coeficientes se define como: $E_c = \sum_{i=0}^q w_i^2$ $0 \leq q \leq N$. Se pueden ordenar los coeficientes w_i en forma descendente de acuerdo a su magnitud e ir incrementando q hasta que la razón de la energía acumulada dividida entre la energía original E alcance un cierto valor (en este trabajo la razón $\frac{E_c}{E} = 0,999999$). Los $wh - q$ coeficientes restantes se igualan a cero (coeficientes irrelevantes) y con ello se obtiene un esquema de compresión con reconstrucción casi perfecta⁴. Lo que sigue es calcular el mejor filtro que maximiza el número de coeficientes irrelevantes (se minimiza q y la entropía de los coeficientes de transformación). En este trabajo se describe un método que pretende acelerar la convergencia hacia el filtro evolutivo óptimo para cada imagen procesada. En la Sección 4 se describe el método propuesto y las propiedades de las parametrizaciones de los filtros que marcaron la pauta para su diseño.

4. Algoritmos Genéticos y Compresión de Imágenes

En [12] se usa un algoritmo genético (AG) para calcular el mejor filtro evolutivo considerando como función de aptitud el error cuadrático medio obtenido

² Se usa el formato PGM, siglas en inglés de Mapa de Grises Portable.

³ En forma equivalente se maximiza el número de coeficientes irrelevantes con valor absoluto cercano a cero.

⁴ Los filtros son diseñados para proveer reconstrucción perfecta, sin embargo puede haber errores de precisión con las operaciones de punto flotante.

al reconstruir una imagen usando la Codificación de Árbol de Ceros [13]. En la Sección 3 se describió un método alternativo que no requiere reconstruir la imagen transformada para calcular el mejor filtro evolutivo sino que se maximiza el número de coeficientes irrelevantes. En [10] un AG optimiza un único parámetro para los filtros de longitud 4, y en [11] se optimizan filtros de longitud 6, 8 y 10, y se hace hincapié en el incremento de la complejidad según la longitud de los filtros toda vez que se incrementa el número de parámetros y se introducen restricciones no lineales. El manejo de C restricciones con un AG puede hacerse mediante el método de penalización que básicamente agrega un valor negativo igual a $-\frac{n}{C}K$ al individuo que no cumple las $(C - n)$ restricciones en un problema de maximización, n es el número de restricciones satisfechas y K es una constante positiva suficientemente grande. La función de aptitud para el AG que maximiza el número de coeficientes irrelevantes e incluye el criterio de penalización es:

$$MAX_{aptitud} = (wh - q) + K(1 - \frac{r}{C}) \quad (8)$$

donde r es el número de restricciones satisfechas. El principal problema al que se enfrenta el diseño de filtros personalizados es el tiempo de ejecución invertido. En [11] se reporta un promedio de 165 y 634 segundos para filtros de longitud 8 y 10 respectivamente. Ante tal problema se han explorado las propiedades de las parametrizaciones reportadas en [8] y se encontró que las soluciones de los filtros de longitud cuatro cumplen las condiciones impuestas en las parametrizaciones de filtros de longitud seis, las de longitud seis cumplen las condiciones de las parametrizaciones de filtros de longitud ocho, y así sucesivamente. Esto significa que un filtro de longitud cuatro $H_4 = \{h_4[0], h_4[1], h_4[2], h_4[3]\}$ al cual se le completa la longitud a seis con valores en cero digamos $H_6 = \{h_4[0], h_4[1], h_4[2], h_4[3], 0, 0\}$ es solución para un filtro de longitud 6, y lo mismo ocurre con filtros más largos. Esta propiedad es particularmente útil para proponer filtros de longitud 8 y 10 en donde crear soluciones que cumplan las restricciones (5) y (7) no es tarea fácil. Así surgió la idea de ejecutar un AG4 que optimice el parámetro α_4 del filtro de longitud cuatro, un segundo AG6 que optimice los parámetros α_6 y β_6 de los filtros de longitud seis, un tercer AG8 que optimice los parámetros α_8 , β_8 , γ_8 y θ_8 de filtros de longitud ocho, y un cuarto AG10 que optimice los parámetros α_{10} , β_{10} , γ_{10} , θ_{10} y δ_{10} de los filtros de longitud diez, e ir "exportando" los individuos de AG en AG según la longitud de los filtros en forma ascendente.

Los cuatro AGs se ejecutan simultáneamente y en cuanto uno de ellos tiene un cambio en su individuo más apto lo introduce reemplazando el individuo menos apto de otra población en la secuencia siguiente: del AG4 para el filtro de longitud 4 a la población del AG6, del AG6 para el filtro de longitud 6 a la población del AG8, del AG8 para el filtro de longitud 8 a la población del AG10. La premisa es que al ser más sencillo encontrar soluciones para filtros cortos se alcanzará más rápido una solución a problemas de filtros más largos con más parámetros. Los filtros Daubechies D4, D6, D8 y D10 se usan para inicializar individuos de las poblaciones de AG4, AG6, AG8 y AG10 por dos razones principales: primera, porque maximizan los momentos de desvanecimiento, y segunda, porque cumplen las condiciones de las parametrizaciones respectivas. La segunda condición es

particularmente importante en las parametrizaciones de filtros de longitud 8 y 10 ya que al inicializar aleatoriamente los individuos es altamente probable que no cumplan las restricciones no lineales (ec (5) y ec. (7)).

Para realizar la codificación de un filtro de longitud cuatro $H4=\{h4[0], h4[1], h4[2], h4[3]\}$ en la parametrización de longitud seis se requiere usar las siguientes expresiones:

$$\begin{aligned}\alpha_6 &= \text{acos}(2\sqrt{2}(1/4 - h6[2])) \\ \beta_6 &= \text{atan}\left(\frac{h6[1]-h6[5]}{h6[0]-h6[4]}\right)\end{aligned}\quad (9)$$

Para realizar la codificación de un filtro de longitud seis $H6=\{h6[0], h6[1], h6[2], h6[3], h6[4], h6[5]\}$ en la parametrización de longitud ocho se usan las siguientes expresiones:

$$\begin{aligned}\alpha_8 &= \text{acos}(2\sqrt{2}(h8[0] + h8[4] - 1/4)) \\ \gamma_8 &= \text{atan}\left(\frac{-h8[6]+1/8-\frac{\cos(\alpha_8)}{4\sqrt{2}}}{h8[0]-1/8-\frac{\cos(\alpha_8)}{4\sqrt{2}}}\right) \\ \beta_8 &= -\text{acos}\left(2\sqrt{2}\frac{h8[0]-1/8-\frac{\cos(\alpha_8)}{4\sqrt{2}}}{\cos(\gamma_8)}\right) \\ \theta_8 &= \text{acos}\left(2\sqrt{2}\frac{h8[1]-1/8-\frac{\sin(\alpha_8)}{4\sqrt{2}}}{\sin(\beta_8)}\right)\end{aligned}\quad (10)$$

y para codificar un filtro de longitud ocho $H8=\{h8[0], h8[1], h8[2], h8[3], h8[4], h8[5], h8[6], h8[7]\}$ en la parametrización de longitud diez se usan las expresiones:

$$\begin{aligned}\alpha_{10} &= \pi - \text{asin}((1/4 - (h10[3] + h10[7]))2\sqrt{2}), \text{ o} \\ \alpha_{10} &= \text{asin}((1/4 - (h10[3] + h10[7]))2\sqrt{2}) \\ \theta_{10} &= \text{atan}\left(\frac{(h10[3]-1/8+\sin(\alpha_{10})/(4\sqrt{2}))}{(-h10[5]+1/8+\sin(\alpha_{10})/(4\sqrt{2}))}\right) \\ \beta_{10} &= \text{asin}(((h10[3] - 1/8 + \sin(\alpha_{10})/(4\sqrt{2}))2\sqrt{2})/\sin(\theta_{10}))\end{aligned}$$

$$\begin{aligned}\gamma_{10} &= \text{asin}(((1/8 - 1/(4\sqrt{2})) * \cos(\alpha_{10}) - h10[6])2\sqrt{2})/(\cos(\beta_{10}))) \\ r_{10} &= \sqrt{0,5 - ((h10[2]^2 + h10[3]^2 + h10[4]^2 + h10[5]^2 + h10[6]^2 + h10[7]^2))} \\ \delta_{10} &= \text{acos}\left(\frac{(h10[0]-h10[8])}{r_{10}}\right)\end{aligned}\quad (11)$$

La relación entre los algoritmos genéticos AG4, AG6, AG8 y AG10 "encadenados" se ilustra en la Figura 1, en donde también se muestra la codificación de un individuo correspondiente al filtro Daubechies-4 codificado con α_4 en 30 bits. Para fines ilustrativos ese mismo individuo se exporta mediante (9) para codificarlo en AG6 donde se codifica con $\alpha_6\beta_6$. Con (10) se exporta a AG8 en donde cumple la restricción (5), y por lo tanto está en región factible de AG8 y no se penaliza. Finalmente usando (11) se exporta a AG10 y cumple la restricción (7) así que también está en la región factible de AG10.

Los AG (no tradicionales) usados tienen las siguientes características [14]: una población de tamaño par P , con población temporal de tamaño $2P$. Cruzamiento en pares entre el i -ésimo y el $(P-1-i)$ -ésimo elementos ordenados del más apto al menos apto con $i \in [0, P-1]$. Selección determinista, los P individuos más aptos de la población temporal, y cruzamiento anular.

El Algoritmo 1 describe el método propuesto que calcula el mejor filtro ortogonal evolutivo (Évolet) que maximiza el número de coeficientes irrelevantes para una imagen en escala de grises.

Algoritmo 1 Algoritmo que calcula el mejor Évolet para una imagen PGM

ENTRADA: IMAGEN PGM de wh pixeles

SALIDA: FILTRO DE LONGITUD 10 ORTOGONAL

repeat

 Inicializar poblaciones de AG4, AG6, AG8 y AG10

 Inicializar individuos más aptos I4, I6, I8, I10 con filtros Daubechies

 Aplicar cruzamiento, mutación, evaluación, ordenamiento en AG4

 Seleccionar al individuo I4 más apto de AG4 que codifica α_4

 Calcular el filtro H4 de longitud 4 usando α_4 (ver ec. (2))

if Aptitud(I4) > Aptitud(I4 previo) **then**

 I4 previo \leftarrow I4

 Calcular el valor de α_6, β_6 para AG6 con $H6=H4 \cup \{0, 0\}$ (ver ec. (9))

 Individuo menos apto de AG6 \leftarrow Individuo más apto I4 de AG4

end if

 Aplicar cruzamiento, mutación, evaluación, ordenamiento en AG6

 Seleccionar al individuo I6 más apto de AG6 que codifica α_6, β_6

 Calcular el filtro H6 de longitud 6 usando α_6, β_6 (ver ec. (3))

if Aptitud(I6) > Aptitud(I6 previo) **then**

 I6 previo \leftarrow I6

 Calcular el valor de $\alpha_8, \beta_8, \gamma_8, \theta_8$ para AG8 con $H8=H6 \cup \{0, 0\}$ (ver ec. (10))

 Individuo menos apto de AG8 \leftarrow Individuo más apto I6 de AG6

end if

 Aplicar cruzamiento, mutación, evaluación, ordenamiento en AG8

 Seleccionar al individuo más apto I8 de AG8 que codifica $\alpha_8, \beta_8, \gamma_8, \theta_8$

 Calcular el filtro H8 de longitud 8 usando $\alpha_8, \beta_8, \gamma_8, \theta_8$ (ver ec. (4))

if Aptitud(I8) > Aptitud(I8 previo) **then**

 I8 previo \leftarrow I8

 Calcular el valor de $\alpha_{10}, \beta_{10}, \gamma_{10}, \theta_{10}, \delta_{10}$ para AG10 usando $H10=H8 \cup \{0, 0\}$ (ver ec. (11))

 Individuo menos apto de AG10 \leftarrow Individuo más apto I8 de AG8

end if

 Seleccionar al individuo más apto I10 de AG10 que codifica $\alpha_{10}, \beta_{10}, \gamma_{10}, \theta_{10}, \delta_{10}$

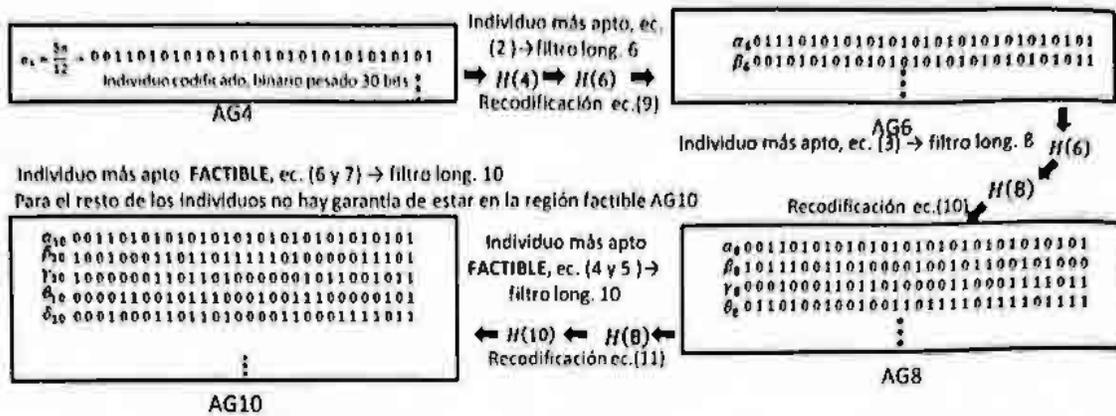
until (Alcanzar MAXGENERACIONES)

return Solución: El mejor filtro evolutivo ortogonal H10 calculado con I10 y ec. (6).

5. Experimentos

Las imágenes usadas en los experimentos son ampliamente conocidas en el tema de procesamiento de imágenes con los nombres de "Baboon", "Barbara", "Clegg", "Frymire", "Lena", "Monarch", "Peppers" y "Sail" con dimensiones

Figura 1. Algoritmos genéticos encadenados para calcular Evolets.



$w=64$, $h=64$ coeficientes/píxeles. Cada parámetro se codificó en binario pesado con 30 bits de precisión en cada individuo de los algoritmos genéticos. Sean P_m la probabilidad de mutación, P_c la probabilidad de cruzamiento, G el número de generaciones, y P el tamaño de la población en los AG de los experimentos.

Experimento 1. Consistió en ejecutar de forma independiente los algoritmos AG4, AG6, AG8 y AG10 para maximizar la función de aptitud descrita en la ec. (8) con las siguientes condiciones, AG4: $P_m = 0,02$, $P_c = 0,97$, $P = 4$, $G = 60$; AG6: $P_m = 0,02$, $P_c = 0,97$, $P = 6$, $G = 60$; AG8: $P_m = 0,01$, $P_c = 0,97$, $P = 16$, $G = 60$, $K = 1000000$; AG10: $P_m = 0,01$, $P_c = 0,97$, $P = 40$, $G = 60$, $K = 1000000$.

El valor de K se ha elegido suficientemente grande respecto al valor máximo de las aptitudes de los individuos sin penalización que están en $[0, 4096)$.

Experimento 2. Consistió en ejecutar la implementación del Algoritmo 1 (AGs encadenados, AG4-6-8-10) con las mismas condiciones que en el experimento 1. La Tabla 1 reúne los resultados experimentales de ambos experimentos y muestra el número de coeficientes alcanzado por los filtros *Daubechies* D2 (Haar), D4, D6, y D8, los Evolets usando AG4, AG8 y AG10 en forma independiente, así como el método propuesto para filtros de longitud diez. Como se puede observar tanto el AG10 como AG4-6-8-10 convergen aproximadamente a la mismas soluciones, en 3 de 8 casos AG4-6-8-10 mejora el número de coeficientes irrelevantes, en 4 casos convergen al mismo resultado, y sólo en un caso (Peppers) AG4-6-8-10 queda por debajo de AG10, con los mismos parámetros del experimento 1.

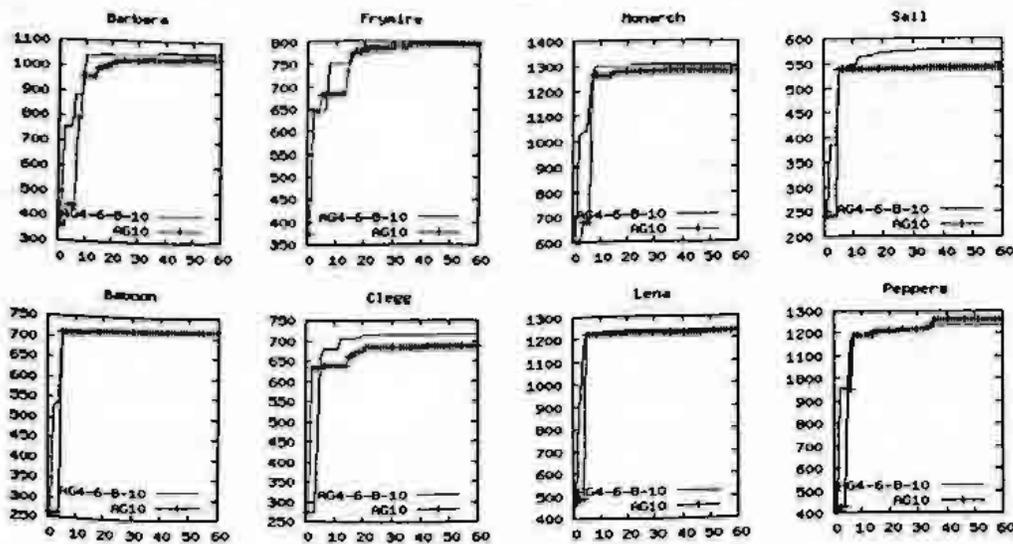
Experimento 3. Este último experimento atiende el tema fundamental de este trabajo, que es lograr la convergencia a la solución óptima lo más rápido posible. La Figura 2 muestra el desempeño de AG10 y de AG4-6-8-10 usando las ocho imágenes de prueba. El eje vertical indica el número de coeficientes irrelevantes (llevados a cero) y el eje horizontal es el número de generaciones ejecutadas. Obsérvese que en las gráficas los AGs encadenados tienden a estar por encima de AG10 y por lo tanto se puede decir que, en efecto, los primeros

convergen más rápido que el AG10. Respecto al tiempo de ejecución el tiempo promedio requerido para 60 generaciones en cada imagen de prueba es 1295.375 segundos con desviación estándar 18.3 de que da un promedio de 21.5 segundos por cada generación⁵. Es posible estimar el número de generaciones y tiempo que requieren los AGs encadenados para igualar o mejorar el valor máximo promedio alcanzado por AG10 (955.625) en 60 generaciones y que son 20 generaciones en aproximadamente 430 segundos.

Tabla 1. Resultados experimentales para 1. Baboon, 2. Barbara, 3. Clegg, 4. Frymire, 5. Lena, 6. Monarch, 7. Peppers and 8. Sail.

Imagen	Baboon	Barbara	Clegg	Frymire	Lena	Monarch	Peppers	Sail
Filtro	$wh - q$							
Haar/D2	214	327	272	373	419	413	371	228
D4	220	348	251	279	464	567	392	240
D6	261	359	251	231	443	595	385	231
D8	215	320	263	227	433	594	391	216
Evolet/AG4	244	354	278	373	473	579	403	256
Evolet/AG6	265	374	284	375	466	603	411	262
Evolet/AG8	661	1042	689	667	1218	1291	1227	534
Evolet/AG10	716	1035	717	796	1239	1302	1270	570
Evolet/AG4-6-8-10	717	1052	717	796	1239	1302	1238	576

Figura 2. AG4-6-8-10 vs AG10 al maximizar el número de coeficientes wavelet irrelevantes (eje Y: número de coeficientes cero, eje X: número de generaciones).



⁵ Los datos numéricos de la gráfica, de los experimentos con cada imagen, así como las gráficas de los Evolets, no se incluyen por cuestiones de espacio pero están disponibles con petición a los autores.

6. Conclusiones y trabajo futuro

Es posible reducir el tiempo requerido para resolver el problema de calcular el mejor Evolet para una imagen usando varios algoritmos genéticos encadenados aprovechando las propiedades de las parametrizaciones de filtros basados en wavelets. Es evidente que los filtros "estándar" no proveen la mejor alternativa ante un conjunto diverso de problemas que incluyen a la compresión de imágenes, y por tanto se establece el compromiso entre tiempo invertido y la compresión lograda. La alternativa presentada en este trabajo resulta prometedora ya que se ha acelerado la convergencia a la solución global. En trabajos futuros se buscará disminuir el tiempo de cómputo a través de una versión paralela donde los algoritmos genéticos participantes se ejecuten en procesadores o computadoras distintas, lo cual permitiría experimentar con filtros más largos y con imágenes de mayor dimensión. También, como trabajo futuro, se plantea hacer un análisis estadístico con un mayor número de imágenes.

Referencias

1. Mallat, S.: A wavelet tour of signal processing. Academic Press Inc. (1998)
2. Daubechies, I.: Ten lectures on wavelets. Society for Industrial and Applied Mathematics, Philadelphia, PA, USA (1992)
3. Walker, J.: A primer on wavelets and their scientific applications, 2nd edition. Chapman Hall/CRC, Taylor and Francis Group (2008)
4. Shannon, C.: A mathematical theory of communication. Bell System Technical Journal **27** (July 1948) 379-423
5. Wang, H., Peng, L.: Parameterizations of univariate wavelet tight frames with short support. Communications in Nonlinear Science and Numerical Simulation **11** (2006) 663-677
6. Hehong, Z., Tewfik, A.: Parametrization of compactly supported orthonormal wavelets. IEEE Transactions on signal processing **41**(3) (march 1993) 1428-1431
7. Schneid, J., Pittner, S.: On the parametrization of the coefficients of dilation equations for compactly supported wavelets. Computing **51** (1993) 165-173
8. Ming-Jun, L., Roach, W., D.: Parameterization of univariate orthogonal wavelets with short support. Aproximation theory X. Vanderbilt Univ. Press (2002)
9. Beylkin, G., Coifman, R., Rokhlin, V.: Fast wavelet transforms and numerical algorithms. Commun. Pure Appl. Math. **44** (1991) 141-183
10. Herrera, O.: On the best evolutionary wavelet based filter to compress a specific signal. Springer LNCS 6438 Advances in Soft Computing **1**(1) (November 2010) 394-405
11. Herrera, O.: Optimization of parameterized compactly supported orthogonal wavelets for data compression. Springer LNCS 7095 **1**(1) (November 2011) a ser publicado
12. Hereford, J.M., Roach, D.D.W., Pigford, R.: Image compression using parameterized wavelets with feedback. SPIE (2003) 267-277
13. Shapiro, J.: Embedded image coding using zerotrees of wavelet coefficients. IEEE Trans. Signal Processing. **41** (December 1993) 3445-3462
14. Kuri, A.: A Comprehensive Approach to Genetic Algorithms in Optimization and Learning. National Polytechnic Institute, Mexico (1999)

Detección de hipovigilancia en un automovilista: Utilización de wavelets para el análisis del ángulo del volante y posición lateral

Rodolfo Ibarra-Orozco¹, Miguel González-Mendoza², Neil Hernández-Gress¹,
Juan Carlos López-Pimentel¹, Victor Ramos-Fon Bon¹, and Cristian
Pon-Gómez¹

¹ Universidad Politécnica de Chiapas

² Instituto Tecnológico y de Estudios Superiores de Monterrey
Campus Estado de México
{ngonza, ngress}@itesm.mx

Abstract. Este artículo describe nuestra experiencia en el uso de wavelets para la extracción de características en señales que apoyen en la detección de hipovigilancia en un automovilista. Las señales utilizadas para dicho análisis son las variables mecánicas ángulo del volante y posición lateral del automóvil.

Esta extracción de características tiene la finalidad de alimentar un sistema de fusión de datos que sea capaz de realizar un diagnóstico con la información que esta extracción le proporcione. De esta forma se espera un diagnóstico más robusto que el realizado por un solo sensor.

El artículo presenta primero una introducción al uso de wavelets para filtrado y extracción de característica, posteriormente se revisa su uso para el análisis de la posición lateral y ángulo del volante de un automóvil. Finalmente, presenta las conclusiones y trabajo futuro.

Keywords: wavelets, detección de hipovigilancia

1 Introduction

Un operador de un sistema hombre-máquina es capaz de reconocer cuando se encuentra en un estado de somnolencia, pero no necesariamente traslada esta introspección en predicciones precisas, tales como cuánto tiempo estarán sus ojos cerrados o cuándo estará experimentando señales de adormecimiento o cuándo tendrán una fase de sueño incontrolado. En este trabajo abordamos la detección de hipovigilancia en un sistema hombre-máquina, específicamente en el de conductor-automovilista.

Un médico experto en sueño puede generar diagnósticos de la evolución del sueño de un conductor por simple supervisión visual pero que están altamente influenciados por muy diversas variables contextuales que son muy difíciles o imposibles de medir. En este sentido, la tecnología puede ofrecer la generación de una alerta más pronta y confiable que indique el deterioro del nivel de desempeño y del nivel de sueño de un operador, antes de que tal deterioro conlleve

a una catástrofe. La metodología propuesta en [1] utiliza variables mecánicas del automóvil (posición lateral y ángulo del volante) para predecir el estado de vigilancia del conductor. En este artículo se describe el análisis realizado mediante el uso de bancos de filtros de wavelets para obtener las características principales de las señales de las variables utilizadas.

2 Filtrado y extracción de características

2.1 Análisis en tiempo-frecuencia

La mayoría de las señales de fenómenos del mundo real son no estacionarias. Ejemplos de estas son las señales generadas por electroencefalogramas, electroculogramas, voz, sonidos de animales, etc. Estas señales varían con el tiempo su intensidad y su contenido en frecuencia. Así, si se quiere saber en que momento (tiempo) ocurrió una frecuencia, se necesitan otras técnicas para el análisis.

En esta sección, se describen herramientas matemáticas para la descripción de señales no estacionarias. Este tipo de análisis es llamado análisis en tiempo-frecuencia. En particular, describimos el uso de la transformada en Wavelets. Para una referencia más amplia se recomienda consultar [6] y [4].

Wavelets. Se puede pensar en las wavelets como oscilaciones que están localizadas al mismo tiempo en tiempo y frecuencia. Una representación de una función o señal basada en estas oscilaciones localizadas es atractiva porque los coeficientes asociados con cada wavelet revelan detalles en tiempo y frecuencia. Un coeficiente pequeño significa una pequeña variación en la función o señal en la vecindad de esa particular pequeña oscilación, mientras que coeficientes con un valor alto indican un cambio apreciable en la señal o función, [2].

Para analizar estructuras de la señal de muy diferentes tamaños, es necesario utilizar átomos en tiempo-frecuencia con diferentes soportes en el tiempo. La transformada en wavelets descompone a la señal en wavelets dilatadas y trasladadas. Una wavelet es una función $\psi \in L^2(\mathbb{R})$ con media cero:

$$\int_{-\infty}^{\infty} \psi(t) dt = 0 \quad (1)$$

Una familia de átomos en tiempo-frecuencia es obtenida al escalar ψ con s y trasladándola con u .

$$\psi_{u,s}(t) = \frac{1}{\sqrt{s}} \psi\left(\frac{t-u}{s}\right) \quad (2)$$

La transformada en wavelets de $f \in L^2(\mathbb{R})$ en el tiempo u y en la escala s es:

$$Wf(u, s) = \int_{-\infty}^{\infty} f(t) \frac{1}{\sqrt{s}} \psi^*\left(\frac{t-u}{s}\right) dt \quad (3)$$

Hay dos características importantes del análisis con wavelets que lo hacen atractivo para esta investigación:

- **Compresión de datos / limpieza de la señal** Una buena base de funciones debe de ser capaz de representar la función con el menor número de bases posible. Para una función arbitraria, un número grande de coeficientes de wavelets son pequeños o cero. Esto en contraste con la transformada de Fourier, donde las discontinuidades llevan a valores de coeficientes mayores. Coeficientes en cero o con valores muy pequeños contribuyen muy poco en la reconstrucción de la señal, por lo que pueden ser eliminados y con eso se realiza una buena compresión de datos. Esta supresión de coeficientes realiza una buena limpieza de señales con ruido.
- **Extracción de características:** La descomposición en wavelets identifica escalas altas, medias y bajas en la función (análisis en multiresolución), [4]. Coeficientes con valor no cero pueden estar relacionados con el valor de la derivada de la función en una región en particular. Esta relación es útil para la extracción de características, los coeficientes con valor grande identifican gradientes altos o discontinuidades en la función.

La transformada en wavelets continua La transformada en wavelets continua (Continuous Wavelet Transform, CWT) se basa en la convolución de una wavelet madre ψ que es dilatada (s) y defasada (τ), (ecuación 4) para tener una aproximación y descomposición de la señal original. La wavelet es desplazada a lo largo de la señal y para cada posición el espectro es calculado. Luego este proceso es repetido muchas veces con una ventana ligeramente más corta o larga en cada nuevo ciclo. Finalmente se obtiene una colección de representaciones tiempo-frecuencia de la señal a diferentes resoluciones o escalas (s), (ecuación 5).

$$\psi_s^\tau(t) = \frac{1}{\sqrt{s}} \psi\left(\frac{t-\tau}{s}\right) \quad (4)$$

$$C(s, \tau) = \langle f, \psi_{s,t} \rangle = \frac{1}{\sqrt{|s|}} \int f(t) \psi\left(\frac{t-\tau}{s}\right) \quad (5)$$

Esta transformación es en teoría infinitamente redundante, pero puede ser útil en el reconocimiento de ciertas características de la señal

La transformada en wavelets discreta y multiresolución En una transformada de wavelets discreta, una wavelet es trasladada y dilatada únicamente por valores discretos. Frecuentemente esta dilatación es a una potencia de 2 (llamada diádica). Así, la wavelet es utilizada de la forma:

$$\psi_s^{2^k t+l} \quad (6)$$

donde k y l son números enteros.

Con la teoría de multiresolución, Stéphane Mallat relacionó las wavelets ortogonales con los filtros utilizados en procesamiento de señales. En este método se utiliza una nueva función, la función de escala, la cual produce una serie de

imágenes de la señal, cada una diferente por un factor de dos de la señal previa. En una dirección, estas imágenes sucesivas aproximan la señal con cada vez mayor precisión, acercándose a la original. En la otra dirección, se acercan a cero, conteniendo cada vez menor información.

La función de las wavelets en este método es la de obtener la diferencia de información entre dos resoluciones.

A las señales obtenidas al aplicarles la función de escala se les llama aproximaciones y a las obtenidas al aplicarles la wavelet se les llama detalles.

La multiresolución puede aproximar cualquier función sin redundancia y con precisión arbitraria si son cumplidas las siguientes cuatro condiciones:

- La función de escala debe ser ortogonal a sus traslados por enteros.
- La señal, descompuesta a una resolución dada j , contiene toda la información de esa misma señal descompuesta a una resolución menor $j - k$.
- La función 0 es el único objeto común en todos los espacios V_j .
- Cualquier señal puede ser aproximada con una precisión arbitraria.

La multiresolución es descrita primero por los subespacios V_j y W_j . Los espacios de escala V_j . W_j es la diferencia entre V_j y V_{j+1} . La suma de V_j y W_j es V_{j+1} . Cada V_j está contenido en el siguiente subespacio V_{j+1} . Una función en un subespacio está en todos los subespacios más altos (finos).

$$V_0 \subset V_1 \subset \dots \subset V_j \subset V_{j+1} \subset \dots \quad (7)$$

Una función $f(t)$ en el espacio completo tiene una pieza en cada subespacio. Esas piezas contienen más y más información de la información completa en $f(t)$. La pieza en V_j es $f_j(t)$. Un requerimiento en la secuencia de subespacios es la completitud:

$$f_j(t) \rightarrow f(t) \text{ as } j \rightarrow \infty \quad (8)$$

Visto desde el punto de vista en que

$$V_j \oplus W_j = V_{j+1} \quad (9)$$

Cada función en V_j es la suma de dos partes ortogonales, f_j en V_j y Δf_j en W_j .

El espacio W_j contiene términos de grado $j+1$. Estos son ortogonales a todos los términos de grado $\leq j$.

$$V_0 \oplus W_0 = V_1 \quad \text{y} \quad V_1 \oplus W_1 = V_2 \quad (10)$$

De donde se tiene que:

$$V_0 \oplus W_0 \oplus W_1 = V_2 \quad (11)$$

Agregando más detalles se tiene:

$$V_0 \oplus W_0 \oplus W_1 \oplus \dots \oplus W_j = V_{j+1} \quad (12)$$

para las funciones en estos subespacios, esta ecuación es:

$$f_0(t) + \Delta f_0(t) + \Delta f_1(t) + \dots + \Delta f_j(t) = f_{j+1}(t) \quad (13)$$

La construcción de wavelets ha tenido éxito encontrando V_j primero, es decir, la función de escala $\psi(t)$. Las translaciones de esta wavelet $\psi(t - k)$ caen en V_0 . Reescalando por 2^j se obtiene V_j . Entonces los espacios de la wavelet W_j son las diferencias entre V_{j+1} y V_j . Las funciones en W_j son los detalles en la escala j .

De manera similar para los bancos de filtros, primero se diseña el filtro pasa bajas y después se diseña el filtro pasa altas, que son análogos a la función de escala y a la función wavelet, respectivamente.

3 Detección de Singularidades usando el módulo máximo de la transformada en wavelets

En la mayoría de las señales de información, desde electrocardiogramas hasta señales de radar, la información relevante está contenida en fenómenos transitorios o picos en la señal. Así, si se realiza una descomposición de la señal en bloques elementales, los cuales estén bien identificados tanto en tiempo como en frecuencia, y utilizando la transformada en wavelets es posible caracterizar las regularidades locales de la señal, [3]. Mallat, [3], demostró que el módulo máximo de la transformada en wavelets detecta todas las singularidades de una función y aportó un conjunto de estrategias para medir las regularidades de Lipschitz.

3.1 Selección de la wavelet a utilizar

En general, hay dos tipos de seleccioes a realizar: el sistema de representación (continua, discreta) y las propiedades de las wavelets (número de momentos de desvanecimiento, el grado de regularidad y la selectividad en frecuencia).

Sistema de representación. La ortogonalidad confiere concisión y velocidad al precio de escalas arbitrarias y posiblemente dificultad en el análisis, especialmente en el reconocimiento de patrones. Las transformadas continuas tienen, en general, la ventajas y desventajas inversas.

Las wavelets biortogonales son otra opción. Estas proveen reconstrucción perfecta de la señal, sin redundancia, usando dos pares de wavelets: un par para descomposición y el otro para la reconstrucción. Esto hace posible tener wavelets simétricas con soporte compacto, una combinación que no es compatible con la ortogonalidad, excepto en el caso de la wavelet Haar.

Para algunas aplicaciones, como el análisis numérico, esta simetría no es importante, pero puede ser útil en procesamiento de imágenes, ya que los errores en cuantización pueden ser menos evidentes cuando se usan wavelets simétricas.

Regularidad. El orden de regularidad de la wavelet es el número de sus derivadas continuas. Codificando una imagen con una función discontinua como la wavelet Haar produce una imagen discontinua.

Al incrementar el orden de la regularidad incrementa la longitud del soporte, lo que hace que los cálculos sean mayores.

Momentos de desvanecimiento. El número de momentos de desvanecimiento de termina lo que la wavelet no puede ver. Una wavelet con un momento de desvanecimiento no ve funciones lineales: el coeficiente de la wavelet es cero. Dos momentos de desvanecimiento hacen que la wavelet no vea funciones cuadráticas y así sucesivamente. Con una wavelet con muchos momentos de desvanecimiento se obtienen coeficientes con valores pequeños cuando se analizan bajas frecuencias; los coeficientes miden la semejanza entre la wavelet y la señal, y una wavelet que oscila rápidamente no se asemeja a ondas de baja frecuencia.

Una wavelet tiene m momentos de desvanecimiento si y sólo si puede generar polinomios de grado menor o igual a m . Mientras esta propiedad es utilizada para describir el poder de aproximación de las funciones de escala, en el caso de las wavelets se usa también para conocer la posibilidad de caracterizar el orden de las singularidades.

El número de momentos de desvanecimiento está determinado por los coeficiente $h(n)$ del filtro h que se encuentra caracterizado en la función de escala.

El efecto práctico de los momentos de desvanecimiento es concentrar la información de la señal en un número relativamente pequeño de coeficientes.

Selectividad de frecuencia. En el análisis de Fourier, la función analizadora es una senoide de frecuencia precisa, que calcula un coeficiente que se refiere exclusivamente a esa frecuencia. Una wavelet se compone de una mezcla de frecuencias, por lo que los coeficientes de la wavelet se refieren también a una mezcla de frecuencias. En el mejor de los casos, uno quisiera tener wavelets bien localizadas o concentradas en frecuencia y tiempo: wavelets muy selectivas en frecuencia con un soporte muy compacto.

Mientras la complejidad de la wavelet sea mayor, es decir, tenga un número mayor de momentos de desvanecimiento, el soporte de la wavelet es más grande y por lo tanto crece la complejidad para calcular los coeficientes de la wavelet. Por otro lado, mientras se tenga un mayor número de momentos de desvanecimiento, se podrá aproximar una función suave, por lo que los coeficientes de la wavelet serán pequeños, lo que significa que la calidad de la reconstrucción al suprimir coeficientes será mejor que la obtenida con una wavelet con menor número de momentos de desvanecimiento.

La selección de la wavelet a utilizar depende tanto de la señal a analizar y del propósito del análisis, como de la experiencia previa del usuario. No existe un consenso general de que tanto se deba trabajar para encontrar la mejor wavelet para un aplicación dada y tampoco existe guías para realizar tal decisión. Sin embargo, existen algunas propiedades que son deseables para ciertas aplicaciones. Por ejemplo, el uso de wavelets con cinco o seis momentos de desvanecimiento son útiles para análisis numérico, pero no tienen sentido para procesamiento de imágenes.

Debido a que estamos utilizando las wavelets para detectar singularidades en la señal, no estamos interesados en la reconstrucción de la señal. Nuestro principal interés es detectar las singularidades en un tiempo muy corto. Para lograr esto, se prefiere el uso de wavelets ortogonales con soporte compacto. De esta manera, al realizar el análisis de la señal, empezamos con las familias de

wavelets que cumplan estos requisitos y, si no se obtienen los resultados esperados, incrementamos el número de momentos de desvanecimiento para aproximar mejor la señal y así poder encontrar los cambios abruptos.

4 Análisis de la descomposición en wavelets de una señal

4.1 Análisis de la posición lateral y el ángulo de volante de un automóvil

En esta sección se ejemplifica el análisis de las señales del ángulo de volante y de la posición lateral. Tras un análisis con diferentes wavelets (Haar, Gaussian y Daubechies), se seleccionó la wavelet db4, debido a que, como se muestra posteriormente en este artículo, nos mostró información relevante de la señal para nuestro análisis.

Primero, una señal del ángulo del volante, así como su distribución de frecuencia se muestra en la figura 1. La frecuencia de muestreo de la señal es de 10 Hz. La respuesta en frecuencia de la señal muestra su mayor concentración en el rango de frecuencia de 0 a 1 Hz. La respuesta en frecuencia de los detalles de las wavelets es también mostrada en la figura para mostrar su distribución en frecuencia. Se puede ver que los detalles en los primeros niveles contienen las frecuencias más altas de la señal. De esta figura se puede observar que los primeros dos detalles contienen muy baja energía (0-18). El rango de frecuencia que contiene la señal del ángulo del volante es de 0 - 3 Hz, [5], por lo que se puede concluir que estos detalles contienen únicamente ruido y pueden ser eliminados del análisis.

El tercer detalle tiene mayor energía (0-80), pero todavía tiene una baja influencia en la distribución de frecuencia de la señal original. En estos detalles, un módulo máximo de la wavelet indica cambios en alta frecuencia, relacionados con cambios abruptos en el ángulo del volante que puede ser debido a la corrección repentina del ángulo del volante debido a que el conductor estaba a punto de dormir.

A escalas medias, en este caso consideraremos los niveles de detalle 4 a 6, con una energía entre 0 y 300, el módulo máximo de la wavelet indica cambios en la dinámica de la señal a frecuencias medias, indicando que una corrección normal ha ocurrido.

Finalmente, los últimos niveles de los detalles de la descomposición, como se puede ver en las figuras 2 and 3, se encuentran correlacionadas con la señal de la posición lateral del automóvil. Esta correlación es de 46% después de una corrección de fase de la señal. Esto es debido a los componentes del camino.

La figura 4 (a) y (b) muestra los detalles de el nivel 5 de descomposición de un conductor alerta y de un conductor somnoliento, respectivamente. Este nivel de análisis es muy importante para nuestro análisis porque contiene el rango de frecuencia donde ocurren las correcciones en el ángulo del volante. De esta figura se puede observar que el módulo máximo de la wavelet tiene valores mayores en un conductor alerta que en un conductor somnoliento. En este ejemplo, el

promedio de los valores del módulo máximo de la wavelet es de 0.52 para el conductor alerta y 0.34 para el conductor somnoliento.

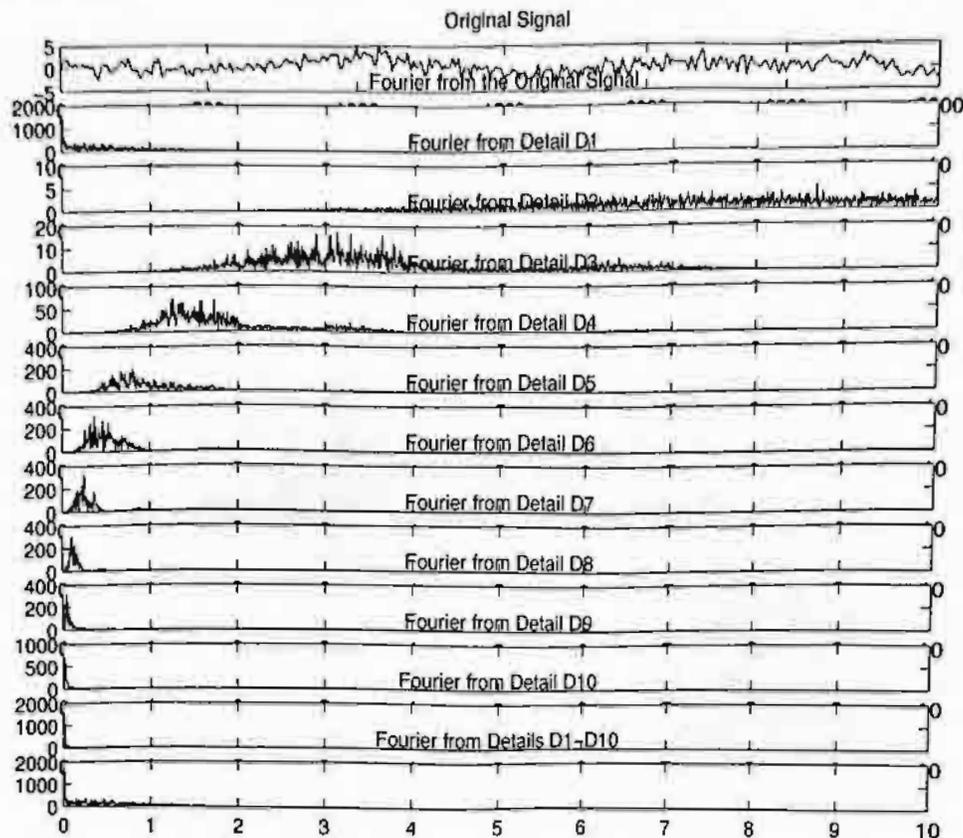


Fig. 1. Señal del ángulo del volante, su transformada de Fourier y la transformada de Fourier de los detalles de la transformada en wavelets

5 Conclusiones y trabajo futuro

En este artículo se muestra que es factible utilizar la transformada en wavelets para extraer características de las señales de posición lateral y ángulo del volante de un vehículo que apoyen a realizar la detección de un estado de hipovigilancia del conductor del vehículo. El trabajo futuro es muy extenso y consiste principalmente, a grandes rasgos de los siguientes puntos:

- Realizar estudios similares con otras señales tanto fisiológicas y de comportamiento del conductor.
- Hacer una fusión de datos efectiva de las señales que nos proporcionen la mayor información. Es necesario establecer el tipo de técnica más apropiado para esta fusión. Específicamente si se realiza con una técnica basada en datos o en datos y conocimiento.
- Establecer una metodología robusta para la validación de resultados.

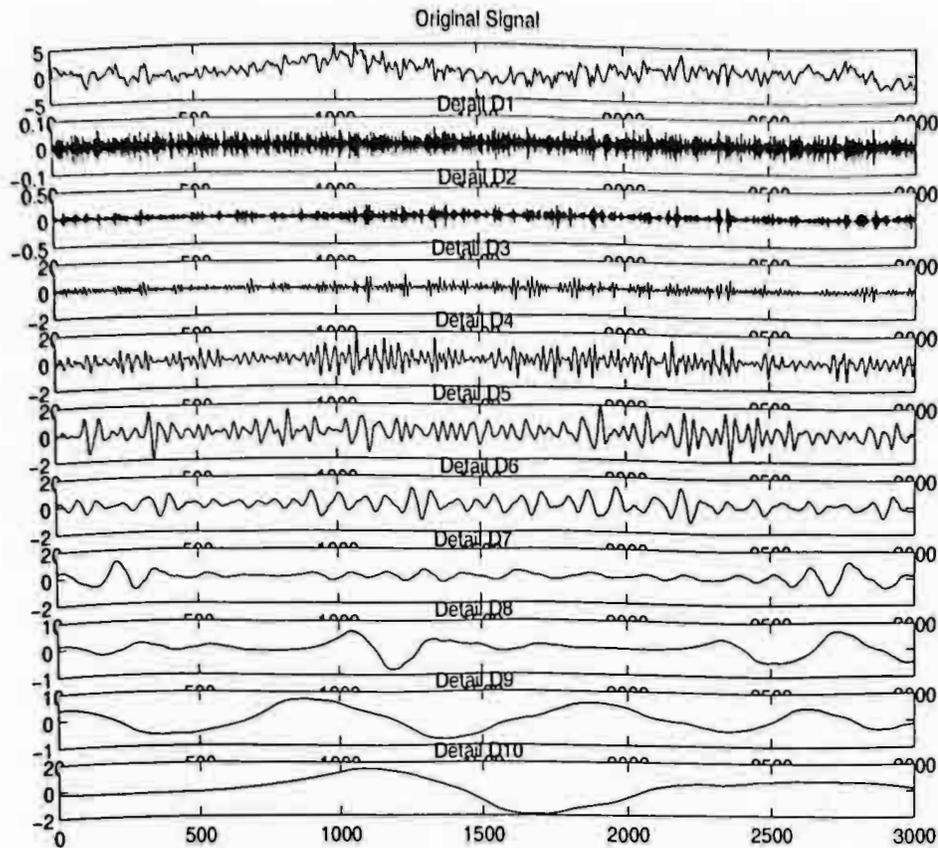


Fig. 2. Detalles de la transformada en wavelets utilizando la wavelet db4

References

1. Rodolfo Ibarra-Orozco, Miguel González-Mendoza, Neil Hernández-Gress, Frederik Diederichs, and Juha Kortelainen. Towards a ready-to-use drivers vigilance monitoring system. *International Conference on Computational Intelligence for Modelling, Control and Automation*, 2008.
2. G.J. Lord, E. Pardo-Igúzquiza, and I.M. Smith. A practical guide to wavelets for metrology. *NPL report CMSC 02/00*, 2000.
3. S. Mallat and L. Ilwang. Singularity detection and processing with wavelets. *IEEE Transactions on information Theory*, 1992.
4. Stéphane Mallat. *A Wavelet Tour of Signal Processing*. Academic Press, 1999.
5. Prashant S. Rao, David Roccaforte, and Ron Campbell. Developing an adams model of an automobile using test data. *Proceedings of the 2002 SAE Automotive Dynamics and Stability Conference and Exhibition*, 2002.
6. Gilbert Strang and Truong Nguyen. *Wavelets and Filter Banks*. Wellesley-Cambridge Press, Ellesley MA, 1997.

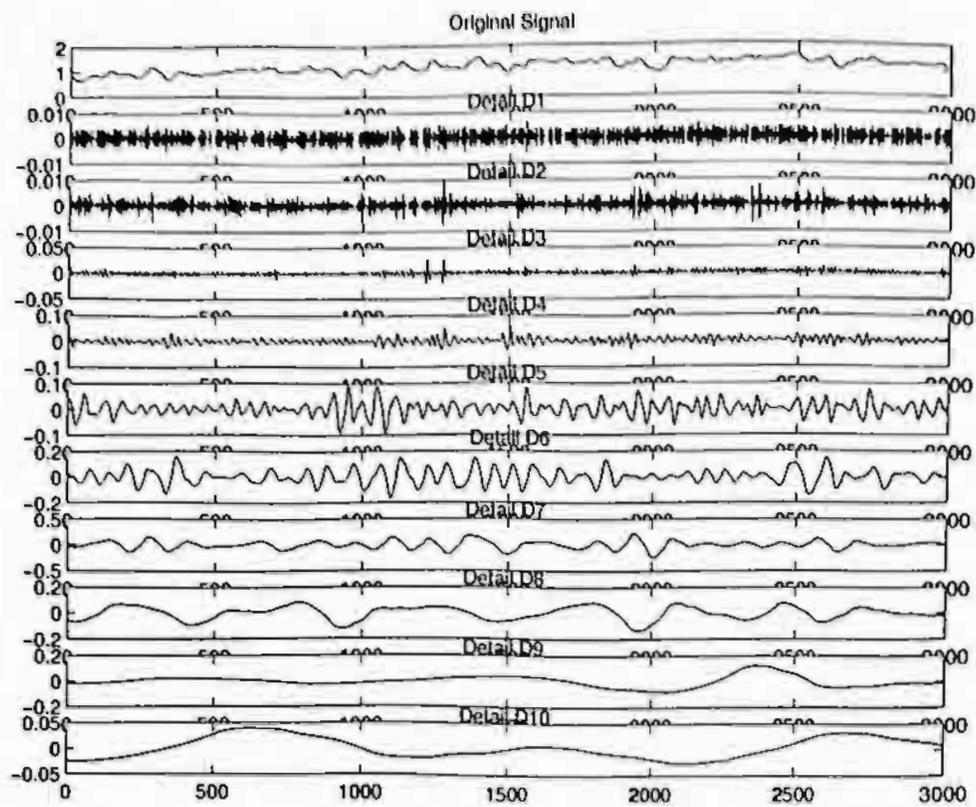


Fig. 3. Detalles de la señal de la posición lateral utilizando db4

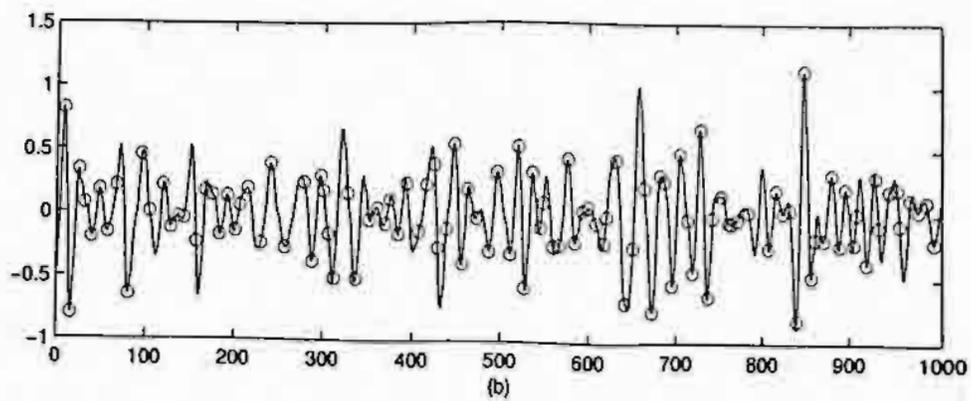
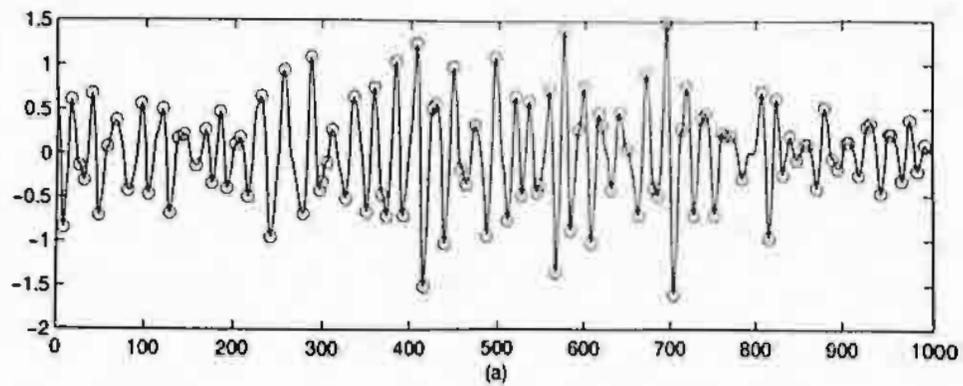


Fig. 4. Detalles del quinto nivel de descomposición de a) un conductor alerta y b) un conductor somnoliento

Modelo de Evaluación de Puntos Compromiso para la Banca en México

Victor Mata-Mata¹, Miguel González-Mendoza¹

¹ Tecnológico de Monterrey Campus Estado de México, Carr. Lago de Guadalupe Km 3.5, Atizapán de Zaragoza, Estado de México, México
{A01168799, mgonza}@itesm.mx

Abstract. El sistema bancario en México utiliza diversas soluciones para determinar el grado en que una transacción financiera es fraudulenta, basando sus cálculos en un conjunto de parámetros señalados por cada entidad financiera. Este proceso de identificación es realizado por los bancos, solamente desde una perspectiva emisora dejando a la parte adquirente sin soluciones que permitan no sólo detectar transacciones con fraude, sino más aun los puntos (POS - ATM) donde la información del tarjetahabiente fue posiblemente comprometida. El objetivo de este trabajo es plantear un modelo que permita rastrear en la historia transaccional de los bancos, aquellos puntos comunes donde se han encontrado transacciones confirmadas con fraude, para sugerir a los bancos puntos compromiso (comercios, terminales punto de venta o cajeros automáticos) que presentan anomalías en el manejo de información y muy posiblemente donde se clono el plástico del cliente.

Keywords: FIID, BIN, Emisor, Adquirente, Switch transaccional.

1 Introducción

A la fecha existen poco más de sesenta entidades financieras en México con plásticos de débito y crédito en el mercado nacional e internacional; sumando para tarjetas de crédito en circulación 11,995 millones de pesos corrientes al cierre de Julio para México [1]. Debido al gran número de bancos, tarjetas y el volumen financiero que representa es imposible que todas las entidades estén pendientes de los movimientos que tienen sus plásticos, no sólo en sus comercios sino también en los bancos competidores. Por tal motivo existen empresas para el procesamiento de transacciones bancarias, en donde el Switch transaccional [10] Mexicano es el líder en Latinoamérica respaldado por marcas como VISA o MasterCard a nivel mundial.

El rol de de un Switch transaccional [10] consiste en recibir todas las transacciones electrónicas sin distinción de banco, comercio, etc. y redirigir la transacción al banco correspondiente generando diariamente cortes monetarios para realizar un proceso de compensación y liquidación entre cada banco. Por tal motivo, no sólo los bancos sino

los mismos operadores de transacciones financieras, requieren innovar en sus procesos operativos para el monitoreo de transacciones potencialmente riesgosas.

1.1 Puntos compromiso en la Banca

Las áreas de prevención de fraude son unidades de trabajo encargadas de la detección de transacciones fraudulentas, bloquear estas tarjetas ya marcadas y en consecuencia, realizar una serie de actividades operativas que surgen como resultado de un fraude. En este sentido es importante hacer notar que las áreas de prevención intentan anticiparse lo más posible al fraude, deteniendo el tipo de tarjeta ya identificada entre muchas otras actividades; pero la raíz del problema se encuentra aún más atrás. Esta problemática cobra aún más relevancia al hacer notar que de acuerdo a la Asociación de Bancos de México (ABM), se estimó que durante 2010 los fraudes con tarjetas de crédito, débito y otros medios de pago ascendieron a más de 600 millones de pesos.

Para realizar una tarea verdaderamente preventiva en tema de fraude se requiere determinar los puntos físicos donde se comprometió la información del tarjetahabiente (puntos compromiso), los cuales son comercios donde se realizó una transacción en terminales punto de venta (POS) o cajeros automáticos (ATM) y por diversos medios se obtuvo la información confidencial del cliente. Bajo este contexto se puede indagar donde se clonó la tarjeta y por lo tanto, donde obtuvo el defraudador los datos confidenciales propios de la tarjeta.

1.2 Detección de fraudes Emisor y Adquirente

Los procesos de monitoreo y detección de fraudes en diversas entidades financieras, están basadas en dos componentes: el conocimiento de sus expertos y herramientas de diagnóstico. El primer componente se presenta más frecuentemente en bancos con bajo - medio volumen transaccional, donde el proceso de toma de decisión se basa en el análisis de expertos; los cuales revisan el comportamiento transaccional y bajo su criterio dictaminan si una aclaración o transacción es fraude o no. El proceso para decidir si una transacción es fraude, se basa en gran medida en el tipo de banco, características del BIN [9] y a la experiencia del analista mientras revisa las transacciones en su equipo de trabajo. El segundo componente, se presenta más en organismos financieros con un volumen transaccional alto, debido a los altos costos en licencias e infraestructura. Estos bancos realizan los procesos de prevención y análisis por medio de herramientas computacionales que monitorean en tiempo real el comportamiento transaccional. Estas soluciones realizan una evaluación solamente parcia de las transacciones; esto quiere decir que las herramientas sugieren un conjunto reducido de transacciones peligrosas, que requieren la evaluación final del experto.

Como se puede apreciar el conocimiento del analista dentro de la organización es clave dentro del proceso, por tal motivo existen un sin número de esfuerzos por disminuir las operaciones con fraude. Entre estos ejemplos se encuentra la circular que emitió la Comisión Nacional Bancaria y de Valores (CNBV), la cual estipula que todas las instituciones deberán migrar sus tarjetas a tecnología de Chip o en tal caso absorber el quebranto ante un fraude y abonar las transacciones que el tarjetahabiente no reconozca en un plazo de 48 horas.

Este ejemplo muestra el esfuerzo por mitigar el problema, más sin embargo todas estas políticas bancarias se centran en bancos emisores; pero para un banco en su rol adquirente no existen políticas bien definidas que ayuden a resolver el problema aún más de raíz. Bajo este escenario se tiene la motivación de ofrecer un modelo que sirva de apoyo, en una detección más ágil de puntos comprometido. Esto puede llevarse a cabo mediante un consolidado de la información de diversas instituciones financieras, generando con ello un modelo de detección de puntos comprometidos para la banca adquirente en México, cuya principal contribución es que a la fecha dicho modelo no se tiene.

1.3 Objetivos del trabajo de investigación

Objetivo: Plantear un modelo que permita a la banca en México, localizar de forma más ágil los sitios donde la información del tarjetahabiente fue comprometida, por medio de un análisis de información empleando métodos de Inteligencia Artificial. El modelo no contempla una implementación o producto final para los bancos o el switch transaccional [10]; solamente un estudio que demuestre que la solución planteada es computacionalmente instrumentable.

Hipótesis: Utilizando métodos de Inteligencia Artificial como arboles de decisión y procedimientos de clustering, es posible identificar históricamente puntos comunes de transacción (comercios), donde el patrón de dichos comercios corresponde a los sitios donde el tarjetahabiente utilizó su plástico antes de que se realizara el fraude y por ende, se le puede asignar un índice que permita identificar más fácilmente al banco aquellos comercios donde se comprometió la información del tarjetahabiente.

La metodología propuesta para desarrollar este trabajo es la siguiente:

1. Obtener los datos históricos de fraudes para algunas tarjetas.
2. Realizar un análisis de los datos para determinar el tipo de información con la que se está trabajando (generación de catálogos e interpretación de valores con el banco).
3. Realizar un análisis exploratorio de los datos por medio del uso de Arboles de decisión, que permita identificar los campos que aportan mayor información e ignorar para este análisis aquellos cuya aportación en cuanto a información sea bajo.
4. Aplicar algoritmos de clasificación que permita identificar patrones en la información.

5. Filtrar aquellos patrones de negocio identificados en base a reglas del negocio (Reuniones de trabajo con el banco).
6. Realizar un reporte de hallazgos (patrones, tendencias, etc.) para presentar las capacidades y beneficios del modelo a la Asociación de Banca en México (ABM).

2 Procesamiento y análisis de la información

2.1 Identificación de campos clave – Árboles de decisión

En la actualidad las bases de datos empresariales, manejan información estratégica del negocio, gerencial y operativa. Una cantidad importante de esta información no es del todo relevante para este tipo de análisis. Por ejemplo, las bases de datos bancarias manejan muchos campos para controlar la hora a la que llega una transacción, la hora en se compensa o las condiciones de entrega satisfactoria de la información para un banco emisor. Existen también campos para gestionar que la trama de datos sea consistente, etc. Todos estos campos no son realmente relevantes para el tipo de modelo planteado en este trabajo; más sin embargo, ¿Cómo determinar que campos son valor y cuáles pueden ser omitidos?

Los métodos para la detección de transacciones fraudulentas utilizados por diversos autores [3], se centran en realizar un estudio sobre los datos operativos; más sin embargo en la realidad los bancos efectúan análisis en nubes de datos más pequeños denominados Casos. Los casos se utilizan para monitorear segmentos del mercado que son considerados riesgosos y de posibles pérdidas. Posteriormente sobre estos subconjuntos selectos es posible aplicar algoritmos más robustos y que demandan menores recursos por parte de la organización.

Las herramientas de procesamiento y análisis de fraudes están basadas en algoritmos como redes bayesianas [4], modelos de Markov [5], maquinas de soporte vectorial, arboles de decisión entre otros. Más sin embargo, estos algoritmos son globales e implementados sobre una caja cerrada para el cliente final y con una baja personalización. Esto significa que los bancos requieren ingresar las particularidades de su negocio a la herramienta final, mediante reglas de negocio que definen precisamente el subconjunto de transacciones que serán tratadas con mayor detalle. En resumen, es necesario definir y optimizar los criterios para la generación de subconjuntos y por lo tanto, las islas generadas utilizadas por las herramientas de los bancos tendrán la mayor cantidad de información posible en un menor número de datos. Para este proceso se utilizan arboles de decisión con el objetivo de identificar de forma más precisa que campos pueden ser descartados del análisis, esta selección de campos se complementa con el conocimiento del experto en el negocio.

2.2 Identificación de patrones – Algoritmos de clasificación

El objetivo final de este proyecto, es realizar un proceso de análisis sobre las transacciones fraudulentas y la operación bancaria para encontrar patrones que permitan hacer más eficiente la toma de decisiones [6], en temas de prevención de fraude y generación de casos más precisos para las áreas de prevención.

En ese sentido se aplicaran diversos algoritmos, como K-Means, Support Vector Machine, etc. a los datos para intentar identificar patrones en el mar transaccional [7]. El conjunto de datos estará constituido de las M transacciones anteriores en un periodo de tiempo T, a partir de cada transacción fraudulenta. En este sentido, el universo consistirá en obtener las N transacciones reportadas como fraude de los bancos participantes en el proyecto y obtener las M transacciones anteriores, que claramente indican los comercios donde la persona ha utilizado su tarjeta y por tanto, son posibles candidatos de puntos compromiso.

A partir del primer conjunto de patrones identificados, se realizará un filtro que permita excluir comportamiento ya definido y conocido por los bancos; referente a geografía laboral, localización de terminales ATM/POS y hábitos identificados. La figura 1, ilustra el comportamiento de consumidores que pudieron sufrir fraude y tener un comportamiento transaccional histórico similar.

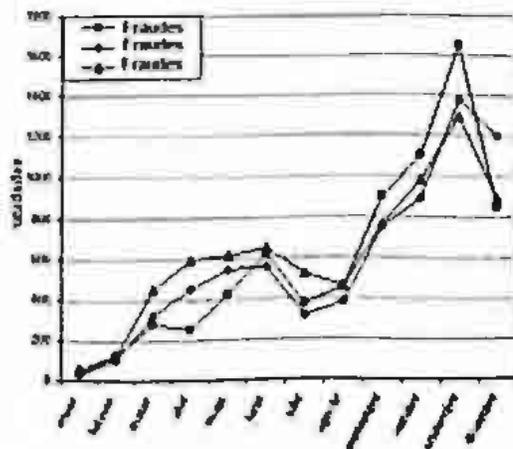


Figura 1. Comportamiento histórico de tarjetahabientes con Fraude

Como se ha mencionado este trabajo pretende ofrecer, primeramente un modelo que permita a los bancos identificar de forma más rápida y oportuna los sitios donde se comprometió la información del tarjetahabiente; analizando puramente transacciones fraudulentas y su respectiva historia. En segundo término, realizar un análisis de la operación completa del banco que apoye de forma directa al área de prevención de fraudes en la generación de reglas de negocio más precisa permitiendo con esto una reducción en tareas operativas para el área de prevención y de aclaraciones.

Para el desarrollo de este trabajo de investigación se tiene:

1. Información operacional de 2 años para un banco. Aproximadamente 10 millones de transacciones. Los datos ya pasaron por una fase de validación y limpieza tal que se garantiza:
 - a. Toda transacción tiene un número de cuenta, secuencia y números validos.
 - b. Sólo se consideran transacciones, cuyo tipo de mensaje sea con valor 2XX y 4XX. Debido a la definición del banco.
 - c. La explotación de información para el rastreo de transacciones, se realiza en un ambiente Data Warehouse; por lo que el switch transaccional [10] garantiza que sólo contiene transacciones válidas.
2. Contratos de confidencialidad y derechos con la entidad financiera.
3. Análisis de información de las fuentes de datos donde reside la información. Las fuentes son: base operacional del banco inicial y Data Warehouse del switch transaccional [10] en México.
4. Documentación de las fuentes de datos, generación de catálogos homologados y reglas de tratamiento – limpieza de la información.

Debido a la naturaleza del proyecto y sector empresarial, se tiene planeado realizar un seguimiento y cierre de actividades (asignación de responsabilidades, definición de objetivos, entrega, etc.), controlados por plan de trabajo realizado en conjunto con los bancos.

3 Trabajo futuro

Como trabajo futuro para lograr finalizar este proyecto, se requiere concluir con la validación de reglas de negocio aplicadas a la información, terminar la exclusión de campos de las fuentes de datos, automatizar los procesos para generar la historia de las M transacciones anteriores al fraude cometido, procesar la información resultante por medio de análisis de clustering y visualización geográfica, determinando con ello posibles puntos compromiso de la información.

Al final del proyecto se entregará un documento con los resultados del estudio, el cual será incluido en los procesos operativos de diversas áreas del banco, con el objetivo de mejorar los procesos de negocio actuales, lograr una reducción de casos alertados por posibles fraudes y en consecuencia una disminución en el trabajo operativo para el área de aclaraciones bancarias.

En resumen, para poder continuar con actividades y finalizar este proyecto se requieren resolver diversos puntos, como lo son:

1. Aplicar algoritmos de clasificación a los datos del banco, que permita identificar patrones en la información.
2. Identificar patrones de comportamiento relevantes para el banco en base a reglas del negocio. Para identificar estos patrones se manejarán diversas reuniones de trabajo con el banco.
3. Acceder a un listado de transacciones confirmadas con fraude. Esta lista contempla sólo dos campos: número de la transacción y el número de cuenta del tarjetahabiente. Se espera tener la información de diversas entidades financieras (4 bancos al final del proceso).
4. Realizar un reporte de hallazgos (patrones, tendencias, etc.) que presente las capacidades y beneficios del modelo. Este reporte se entregará a:
 - a. Asociación de Banca en México (ABM).
 - b. Switch transaccional [10] de México.

4 Conclusiones

La determinación precisa de las variables sobre las que las entidades financieras deben enfocarse para detectar un determinado tipo de comportamiento y, la implementación de un modelo que apoye al analista experto en la detección de puntos comprometidos, son dos de los objetivos críticos de este trabajo; esperando con ello realizar una mejora en las políticas de control fraudulento con los bancos. Las actividades mencionadas son precisamente las tareas en las que actualmente se labora, así como el inicio de búsqueda de patrones en la información y entablar acuerdos con los bancos para determinar las premisas a nivel de negocio que requiere el proyecto.

Es importante señalar que un impacto positivo en la reducción del fraude, puede ser posible gracias al apoyo de modelos que permitan bosquejar el comportamiento transaccional; pero más aún sólo puede llevarse a cabo si existe un cambio en los procedimientos de los bancos para el manejo de información y distribución pertinente de datos críticos entre las diversas entidades financieras.

5 Agradecimientos

Deseo expresar mis agradecimientos al cuerpo docente del ITESM Campus Estado de México - Dr. Miguel González Mendoza, por su apoyo en el comienzo de este proyecto. De la misma manera a las entidades financieras en México, ya que sin su valioso apoyo y facilidades al acceso de información no podría ser posible la realización de este trabajo. Y más aun a mi familia, amigos y colaboradores por su interés y apoyo en este camino que comienza.

6 Referencias

- [1] Información pública obtenida a partir de la Comisión Nacional de Banca y Valores, CNBV a Julio de 2011.
- [2] Información pública obtenida a partir de la Asociación de Bancos de México – ABM.
- [3] Credit Card Fraud Detection System. V. Filippov Institute of Control Sciences Moscow, Russia Email: filippoys@umail.ru L. Mukhanov Insitute of Electronic Controlling Machines Moscow, Russia Email: lmukhanov@fraudprevention.ru
- [4] A Comprehensive Survey of Data Mining-based Fraud Detection Research. School of Business Systems, Faculty of Information Technology, Monash University, Clayton campus, Wellington Road, Clayton, Victoria 3800, Australia.
- [5] Credit Card Fraud Detection Using Hidden Markov Model Abhinav Srivastava, Amlan Kundu, Shamik Sural, Senior Member, IEEE. and Arun K. Majumdar, Senior Member, IEEE.
- [6] Neural Fraud Detection in Credit Card Operations. José R. Dorronsoro, Francisco Ginel, Carmen Sánchez, and Carlos Santa Cruz.
- [7] Research on Credit Card Fraud Detection Model Based on Distance Sum. Wen-Fang YU Computer Science and Information Engineering College Zhejiang Gongshang University Hangzhou, China Ywf_1@163.com.
- [8] TORO OCAMPO, ELIANA M.; MEJÍA GIRALDO, DIEGO A.; SALAZAR ISAZA, HAROLD 'PRONÓSTICO DE VENTAS USANDO REDES NEURONALES' *Scientia Et Technica*, Vol. X, Núm. 26, diciembre-sin mes, 2004, pp. 25-30 Universidad Tecnológica de Pereira, Colombia
- [9] Credit Card BIN Numbers Database | Bank Identification Number <http://www.binbase.com/visa.html>
- [10] Switch Transaccional en México (PROSA – Eglobal) <http://www.banxico.org.mx/sistemas-de-pago/material-educativo/intermedio/%7BA7F8011B-7F77-F211-B48B-63DE4DE743D1%7D.pdf>
<http://www.banxico.org.mx/sistemas-de-pago/material-educativo/intermedio/%7B8BE6E2D1-95B2-4161-E81E-3EA205742B60%7D.pdf>

Para mayor referencia sobre el tema tratado se pueden consultar los siguientes enlaces, pertenecientes a organismos de alto impacto en el mercado transaccional de México.

1. <http://usa.visa.com>
2. <http://www.banxico.org.mx/>
3. <http://www.cnbv.gob.mx>
4. <http://www.prosa.com.mx>

Algoritmos Fuzzy C-Means robustos basados en un estimador-RML para segmentar imágenes ruidosas

Dante Mújica-Vargas¹, Francisco J. Gallegos-Funes²

Escuela Superior de Ingeniería Mecánica y Eléctrica
Instituto Politécnico Nacional de México.
Av. IPN s/n. Edificio Z, acceso 3, 3^{er} piso: SEPI-Electrónica, Col. Lindavista, 07738,
México D. F. México, Tel./Fax (5255)57296000 ext. 54622
¹adorinam@yahoo.com.mx, ²fcogf@hotmail.com

Resumen. La segmentación de imágenes es un paso crucial en diferentes aplicaciones de análisis de imágenes. Hasta ahora no existe un método general para segmentar apropiadamente todo tipo de imágenes considerando si están degradadas o libres de ruido. En este artículo, proponemos modificar el algoritmo de agrupamiento Fuzzy C-Means y su variante FCM_S1 usando un estimador-RML (combinación de estimadores-R, estimadores-M y estimadores-L). La idea de nuestro método es obtener algoritmos de agrupamiento capaces de segmentar imágenes con diferentes tipos y niveles de ruido. El desempeño de los algoritmos propuestos es probado con reales. Los resultados experimentales muestran que los algoritmos propuestos son más robustos ante la presencia de ruido y más efectivos que los algoritmos comparativos.

Palabras claves: estimador-RML, Fuzzy C-Means, FCM_S1, segmentación, ruido.

1 Introducción

La segmentación de imágenes es el proceso de separar una imagen en varias regiones disjuntas cuyas características tales como la intensidad, color, textura, etc., son significativamente diferentes con respecto a la misma característica. Es un paso importante en actividades de visión por computadora y ha sido ampliamente investigado en el campo del procesamiento de imágenes [1]. Numerosas técnicas de segmentación han sido desarrolladas y reportadas en la literatura. Pero, no existe un método general capaz de desarrollar a la perfección la tarea de segmentación para todo tipo de imágenes.

El agrupamiento difuso como un método de segmentación ha sido ampliamente estudiado y exitosamente aplicado a la segmentación de imágenes. Entre los métodos de agrupamiento difuso, el algoritmo Fuzzy C-Means (FCM) [2] la técnica más popular debido a su simplicidad, facilidad de programación y porque puede retener mucho mayor información que los métodos duros (K-Means). Aunque los métodos de agrupamiento difuso trabajan bien en la mayoría de imágenes libres de ruido, tienen una seria limitación cuando las imágenes están degradadas con ruido, debido a que no

incorporan ningún tipo de información acerca del contexto espacial, haciéndoles sensibles al ruido o a los datos atípicos. Entonces, es necesario modificar la función objetivo de estos algoritmos para incorporar información local de la imagen obteniendo así mejores resultados.

En este artículo, proponemos modificar los algoritmos de agrupamiento Fuzzy C-Means y su variante FCM_S1 [3], [4] usando un estimador-RML [5], [6], [7], [8]. El objetivo es obtener algoritmos para segmentar imágenes con diferentes condiciones de ruido. Los resultados experimentales muestran que los algoritmos propuestos son más efectivos y robustos que los algoritmos de referencia.

2 Método Propuesto

2.1 Estimador-RML

Como fue propuesto en [7] y [9], el estimador mediano tipo-M (MM) puede mejorar las propiedades de los estimadores M y R por si solos.

$$\theta_{medM} = med\{X_i \tilde{\psi}(X_i - med\{\vec{X}\}), i = 1, \dots, N\} \quad (1)$$

donde, X_i es la muestra de datos. \vec{X} es el vector de datos, y $\tilde{\psi}$ es la función de influencia. La robustez del filtro-L es mejorada por medio del estimador-RM (1). La representación del filtro-L es

$$\theta_L = \sum_{i=1}^N a_i X_{(i)} \quad \text{con} \quad a_i = \int_{i-1/N}^{i/N} h(\lambda) d\lambda / \int_0^1 h(\lambda) d\lambda \quad (2)$$

donde, $X_{(i)}$, $i=1, \dots, n$ es la i -th estadística de orden de la muestra de datos, a_i , $i = 1, \dots, N$ son los coeficientes de ponderización del filtro, y $h(\lambda)$ es la función de distribución de probabilidad del ruido $[0,1] \rightarrow R$ satisfaciendo la condición $\int_0^1 h(\lambda) d\lambda \neq 0$.

Para introducir el estimador-RM [9] en el esquema del filtro-L, la muestra de datos ordenada del filtro-L debe ser presentada como una función de una función de influencia [9]

$$\theta_L = \sum_{i=1}^N a_i \psi(X_i) X_i \quad (3)$$

$$\psi(u) = \begin{cases} c & |u| \leq r \\ 0 & \text{en cualquier otro caso} \end{cases}$$

donde, $N = (2L + 1)^2$ es el tamaño de la ventana de filtrado, $\psi(X_i) X_i$ es la muestra de datos ordenada, $\psi(u)$ es la función de influencia, c es una constante, y r está relacionada con el rango de $\psi(u)$. Entonces, el filtro-RML puede ser obtenido fusionando el filtro-L (3) y el estimador-RM (1)

$$\theta_{RML} = \frac{\text{med}\{a_i[X_i\psi(X_i - \text{med}\{\bar{X}\})]\}}{a_{\text{med}}} \quad (4)$$

donde, $X_i\psi(X_i - \text{med}\{\bar{X}\})$ son los pixeles seleccionados en concordancia con la función de influencia en la ventana deslizante de filtrado, los coeficientes a_i son calculados usando las funciones de distribución Laplaciana, Uniforme y Exponencial en $h(\lambda)$, y a_{med} es la mediana de los coeficientes a_i usada como una constante de escalamiento. La tabla 1 muestra la función de influencia usada en este artículo.

Tabla 1. Función de influencia usada en el estimador-RML.

Función de Influencia	Formula
Hampel's three part redescending	$\psi_{\alpha,\beta(r)}(x) = \begin{cases} x & 0 \leq x < \alpha \\ \alpha \text{sgn}(x) & \alpha \leq x < \beta \\ \alpha \frac{r- x }{r-\beta} \text{sgn}(x) & \beta \leq x < r \\ 0 & r \leq x \end{cases}$

Para mejorar las propiedades de supresión de ruido impulsivo del estimador-RML un detector de impulsos es usado

$$[(\text{rango}(x_{ij}) \leq s) \vee (\text{rango}(x_{ij}) \geq N - s)] \wedge (|x_{ij} - \text{med}\{x_n\}| \geq U_2) \quad (5)$$

donde, x_{ij} es el pixel central en la ventana de filtrado, $s > 0$ y $U_2 \geq 0$ son umbrales, N es la longitud de la muestra de datos y $\text{med}\{x_n\}$ es la mediana de los pixeles en la ventana de filtrado.

2.2 Algoritmos robustos de agrupamiento difuso

Fuzzy C-Means es un método para clasificación de datos, donde cada dato pertenece a una clase en algún grado, lo cual es especificado por un valor de pertenencia. Este algoritmo desarrolla la iteración de dos condiciones indispensables para minimizar la siguiente función objetivo

$$J_f(U, V, X) = \sum_{i=1}^N \sum_{j=1}^c u_{ij}^m \|x_i - v_j\|^2 \quad (6)$$

sujeto a $\sum_{i=1}^N u_{ij} > 0 \quad \forall_j$ y $\sum_{j=1}^c u_{ij} = 1 \quad \forall_i$

donde, $X = \{x_i | i = 1, \dots, N\}$ es el conjunto de N vectores de características, c es el número de clases, $m \in [1, \infty)$ es un exponente de ponderización llamado fuzzificador, $\|x_i - v_j\|^2$ es el cuadrado de la distancia Euclidiana, tomada desde el vector de características x_i al centro de la clase v_j y $V = (v_1, \dots, v_j)$ es un vector con todos los centros de las clases. $U = [u_{ij}]$ es una matriz de $N \times c$ denotando la partición c-difusa

condicionada. El valor de u_{ij} es el grado de pertenencia de x_i a la clase v_j . Para cubrir la sensibilidad al ruido del algoritmo Fuzzy C-Means, un estimador-RML es aplicado sobre el vector de características (intensidad del pixel) x_i . Basados en el algoritmo Fuzzy C-Means estándar (6) y el estimador-RML (4) la función objetivo del nuevo algoritmo puede ser escrita como

$$J_{rf}(U, V, \Theta) = \sum_{i=1}^N \sum_{j=1}^c u_{ij}^m \|\theta_{RMLi} - v_j\|^2 \quad (7)$$

sujeto a $\sum_{i=1}^N u_{ij} > 0 \forall_j$ y $\sum_{j=1}^c u_{ij} = 1 \forall_i$

donde, $\Theta = \{\theta_{RMLi} | i = 1, \dots, N\}$ es el estimador-RML aplicado sobre el vector de características. Tomando en cuenta ambas condiciones, la matriz de pertenencia y los centros de las clases pueden ser calculados con las siguientes expresiones

$$v_j = \frac{\sum_{i=1}^N u_{ij}^m \theta_{RMLi}}{\sum_{i=1}^N u_{ij}^m} \quad (8)$$

$$u_{ij} = \frac{\|\theta_{RMLi} - v_j\|^{-\frac{2}{m-1}}}{\sum_{i=1}^c \|\theta_{RMLi} - v_i\|^{-\frac{2}{m-1}}} \quad (9)$$

Una deficiencia del algoritmo FCM_S1 [3], [4] propuesto por Chen y Zhang (10), es que el efecto de la imagen ruidosa puede ser mayor que las imágenes filtradas mediante la media y la mediana, produciendo una segmentación inapropiada.

$$J(U, V, X) = \sum_{i=1}^N \sum_{j=1}^c u_{ij}^m \|x_i - v_j\|^2 + \alpha_1 \sum_{i=1}^N \sum_{j=1}^c u_{ij}^m \|\bar{x}_i - v_j\|^2 + \alpha_2 \sum_{i=1}^N \sum_{j=1}^c u_{ij}^m \|\tilde{x}_i - v_j\|^2 \quad (10)$$

sujeto a $\sum_{i=1}^N u_{ij} > 0 \forall_j$ y $\sum_{j=1}^c u_{ij} = 1 \forall_i$

donde, \bar{x}_i y \tilde{x}_i son la media y la mediana en la vecindad de los pixeles alrededor de x_i , respectivamente. Los parámetros α_1 y α_2 controlan el efecto de la media y la mediana de la vecindad de pixeles. Por lo tanto proponemos modificar la función objetivo de éste algoritmo cambiando los términos x_i , \bar{x}_i y \tilde{x}_i por los términos $\theta_{RML} - Uniforme$, $\theta_{RML} - Exponencial$, y $\theta_{RML} - Laplaciana$ para obtener una segmentación más robusta.

$$J_{raf}(U, V, \Theta) = \sum_{i=1}^N \sum_{j=1}^c u_{ij}^m \|\theta_{RML-Ui} - v_j\|^2 + \alpha_1 \sum_{i=1}^N \sum_{j=1}^c u_{ij}^m \|\theta_{RML-Ei} - v_j\|^2 \quad (11)$$

$$+ \alpha_2 \sum_{i=1}^N \sum_{j=1}^c u_{ij}^m \|\theta_{RML-Li} - v_j\|^2$$

$$\text{sujeto a } \sum_{i=1}^N u_{ij} > 0 \quad \forall_j \text{ y } \sum_{j=1}^c u_{ij} = 1 \quad \forall_i$$

donde, θ_{RML-U} , θ_{RML-E} y θ_{RML-L} son los estimadores- θ_{RML} (4) calculados usando las distribuciones Uniforme, Exponencial y Laplaciana respectivamente, α_1 y α_2 son los parámetros que controlan el efecto de los términos, θ_{RML-E} y θ_{RML-L} . Minimizando la expresión (11) la matriz de pertenencia y los centros de las clases pueden ser obtenidos a partir de las siguientes expresiones

$$v_j = \frac{\sum_{i=1}^N u_{ij}^m [\theta_{RML-U} + \alpha_1 \theta_{RML-E} + \alpha_2 \theta_{RML-L}]}{(1 + \alpha_1 + \alpha_2) \sum_{i=1}^N u_{ij}^m} \quad (12)$$

$$u_{ij} = \frac{\left[\|\theta_{RML-U} - v_j\|^2 + \alpha_1 \|\theta_{RML-E} - v_j\|^2 + \alpha_2 \|\theta_{RML-L} - v_j\|^2 \right]^{-\frac{1}{m-1}}}{\sum_{i=1}^c \left[\|\theta_{RML-U} - v_i\|^2 + \alpha_1 \|\theta_{RML-E} - v_i\|^2 + \alpha_2 \|\theta_{RML-L} - v_i\|^2 \right]^{-\frac{1}{m-1}}} \quad (13)$$

3 Resultados Experimentales.

3.1 Medidas de desempeño

Para medir objetivamente la calidad de las segmentaciones producidas, en este artículo se consideran 4 índices de evaluación [10]. El primero es el *Probabilistic Rand Index* (PRI), este índice compara los resultados obtenidos mediante los algoritmos empleados y un conjunto de imágenes segmentadas manualmente. Dado que no hay una salida única que sea totalmente correcta, considerando múltiples resultados se mejora la comparación y se toma en cuenta la variabilidad de la percepción humana. El PRI está basado en la ponderización no uniforme de un par de píxeles como una función de la variabilidad en el *ground-truth*. El conjunto del *ground-truth* está definido como $\{G_1, G_2, \dots, G_L\}$ donde L es el número de imágenes segmentadas manualmente. Dado S ser la segmentación provista por el algoritmo empleado, $l_i^{G_k}$ la etiqueta del pixel x_i en la k -th imagen segmentada manualmente y l_i^S la etiqueta del pixel x_i en el algoritmo empleado. Entonces, el PRI está definido por

$$PR(S, G_k) = \frac{2}{N(N-1)} \sum_{i,j,l < j} (p_{ij}^{c_{ij}} (1 - p_{ij})^{1-c_{ij}}) \quad (14)$$

donde N es el número de píxeles, c_{ij} es una función booleana denotando si l_i^S es igual a $l_i^{G_k}$ y p_{ij} es el valor esperado de una distribución de Bernoulli para el par de píxeles. El valor de PRI está en el rango de $[0, 1]$, donde valores altos indican una gran similitud entre las imágenes segmentadas por los algoritmos analizados y el *ground-truth*. El segundo índice es la métrica *Variation of Information* (VOI). Este parámetro mide la suma de ganancia y pérdida de información entre dos agrupamientos pertenecientes a la red de posibles particiones. Está definido por

$$VOI(S, G_k) = H(S) + H(G_k) - 2I(S, G_k) \quad (15)$$

donde H es la entropía $-\sum_{i=1}^c \frac{n_i}{n} \log \frac{n_i}{n}$, n_i es el número de puntos pertenecientes a la i -th clase. El término I es la información mutua entre dos clases, y está definido por

$$I(S, G_k) = \sum_{i=1}^c \sum_{j=1}^c \frac{n_{i,j}}{n} \log \frac{n_{i,j}}{n} \frac{n_i}{n} \frac{n_j}{n} \quad (16)$$

donde $n_{i,j}$ es el número de puntos en la intersección de la clase i de S y j de G_k . La medida VOI es una distancia, por lo tanto el valor más pequeño de éste significa que la segmentación obtenida es la más cercana a la del ground-truth. El tercer índice es el *Global Consistency Error* (GCE) el cual evalúa en qué medida la segmentación puede ser vista como el perfeccionamiento de los otros. Una medida de error en cada pixel x_i se define por

$$E(S, G_k, x_i) = \frac{|R(S, x_i) \setminus R(G_k, x_i)|}{|R(S, x_i)|} \quad (17)$$

donde $|\cdot|$ es la cardinalidad, \setminus es la diferencia de conjuntos, y $R(S, x_i)$ es el conjunto de pixeles correspondientes a la región en la segmentación de S que contiene el pixel x_i . La medida GCE obliga a todos los refinamientos locales a estar en la misma dirección, es entonces definida por

$$GCE(S, G_k, x_i) = \frac{1}{n} \min \left(\sum_{i=1}^n E(S, G_k, x_i), \sum_{i=1}^n E(G_k, S, x_i) \right) \quad (18)$$

El valor de GCE más cercano a cero representa una mejor segmentación de S con respecto al ground-truth G_k . El último índice es el *Boundary Displacement Error* (BDE), el cual evalúa el error de desplazamiento promedio de pixeles en los bordes entre dos imágenes segmentadas a través del cálculo de la distancia entre el pixel y el pixel más cercano en la otra segmentación. Dado un punto arbitrario x_i de S , el índice BDE usa la distancia Euclídea desde x_i a todos los puntos de G_k . Una distribución de la distancia $D_S^{G_k}$ es obtenida sumando la distancia de todos los puntos de S . El índice BDE es finalmente calculado mediante la siguiente expresión

$$BDE(S, G_k) = \frac{1}{2} (D_S^{G_k} + D_{G_k}^S) \quad (19)$$

3.1 Evaluación con el Berkeley Image Segmentation Benchmark

El desempeño de los métodos propuestos es evaluado usando un subconjunto de imágenes tomadas del *Berkeley Image Segmentation BSD500*. Éste provee un conjunto de imágenes reales junto con sus mapas de segmentación proporcionada por diferentes individuos. Las imágenes fueron degradadas con diferentes niveles de ruidos Gaussiano, Impulsivo y una mezcla de ambos. Los resultados obtenidos son

comparados con los algoritmos de referencia FCM_S1, NCut, y NL_SSC [3], [4], [10], [11]. Los parámetros utilizados en nuestros algoritmos fueron establecidos como: $c=2$ (en la imagen 42049) y $c=3$ (resto de imágenes), $m=2$, $\epsilon=1e-6$ en la etapa de agrupamiento (inicializada aleatoriamente); $r=5$, $\alpha=0.16r$, $\beta=0.8r$ en la estimación-RML, $s=4$ y $U_2=5$ en la detección impulsiva, $\alpha_1=0.8$ y $\alpha_2=0.4$ para el algoritmo RDFCM. La Fig. 1 muestra las imágenes originales a ser segmentadas.

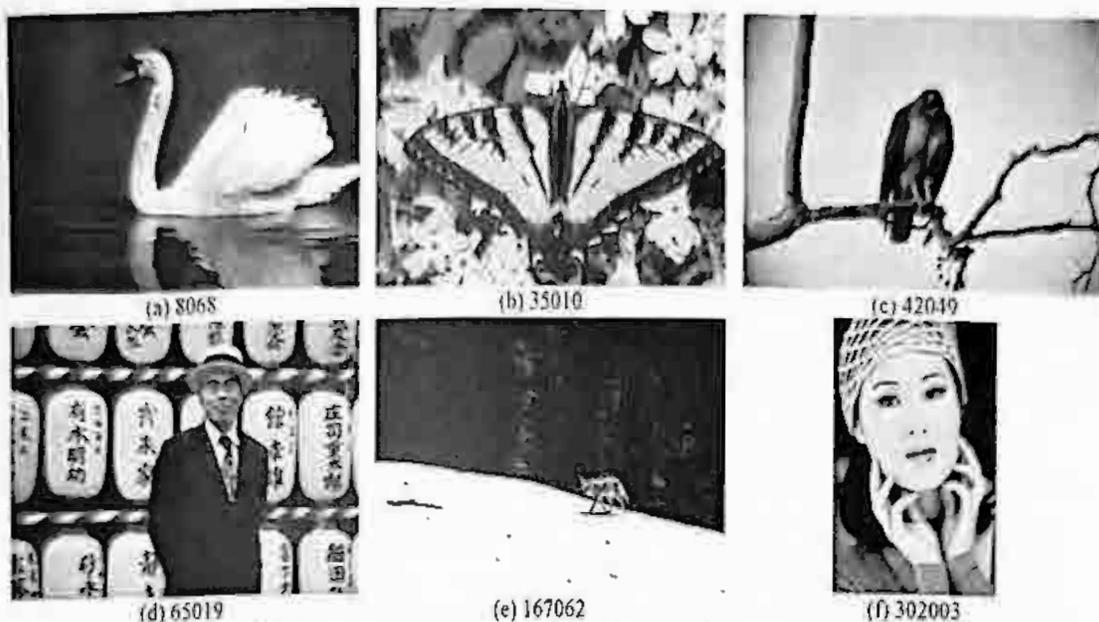


Fig. 1. Imágenes en escala de grises para segmentar.

La tabla 1 resume los índices promedio del desempeño calculados para los algoritmos y diferentes condiciones de ruido.

Tabla 1. Medidas promedio del desempeño en las imágenes utilizadas.

Algoritmo	Índice	Gaussiano		Sal y Pimienta		Mezcla	
		$\sigma=10$	$\sigma=20$	10%	15%	$\sigma=10 + 10\%$	$\sigma=20 + 15\%$
FCM_S1	PRI	0.73	0.71	0.71	0.70	0.71	0.70
	VOI	2.53	2.62	2.79	2.90	2.79	2.91
	GCE	0.19	0.20	0.24	0.27	0.25	0.27
	BDE	7.85	11.14	14.50	15.03	14.52	15.03
NCut	PRI	0.73	0.70	0.73	0.73	0.73	0.70
	VOI	2.53	2.64	2.53	2.53	2.53	2.66
	GCE	0.19	0.20	0.19	0.19	0.19	0.20
	BDE	7.46	11.58	7.38	7.58	7.92	12.03
NL_SSC	PRI	0.73	0.71	0.73	0.73	0.73	0.71
	VOI	2.52	2.61	2.53	2.53	2.53	2.64
	GCE	0.19	0.20	0.19	0.19	0.19	0.20
	BDE	7.53	10.87	7.55	7.69	7.80	11.45
RDFCM	PRI	0.83	0.81	0.80	0.79	0.80	0.78
	VOI	1.85	1.99	1.98	1.99	1.98	2.03
	GCE	0.15	0.16	0.16	0.17	0.16	0.17
	BDE	4.83	4.88	4.86	4.94	4.94	5.31
RDFCM	PRI	0.82	0.80	0.81	0.79	0.80	0.78
	VOI	1.98	1.99	1.98	1.99	1.98	2.00
	GCE	0.16	0.17	0.16	0.18	0.17	0.17
	BDE	4.77	4.85	4.82	5.01	4.80	5.37

Los resultados visuales de algunas de las imágenes segmentadas son mostrados en Fig. 2, Fig. 3, Fig. 4 y Fig. 5

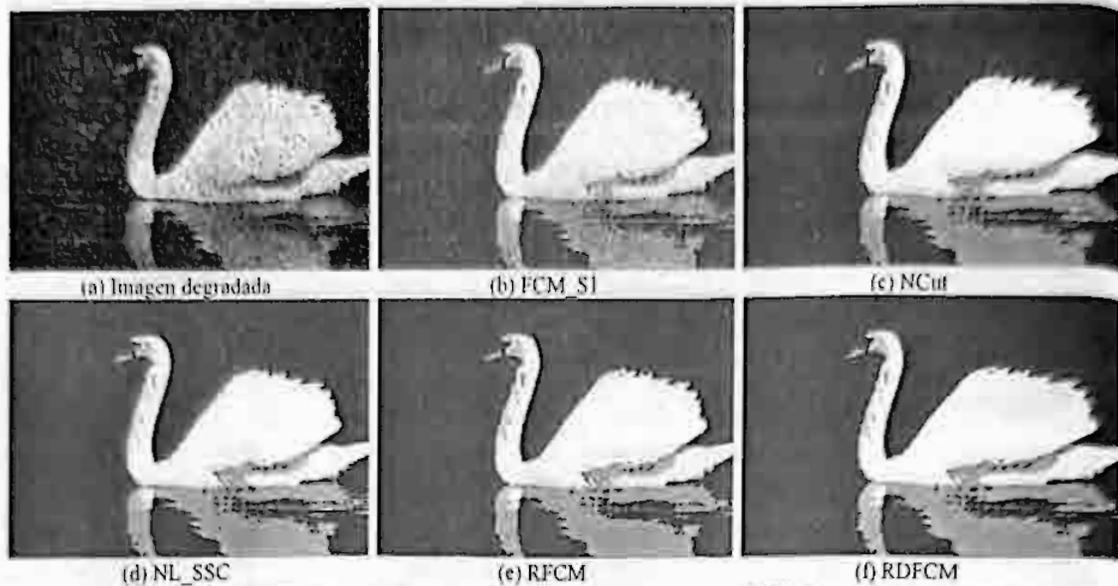


Fig. 2. Imagen 8068 degradada con $\sigma=20$ de ruido gaussiano y 15% de ruido impulsivo.

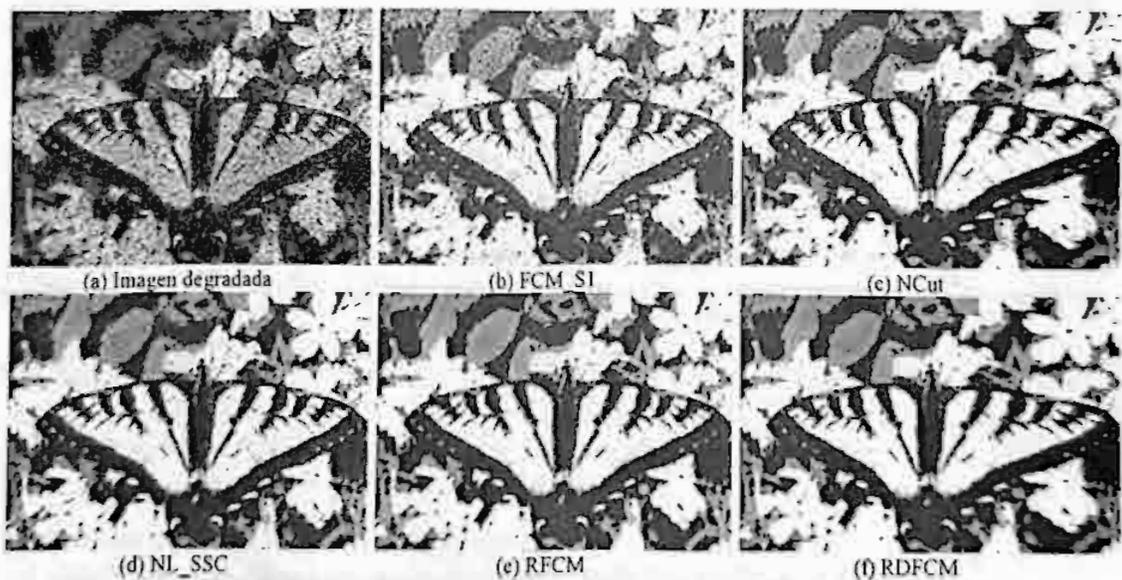
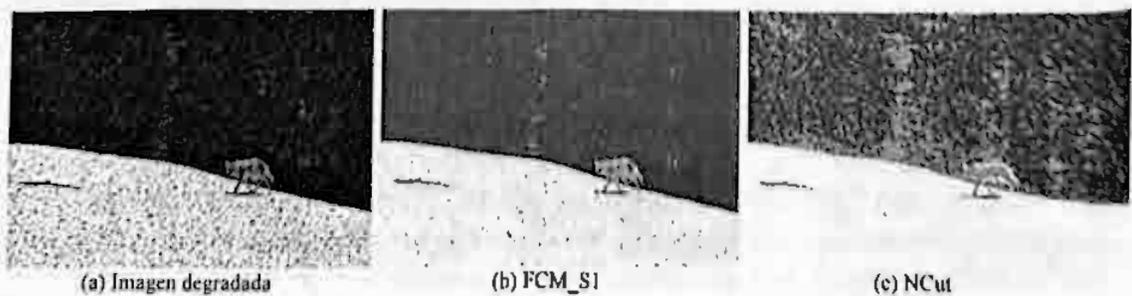


Fig. 3. Imagen 35010 degradada con $\sigma=20$ de ruido gaussiano y 15% de ruido impulsivo.





(d) NL_SSC

(e) RFCM

(f) RDFCM

Fig. 4. Imagen 167062 degradada con $\sigma=20$ de ruido gaussiano y 15% de ruido impulsivo.



(a) Imagen degradada



(b) FCM_S1



(c) NCut



(d) NL_SSC



(e) RFCM



(f) RDFCM

Fig. 5. Imagen 302003 degradada con $\sigma=20$ de ruido gaussiano y 15% de ruido impulsivo.

4 Discusión de resultados.

Los resultados experimentales mostraron que los algoritmos propuestos obtuvieron mejores resultados que los algoritmos comparativos. Para los métodos propuestos el índice PRI tienen los valores más altos y alrededor de 0.8, en cambio los demás índices tienen valores mucho menores que los obtenidos por los algoritmos comparativos. También se observó que la incorporación del estimador-RML sobre el algoritmo FCM_S1 le dio mayor capacidad para segmentar imágenes ruidosas en comparación de su versión original.

5 Conclusiones

Este artículo presentó un método para incorporar el estimador-RML dentro del algoritmo estándar Fuzzy C-Means y su variante FCM_S1, dándoles mayor robustez para segmentar imágenes ruidosas. El desempeño de los métodos propuestos fue mejor que el de los algoritmos de referencia. Sin embargo, para darles mayor capacidad para trabajar con imágenes a color así como otros tipos y niveles de ruido deben ser modificados y optimizados.

Agradecimientos

Los autores dan las gracias al Instituto Politécnico Nacional de México y al Conacyt por los apoyos brindados.

Referencias

1. F. Yang, T. Jiang., Pixon-based Image Segmentation with Markov random fields. National Laboratory of Pattern Recognition, Institute of Automation, Chinese Academy of Sciences, Beijing, China. (2002)
2. Bezdek J.C., Pattern Recognition with Fuzzy Objective Function Algorithms. Plenum Press. New York (1981)
3. Z. Wang, R. Lu., A New Algorithm for Image Segmentation Based on Fast Fuzzy C-Means Clustering. School of Electronics and Information Engineering, Southwest University, Chongqing, China. (2008)
4. W. L. Cai, S. C. Chen, D. Q. Zhang., Fast and robust fuzzy c-means clustering algorithms incorporating local information for image segmentation. Department of Computer Science & Engineering. Nanjing University of Aeronautics & Astronautics. Nanjing, China. (2007)
5. F. Gallegos-Funes, Jose Varela-Benitez, V. Ponomaryov, Rank M-Type L (RM L)-Filter for Image Denoising, IEICE Trans. Fundamentals of Electronics, Communications and Computer Sciences, E91-A(12), 3817-3819, (2008)
6. F. J. Gallegos-Funes, V. Ponomaryov. Real-time image filtering scheme based on robust estimators in presence de noise impulsive. Real Time Imaging, 8(2), 78-90, (2004)
7. F. J. Gallegos-Funes, J. L. Varela-Benitez, V. Ponomaryov. Real-time image processing based on robust linear combinations of order statistics. Proc. SPIE Real-Time Image Processing 2006, Vol. 6063 177-187, (2006)
8. J. L. Varela-Benitez, F. J. Gallegos-Funes, V. Ponomaryov. RML-filters for real time imaging. Proc. IEEE 15th International Conference on Computing, CIC 2006, 43-48, (2006)
9. Gallegos-Funes, V. Ponomaryov, and J. De-La-Rosa. ABST M-type K-nearest neighbor (ABSTM-KNN) for image denoising, IEICE Trans. Funds. Electronics Comms. Computer Science, vol. E88A, no. 3, pp. 798-799, (2005)
10. H. Le Capitaine, C. Frélicot. A fast fuzzy c-means algorithm for color image segmentation. Laboratoire Mathématiques, Image Applications, Université de la Rochelle. France. (2011)
11. H.Q. Liu., L.C. Jiao., F. Zhao. Non-local spatial spectral clustering for image segmentation. Elsevier. Neurocomputing 74, 461-471, (2010)
12. H. Q. Liu., L.C. Jiao., F. Zhao. A novel fuzzy clustering with non-local adaptive spatial constraint for image segmentation. Elsevier. Signal Processing 91, 988-999, (2011)

Sistema de agentes para control y protección de políticas de seguridad aplicadas a equipos de cómputo con ambiente Windows-Tecnología NT

Pedro Rojas¹, Mario Medina¹, Guadalupe Cota¹,

¹Departamento de Matemáticas - Ciencias de la Computación,

Universidad de Sonora,

Rosales s/n, Col. Centro, Hermosillo, México

pedro.rojaslo50@gmail.com, maario.medina@hotmail.com, lcota@gauss.mat.uson.mx

Resumen. La problemática 'Zero Day' es el período en el cual surge 'malware' que no se detecta por software utilizado como herramienta de protección, pero también involucra el surgimiento de técnicas de 'Ingeniería Social' que se utilizan para vulnerar recursos de un sistema de cómputo, práctica que se lleva a cabo excediendo privilegios o violentando los de otros al acceder a información o recursos del sistema en forma no autorizada. Para coadyuvar en la búsqueda de alternativas de solución a esta problemática, en este trabajo presentamos una propuesta basada en tecnología de agentes para la plataforma Windows-Tecnología NT, ya que es reportada como una de las más inseguras de su tipo, que consiste en aplicar un modelo de confianza para conceder o denegar permisos en base a la detección del uso y abuso de privilegios.

Palabras clave: Seguridad Computacional. Políticas. Agentes.

Introducción

En materia de seguridad computacional se ha venido presentando en tiempo real el problema conocido como 'Zero day', que se identifica como el período en el cual surgen amenazas que no pueden controlarse eficientemente con herramientas de protección como software antivirus, sistemas detectores de intrusos, etc., dado que estas no son capaces de reconocer patrones de comportamientos anormales que no estén registrados previamente en su base de conocimientos.

En el tema de seguridad computacional, algunos autores sugieren iniciar con la definición de los requisitos básicos que permitan la identificación de posibles amenazas, es decir, analizar qué significa el llamar a un sistema de cómputo 'seguro' [4]. A este respecto en [1], se menciona que en general, un sistema seguro debe ser capaz de controlar a través del uso de características específicas de seguridad el acceso a la información, de tal forma que solamente los individuos o procesos autorizados con suficientes privilegios tengan acceso a leer, escribir, crear, modificar o eliminar información, logrando así un control de acceso seguro y un sistema informático confiable.

Tomando en cuenta que el tema de seguridad computacional abarca una gran variedad de áreas de aplicación, en este artículo nos enfocaremos al diseño de un modelo para protección de políticas en equipos de cómputo con ambiente Windows-NT, utilizando como soporte una plataforma de agentes y la aplicación de un modelo que permite tener el control sobre los perfiles de usuarios aplicando un factor de confianza para incrementar o disminuir los privilegios de usuarios del sistema,

Cabe aclarar que este proyecto se encuentra en etapa de desarrollo y se sustenta en la arquitectura ARSEC-AMS (Security Architect- Multiagent System) [2], la cual permite diseñar esquemas de comunicación y razonamiento de agentes, establecer criterios de seguridad y programar agentes reactivos.

2 Problemática

Hoy en día las políticas de seguridad son un requisito fundamental en cualquier organización, las cuales se aplican para buscar un óptimo funcionamiento y desempeño de los integrantes de las mismas. Sin embargo, existen casos donde las políticas impuestas no son lo suficientemente buenas y tienden a fallar o ser inútiles. Aunado a esto, hay que tomar en consideración que el imponer demasiadas restricciones en un sistema, puede provocar que los usuarios no realicen su trabajo cómoda y eficientemente, ya sea porque no se les proporcionan las herramientas necesarias para su labor, o simplemente por el hecho de llegar a un grado de desconfianza que propicia que las relaciones laborales en una organización se vuelvan hostiles.

Por otra parte, algunos problemas que hacen que las políticas de seguridad no surtan el efecto adecuado pueden ser [3]:

- Por desconocimiento de la necesidad de implementar una política.
- Por no tener un sistema de mantenimiento de políticas.
- Por tener políticas demasiado complejas.

El no existir políticas implementadas en el sistema y que están sean requeridas es un problema serio para una organización, ya que afecta los objetivos de la misma y permite que usuarios no autorizados realicen acciones que no les competen.

Muchos organismos cometen el error de confiar en sus empleados, y por ello imponen restricciones a sus sistemas de cómputo en forma verbal, lo cual permite que el empleado tome la decisión de obedecerlas o de ignorarlas y acceder a archivos que contienen información que no son de su incumbencia, o peor aún, implementar algún tipo de acción perjudicial para la empresa que podría culminar con la pérdida de información confidencial. Es por ello es necesario analizar constantemente cuales son las necesidades actuales de la organización e imponer las políticas pertinentes a través de reglamentos que sean conocidos por todos los involucrados y detectar a tiempo posibles irregularidades.

Por otro lado, hay que considerar que también existen organismos que tienden a darle poca importancia al aspecto de seguridad en sus sistemas de cómputo, lo que ocasiona que se desconozca la necesidad de implementar políticas para mejorar el desempeño de una organización, y esto conlleva a generar problemas y a tener falsas suposiciones como las que a continuación se mencionan:

- “En mi organización no existen amenazas”

- “Con un antivirus basta para protegerme”
- “Un firewall nos da toda la seguridad para protegernos de ataques”

Las anteriores suposiciones pueden generar problemas debido a que se deja el equipo vulnerable a la recepción de ataques informáticos de cualquier tipo (como ataques ‘Zero Day’ por ejemplo). Es por ello que consideramos que suponer que una organización no tiene amenazas es un grave error, dado que no se tiene garantía alguna de que tal apreciación sea real. De igual forma, el hecho de utilizar software antivirus o de tipo ‘firewall’ no es suficiente para suponer que se está totalmente protegido, ya que día a día surgen nuevas amenazas y no se puede tener una confianza plena en que éstas serán detectadas en forma total por herramientas de protección como las antes mencionadas.

Otros de los problemas que se presentan en la ejecución de políticas de seguridad, cuando éstas ya están implementadas, es la falta de mantenimiento de las mismas porque no se revisan periódicamente, lo que ocasiona que con el paso del tiempo, las políticas tiendan a no ajustarse a las necesidades de seguridad de la organización. En este aspecto, cabe mencionar que es recomendable sostener un plan de monitoreo constante sobre las políticas existentes, y ajustarlas al comportamiento de los individuos cuando sea necesario.

Tomando como base lo antes expuesto, nuestra propuesta va enfocada al desarrollo de políticas fáciles de implementar y de comprender, y que las mismas sean monitoreadas constante y automáticamente por un grupo de agentes inteligentes con el fin de mantener el control y protección en tiempo real del equipo de cómputo.

3 Políticas

Los sistemas de cómputo con ambiente Windows-Tecnología NT, permanentemente consultan a un registro que contiene información relevante para su operación y funcionamiento, por lo que una mala configuración de éste causa errores y fallos inesperados en el sistema. Por lo tanto, el registro es el núcleo de los sistemas de políticas y requieren de un monitoreo continuo para detectar intrusos o acciones no autorizadas por los usuarios.

3.1 Definición

Existen en la literatura varias definiciones para el término “política”, entre los cuales podemos destacar las siguientes:

- “Una política es una declaración general de principios que presenta la posición de la administración para un área de control definida” [3].
- “Una política de seguridad es un conjunto de reglas y estrategias que definen la manera en que una organización maneja, administra, protege, y asigna recursos para alcanzar el nivel de seguridad definido como objetivo” [4].

Tomando en cuenta lo anterior, podemos considerar que las políticas se crean con el propósito de que surtan aplicación en un cierto plazo, y controlen el desarrollo de reglas y criterios concretos que afronten situaciones específicas. Las políticas son abordadas por estándares, procedimientos y guías de gestión.

3.2 Políticas de seguridad

Existen diferentes tipos de políticas y cada una de ellas sirve para propósitos específicos [4]:

- Seguridad informática.
- Identificación y autenticación.
- Propiedad de la información.
- Control de acceso y autorizaciones.
- Confidencialidad e integridad de la información.
- Detección de incidentes y respuestas.
- Administración de seguridad.
- Aceptación de riesgos.

Todos los puntos anteriores cumplen metas en conjunto para lograr una buena administración de los recursos del sistema y tener un control sobre usuarios o aplicaciones [4].

3.3 Desarrollo de políticas de seguridad

Cada sistema de software debe tener un ciclo de vida para definir los plazos para el desarrollo, implementación, mantenimiento y supresión de políticas que dejan de ser útiles. Un esquema general para el ciclo vital de las políticas de seguridad se muestra en la Figura 1, donde se pueden detectar 11 etapas de desarrollo agrupadas en cuatro fases [10].

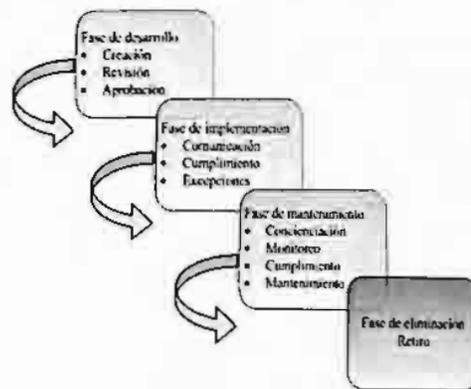


Fig. 1 Ciclo de vida de una política de seguridad

El procedimiento aplicado para el ciclo de vida, como se muestra en la Figura 1 es el siguiente:

1. Fase de desarrollo: La política es creada y aprobada.
2. Fase de Implementación: Se pone en marcha la política para revisar si se generan excepciones.
3. Fase de mantenimiento: Se comienza con la aplicación de la política sobre los usuarios, además los administradores deben garantizar su cumplimiento y mantener constante actualización.
4. Fase de eliminación: La política ya no es necesaria y se desecha.

Cada una de las etapas anteriores define en general el comportamiento ideal para

el ciclo de vida de las políticas de seguridad [3], [10]. Por tal razón, para resolver los problemas que surgen con frecuencia en las organizaciones se requiere del diseño de un conjunto de políticas y estrategias, aplicando las fases y conceptos antes mencionados.

4 Metodología aplicada

Nuestro modelo se sustenta en el desarrollo de un sistema basado en tecnología de agentes y la arquitectura ARSEC-AMS (ver Figura 2)[2], la cual se estructura en módulos que permiten aplicar conceptos de seguridad.

La arquitectura ARSEC-AMS es organizada en los siguientes módulos: el intérprete que se encarga de filtrar los mensajes entrantes y salientes; el módulo pizarra que se utiliza como elemento de comunicación del sistema de agentes; el módulo deliberativo-cognitivo que permite implementar la representación del conocimiento; el módulo de seguridad donde se pueden establecer criterios y niveles de seguridad, registro de alertas, definición de instrucciones, etc.; y el módulo reactivo a través del cual se pueden implementar acciones que respondan a cierto tipo de eventos, las cuales son previamente definidas por el administrador o el diseñador del sistema.

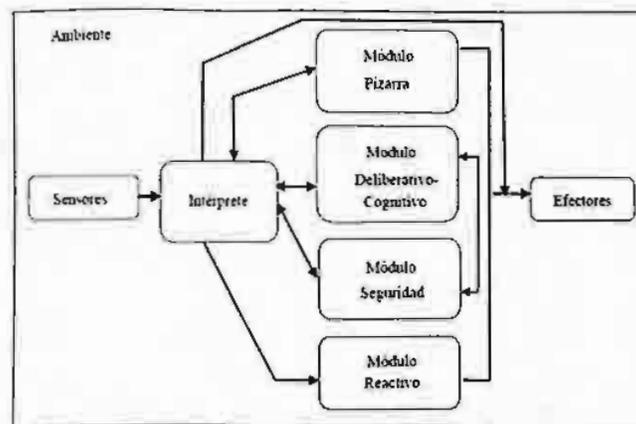


Fig. 2. Arquitectura ARSEC-AMS.

El modelo está basado en la jerarquía de agentes implementada en la arquitectura ARSEC-AMS [2], que se describe a continuación:

- **Agente Coordinador:** “Desempeña una actividad especializada, destinado a coordinar las operaciones que permitan aplicar los niveles de seguridad en el sistema, y coordinar el trabajo de los agentes local y de red [2].”
- **Agente Local:** Es asignado a las actividades primarias, como monitorear los mensajes enviados y recibidos, revisar el comportamiento de los usuarios a nivel local en el sistema y aplicar instrucciones giradas por agentes de tipo Coordinador [2].
- **Agente de red:** Su función es la de monitorear eventos relacionados con la información que se transmite a través paquetes de red [2].

4.1 Esquema funcional y elementos de comunicación del sistema de agentes

Considerando que el buen desempeño de un sistema de seguridad, debe tener una forma de registrar los eventos que ocurran en su ambiente, los cuales se tomarán como base para el procedimiento de prueba y razonamiento en la toma de decisiones, en este trabajo se propone utilizar una "pizarra" como elemento básico del esquema funcional del sistema de agentes.

La arquitectura de pizarra es un medio de comunicación en un sistema de agentes, una fuente de conocimiento y un mecanismo de control [5]. En ARSEC-AMS el módulo de pizarra BLACKBOARD-ECRE (Blackboard – Elemento de control para el registro de Eventos) [2] que se muestra en la Figura 4, se incluye como un elemento de registro de la información generada en tiempo real, que es una fuente de conocimiento que se utiliza en el proceso de análisis y la identificación de patrones que pueden ser considerados como posibles ataques.

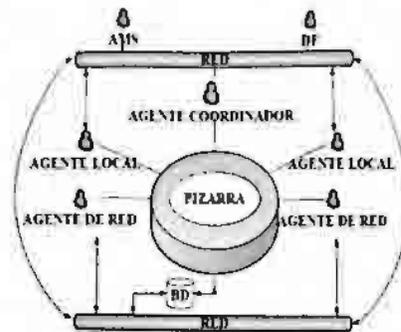


Fig. 4. Jerarquía de agentes.

5 Modelo de confianza para control de perfiles de usuario

El concepto de confianza puede tener varias definiciones o interpretaciones. Mayer la define como la voluntad de una persona vulnerable a las acciones de otro, basado en la expectativa que ejecutará una importante acción en particular sin tener que ir al control y seguimiento de dicha acción [6]. Sin embargo, en equipos de cómputo no es posible aplicar esta definición de confianza sin un factor que permita evaluar el desempeño de un usuario relacionado con el perfil asignado.

Una manera de definir lo que es confianza respecto a los temas tratados es:

“La confianza es la seguridad o verdadera esperanza que tenemos de otro individuo o algo en particular”

Y esta será la base de la originalidad de este trabajo, ya que se tiene la intención de implementar un sistema que permita manejar el factor de confianza y aplicarlo a los perfiles de usuarios, sin tener que restringirlo de principio, otorgándole al contrario, el máximo grado de confianza que se pueda tener, de tal forma que éste se reduzca automáticamente conforme el usuario infrinja las reglas previamente establecidas, disminuyendo los privilegios asignados, evitando así caer en el error de hostigar a un usuario que no ha realizado daño alguno al sistema, provocando comportamientos hostiles derivados de una muestra de descontento, realizando acciones dañinas que no habían sido premeditadas.

5.1 Control de acceso

La autenticación ayuda a establecer quién es el individuo que accede a un sitio. Si consideramos el área de seguridad computacional nos referimos a quien es el usuario o programa que está intentando irrumpir en nuestro sistema.

La autorización nos dice lo que podemos hacer dentro de nuestro sistema, es decir, de que privilegios podemos gozar mientras estemos dentro de una sesión. Ambos conceptos parecen estar ligados indiscutiblemente, pero no siempre es así.

Los dos conceptos anteriores frecuentemente están vinculados de una manera poco clara y son independientes. Aun así, cuando intentamos acceder a un recurso de información a través de la red sin necesidad de usar protección física, existen tres actividades que están vinculadas: Autenticación (quien soy), autorización (lo que puedo hacer) y el registro de auditoría (lo que he hecho) [7]. En el caso de las políticas de seguridad, es fundamental no perder de vista estos conceptos, ya que son la clave para la estabilidad del monitoreo y mantener controladas las actividades de los usuarios de un sistema.

5.2 Asignación de pesos a las políticas de seguridad

Como parte de esta propuesta, se plantea utilizar una medida de peso en el factor de confianza. Para ello utilizamos la serie de números primos que permiten identificar a cada una de las políticas, aplicando el criterio de asignar menor peso a las políticas que tienen mayor grado de restricción, y por lo tanto, cuanto más grande sea el peso de la política que se infrinja, menor será el impacto del castigo proporcionado. Y si uno se pregunta por qué los números primos, es por esta razón: simplemente porque la sumatoria de n números primos todos distintos, siempre será diferente si n cambia y solo habrá una combinación posible para que la suma de un conjunto de números primos den como resultado X valor, es decir, un 8 solo puede darse como resultado de $3+5$ en numeración prima, y un 26 solo puede darse con la suma de $2+3+5+11$.

A continuación se muestra como ejemplo de lo planteado anteriormente, una tabla con la asignación de pesos a una serie de políticas:

- Políticas permanentes de usuarios, excepto para el administrador:

Índice (i)	Políticas Permanentes PP()	Peso-privilegio(NP)
1	Mostrar archivos del sistema y ocultos	2
2	Habilitar cambios en la configuración del sistema	3
3	Acceso al registro de Windows	5

- Lista de privilegios controlados:

Índice (i)	Políticas Dinámicas PD()	Peso-privilegio(NP)
1	Borrar historial de internet	7
2	Acceso a consola MS-DOS	11
3	Permitir instalación de aplicaciones	13

Índice (i)	Políticas Dinámicas PD()	Peso-privilegio(NP)
4	Acceso al administrador de tareas	17
5	Acceso al panel de control	19
6	Acceso a configuración de red	23
7	Acceso a sistema de configuración de hardware	29

- NP → Número primo que se asigna a cada privilegio.
- PP → Valor alfanumérico que describe a una política permanente, que en conjunto constituyen a un grupo de MGPP.
- PD → Valor alfanumérico que representa a una política dinámica, que en conjunto constituyen un grupo de MGPD.

5.3 Evaluación del Perfil de usuarios

Para medir o evaluar de alguna manera el nivel de privilegios que tiene un usuario, es decir, tener la posibilidad de reconocer las violaciones a privilegios otorgados, se ha diseñado una serie de operaciones que nos permitirán calcular y guardar en variables el valor del perfil que el sistema de agentes ha asignado a un usuario conforme al desempeño y respeto al grado de confianza otorgado en forma inicial.

MGPD → Máximo grado del factor de confianza con políticas dinámicas.

$$MGPD = 1$$

$$MGPD = \prod_{i=1}^n PD(NP_i) \quad (1)$$

MGPP → Máximo grado del factor de confianza con políticas permanentes.

$$MGPP = 1$$

$$MGPP = \prod_{i=1}^n PP(NP_i) \quad (2)$$

PR → Variable auxiliar para verificar si el privilegio que se analiza se relaciona con una política de restricción.

$$PR = MGPD \bmod PD(NP_i) \quad (3)$$

$$PR = \begin{cases} 0 \rightarrow \text{Privilegio NO restringido} \\ \text{otro caso} \rightarrow \text{privilegio restringido} \end{cases}$$

MGFC → Máximo grado del factor de confianza general.

$$MGFC = MGPP + MGPD \quad (4)$$

I → Valor de incidencia en quebrantamiento de políticas.

P → Valor de penalización por violentar políticas establecidas.

Este peso debe estimarse en base a:

PA → Perfil actualizado

El sistema de agentes debe vigilar que MGFC no sea modificado. En caso de que MGFC haya sido modificado sin autorización, el sistema de agentes deberá tomar las acciones adecuadas para estimar PA y restablecer valores a MGFC, realizando las acciones necesarias para restringir los privilegios que hayan sido afectados.

4.4 Modelo de confianza para control de perfiles de usuarios

Cabe aclarar que no se han encontrado referencias de algo igual o parecido al modelo que a continuación se propone, donde se contempla un procedimiento para medir el grado de respeto que un usuario tiene en base a la confianza depositada.

La fórmula planteada para efectos de aplicar esta idea es la siguiente:

$$E(U) = F - P \quad (5)$$

Donde los términos son:

U → Identificador de usuario que es evaluado.

E → Evaluación de perfiles de usuarios.

F → Factor de confianza que permite verificar el comportamiento de un usuario.

Sirve como medida para valorar la confianza que se ha depositado en el usuario a través de los permisos y privilegios otorgados en el sistema.

M → Medida del factor de confianza registrado cuando se detectan infracciones a las reglas establecidas.

N → Número de registros de ocasiones en que se ha detectado que se infringen las políticas establecidas.

P → Promedio de la valoración del comportamiento del usuario por períodos.

Para obtener P se aplica la siguiente fórmula:

$$P = \frac{1}{N} \sum_{i=1}^J M \quad (6)$$

Por último, para valorar el resultado obtenido en E, se pueden tomar como base los siguientes criterios:

- Si $E = 0$, se estima que el usuario tiene un comportamiento perfectamente estable, ya que el factor de confianza no ha variado en el periodo que se evalúa.
- Si $E > 0$ se estima que el usuario ha defraudado la confianza otorgada porque el valor ideal o perfecto para E es 0. Además, es un indicativo que los valores de confianza de F han disminuido y por lo tanto los privilegios del usuario han sido restringidos debido a infracciones detectadas por el sistema de agentes.
- Si $E < 0$ debe revisarse muy bien el resultado y el estado del equipo, ya que el sistema no permite esta posibilidad porque el factor de confianza F tiene un tope establecido y el promedio P no debe exceder nunca el valor de F.

7 Conclusiones

En este trabajo se abordaron conceptos fundamentales y básicos que se requieren para elaborar políticas de seguridad confiables que nos ayuden a mantener íntegros los recursos y acciones en equipos de cómputo con ambiente Windows-Tecnología NT y también se analizó el hecho de que comúnmente en las organizaciones no se da mucha importancia a la implementación de políticas correctas, o a políticas que puedan protegerlos de problemas relacionados con la ingeniería social y seguridad computacional, es decir, tener posibilidades de saber que usuarios no autorizados cambian la configuración de las políticas haciendo a sus sistemas vulnerables a los ataques cibernéticos. Además de lo anterior se realizó un estudio de cuáles son las herramientas de software y que novedades existen hoy en día para atacar los diversos problemas que conlleva el manejo de políticas relacionadas con perfiles de usuarios en este tipo de sistemas, y a pesar de que existen muchas iniciativas al respecto, se encontró que a la fecha es un problema que no se ha podido controlar eficientemente.

Es por lo antes expuesto que proponemos utilizar tecnología de agentes para buscar una alternativa de solución, ya que es una técnica que ha mostrado efectividad en la solución de otro tipo de problemas y podría convertirse en una línea de investigación que permita valorar el comportamiento de un sistema de agentes para cuidar y controlar la aplicación de políticas en sistemas con ambiente Windows-Tecnología NT. La ventaja de aplicar esta técnica es que un conjunto de agentes trabaja en al mismo tiempo para aplican acciones relacionadas con el cuidado interno de los equipos de cómputo, aplicando un modelo de confianza que permita medir que tan confiable es un usuario en relación con el grado de confianza que se ha depositado en él en un principio y los cambios que presente en el transcurso del tiempo.

Referencias

1. Departamento de control de calidad y auditoría informática, Vista General del Libro Naranja, 2000, páginas 7-8. <http://gseguridad.unicauca.edu.co/documentos/orangeb.pdf>
2. Cota O. María de Guadalupe & Soto B. J. Pablo. "Architecture for design and development of security Systems based on agent technology". WorldComp'11 – The 2011 World Congress in Computer Science, Computer Engineering, an Applied Computing (ICAI'11 – International Conference on Artificial Intelligence). Las Vegas, Nevada USA. Julio 2011. (<http://www.world-academy-of-science.org/worldcomp11/ws>)
3. Patrick D. Howard, «The Security Policy Life Cycle: Functions and Responsibilities, Information Security Management Handbook», CRC Press LLC, 2003.
4. Alejandro Núñez Sandoval, «Taller de desarrollo de políticas de seguridad», Universidad Nacional Autónoma de México, 2005, pp. 5.
5. Wooldridge M. & Jennings N.R. Intelligent agents: Theory and practice. The Knowledge Engineering Review. Vol. 10 (2). 1995. pp. 116–152
6. Mayer, Roger; Davis, James H.; Shoorman, F. David, An Integration model of organizational trust, Academy of Management Review. (2007).
7. Luis Rodríguez Berzosa, Control de accesos: De la era mainframe a las PKI, <http://www.iec.csic.es/criptonomicon/articulos/expertos69.html>.
8. <http://www.spywarevoid.com/remove-olmarik-trojan-win32olmarik-removal.html>
9. <http://www.fasterpcclean.com/es/eliminar-olmarik-trojan>.
- 10 Michael McInerney, Windows NT Security, Editorial Prentice Hall, 2000, pp. 120-121

Hacia la mejora de los algoritmos SVM y SNN para arquitectura GPGPU-CUDA

Israel Tabarez-Paz¹, Neil Hernández-Gress¹, Miguel González-Mendoza¹

¹Tecnológico de Monterrey

Campus Estado de México
Atizapán de Zaragoza, Estado de México, México
{a01104374, ngress, mgonza}@itesm.mx

Abstract. En este artículo presentamos un estudio comparativo de las Máquinas de Soporte Vectorial (SVM) con la red neuronal Spiking Neural Network (SNN). En el primer caso se tiene el problema de que los requerimientos computacionales aumentan exponencialmente al utilizar Programación Cuadrática (QP). Por esta razón se han diseñado otros métodos de optimización que dividen el problema en subproblemas tal como el Método de Optimización Mínima Secuencial (SMO, por sus siglas en inglés), Programación Lineal (PL), y sus variantes como LS – SVM, entre otros. Por otro lado, en el caso de SNN, su principio de funcionamiento se apega al funcionamiento de una neurona biológica de un mamífero. De esta manera se tienen tres modelos matemáticos que representan el principio de funcionamiento de la neurona de la red SNN. En el caso de SNN la dimensión de la variable de entrada puede ser alta. Se muestran algunos resultados obtenidos en MatLab en un ambiente secuencial (CPU) y se comparan las metodologías mencionadas. Finalizamos presentando las consideraciones necesarias para implementar ambos algoritmos en una arquitectura paralela GPGPU – CUDA.

Keywords: Spiking Neural Networks, Máquinas de Soporte Vectorial, Clasificación.

1 Introducción

En la segunda mitad del siglo XX, comenzaron a desarrollarse Redes Neuronales Artificiales (RNA), que son consideradas como la primera generación, tales como la de MC Culloc (1943), Hebb (1949), Perceptrón (1957), Adaline y Madaline (1959), Cognitrón (1980) y Hopfield (1982), entre las más conocidas. Sin embargo, fue hasta 1982 en que Hinton y William desarrollaron el algoritmo de retropropagación dando apertura a la segunda generación de las redes neuronales. También, en la década de 1930, los científicos británicos Alan Lloyd Hodgkin (1914-1998) y Andrew Fielding Huxley (nacido en 1917) iniciaron el modelo matemático que hoy se conoce con su mismo nombre Hodgkin – Huxley [15]. Su estructura es de un sistema de cuatro ecuaciones diferenciales en derivadas parciales, acopladas, no lineales, dependientes del espacio y del tiempo. Con este modelo se logró describir la generación y propagación del potencial de acción en el axón gigante de calamar, como así también explicar sus principales propiedades. Este modelo es aplicado al modelo de red

neuronal SNN, por su semejanza con el funcionamiento de las neuronas de los mamíferos. Fue en la década de 1990 que comenzó el desarrollo de las SVM [23] para solución de los problemas linealmente separable y no linealmente separables aunque Vapnik lo había inventado desde 1979 [21]. Con esta metodología se logró calcular la arquitectura de la red. En la primera década del siglo XXI, Bohte [18] y Olaf [16] comenzaron desarrollar la tercera generación de las redes neuronales artificiales al aplicar el método de Spikeprop para SNN, que está basado en el algoritmo de retropropagación. En la misma época, Izhikevich [19] diseñó un modelo matemático a partir del modelo de Hodgkin – Huxley pero con dos ecuaciones diferenciales que también explican el comportamiento de las neuronas de los mamíferos.

Por otra parte, la programación paralela es una técnica de programación en que muchas instrucciones se ejecutan simultáneamente [29]. Sin embargo, la programación paralela puede ser de tres niveles distintos que son: a nivel bit, nivel instrucción y de tareas. De los cuales, en nuestro caso nos enfocaremos al nivel instrucción. En general, el modelo de programación paralela ayuda a dividir el problema en múltiples subproblemas por lo que pueden resolverse de forma independiente y concurrente. A los subproblemas se les aplica un nivel de granularidad, de tal manera que cada uno de ellos puede resolverse simultáneamente con múltiples instrucciones ejecutadas también en paralelo.

De acuerdo con [29] los chips de gráficos fueron evolucionando a través de los años desde la ejecución de funciones fijas hasta que se fueron haciendo más programables. A principios del siglo XXI, investigadores empezaron a usar las GPUs para ejecutar aplicaciones de cálculo obteniendo un alto rendimiento computacional y como consecuencia se logró aumentar la velocidad de ejecución para una gran variedad de aplicaciones científicas. Por esta razón, los lenguajes de programación utilizados en programación paralela fueron los usados en gráficos como OpenGL y Cg. Es así que se comenzó a desarrollar el hardware para soporte de los lenguajes C, C++ y Fortran y así se desarrolló la posibilidad de aprovechar el gran rendimiento de las GPUs para usos científicos. Finalmente, se implementó la para arquitectura CUDA.

En la figura 1 se muestra el modelo de programación del hardware de CUDA para el procesamiento de tareas y procesos con diferentes niveles de granularidad. El programador puede desarrollar su aplicación a través de diferentes lenguajes de alto nivel como C, C++ y Fortran, etcétera.

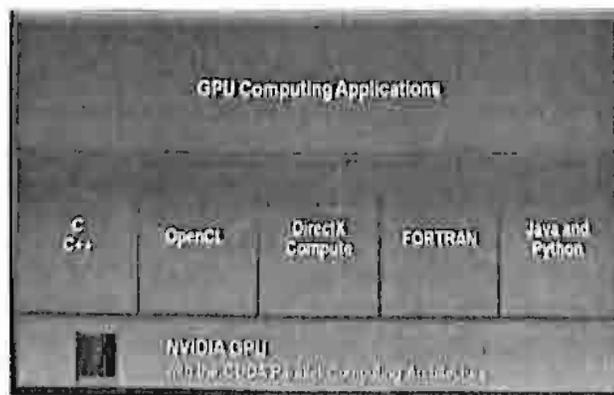


Fig. 1. Arquitectura paralela y modelo de programación en CUDA.

Los objetivos de este trabajo son, primero, analizar la velocidad de aprendizaje, uso de recursos y eficiencia de la metodología Spikeprop de SNN para compararlas con metodología SVM; y segundo, para el caso particular de las SNN, es posible plantear las mejoras en el algoritmo en donde se tienen las siguientes áreas de oportunidad: los rangos de sus parámetros aún no se han determinado, tampoco se conoce su complejidad computacional, no se tiene un método para calcular la arquitectura óptima de acuerdo a cada problema, y no se sabe que tan eficiente es respecto a SVM que en este caso, de acuerdo con Platt [21], es rápida, eficiente y asegura el valor mínimo global.

Se seleccionó SNN debido a las áreas de oportunidad que se tienen, las cuales ya se mencionaron en el párrafo anterior. También porque es muy probable que el algoritmo tenga velocidad de aprendizaje, eficiencia y ahorro de recursos de memoria comparables con SVM. Además, SVM depende de un optimizador y que en muchas ocasiones la complejidad computacional es alta y en SNN aún no está determinada su complejidad, asimismo SVM tiene un número limitado de funciones kernel y SNN no depende de una función Kernel.

Por otro lado, respecto al algoritmo Spikeprop, utiliza como función de activación una sumatoria de exponentes [16] en la entrada de cada neurona, y esto representa el impulso eléctrico en cada neurona. Por otro lado, de acuerdo con Bhuiyan el modelo de Izhikevich es más eficiente que el modelo Hodgkin – Huxley [26], aunque éste último es considerado como el más parecido al funcionamiento de los mamíferos, que es un modelo sujeto a cuatro ecuaciones diferenciales y veinticinco parámetros comparado con el de Izhikevich que contiene cuatro parámetros y dos ecuaciones diferenciales. Comparando con Spikeprop, este algoritmo es más sencillo y fácil de implementar, ya que tiene como parámetros la matriz de pesos y el tiempo, y como función de activación una sumatoria, por lo que es más simple que una ecuación diferencial. La desventaja principal de utilizar ecuaciones diferenciales es que dificulta la portabilidad en las aplicaciones al implementarse en una microcomputadora, como un microcontrolador, GPU o FPGA, pues requiere de complejos métodos matemáticos de solución que no pueden ser portables en algún hardware específico.

Por las razones del párrafo anterior nos enfocamos a SNN. Es así que presentamos algunas consideraciones para mejorar la velocidad de aprendizaje y eficiencia de SNN y SVM, entre ellas se sugiere utilizar una arquitectura GPGPU – CUDA de programación paralela. También, la velocidad de aprendizaje se puede mejorar al utilizar otra función de activación y sobre todo encontrando los rangos de los valores de los parámetros que considera esta metodología. Como consecuencia, esto influirá en la eficiencia, sin embargo otra consideración para mejorar la eficiencia es probablemente la codificación de la señal de entrada. Esto depende del resultado de los experimentos.

2 Estado del Arte

Con los siguientes trabajos se observan varios avances significativos en el caso de los métodos de aprendizaje SNN (Spiking Neural Networks) y SVM (Support Vector Machine) (ídem).

Izhikevich [3], diseña un modelo híbrido para SNN para combinar los métodos numéricos continuo y discontinuo.

Yudanov [2], implementa un sistema híbrido para SNN usando integración numérica de Parker Sochacki (PS) con orden adaptativo. La implementación es validada comparando la implementación entre un GPU y un CPU.

Bohte [18] realiza una comparación entre diferentes algoritmos, entre ellos Spikeprop. El trabajo reporta que con menos iteraciones se logró mayor precisión. Sin embargo, no midió el tiempo de entrenamiento ni la complejidad computacional. Por otro lado, presenta la teoría del funcionamiento de SNN dejando ver la diferencia con respecto a las redes neuronales artificiales (RNA) tradicionales, en dónde presenta la solución al problema XOR con esta red, pero no hace alguna comparación en los parámetros de tiempo y arquitectura de SNN con otras. Tampoco, presenta resultados con bases de datos más complejas.

De acuerdo con Xin Jin [4] utilizó un SpiNNaker chip que contiene varios procesadores ARM968 con una velocidad de 200MHz cada uno. Comparó estos resultados con los obtenidos en MatLab. Es así que se evaluaron tres tiempos: estado de actualización de la neurona, actualización del arreglo de entradas, procesamiento de una entrada nueva. Sin embargo, no realizó experimentos con otras bases de datos ni comparó con SVM.

También, Olaf [16] muestra una panorama amplio del funcionamiento de SNN y el problema de la codificación de una entrada analógica. Una de sus aportaciones es que utiliza SNN para el reconocimiento de dos palabras de una sílaba a través del movimiento de los labios consiguiendo la identificación de las palabras.

Schrauwen [13] menciona identificar algunos errores en el algoritmo Spikeprop corrigiéndolos al agregar el ajuste de los parámetros de tiempo y umbral. Sin embargo no procesa diversas bases de datos, ni tampoco el tiempo de entrenamiento con otras redes neuronales.

Pavlidis, et. al [14], aplica algoritmos de evolución usando SNN a la base de datos Iris y compara los resultados obtenidos con otras metodologías. Pero no compara sus resultados con SVM, además no calcula la eficiencia obtenida.

Catanzaro [12] se enfoca a SVM y aplica el algoritmo de Optimización Mínima Secuencial (SMO) para comparar tiempos y precisión de clasificación de un procesador GPU con las librerías para SVM de MatLab.

Sergio Herrero [1] continua el trabajo de Catanzaro. Realiza la comparación en el tiempo de entrenamiento y precisión en la clasificación utilizando dos tarjetas GPU y las librerías para SVM de MatLab.

Bhuiyan [6] compara los dos modelos de SNN, Izhikevich y Hodgkin Huxley, que son aplicados reconocimiento de caracteres.

Stewart y Bair [9], aplica el método de Runge – Kutta para los modelos de Izhikevich y Hodgkin Huxley, después compara dichos métodos determinando que el segundo es más eficiente.

Thomas y Luk [10], simularon hasta 1024 neuronas en un FPGA el modelo de Izhikevich.

Nageswaran [11], demuestra la eficiencia del modelo de Izhikevich.

Izhikevich [19] simula en MatLab el modelo de Hodgkin Huxley. Asimismo, logra ejecutar hasta 10000 neuronas y 1 000 000 de sinapsis.

Papadonikolakis [7], quien se ha enfocado al mejorar la velocidad y eficiencia de SVMs utilizando varios métodos, también compara estos dos parámetros entre un GPU y un FPGA programando el algoritmo de Gilbert.

Carpenter [8] presenta un software aplicado a SVM para resolver los problemas de predicción, clasificación y regresión. Para ello, utilizó el software desarrollado por Catanzaro.

González, et al [17], quien se enfoca a comparar el tiempo de entrenamiento de diferentes metodologías utilizadas en SVM, utilizando diversas bases de datos de UCI. En este caso, los resultados obtenidos en este trabajo fueron que en el caso de la base de datos Iris, el peor tiempo fue aproximadamente de 19.6 segundos con SVM QP. Sin embargo, en este caso no se especifica la precisión ni compara con SNN. Aunque por la naturaleza de SVM sólo pueden ser elegidos dos conjuntos de entrada a clasificar, sí se pudieron clasificarse más de dos conjuntos con un grado de error pequeño. En contraste, en SNN pueden ser seleccionados más de dos conjuntos de entrada a la red neuronal.

3 Resultados en Matlab

Se implementaron los algoritmos Spikeprop de SNN y SVM en un CPU en MatLab y se compararon ambos resultados obtenidos, los cuales se muestran en la tabla 1.

En los resultados de la tabla 1, para SVM existen dificultades para clasificar multiconjuntos, por lo que se tienen que implementar técnicas para ir discriminando cada conjunto a medida que es clasificado. Es decir, sólo se puede clasificar solo un conjunto a la vez. Posteriormente se elige otro conjunto para ser clasificado y se actualiza la salida esperada correspondiente, y así sucesivamente. De esta manera, como resultado en cada nueva clasificación se va reduciendo el número de instancias. Como consecuencia, el tiempo de aprendizaje se va reduciendo significativamente. Este procedimiento se repite dependiendo del número de clases que se tenga.

Por otro lado, en caso de SNN se tiene la ventaja de que se pueden clasificar multiconjuntos al mismo tiempo, sin embargo la eficiencia depende de la arquitectura de red aplicada y de la codificación de la entrada al espectro del tiempo. Al principio se le indica el grado de error deseado, ya que el tiempo de procesamiento es muy lento. En cuanto al valor de error mostrado en la misma tabla, se refiere al error cuadrático medio en el caso de SNN.

Tabla 1. Resultados previos en MATLAB

	No. De Instancias	Tiempo de aprendizaje (s)	Error (Arquitectura)
IRIS			
		CLASIFICACIÓN 1 SVM	
SVM	150	0.0310	0 (14 SV)
		CLASIFICACIÓN 2 SVM	
SVM	50	0.000	0 (8 SV)
SNN	150	80.5036	4 (4,4,1 neuronas)
Car Evaluation			
		CLASIFICACIÓN 1 SVM	
SVM	1728	32.642	0 (735 SV)
		CLASIFICACIÓN 2 SVM	
SVM	1343	18.391	0 (245 SV)
		CLASIFICACIÓN 3 SVM	
SVM	1275	17.625	1 6 0 (139 SV 6 138 SV)
SNN	1728	69.513	727.9563 (7,5,4 neuronas)
Breast Cancer Wisconsin			
SVM	683	3.906	0 (98 SV)
SNN	683	32.421	128.78 (10,5,1)
HEPATITIS			
SVM	155	1.516	0 (144)
SNN	155	1.906	4.1590 (20,5,1 neuronas)

Asimismo, se observa que SVM es más eficiente y veloz con MatLab. En general, se requiere mejorar el tiempo de aprendizaje, de la metodología SNN, porque como observamos el tiempo depende completamente de los parámetros de entrada mencionados anteriormente y los valores iniciales de los pesos, los cuales deben ser diferente de cero. Si se inicializan los pesos en cero, entonces no habrá aprendizaje y el algoritmo no funciona. Por otro lado, los valores iniciales del umbral, retardo y tiempo de activación de la neurona, fueron inicializados de manera aleatoria y también como resultado de las observaciones realizadas en los experimentos. Es decir, de acuerdo con la amplitud de los impulsos, se seleccionaron dichos parámetros.

Finalmente, las observaciones realizadas dirigen esta investigación a mejorar el algoritmo en los aspectos dados en el párrafo anterior. Una ventaja importante que se observa es que la salida precisa lo que nos da un panorama más amplio de las aplicaciones relacionadas a la metodología, como reconocimiento de patrones.

5 Consideraciones y Trabajo a Futuro

A continuación se dan algunas consideraciones para la mejora de metodología SNN, de acuerdo con los resultados obtenidos anteriormente:

- Es más fácil aplicar Spikeprop por ser el método que tiene menos parámetros a considerar, además por la sencilla función de activación que utiliza, lo cual nos indica que es muy probable de aplicar en tecnología portable.

- Es necesario encontrar el rango de los pesos de las conexiones del algoritmo Spikeprop.
- Es necesario encontrar el rango del tiempo de la función de activación que influye en la amplitud de la función de activación.
- Es necesario encontrar el rango del umbral de la función de transferencia.
- Es necesario considerar otras funciones de transferencia ya que en el modelo que Bothe [18] nos presenta tiene como desventaja el lento tiempo de aprendizaje.
- Es necesario considerar la programación en arquitectura GPGPU – CUDA para reducir el tiempo de aprendizaje.

Por otro lado, respecto a SVM se tienen las siguientes consideraciones de mejora, de acuerdo con los resultados obtenidos:

- Seleccionar SMO por la sencillez en la programación, rapidez y eficiencia con respecto a otros algoritmos. Teóricamente, el método QP asegura el valor óptimo global, sin embargo en la práctica no se ha desarrollado algún software por la dificultad que representa y la exactitud de la selección de los datos, y en el caso de SMO es un método rápido y eficiente [21] que aproxima al valor del óptimo global y en la práctica si se ha desarrollado software.
- Para comparar Spikeprop con SMO, es necesario considerar la programación en la misma arquitectura GPGPU – CUDA.

6 Conclusiones

En este trabajo se concluye que bajo una misma plataforma de hardware utilizando un CPU y el software MatLab SVM es más eficiente al obtenerse menos errores que con SNN, y además la velocidad del tiempo de aprendizaje de SVM es mayor que con SNN. Sin embargo de acuerdo con las características de cada método vistas anteriormente, nos indican que es muy probable que, utilizando una plataforma de hardware GPU y agregando las mejoras en las áreas de oportunidad mencionadas en el documento, la metodología Spikeprop de SNN sea comparable con SMO de SVM.

7 Bibliografía

- [1] Sergio Herrero-Lopez, John R. Williams, Abel Sanchez.: Parallel Multiclass Classification using SVMs on GPUs, November 2010.
- [2] Yudanov, D.; Shaaban, M.; Melton, R.; Reznik, L.: GPU-Based Simulation of Spiking Neural Networks with Real-Time Performance & High Accuracy, Feb 2010.
- [3] Izhikevich, E.M.: "Hybrid spiking models", vol. 368, issue 1930, pp. 5061-5070, Nov 2010
- [4] XIN JIN: Parallel Simulation of Neural Networks on Spinnaker Universal Neuromorphic Hardware, University of Manchester, 2010
- [5] Papadonikolakis, M.; Bouganis, C.: "A novel FPGA-based SVM classifier." Field-Programmable Technology (FPT), 2010 International Conference on , vol., no., pp.283-286, 8-10 Dec. 2010
- [6] Bhuiyan, M.A.; Pallipuram, V.K.; Smith, M.C.: "Acceleration of spiking neural networks in emerging multi-core and GPU architectures," Parallel & Distributed Processing, Workshops and Phd Forum (IPDPSW), 2010 IEEE International Symposium on , vol., no., pp.1-8, 19-23 April 2010

- [7] Papadonikolakis, M.; Bouganis, C.-S.; Constantinides, G.: "Performance comparison of GPU and FPGA architectures for the SVM training problem," *Field-Programmable Technology, 2009. FPT 2009. International Conference on*, vol., no., pp.388-391, 9-11 Dec. 2009
- [8] Carpenter, A.: "CUSVM: A CUDA IMPLEMENTATION OF SUPPORT VECTOR CLASSIFICATION AND REGRESSION", Jan. 2009
- [9] Stewart R.D., Bair W.: "Spiking neural network simulation: numerical integration with the Parker-Sochacki method", Jan. 2009
- [10] Thomas, D.B.; Luk, W.: "FPGA Accelerated Simulation of Biologically Plausible Spiking Neural Networks." *Field Programmable Custom Computing Machines, 2009. FCCM '09. 17th IEEE Symposium on*, vol., no., pp.45-52, 5-7 April 2009
- [11] Nageswaran, J.M., Dutt, N.; Krichmar, J.L.; Nicolau, A.; Veidenbaum, A.; , "Efficient simulation of large-scale Spiking Neural Networks using CUDA graphics processors," *Neural Networks, 2009. IJCNN 2009. International Joint Conference on*, vol., no., pp.2145-2152, 14-19 June 2009
- [12] Bryan Catanzaro, Narayanan Sundaram, Kurt Keutzer: *Fast Support Vector Machine Training and Classification on Graphics Processors*, International Conference on Machine Learning, Helsinki, Finland, 2008.
- [13] Benjamin Schrauwen, Jan Van Campenhout, *Improving SpikeProp: Enhancements to An Error-Backpropagation Rule for Spiking Neural Networks*, Ghent University ELIS-PARIS (2006)
- [14] N.G. Pavlidis, D.K. Tasoulis, V.P. Plagianakos, G. Nikiforidis, M.N. Vralatis: *Spiking Neural Network Training Using Evolutionary Algorithms*, IEEE International Joint Conference on, pp: 2196 – 2194, vol. 4 ,december 2005.
- [15] Seijas Fossi C., Caralli D' Ambrosio A.: *Uso de las máquinas de soporte para la estimación del potencial de acción celular*, Revista de Ingeniería UC. Vol. 11, N° 1, 56 – 61 (2004).
- [16] Olaf Booij.: *Temporal Pattern Classification using Spiking Neural Networks*, Intelligent Sensory Information Systems Informatics Institute Faculty of Science Universiteit van Amsterdam (2004).
- [17] González-Mendoza M., Hernández-Gress N, André Titli: *Quadratic Optimization fine tuning for the learning phase of SVM*, (2004).
- [18] Sander M. B.: *Spiking Neural Networks*, Universiteit Leiden (2003).
- [19] Izhikevich, E.M.: "Simple model of spiking neurons," *Neural Networks, IEEE Transactions on*, vol.14, no.6, pp. 1569- 1572, Nov. 2003
- [20] Sander M. Bohte, Joost N. Kok, Han La Poutre: *Error-backpropagation in temporally encoded networks of spiking neurons*, *Neurocomputing* 48, pp: 17 – 37, (2002).
- [21] Platt C. J.: *Sequential Minimal Optimization: A fast Algorithm for Training Support Vector Machines* (1998).
- [22] Osuna E., Freund R., Girosi F.: *An Improved Training Algorithm for Support Vector Machines*, In *Proc. of IEEE NNSP'97*.
- [23] Vapnik V., Cortes C.: *Support Vector Networks* (1995).
- [24] L.A. Zadeh: *Fuzzy Sets*, *Inform. Control*, Vol. 8, pp: 338 – 353 (1965).
- [25] Masoud Mohammadian: *Computational intelligence for modelling, control & automation: neural*.
- [26] A. L. Hodgkin and A. F. Huxley: *A quantitative description of membrane current and its application to conduction and excitation in nerve*, *J. Physiol.* (1952).

8 Referencias

- [27] <http://www.svms.org>
- [28] <http://archive.ics.uci.edu/ml/>
- [29] <http://www.nvidia.com>

Análisis del historial climático del Valle de Toluca con minería de datos

1J. C. Torres Arellano, 1R. M. Valdovinos Rosas, 2R. Alejo, 1D. Calderón

1Universidad Autónoma del Estado de México
Centro Universitario Valle de Chalco, Av. Hermenegildo Galeana 3, 56615
Valle de Chalco, México.

2Instituto Tecnológico de Estudios Superiores de Jocotitlan, Carretera Toluca-
Atlacomulco Km 44.8, col. Ejido de San Juan y San Agustín, Jocotitlan, 50700,
México.

e-mail: atej22@yahoo.com.mx, li_rmvr@hotmail.com, ralejoll@hotmail.com,
dcatal@hotmail.com.

Resumen: Hoy en día el cambio climático es un tema relevante para nuestra supervivencia. Para entender lo que ocurre y poder revertir los efectos, es necesario acudir a herramientas que permiten monitorear parámetros atmosféricos y meteorológicos, en los que se busca encontrar patrones de comportamiento que arrojen a la búsqueda de estrategias de solución. En la República Mexicana se dispone de una amplia red automática de monitoreo atmosférico (RAMA) que realiza mediciones atmosféricas para determinar los cambios que han tenido los parámetros meteorológicos. En este trabajo se propone aplicar algoritmos de minería de datos a la RAMA Toluca para identificar el patrón climático que se ha presentado en el Valle de Toluca.

Palabras claves: historial climático, minería de datos, agrupamiento, WEKA.

1 Introducción

La Zona Metropolitana del Valle de Toluca (ZMVT) es la segunda conurbación en importancia del Estado de México después de la Zona Metropolitana del Valle Cuautitlán-Texcoco. Está considerada como uno de los poblados más importantes de la región Centro del País. En esta metrópoli viven alrededor de 1.3 millones de habitantes; posee importantes parques industriales, y anualmente arriban a ella cientos de personas con el fin de desarrollar un proyecto de vida [9]. Entender lo factores que controlan el clima y las causa del cambio climático permite evaluar cuanto de este cambio es producto de la variación natural del sistema climático y hasta donde es una

¹ Juan Carlos Torres Arellano Centro Universitario UAEM Valle de Chalco, México. Calle Hermenegildo Galeana No. 3 col. María Isabel Valle de Chalco, Estado de México Tel. 57 30 47 00 correo electrónico: atej22@yahoo.com.mx

Rosa María Valdovinos Rosas Centro Universitario UAEM Valle de Chalco, México. Calle Hermenegildo Galeana No. 3 col. María Isabel Valle de Chalco. Estado de México Tel. 57 30 47 00 correo electrónico: li-rmvr@hotmail.com

consecuencia de la actividad humana. Debido a la influencia que las variaciones meteorológicas y las condiciones topográficas locales tienen en los ecosistemas, se pone de manifiesto la importancia de analizar el historial climático del Valle de Toluca para definir el comportamiento de las variables meteorológicas en esta región. Por esta situación, en este estudio se acude a dos algoritmos ampliamente estudiados en minería de datos para identificar el comportamiento que los datos del historial climático del Valle de Toluca tienen durante el periodo de 2001-2003 en las estaciones de monitoreo Centro (Ce), San Lorenzo Tepaltitlán (SL) y San Mateo (SM).

La estructura de este documento cuenta con las siguientes secciones. La sección I describe los antecedentes involucrados en el sistema climático del Valle de Toluca y como apoyo la minería de datos en el proceso de análisis orientado al estudio de bases de datos con información del clima, en la sección II se comentan algunos trabajos relacionados con el comportamiento meteorológico del Valle de Toluca con herramientas de minería de datos, en la sección III se describe la metodología empleada dentro de la investigación para analizar el comportamiento meteorológico de dicho sistema, en la sección IV se muestran las conclusiones obtenidas y se concluye con la bibliografía.

1.1 El clima

El clima terrestre es controlado por el balance energético que existe entre los diversos factores que componen el sistema climático. La tierra intercepta la radiación del sol y esta energía es la que en última instancia controla el clima del planeta. [6]. Cerca de un tercio de la energía solar que la tierra recibe es reflejada de vuelta al espacio; el resto es absorbida por las diferentes partes del sistema climático: la atmósfera, los océanos, las capas de hielo, los continentes y las diversas formas de vida. Cada una de estas partes ayuda a distribuir la energía, disminuyendo los contrastes entre las áreas donde la mayor parte de la energía es recibida y las regiones donde la mayor parte de esta se reflejan. La tierra también manda energía de vuelta al espacio, este proceso se llama radiación. Sin embargo, existe una gran diferencia entre la energía que se recibe del sol y la que se emite desde la tierra, ya que el primero emite radiaciones de onda corta (en su mayoría rayos UV), mientras la superficie terrestre irradia energía en forma de calor, es decir radiaciones de onda larga (rayos infrarrojos). Como parte del proceso de radiación, un porcentaje de la energía de onda larga es reabsorbida por el sistema climático y reenviada en un proceso denominado efecto invernadero, el resto se disipa en el espacio [6] (ver Fig. 1).



Fig. 1. Esquema del efecto invernadero [6].

A largo plazo existe un balance entre la energía que se recibe y la que se emite; sin embargo, éste puede alterarse si ocurre cualquier cambio en los factores que controlan el proceso, provocando alteraciones en el clima. El resultado es que a lo largo de la historia de la tierra el clima ha cambiado múltiples veces. Algunos de estos cambios se han dado a escala global, mientras que otros han sido de escala hemisférica o regional y varios son los factores naturales y humanos que han estado involucrados.

1.2 Meteorología en el Valle de Toluca

Ubicada en la porción central del Estado de México, la Zona Metropolitana del Valle de Toluca (ZMVT) se extiende sobre una superficie de 1,208.55 km², en ella se ha observado una transformación paulatina en sus actividades económicas, pasando de ser una economía rural a una industrial y de servicios. Para darse una idea de la complejidad de la metrópoli, una tercera parte de su territorio tiene uso urbano; en ella se asientan 4,291 unidades manufactureras, 26,525 comercios, 14,665 establecimientos de servicio y posee un parque vehicular mayor a 260 mil vehículos. Tanto las fuentes fijas como móviles satisfacen su demanda energética, además de la energía eléctrica, mediante el consumo de combustibles fósiles, lo cual implica la emisión de contaminantes y un impacto adverso en la calidad del aire. [8]

Por lo anterior se destaca que la ZMVT cuenta con la infraestructura necesaria para la medición, registro y procesamiento de los datos de calidad del aire y los parámetros meteorológicos básicos que influyen en la dispersión, transporte, transformación y deposición de los contaminantes del aire [8]. A través de la Red Automática de Monitoreo Atmosférico de la Zona Metropolitana del Valle de Toluca (RAMA-ZMVT) que es operada por la Secretaría del Medio Ambiente del Gobierno del Estado de México desde 1993, a través de la Dirección General de Prevención y Control de la Contaminación Atmosférica. La RAMA-ZMVT consta de un Centro de Control y siete estaciones remotas de monitoreo distribuidas en la metrópoli, asimismo, está integrada por tres subsistemas operativos: 1) Red Automática de

K-Medias

Debido a su simplicidad, uno de los algoritmos más utilizados es el K-Medias. K-Medias es un algoritmo de clustering, el cual estructura grupos de objetos con los datos que tienen varios atributos en varias clases solo que los objetos que pertenecen a la misma clase sean similares y aquellos que no pertenecen se encuentren en clases diferentes.

El objetivo de K-Medias es determinar un número entero de particiones que optimice una función criterio: esta optimización es a través de un proceso iterativo, representa cada una de las agrupaciones (clúster) por la media de sus puntos, es decir, por su centroide; primero reasigna todos los puntos a sus centroides más cercanos, y en segundo lugar recalcula los centroides de los nuevos grupos creados en el anterior. El proceso continúa hasta alcanzar un criterio de termino [14].

La representación mediante centroides tiene la ventaja de tener un significado gráfico y estadístico inmediato. La suma de las discrepancias entre un punto y su centroide, expresado a través de la distancia apropiada, se usa como función objetivo. La función objetivo, suma de los cuadrados de los errores entre los puntos y sus centroides respectivos, es igual a la varianza total dentro del propio centroide [14].

EM

EM pertenece a una familia de modelos que se conocen como Finite Mixture Models, los cuales se pueden utilizar para segmentar conjuntos de datos. Se trata de obtener la FDP (Función de Densidad de Probabilidad) desconocida a la que pertenecen el conjunto completo de datos. Cada cluster se corresponde con las respectivas muestras de datos que pertenecen a cada una de las densidades que se mezclan. Se pueden estimar FDP de formas arbitrarias, utilizándose FDP normales n-dimensionales, t-Student, Bernoulli, Poisson, y log-normales [14].

El algoritmo EM, procede en dos pasos que se repiten de forma iterativa:

Expectation: Utiliza los valores de los parámetros, iniciales o proporcionados por el paso Maximization de la iteración anterior, obteniendo diferentes formas de la FDP buscada.

Maximization: Obtiene nuevos valores de los parámetros a partir de los datos proporcionados por el paso anterior.

Después de una serie de iteraciones, el algoritmo EM tiende a un máximo local de la función L. Finalmente se obtendrá un conjunto de clusters que agrupan el conjunto de proyectos original. Cada uno de estos cluster estará definido por los parámetros de una distribución normal [14].

2 Trabajos relacionados

El inventario de la Zona Metropolitana del Valle de Toluca 2004, obtuvo que en ese año en la ZMVT se emitieran poco más de 750 mil toneladas de contaminantes a la atmósfera; con los resultados obtenidos y la información procesada se identificó los principales contaminantes emitidos; así como las principales fuentes que los generan y los municipios con mayor aporte de emisiones. [9]

Por otro lado, en el año 2005 se elaboró un tercer inventario de emisiones de contaminantes considerando el año 2000 como año base e incluyó las partículas menores a 2.5 micrómetros (PM_{2.5}), los compuestos orgánicos Volátiles (COV) y la generación de emisiones domésticas de amoníaco (NH₃). En este inventario se recalcularon las emisiones contaminantes provenientes de las fuentes puntuales (industria) fuentes de área (servicios y comercios) fuentes móviles (vehículos automotores) y fuentes erosivas (agricultura y erosión) así como la incorporación de las emisiones provenientes de la fabricación artesanal de tabique rojo, de la disposición de residuos sólidos municipales y de las actividades metabólicas de vegetales. [9]

En la investigación realizada por el doctor Zidjan [7] se identifican niveles de contaminación por partículas menores a 10 µm (PM₁₀) mediante muestreo pasivo en la zona industrial del Valle de Toluca. El periodo de muestreo fue de junio de 2005 hasta agosto de 2006. Se realizó un análisis comparativo entre la concentración gravimétrica y las condiciones meteorológicas; finalmente se realizó un análisis comparativo entre estaciones de la concentración gravimétrica.

Además se buscaron patrones de referencia para poder discriminar asociaciones de fecha y vincularlos con las estaciones culturales del año, utilizando un algoritmo de asociación entre parámetros meteorológicos y fechas de muestreo (algoritmo K-Medias). Se determinó que en el otoño e invierno la concentración de PM₁₀ rebasa el límite permisible con mayor frecuencia que en las otras estaciones del año. [7] Mediante gráficos se identificó la dependencia de la temperatura y concentración gravimétrica, así como las relaciones con humedad, lluvia y/o velocidad del viento. Se encontró que en el verano hay una disminución de la concentración debido a las abundantes lluvias o bien a la alta humedad relativa.

3 Herramientas y Métodos

Para la realización de pruebas experimentales, se utilizó el sistema libre WEKA (Waikato Environment for Knowledge Analysis), desarrollado en la universidad de Waikato [2]. WEKA es una herramienta que permite la experimentación de análisis de datos mediante la aplicación, análisis y evaluación de las técnicas más relevantes de aprendizaje automático.

Tal como se mencionó en secciones anteriores, se utilizaron los algoritmos de aprendizaje no supervisado K-Medias y EM. En lo que respecta a K-Medias, los parámetros de inicialización son los siguientes: se introdujo el número de cluster máximo como 4 con la finalidad de que en ellos quedaran representadas las estaciones climáticas correspondientes a la primavera, verano, otoño e invierno, posteriormente se busca inicializar la semilla en 10 y se toma un máximo número de iteraciones de 50. En tanto que para el algoritmo EM se consideraron los siguientes criterios: como máximo número de iteraciones 50, un máximo número de cluster 4, y se inicializó la semilla en 10.

Debido a la calidad de los conjuntos de datos, en este análisis se consideraron los datos correspondientes a los años 2001, 2002 y 2003 de las estaciones de monitoreo de la zona Centro (Ce), San Lorenzo Tepaltitlán (SL) y San Mateo (SM), cuyas

variables de estudio comprenden los siguientes aspectos meteorológicos que se muestran en la Tabla 1.

Tabla 1. Variables de estudio de los aspectos meteorológicos.

TMP	Temperatura (°C)
HR	Humedad Relativa (%)
PA	Presión Atmosférica (mmHg)
RS	Radiación solar (w/m ²)
DV	Dirección del Viento (°)
VV	Velocidad del Viento (m/s)

Debido al monitoreo constante que se realiza, por cada hora se obtienen 6 mediciones, las cuales a lo largo de cada año constan de 52560 valores anuales; de esta forma entre las actividades de preprocesamiento se aplicó limpieza de datos y se eliminaron algunas inconsistencias, tales como la existencia de espaciados innecesarios y otros errores de tecleo.

3.1 Resultados

En la experimentación se trabajo con el algoritmo K-Medias y EM y mostro una gran similitud en los resultados obtenidos, es por ello que se muestran las graficas obtenidas con K-Medias.

Las figuras 3, 4 y 5 mostradas en esta sección corresponden a los resultados de los cluster obtenidos de las estaciones de monitoreo Centro, San Lorenzo Tepaltitlán, y San Mateo en los años 2001, 2002 y 2003; en ellas se muestra el comportamiento de las 4 estaciones climáticas del año, invierno, primavera, verano y otoño; en base a los 4 cluster formados con respecto a la temperatura, además se observa la correlación que existe entre la temperatura, la humedad relativa y la radiación solar, en donde se puede destacar que a mayor humedad relativa menor será el incremento de la temperatura y radiación solar; cabe señalar que se generaron una serie de comparaciones con todas y cada una de las variables meteorológicas antes descritas, para determinar el aumento de la temperatura en la zona.



Fig. 3. Zona Centro: periodos 2001, 2002, 2003.



Fig. 4. Zona San Lorenzo Tepaltitlán; periodos 2001, 2002, 2003.



Fig. 5. Zona San Mateo; periodos 2001, 2002, 2003.

Por otro lado, en la Tabla 2 se muestran los porcentajes de los cluster formados, indicando la estación del año que pertenecen, además de identificar cuál fue su temperatura más alta y la temperatura promedio registrada en el año correspondiente marcando un ligero incremento en ellas de 1 grado aproximadamente en cada año.

Tabla 2. Comportamiento de los cluster según su porcentaje y temperaturas anuales.

Centro	Invierno	Primavera	Verano	Otoño	Temperatura mas alta	Temperatura promedio
2001	27%	23%	24%	26%	28.2	14.9
2002	32%	22%	21%	25%	29.4	14.45
2003	25%	23%	22%	30%	30.2	15.95
San Lorenzo Tepaltitlán	Invierno	Primavera	Verano	Otoño	Temperatura mas alta	Temperatura promedio
2001	33%	24%	24%	18%	29.5	14.55
2002	36%	23%	21%	21%	30.2	14
2003	35%	24%	20%	21%	31.4	15.25
San Mateo	Invierno	Primavera	Verano	Otoño	Temperatura mas alta	Temperatura promedio
2001	32%	22%	18%	28%	30	13.45
2002	32%	20%	17%	30%	27.8	11.8
2003	32%	21%	19%	28%	29.9	13.9

Conclusiones y Trabajos Futuros

La ZMVT ha pasado de una economía rural a una industrial y de servicios, aunada a un continuo proceso de urbanización con la consiguiente demanda de servicios y satisfactores a costa de los recursos naturales, provocando como consecuencia diversos problemas ambientales: reducción la frontera agrícola y forestal, invasión de áreas naturales protegidas, deforestación, procesos de erosión, incendios forestales,

quemadas de residuos a cielo abierto, así como las emisiones de contaminantes provenientes de industrias que carecen de sistemas de control y de vehículos automotores con condiciones mecánicas en mal estado, entre otros.

En este trabajo se plantea que la RAMA-ZMVT retoma una gran importancia como estación de monitoreo la cual proporciona información relevante en cuestiones de medición atmosférica y meteorológicas que permiten tomar decisiones que afecten de manera positiva a la comunidad y principalmente mejorar la calidad de los ecosistemas en dicha zona. Para su estudio, se consideraron los datos obtenidos por las estaciones Ce, SL, y SM, en los periodos 2001, 2002 y 2003; con la intención de identificar el comportamiento que la temperatura ha obtenido a lo largo del periodo de estudio y en relación a la ubicación que cada estación de monitoreo tiene.

Derivado de los resultados experimentales, es posible concluir que en la ZMVT se muestra el comportamiento de las 4 estaciones meteorológicas del año que si, efectivamente, están presentes, pero existe una clara tendencia a identificar solo dos, es decir, inviernos-otoños más cortos y templados, y primavera-verano más largos y calurosos. De igual modo, en los resultados obtenidos se observan incrementos en las mediciones de temperatura y de radiación solar con disminución en la humedad relativa de la zona. Estos cambios climáticos alteran la estructura de los ecosistemas propios del Valle de Toluca, un ejemplo son las imágenes captadas en distintos años (Figura 6) [13].



Fig. 6 Cambios visuales del ecosistema en el Valle de Toluca [13].

Por último, el estudio aquí mostrado corresponde a resultados preliminares de un estudio más amplio, por lo que las líneas abiertas de estudio y trabajo futuro comprende la utilización de redes neuronales artificiales para realizar la aproximación del comportamiento meteorológico e incremento de la temperatura han tenido en función del tiempo, además de resolver el problema de datos perdidos mediante la interpolación de datos.

Agradecimientos

Este proyecto fue realizado gracias al apoyo recibido del proyecto 1947/2011 de la UAEM.

Bibliografía

1. Palacios Hernández Francisco. Tesis: Redes Neuronales CMAC como Modelo de Clasificación de Minería de Datos. México. C'INVESTAV. IPN. (2006).
2. Data Mining Software in Java: <http://www.cs.waikato.ac.nz/ml/weka/> Acceso Enero 2010.
3. Chen, M., Han, J., Data mining: An overview from database perspective. IEEE Transactions on Knowledge and Data Eng., (1996).
4. Han, J., Kamber, M.: Data mining: Concepts and techniques. Morgan Kauffmann Publishers. (2001).
5. J. Kittler, Reconocimiento de Patrones de Tratamiento de Imágenes del Instituto de Ingeniería Eléctrica, basado en las notas del curso del Prof. J. Kittler en la Univ. De Surrey. (2002)
6. Programa de las Naciones Unidas para el Medio Ambiente (PNUMA) (2005).
7. Zidjian Aldama Lechuga. Determinación de los Criterios que Definen el Comportamiento del Material Particulado (PM10) en la Zona Industrial del Valle de Toluca. Tesis de Licenciatura, Instituto Tecnológico de Toluca, Toluca México. (2007).
8. Gobierno del Estado de México, Secretaria del Medio Ambiente. Aire Limpio Programa para el Valle de Toluca 2007-2011, Toluca México. (2007).
9. Gobierno del Estado de México, Secretaria del Medio Ambiente, Dirección General de Prevención y Control de la Contaminación Atmosférica, Departamento de Diagnostico. Inventario de Emisiones de la Zona Metropolitana del Valle de Toluca, 2004, Toluca México. (2007).
10. Gobierno del Estado de México.
<http://web.edomexico.gob.mx/portal/page/portal/calidaddelaire>
11. Staines Urías Francisca. Cambio Climático: Interpretando el Pasado Para Entender el Presente. Ensayo Ciencia Ergo Sum, noviembre-febrero, año/vol. 14, número 003 UAEM, Toluca México. (2007-2008)
12. Cuellar R. Fredy, Borja S. Manuel, Sanchez A. Luis, López T. Alejandra. Comportamiento de Partículas Pm10 en las Estaciones Climáticas de la Zona Industrial del Valle de Toluca, Estado de México, Laboratorio de Investigación en Ingeniería Ambiental, Instituto Tecnológico de Toluca, Av. Tecnológico s/n Ex-Rancho La Virgen, Metepec, Estado de México. (2007).
13. Vega Patricia. Se Derrite el Nevado de Toluca, Pierde Agua el Nevado de Toluca, periódico Reforma, México 2008. (2008).
14. Garre Miguel, Cuadrado J. José, Sicilia Miguel A., Rodríguez Daniel, Rejas Ricardo. Comparación de diferentes algoritmos de clustering en la estimación de coste en el desarrollo de software, Dept. de Ciencias de la Computación ETS Ingeniería Informática, Universidad de Alcalá Ctra. Barcelona km 33.6 – 28871 Alcalá de Henares, Madrid, Revista Española de Innovación, Calidad e Ingeniería del Software, Vol.3, No. 1. (2007).

Fermat: Una Red Social de Aprendizaje Inteligente para Matemáticas

Ramón Zatarain-Cabada, M. L. Barrón-Estrada, Jesús Armando Beltrán Verdugo,
Franceli Linney Cibrian Robles, Marsia Irais Quiroz López.

Instituto Tecnológico de Culiacán, Juan de Dios Bátiz s/n, Col. Guadalupe, Culiacán Sinaloa, 80220, México
{rzatarain, lbarron}@itculiacan.edu.mx, {armando.3eltran, Linney11}@gmail.com, marcirais@hotmail.com

Resumen. Las Redes Sociales de Aprendizaje, están cobrando un gran auge en la actualidad, por lo que Fermat, una Red Social de Aprendizaje Inteligente para Matemáticas, integra un Sistema Tutor Inteligente, como una funcionalidad extra que ayuda a los usuarios a mejorar el proceso enseñanza-aprendizaje. El tutor inteligente toma en cuenta tanto los factores cognitivos como los afectivos, y mediante usos de técnicas de inteligencia artificial, brinda a los usuarios una experiencia personalizada y satisfactoria a la hora de tomar los cursos, obteniendo un aprendizaje significativo de los cursos.

Palabras claves: Sistemas Tutor Inteligente, Redes Sociales, Redes Neuronales, Reconocimiento de emociones.

1 Introducción

Las Redes Sociales tienen un gran impacto en la vida cotidiana de muchas personas. En [1] la describe como una estructura de nodos que representan relaciones individuales (o de organizaciones) entre personas de cierto dominio. Por lo que se puede afirmar que, la interacción dinámica que existe les permite compartir diferentes tipos de archivos, comentarios y tópicos [2].

Existen diversos tipos de redes sociales, que se distinguen de acuerdo al enfoque que tienen, por ejemplo las redes sociales de aprendizaje, que utilizan el ambiente colaborativo y herramientas para ayudar a los usuarios en el proceso de aprendizaje. Una herramienta que se puede integrar a las redes sociales de aprendizaje son los Sistemas Tutores Inteligentes.

Los Sistemas Tutores Inteligentes (STI) son programas de computadora que usan una gran cantidad de recursos para soportar el proceso de enseñanza-aprendizaje. Un STI incorpora técnicas de Inteligencia Artificial (IA) y de enseñanza, con el objetivo de crear un ambiente flexible e interactivo que considere los diversos estilos cognitivos de los estudiantes [3]. Los STI juegan un papel importante en el seguimiento tanto del aprendizaje que ha construido el alumno, como en la identificación de las

debilidades, con el objetivo de encontrar estrategias que se adapten al estilo cognitivo del estudiante [3].

En contraste con lo anterior, en la educación formal presencial el tipo de enseñanza es un proceso rígido, donde el profesor no puede adaptarse a los distintos estilos de aprendizaje de cada alumno. Esto trae como consecuencia la deficiencia en el aprendizaje de algunos estudiantes. Como lo demuestran los resultados obtenidos en la Evaluación Nacional de Logro Académico en los Centros Escolares (ENLACE), indican que en México en el año 2010, el 52.6 % de los alumnos de secundaria en la asignatura de matemáticas tiene un aprendizaje insuficiente, mientras que solo el 8.1% de alumnos en primaria, presenta un excelente aprendizaje [4].

Con la finalidad de aminorar el problema de aprendizaje se creó una Red Social de Aprendizaje Inteligente para Matemáticas llamada Fermat (en honor a Pierre Fermat [5]). Fermat contiene un Sistema Tutor Inteligente cuyo principal objetivo es apoyar en el proceso enseñanza-aprendizaje de la educación formal presencial.

Este artículo se organiza de la siguiente manera. En la sección 2 presentamos una arquitectura general de Fermat. La sección 3 da información sobre el análisis y diseño de la red social. La cuarta sección nos explica cómo es que estará constituido el STI. En la sección cinco se hablará sobre las pruebas realizadas a la red social y por último las conclusiones se presentarán en la sección 6.

2 Arquitectura de Fermat

La principal característica de Fermat es su STI, que muestra de manera personalizada el contenido de los cursos a los usuarios, complementando las funcionalidades propias de una red social, como se muestra en la Figura 1.

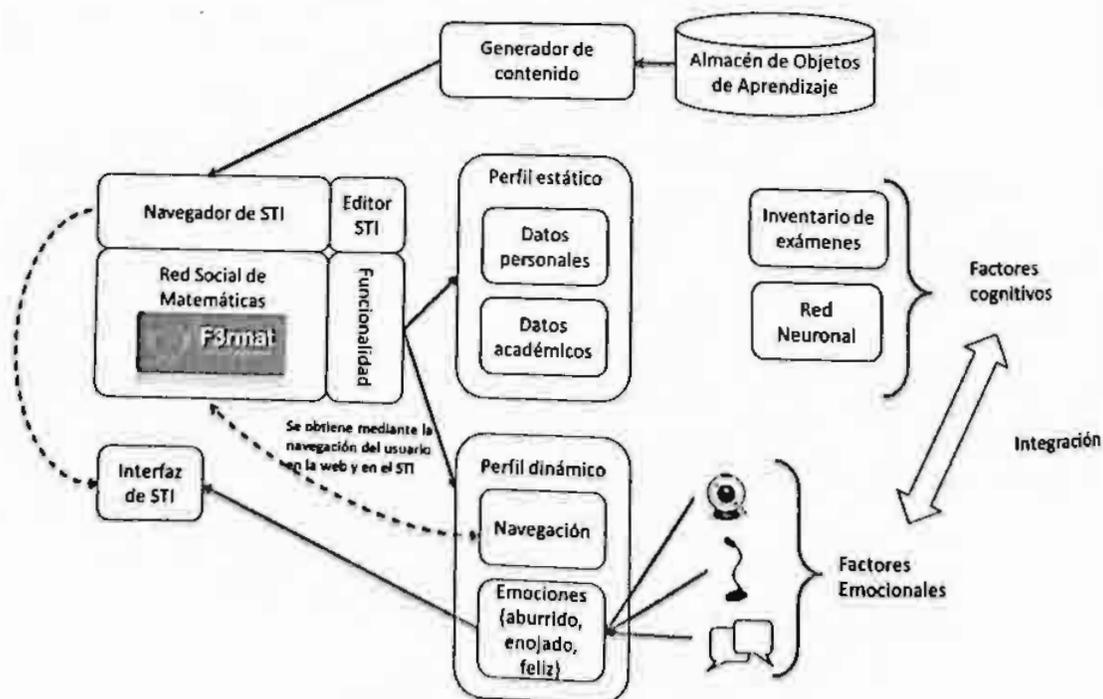


Figura 1. Estructura de Fermat

Los usuarios de la red tienen asociada información personal, académica y de estado de ánimo (perfil), que se obtiene de dos maneras: estática y dinámica. El perfil estático, contiene la información inicial del usuario, como sus datos personales y académicos. El perfil dinámico se actualiza de acuerdo a la interacción del usuario dentro de la red y el STI. En la interacción usuario-Red Social se toman en cuenta los factores cognitivos y los emocionales, esto últimos, de acuerdo con [6], están muy relacionadas con el aprendizaje del estudiante. Por lo tanto, es de vital importancia tomarlo en cuenta junto con los factores cognitivos.

Los factores cognitivos se obtienen de acuerdo al historial de los resultados de exámenes del usuario y el estilo de aprendizaje que va actualizando la red neuronal. Los factores emocionales se adquieren mediante sensores que están monitoreando el estado de ánimo del usuario.

3 Análisis y diseño en Fermat

En la etapa de análisis se determinaron los objetivos y requisitos que tiene la red social, los cuales son fundamentales para entender que características son las que el usuario necesita para poder interactuar y utilizar la red.

Las características fundamentales con las que cuenta Fermat son: Gestionar las cuentas de usuario, gestionar las comunidades, gestionar los cursos, gestionar enlaces de amistades, gestionar envío de mensajes.

Para el diseño de la red social se tomaron en cuenta principalmente cuatro aspectos: modelo de datos, arquitectura, interfaces y componentes.

Los componentes de la red social como: Editor y visor de cursos, administración de los usuarios, objetos de aprendizaje, colaboración de los usuarios en la red y la adaptación de cursos al estilo de aprendizaje. Están relacionados de manera directa para lograr la función principal de la red social, que es crear un aprendizaje colaborativo mediante un sistema tutor inteligente.

Para el diseño de las interfaces se utilizó la herramienta Mockup Builder [7], en la siguiente Figura 2, se muestran los bocetos preliminares que se realizaron para el editor y visor de cursos, como ejemplo del diseño.

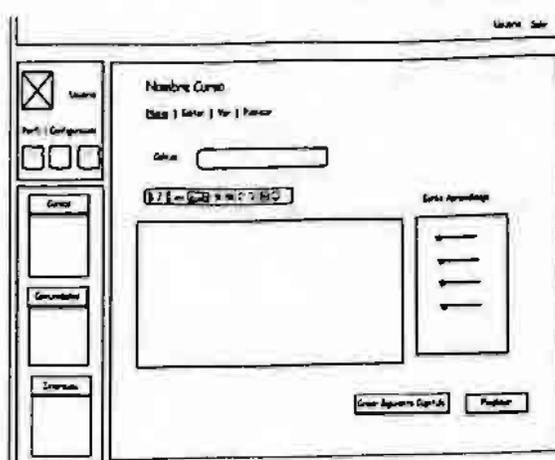


Figura 2. Editor y visor de cursos en Fermat

4 Sistema Tutor Inteligente en Fermat

El sistema tutor inteligente (STI) desarrollado para la red social Fermat, está compuesto por tres componentes, en conjunto son capaces de determinar lo que sabe el estudiante y como va su progreso, ajustándose a las necesidades del mismo [3].

A continuación se numeran los componentes esenciales de un STI:

1. Módulo Experto del conocimiento.
2. Módulo conocimiento del estudiante.
3. Módulo Tutor.

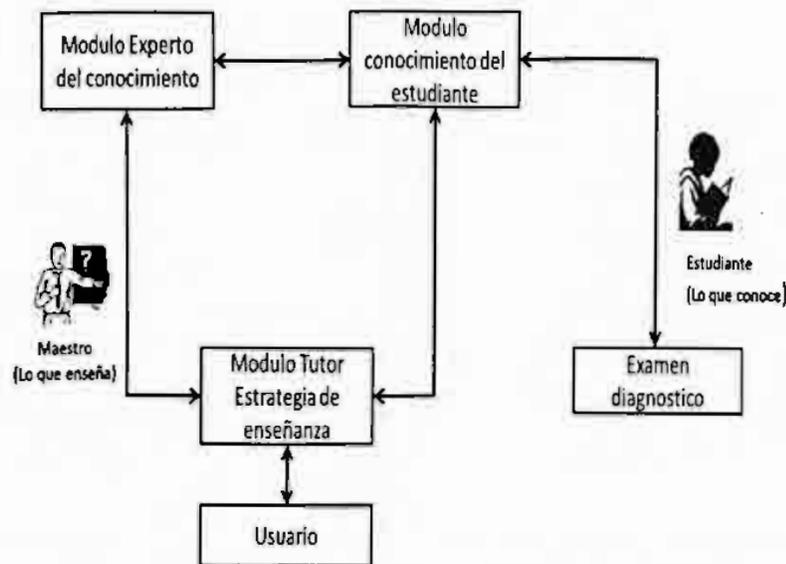


Figura 2. Arquitectura general de sistema tutor inteligente en Fermat

Módulo experto del conocimiento: es aquel que contiene la descripción del conocimiento en la materia de un dominio en particular, proporcionando los conocimientos de lo que se enseña al alumno, para que pueda adquirir las habilidades y conceptos requeridos para un aprendizaje significativo.

Un curso en el STI de Fermat, puede ser visto como un diagrama de árbol, que contiene capítulos y a su vez éstos se componen por temas, como lo muestra en la Figura 3. La totalidad de todos los nodos en el árbol representan el experto en el conocimiento.

Modulo del estudiante: es el encargado de evaluar el rendimiento de los alumnos, para determinar las capacidades cognitivas y habilidades de razonamiento. Proporciona información acerca de lo que sabe el alumno, que es fundamental para el modulo tutor en la selección del estilo de aprendizaje que se adecua al usuario.

En Fermat, la manera en que se percata de cuál es el conocimiento del estudiante, es mediante un examen de diagnóstico. El examen da como resultado lo que el alumno sabe y lo que tiene que reforzar.

El módulo de estudiante, puede ser visto como un subárbol de todo el conocimiento que posee el experto en el dominio como muestra la Figura 3. La representación está basada en un modelo llamado "Overlay", donde el conocimiento del estudiante es un subconjunto del conocimiento del experto, a medida que usa el tutor inteligente el alumno ampliara este subconjunto [8].

Módulo Tutor: Representa las estrategias fundamentales para enseñar los contenidos de un curso dentro de Fermat. Es el responsable de seleccionar el adecuado estilo de aprendizaje y proveer asistencia al estudiante. Es decir, el tutor debe saber responder cuando el alumno no puede contestar una pregunta, o la conteste en forma incompleta.

En Fermat el módulo tutor está basado en el modelo creado por Felder y Silverman, el cual, clasifica las preferencias de los estudiantes en cuatro dimensiones que son: Perecepción, procesamiento, entrada, entendimiento [9]. Un ejemplo es cuando un alumno no puede contestar una pregunta, el tutor le mostrara la explicación de otra forma acerca del tema, tal vez más visual que verbal, más práctico que teórico, etc.

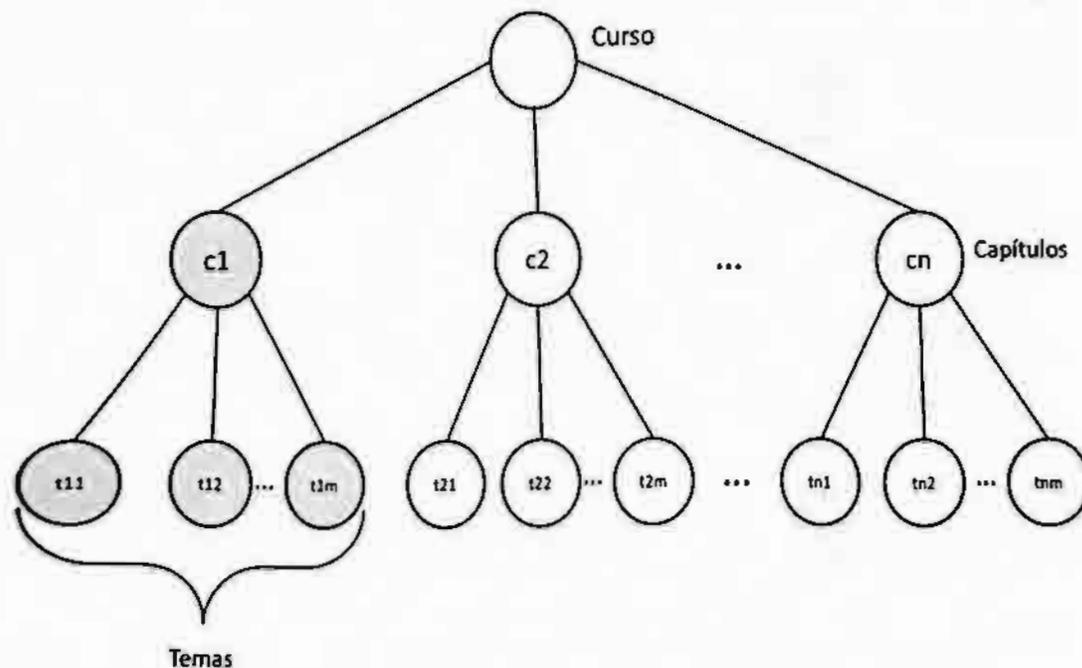


Figura 3. Estructura de los cursos en Fermat. El árbol representa todo el conocimiento del curso. Los círculos de color gris representan lo que el estudiante conoce.

Por otra parte, el tutor integra un módulo que aborda acerca de los estados emocionales de los alumnos en base en su observación, el STI toma algunas medidas que benefician al aprendizaje del estudiante, las emociones se detectaran mediante la expresión del rostro y la voz. A continuación se describe brevemente las manera en cómo se obtendrán dichas emociones.

El método que se utiliza para la detección de emociones visuales está basado en la teoría de Ekman [10], donde se reconocen seis emociones: enojado, disgustado, miedo, felicidad, tristeza, sorpresa.

Para poder determinar la emoción primeramente se toma la imagen y se umbraliza. En base a esta imagen se saca los puntos característicos que permitan minimizar el conjunto de datos de entrada a la red neuronal. La red neuronal que utilizamos son las de "Kohonen" con 20X20 neuronas de entrada y 2 de salida [11].

Para la detección de la emociones con la voz, primeramente se captura la voz a través del micrófono de computadora y se normaliza. A la señal que representa la voz se le aplica la técnica de caracterización de Análisis de componentes principales (ACP). Después mediante el método de SFFS [12] se obtendrá un conjunto óptimo de características, que servirán para alimentar la red neuronal.

En la Figura 4, se puede observar cómo se van integrando las emociones visuales y de sonido mediante una red neuronal. Una vez reconocida la emoción, esta se manda al STI para que reaccione de acuerdo a la emoción, y de ser necesario, cambie su contenido y/o su forma de presentarlo.

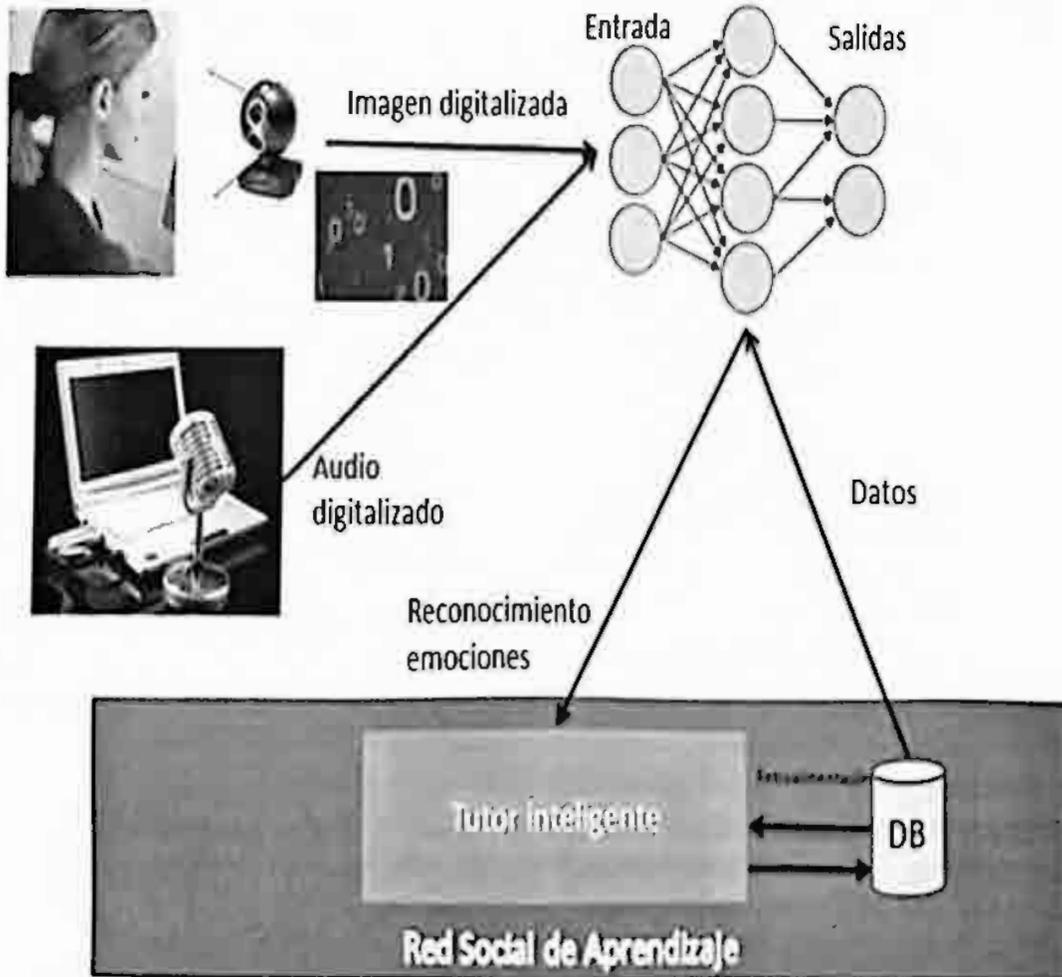


Figura 4. Esquema General del Funcionamiento del Reconocimiento de Emociones y Estilo de Aprendizaje

5 Pruebas

Fermat fue probada con estudiantes de la escuela primaria con alumnos de tercer grado. Durante este periodo se supervisó de manera constante el uso que los estudiantes y el maestro dieron a la red social.

Primeramente los usuarios editaron su perfil, agregaron amigos, se enviaron mensajes. Un maestro creó su una comunidad y agregó a sus alumnos como miembros de ella.

El maestro creó un curso, como se muestra en la Figura 6.

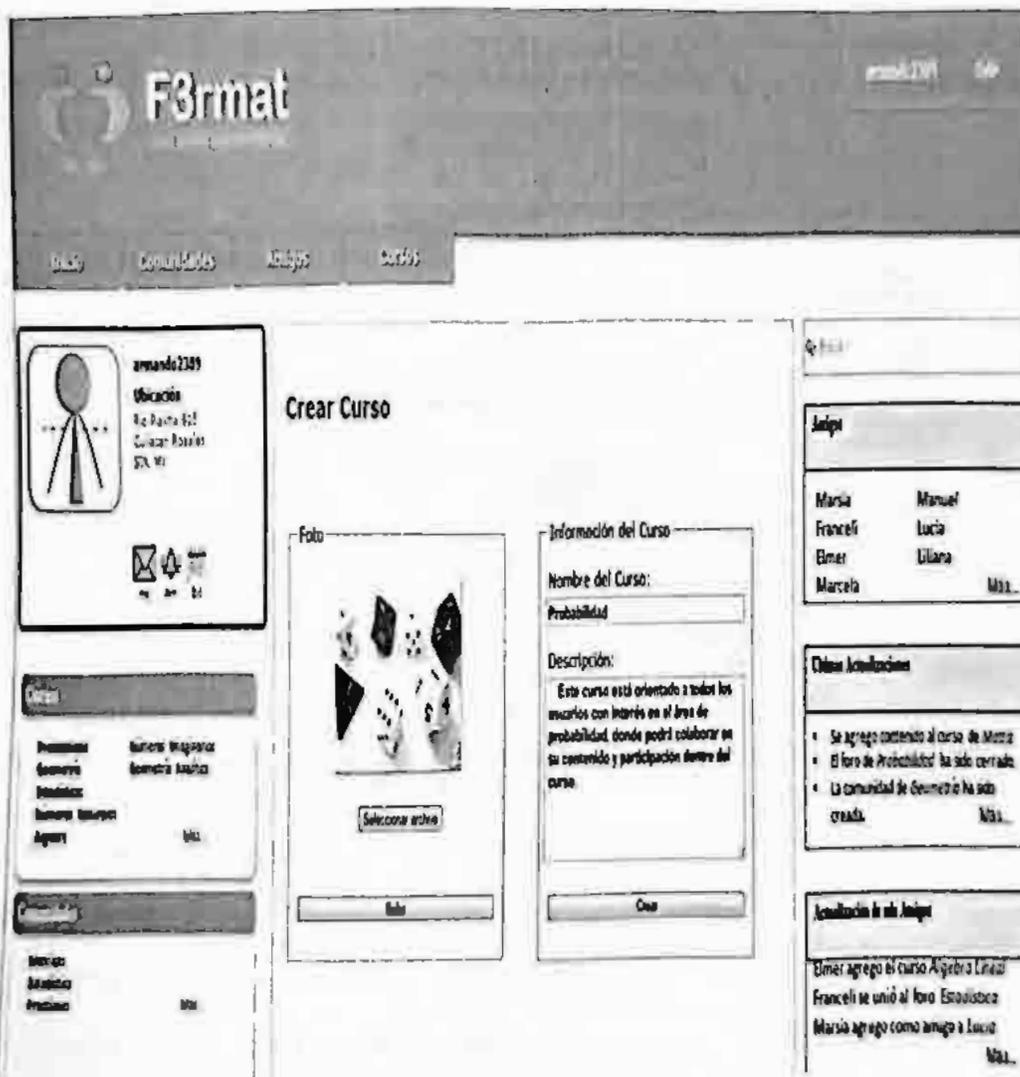


Figura 6. Creación de un curso en Fermat

Los estudiantes, pudieron visualizar el curso como se muestra en la Figura 7.

F3rmat

Agrupamientos de Millares, Centenares, Decenas y Unidades

Para poder contar cifras, hasta la unidad de millares, es importante conocer cuántas unidades, decenas y centenas tiene el número.

Centenas	Decenas	Unidades	Millares
		1	
	1	0	

1 Decena tiene 10 Unidades

Centenas	Decenas	Unidades	Millares
1			
1	0	0	

Siguiente >>

Mis Cursos

- 02 Matemáticas
 - 01 Los Números Naturales
 - 1 Los números Naturales
 - 2 Agrupamiento y Desagrupamiento
 - 3 Lecturas y ejercicios
 - 4 Actividades y ejercicios de la semana
 - 5 Video personal
 - 6 Lecturas y ejercicios de unidades unidades
 - 03 Operaciones
 - 1 Suma
 - 2 Resta
 - 3 Multiplicación
 - 4 División
 - 04 Los Números Enteros
 - 1 Introducción a los Números Enteros
 - 2 Comparación de Números Enteros
 - 3 Características de los enteros
 - 4 Ejercicios de problemas
 - 05 Medios

Inicio Actualizaciones

- Se agregó contenido al curso de Matemáticas
- El foro de Probabilidad ha sido cerrado
- La comunidad de Economía ha sido creada

Actualización de sus Amigos

Emier agregó el curso Álgebra Lineal

Francis se unió al foro Estadística

Marcia agregó como amigo a Lucio

Figura 7. Visualizador de cursos de Fermat

Al finalizar, los estudiantes aplicaron pruebas que actualizaron su subárbol de conocimiento, y el STI, les mostró información nueva en los tópicos que presentó el estudiante algún problema.

Tanto el maestro como los estudiantes quedaron satisfechos con la herramienta.

6 Conclusiones

Las redes sociales son una buena herramienta en donde los usuarios encuentran un espacio de reunión, logrando fomentar las actitudes de colaboración. Agregando a la Red Social un Sistema Tutor Inteligente auxiliamos al profesor con el proceso de enseñanza para que los alumnos obtengan un aprendizaje significativo. Los resultados efectivamente muestran que la Red Social de Aprendizaje Inteligente para Matemáticas Fermat logra auxiliar la educación formal presencial y aminorar el problema de la educación en México. Además el alumno logró sentirse en un ambiente agradable, personalizado, entretenido y sobre todo encontró un espacio para su formación.

7 Referencias

- [1] Liccardi, I.; Ounnas, A.; Pau, R.; Massey, E.; Kinnunen, P.; Lewthwaite, S.; Midy, M.-A. & Sarkar, C. The role of social networks in students' learning experiences *SIGCSE Bull., ACM*, 2007, 39, 224-237
- [2] Calvo Muñoz. Networking, Uso práctico de las redes sociales *Madrid, M. ESIC (Ed.), 2009*
- [3] Stankov, S. ; Glavinic, V.; Rosic M. Intelligent Tutoring Systems in E-Learning Environment: Design, Implementation and Evaluation. Information Science Reference, 2011.
- [4] Secretaria de Educación Pública. (2010). Enlace boletín informativo. [En línea] 2010. [Citado el: 29 de agosto de] http://enlace.sep.gob.mx/ba/docs/boletin_enlaceba2010.pdf.
- [5] Sean, M. The Mathematical Career of Pierre de FERMAT. Princenton University Press. 1994.
- [6] Kort, B., Reilly, R., & Picard, R. W. An Affective Model of Interplay Between Emotions and Learning: Reengineering Educational Pedagogy—Building a Learning Companion. *Proceedings of the IEEE International Conference on Advanced Learning Technologies (ICALT '01)*. IEEE Computer Society, 43-. 2001.
- [7] Mockup Builder @ copyright 2011. <http://mockupbuilder.com/App>
- [8] Günel, K., *Intelligent Tutoring Systems: Conceptual Map Modeling*. LAMBERT Academic Publishing, 2010. p.12
- [9] Graf, S., Rita Viola, S., & Leo, T. Representative characteristics Of Felder-Silverman Learning Styles: An Empirical Model. 2006. 235-242
- [10] Ekman P, Oster, H. Facial expressions of emotion. *Annual Review of Psychology* 30:527-554. 1979.
- [11] Ioanna-Ourania Stathopoulou GAT Visual Affect Recognition. *Frontiers in Artificial Intelligence and Applications*, vol 214. IOS Press. 2010.
- [12] Pudil P, Novovičová J, Kittler J. Floating search methods in feature selection. *Pattern Recognition Letters* 15 (11). 1994.

Clasificación de los métodos para la mejora de WSD y de los métodos de evaluación correspondientes

Alisa Zhila, Alexander Gelbukh

Centro de Investigación en Computación, Instituto Politécnico Nacional, México, D.F.
alisa.zhila@gmail.com, www.gelbukh.com

Resumen. Se presenta una revisión y comparación de los métodos para la evaluación de la desambiguación de los sentidos de las palabras (WSD, por sus siglas en inglés: *Word Sense Disambiguation*). Basándose en esta revisión, se propone una clasificación características de los modelos de WSD que se pueden modificar para mejorar el modelo.

Palabras clave: precisión, recall, inventario de sentidos, traducción automática, procesamiento del lenguaje natural.

1 Introducción

Las palabras en el lenguaje pueden tener varios sentidos, o acepciones: un banco puede ser un mueble, la orilla del río, un conjunto de pesas, o una organización financiera. Tales palabras se llaman *polisémicas*.

En el procesamiento del lenguaje natural (PLN) incluye la tarea de asignar una acepción seleccionada de un inventario de sentidos predeterminado (por ejemplo, un diccionario), a una palabra polisémica en el texto dado. Esta tarea se llama la desambiguación de los sentidos de las palabras (WSD, por sus siglas en inglés: *Word Sense Disambiguation*). Es un proceso muy importante para los sistemas de PLN que realizan el procesamiento del lenguaje a nivel semántico [29]. Sin embargo, es una tarea muy difícil. Aunque los primeros intentos de solucionar este problema se remontan al 1949 [17], la calidad de los métodos actuales no es suficiente [16].

El desempeño de un sistema de WSD se caracteriza por su *cobertura* (*coverage*) C , *precisión* P y *recuerdo* (*recall*) R , los cuales se combinan en la medida acumulativa F :

$$C = \frac{A}{N}, \quad P = \frac{K}{A}, \quad R = \frac{K}{N}, \quad F = \frac{2PR}{P+R},$$

donde N es el número total de las palabras en la entrada, A es el número de las palabras para las cuales el sistema dio alguna respuesta —dado que para ciertas palabras el sistema no propuso ningún sentido: dijo “no sé”— y K es el número de las palabras para las cuales el esta respuesta resultó correcta.

Gale *et al.* [13] estimaron los límites superior e inferior para el desempeño que pueden alcanzar los sistemas de WSD. El límite superior es el grado de acuerdo de anotadores humanos, que es 90%. El límite inferior corresponde al sentido más frecuente: es la proporción de las veces cuando la acepción más frecuente de una palabra es la correcta; es usualmente alrededor del 60% [12].

Hasta el principio de la década de 2000 la WSD fue tratada sobre todo como una tarea aislada, no integrada en alguna aplicación real del PLN. Además, el diccionario WordNet en inglés [11] fue utilizado como el único inventario de sentidos en el área, lo que efectivamente lo convirtió en un inventario de sentidos estándar. Por lo tanto, prácticamente todos los sistemas de WSD se evaluaron en las mismas condiciones, lo que facilitó la comparación del desempeño de los sistemas.

Sin embargo, a partir de mediados de la década de 2000 numerosos autores [1, 2, 16] demostraron que el uso del WordNet como el único inventario de sentidos estándar y la consideración de la WSD como la tarea aislada dirigieron a las soluciones del problema de WSD que se aplicaban mal en las tareas reales del PLN.

Para mejorar el desempeño de los sistemas de WSD, se han sugerido varias medidas: unir las acepciones del WordNet para obtener inventarios de sentidos de granularidad más gruesa [1, 15]; usar las características léxicas de varios idiomas y corpus paralelo [3-6, 18, 19] (este enfoque se conoce como la WSD en varios idiomas o la WSD multilingüe); tratar un problema de WSD como una tarea de traducción de palabras para poder incorporar los módulos de WSD directamente en los sistemas de traducción automática (TA) [2, 8, 9, 10, 27].

Con eso, todas estas modificaciones llevaron a la aparición de ciertas dificultades en la evaluación y comparación del desempeño de los sistemas de WSD, ya que los resultados no eran evaluados con respecto al mismo inventario de sentido estándar.

En este trabajo presentamos nuestra clasificación de las distintas direcciones para la mejora de los métodos para la WSD. A cada una, le corresponde su tipo de evaluación, lo que nos da una clasificación de los métodos para la evaluación de WSD. En las tres siguientes secciones (secciones 2, 3 y 4) analizamos tres diferentes tipos de los métodos o las aplicaciones de WSD, con el propósito de demostrar que en todos ellos se observan las mismas direcciones de mejora y las mismas metodologías de su evaluación. Eso nos lleva a la clasificación que buscamos, la cual la presentamos en la sección 5.

2 Caso de mejoras de la WSD basadas en las modificaciones de los inventarios de sentidos

Uno de los factores que dificultan esta tarea es el hecho de que las acepciones en los diccionarios son discretas, mientras que el significado de una palabra ocupa un diapason continuo [7, 14].

Navigli [15] sostiene que la granularidad de sentidos demasiado fina es uno de los principales obstáculos para el éxito de la WSD. Él presenta un método para reducir el nivel de granularidad del inventario de sentidos WordNet a través del mapeo de las acepciones del WordNet sobre las acepciones de un diccionario con jerarquía semántica elaborada manualmente, el Diccionario Oxford de Inglés.

Como resultado de este mapeo, Navigli obtiene un inventario de sentidos de granularidad gruesa, donde cada acepción corresponde al menos a un sentido del WordNet. La reducción del número de las acepciones resulta el 33,54% (de 60.302 a 40.079 acepciones) y la disminución del grado de polisemia es de 3,14 a 2,09.

Para evaluar el aumento del desempeño de la WSD con el inventario de sentidos de granularidad más gruesa, Navigli ejecuta cinco sistemas de WSD en la tarea

Allwords en inglés de Senseval-3 [20] con el inventario de sentidos nuevo en lugar de WordNet. De esta manera el autor evalúa el aumento del desempeño de los sistemas de WSD elegidos en comparación contra su desempeño con el inventario de sentidos original de granularidad fina.

La tarea Allwords en inglés de Senseval-3 requiere de los sistemas de WSD asignar una acepción de un inventario de sentidos a 2.081 palabras de un conjunto de 301 oraciones de la ficción, noticias y editoriales.

Los sistemas elegidos fueron los tres mejores sistemas de WSD de Senseval-3: GAMBL [21], SenseLearner [22], y KOC [23], y el mejor sistema sin supervisión IRST-DDD [24]. Navigli también incluye un algoritmo de WSD basado en conocimiento que se llama *Structural Semantic Interconnections* [25], ya que éste supera a todos los sistemas que funcionen sin entrenamiento.

Todos los sistemas mostraron un aumento de la medida-F de más de 10% en comparación con sus resultados para la misma tarea con el inventario de sentidos de granularidad fina. El desempeño para la línea base del sentido más frecuente, así como la línea base del sentido al azar también aumentó el 15%.

Snow *et al.* [1] nota que "diferentes aplicaciones de PLN requieren diferentes granularidades de acepciones a fin de aprovechar mejor las diferencias de los sentidos de la palabra, y que para muchas aplicaciones acepciones del WordNet son demasiado finas." En [1] la tarea de la fusión de acepciones se presenta como un problema de aprendizaje supervisado. Como ese tipo de problemas requiere los datos de entrenamiento, los agrupamientos (o los clusters) de acepciones hechos y etiquetados manualmente se usan para entrenamiento. Los autores entrenan un clasificador discriminativo sobre la amplia variedad de características derivadas de la estructura del WordNet, la evidencia basada en corpus, y la evidencia de otros recursos léxicos. Así se obtienen varios WordNets con acepciones agrupadas de granularidad diferente.

Para evaluar el desempeño de la WSD con los inventarios de sentidos de granularidad más gruesa Snow *et al.* aplican un método similar a él de Navigli [15]: "Sabendo las respuestas anteriores de los sistemas en Allwords en inglés de Senseval-3, podemos evaluar los mismos sistemas en el mismo corpus, pero utilizando el inventario de sentidos de granularidad gruesa proporcionado por nuestra taxonomía de acepciones agrupadas". Como prueba de los sistemas se eligieron los 3 sistemas mejores que participaron en Allwords en inglés de Senseval-3: [21, 22, 23].

Snow *et al.* afirman que es obvio que cualquier agrupación de acepciones aumenta el desempeño de los sistemas de WSD. Por lo tanto, en [1] se introduce un inventario de sentidos aleatorio con la agrupación de las acepciones de la misma granularidad que las obtenidas y se realiza la evaluación de cada sistema para estos tres inventarios de sentidos, a diferencia de la comparación hecha por Navigli, donde se compara el desempeño únicamente con los dos inventarios (fino y grueso) de sentidos.

En [1] se muestra que, aunque la WSD con el inventario de sentidos de granularidad más gruesa supera a la WSD con uno de granularidad fina más del 10% de la medida-F, un máximo de sólo el 3,55% (un promedio por los 3 algoritmos de WSD) de mejora de la medida-F se obtiene si se compara contra el inventario de sentidos aleatorio de la misma granularidad. También agregan que este aumento se puede observar sólo con un umbral adecuado para el agrupamiento aleatorio.

La revisión de los artículos sobre el aumento del desempeño de los sistemas de WSD aislados debido al uso de los inventarios de sentidos de granularidad gruesa, muestra que en ese caso para evaluar el desempeño de la WSD se necesitan:

- varios sistemas de WSD existentes,
- un conjunto de datos estándar para el experimento, por ejemplo una tarea Allwords en inglés del Senseval-3,
- los inventarios de sentidos con diferentes granularidades, el de granularidad gruesa siendo derivado del de granularidad fina,
- preferiblemente, un inventario de sentidos aleatorio con la misma granularidad que la del modificado.

Entonces se compara el desempeño de los sistemas de WSD para el mismo conjunto de datos contra diferentes inventarios de sentidos y los resultados de la comparación se presentan como la evaluación de la mejora introducida en un trabajo.

Admitimos que la cantidad de los trabajos observados en esta sección es bastante limitada. Sin embargo, son precisamente los trabajos que demuestran el enfoque más común para la evaluación de resultados en este tipo de investigaciones.

3 Caso de la WSD multilingüe

Una estrecha relación entre la traducción automática (TA) y la WSD puede ser observada a través de dos ramas de los trabajos: textos paralelos mejorando una tarea de la WSD aislada y la WSD mejorando los existentes sistemas de TA. En esta sección analizamos el primer grupo.

Diab [3] analiza la posibilidad de la creación sin supervisión de un corpus etiquetado con sentidos (o acepciones) para las lenguas que carecen de los recursos existentes y su aplicación para WSD supervisada. En su trabajo la autora se basa en el hecho que lexicalización de un concepto, una noción, entre lenguajes tiende a ser consistente, conservando algunos rasgos fundamentales de su semántica, y al mismo tiempo variable a través de las preferencias de traductor, de los contextos y estilos. Diab extrae las características léxicas de los corpus paralelos y las utiliza para aumentar el desempeño de un sistema de WSD supervisado. Para evaluar sus resultados, la autora compara el desempeño de su sistema mejorado con otros sistemas que participaron en el taller de Senseval-2, ejecutándolos en el conjunto de datos estándar Allwords en inglés de Senseval-2.

Ng *et al.* [4], así como Diab, tienen como objetivo reducir el costo de la anotación de los corpus para la WSD supervisada. Sin embargo, su enfoque es algo diferente a lo de Diab. Ellos usan un corpus paralelo de chino-inglés alineado por palabras de forma automática para etiquetar con las correspondientes traducciones un corpus en inglés que se utiliza para el entrenamiento de un sistema de WSD supervisado. En el caso de los sustantivos elegidos para el etiquetado a menudo pasa la fusión de los sentidos que se consideran como acepciones diferentes en el WordNet inglés con la base de la misma traducción en chino.

Se debe notar que este enfoque es opuesto al de Diab ya que ella usa un conjunto de diferentes pero sinónimas traducciones de una palabra polisémica como las características adicionales para la mejor desambiguación de los sentidos de las

palabras. Sorprendentemente, el enfoque de Ng *et al.* es similar a los descritos en la sección 2 debido a la fusión de sentidos.

Por lo tanto, la evaluación de los resultados en [4] se realizó de acuerdo a la metodología de la sección 2: los autores compararon los resultados para los diferentes etiquetados con sentidos –uno con respecto al WordNet inglés original y el otro con respecto a su inventario de sentidos modificado– para un elegido sistema de WSD [26] en un subconjunto estándar de los sustantivos más difíciles de la tarea de la muestra léxica (en inglés: *lexical sample*) inglesa del Senseval-2.

Ide [5] realiza una investigación dirigida a determinar el grado en que los equivalentes de traducción para los diferentes significados de una palabra polisémica inglesa son lexicalizados diferentemente a través de una variedad de idiomas. También determina si esta información puede utilizarse para estructurar o crear un conjunto de distinciones de sentidos útil para las aplicaciones del PLN. Lamentablemente, en este trabajo la autora no investiga la aplicación real de sus resultados a un sistema de WSD, por lo tanto, no se propone una metodología de evaluación de la WSD.

Sin embargo, en los trabajos posteriores Ide *et al.* [18, 19] proporcionan una investigación más profunda de los resultados obtenidos en [5]. Su investigación es similar a [4], ya que los autores explotan las traducciones obtenidos de corpus paralelos para determinar las diferencias en sentidos que pueden utilizarse para el etiquetado de sentidos automático y otras tareas de la desambiguación. En [5] los agrupamientos de las traducciones correspondientes se utilizaron como etiquetas de sentidos para un conjunto de sustantivos polisémicos y compusieron un inventario de sentidos alternativo. El mismo algoritmo del agrupamiento se usó como el algoritmo de WSD. Para el mismo conjunto de sustantivos polisémicos dos anotadores realizaron etiquetado de sentidos manual como un etiquetado de referencia.

Para evaluar los resultados del etiquetado obtenido automáticamente con el algoritmo de WSD elegido, las etiquetas se compararon de forma manual con las proporcionadas por los anotadores humanos para el mismo conjunto de sustantivos. Aunque el porcentaje total de discordancia entre las etiquetas de los sentidos obtenidas automáticamente con las establecidas por los seres humanos es más del 35%, la diferencia entre cada anotador humano y el algoritmo es sólo el 10-13%.

En esta sección para evaluación de los métodos de WSD se han presentado:

- los métodos de WSD referenciales incluyendo el etiquetado manual,
- los conjuntos de datos experimentales estándares, por ejemplo conjuntos de sustantivos polisémicos,
- en el trabajo de Ng *et al.* [4] se usaron diferentes inventarios de sentidos.

En resumen podemos concluir la metodología de la evaluación del desempeño de la WSD depende de las modificaciones concretas realizadas en un trabajo y o depende tanto del tipo de la WSD.

4 Caso de la WSD para la traducción automática

Aunque las tareas de la WSD en varios idiomas y la WSD para la TA pueden parecer similares, la diferencia esencial entre estos dos grupos de los enfoques es que

el primero se centra en el uso de recursos multilingües, que normalmente se utilizan en los sistemas de la traducción automática estadística (SMT, por sus siglas en inglés *Statistical Machine Translation*), para mejorar el desempeño de la WSD, mientras que el segundo se centra en el objetivo de utilizar los modelos de WSD para mejorar la calidad de la traducción.

Vickery *et al.* [2] fueron unos de los primeros en centrarse en el diseño de los sistemas de WSD para el propósito específico de la traducción. En [2] se utilizan los métodos de WSD supervisados y el corpus de los textos paralelos alineados por palabras para resolver la tarea de traducción de una palabra. Luego los autores evalúan sus resultados a través de la tarea de llenar los espacios en blanco con la traducción apropiada, que es esencialmente la evaluación del desempeño de la WSD.

En [2] se comparan cuatro métodos supervisados de WSD. El etiquetado del corpus de entrenamiento con los sentidos se lleva a cabo con los corpus paralelos alineados por palabras, del mismo modo como lo hacen Ng *et al.* [4].

Vickery *et al.* evalúan los resultados de la WSD aislada comparándolos con la línea base del sentido más frecuente. Dado que el objetivo final es medir la mejora de un sistema de TA, proponen una metodología de la evaluación especializada.

En [2] se afirma que los descodificadores disponibles, es decir los sistemas que encuentran la traducción más probable con respecto a algún modelo de probabilidad, actualmente no proporcionan una forma natural de incorporar los resultados de un sistema de la traducción de una palabra. Por lo tanto, para evaluar los resultados se utiliza la tarea de traducción simplificada. La traducción del texto se reduce a la tarea de "llenado de los espacios en blanco": en un texto paralelo se reemplazan las traducciones de las palabras ambiguas con los espacios en blanco. De esta manera la tarea de evaluación de la TA se convierte en la evaluación del desempeño de la WSD.

Resumiendo, a pesar de una formulación diferente de la tarea de Vickery *et al.*, sus métodos de la evaluación son muy similares a los presentados en las secciones 2 y 3.

Specia [8] introduce un nuevo enfoque híbrido de WSD realizado con las técnicas de la programación de la lógica inductiva. El módulo de WSD o, mejor dicho, de la desambiguación de la traducción, introducido tiene el objetivo de aumentar el desempeño de TA, ya que el resultado de este módulo de WSD es una variante de la traducción correcta de una palabra ambigua para un contexto dado. El módulo utiliza las evidencias de los dos idiomas de un par de la traducción y se ha diseñado específicamente para el par portugués-inglés.

Para evaluar el desempeño del módulo, Specia compara el desempeño de su método de WSD contra el desempeño de la WSD de dos algoritmos de aprendizaje – los árboles de decisión y las máquinas de soporte vectorial (en inglés: *Support Vector Machines*, SVM) – en un conjunto de datos estándar.

A pesar de que según Specia este enfoque tiene como objetivo mejorar la TA, ninguna introducción real de su método de WSD en un sistema de TA se llevó a cabo. Por lo tanto, ningunos métodos para la evaluación desarrollados específicamente para la WSD-para-la-TA no se han presentado.

Carpuat y Wu realizaron una serie de trabajos [9, 27–28] en la integración de los resultados de la WSD en los sistemas de SMT. En [9] y [28] efectivamente se pone en práctica la incorporación de un módulo de WSD en un sistema de SMT.

En su trabajo de 2005 [28] Carpuat y Wu muestran que los modelos de SMT actuales tienen limitaciones en la WSD, en comparación con los modelos dedicados a

la WSD, y que la SMT debería beneficiarse de las predicciones hechas por los módulos de WSD.

Para aclarar, en [28], no se introduce un método de WSD novedoso ni se mejoran los existentes. En cambio, se analizan las características de los modelos de SMT relevantes para la tarea de WSD y se experimentan con aplicaciones de los sistemas de SMT a la tarea de WSD.

La evaluación del desempeño se realiza en un conjunto de datos estándar de la tarea de muestra léxica china del Senseval-3. Los modelos de WSD supervisados se entrenaron con el inventario de sentidos HowNet. Con el fin de evaluar las predicciones del modelo de SMT al igual que cualquier modelo de WSD, los autores tuvieron que mapear las traducciones inglesas sobre las acepciones de HowNet.

Resumiendo, los autores lograron reducir sus tareas de la evaluación a la tarea de la comparación de diferentes sistemas de WSD en un conjunto de datos estándar con un inventario de sentidos mutuo.

Su otro trabajo de 2005 [27] ya presenta una integración real de los resultados de la WSD en un sistema de SMT. Este trabajo proporciona un resultado decepcionante: el uso de un modelo de WSD del estado-del-arte en chino para elegir los candidatos de la traducción para un sistema típico de SMT de IBM *no* mejora la calidad de la traducción considerablemente comparando con la del sistema de SMT por sí solo.

En [27] se presentan dos enfoques a la integración de los resultados de la WSD en un sistema de SMT. En el primer enfoque, las predicciones del sentido hechas por el algoritmo de WSD se usan para restringir el conjunto de los posibles sentidos en inglés considerados por el decodificador para cada una de las palabras elegidas para el experimento. En el segundo, las predicciones de la WSD se usan para el pos-procesamiento de la salida del sistema de SMT: en cada frase de salida, la traducción de la palabra observada es reemplazada directamente por la predicción de la WSD. Cuando el sistema de WSD predice más de un candidato, solo una traducción se elige al azar. En ambos casos se utiliza el mismo sistema de WSD basado en el modelo que logró el mejor desempeño en la tarea de la muestra léxica china del Senseval-3, Kernel PCA. El sistema de SMT también no se cambia durante todo el experimento.

Dado que la tarea de Carpuat y Wu en [27] no fue mejorar algún modelo de WSD sino investigar la influencia de la WSD en la calidad de la SMT, los autores emplean un método de evaluación utilizado para evaluar la traducción automática: el sistema de la evaluación de la traducción automática BLEU. Los autores comparan los resultados de las traducciones realizadas por el elegido sistema de SMT por sí solo y por el mismo sistema de SMT con un módulo de WSD integrado.

El último trabajo de la serie de los trabajos de Carpuat y Wu en el área de la integración de los resultados de la WSD en un sistema de SMT que revisamos es el trabajo [9] de 2007. En [9] se muestra que WSD sí mejora la calidad de la traducción de un modelo de SMT típico basado en frases. Sin embargo, para lograr esta mejora, los autores tuvieron que redefinir la tarea de WSD para que coincidiera exactamente con la misma tarea con la cual se enfrentan los sistemas de SMT basados en frases —a saber, la desambiguación de la traducción de las frases.

Por lo tanto, en este trabajo los autores efectivamente presentan un modelo de WSD novedoso, que ellos llaman “el modelo de la desambiguación del sentido de las frases completas compuestas por varias palabras”. En este modelo, en lugar de considerar una sola palabra como un objeto de la WSD, se desambiguan las frases

enteras con un vocabulario de la SMT como el inventario de sentidos. En este caso el modelo de WSD utiliza las mismas definiciones de sentidos y los datos de entrenamiento que el modelo de SMT.

La evaluación se lleva a cabo con dos estándares: tareas de traducción de chino a inglés. Dado que el objetivo de Carpuat y Wu no es evaluar el desempeño de su modelo de WSD modificado, sino la calidad de la traducción, ellos utilizan la metodología estándar de evaluación de la TA y no evalúan la exactitud del modelo de WSD por sí sola. A diferencia de [27], donde se usa sólo una métrica estándar de evaluación de traducción automática BLEU, en [9] los resultados de la SMT se evalúan con ocho métricas diferentes.

Chan y Ng [10], al igual que Carpuat y Wu [27], incorporan un sistema de WSD del estado-del-arte en un sistema de SMT del estado-del-arte, que se llama Hiero. Pero a diferencia de [27] Chan y Ng logran obtener la mejora considerable en el desempeño del sistema de SMT en una tarea real de traducción.

Ya que su trabajo también tiene como objetivo mejorar la calidad de la TA en lugar de aumentar la precisión de la WSD, para evaluar sus resultados Chan y Ng usan la métrica de evaluación de la TA más popular: BLEU.

Resumimos, que la evaluación de los resultados de los trabajos en la rama de WSD para TA puede realizarse en dos modos:

- evaluar el desempeño del modelo de WSD directamente usando los métodos descritos en las secciones 2 y 3,
- evaluar la calidad final de la traducción de los sistemas de TA modificados con los módulos de WSD por medio de las métricas de evaluación de la TA estándares como BLEU.

5 Discusión y las conclusiones

La amplia comparación de una variedad de los trabajos en la mejora del desempeño de la WSD hecha en el artículo presente nos permite obtener varias conclusiones.

Basándonos en la revisión presentada en el artículo, podemos afirmar que las formas de la evaluación de la mejora de un modelo de WSD dependen de las modificaciones que llevan la dicha mejora y de objetivos del uso de la WSD. Los tipos de modificaciones de los modelos de WSD pueden ser:

- la modificación del mismo algoritmo de WSD o introducción de un modelo o algoritmo de WSD novedoso como en los trabajos [2, 3, 8, 27],
- la modificación del inventario de sentidos contra el que se realiza la desambiguación como en los trabajos [1, 4, 15, 18, 19],
- varias modificaciones en el modelo de WSD con el fin de mejorar el desempeño de algún sistema de PLN más general, por ejemplo, los sistemas de TA, como en los trabajos [2, 9, 10, 27].

Se debe notar que no todos los trabajos, que clasifican sus modificaciones de una cierta manera, de hecho realizan la modificación declarada. Como podemos ver del análisis de la investigación conducida en el trabajo [8], la autora declara su modelo de WSD como uno diseñado especialmente para la TA, pero finalmente no propone

método para integrarlo en un sistema de SMT ni metodología de la evaluación de la mejora de la traducción debida a su modelo de WSD.

Diferentes tipos de modificaciones de la WSD requieren diferentes metodologías de la evaluación.

1. Así los modelos de WSD que modifican el mismo algoritmo de WSD o introducen un modelo o algoritmo de WSD novedoso para evaluarse deben comparar su desempeño contra los mejores métodos de WSD en la misma tarea (un conjunto de datos estándar) con el mismo inventario de sentidos para todos los métodos.
2. Los modelos de WSD que modifican el inventario de sentidos deben comparar el desempeño de por lo menos un algoritmo de WSD existente usando el inventario original y modificado en un conjunto de datos estándar.
3. Los modelos de WSD que tienen como su objetivo la mejora de un sistema de PLN más general deben emplear los métodos para medir el desempeño específicamente de ese sistema.

En todos los puntos también es preferible incluir como una posición de la comparación una modificación aleatoria con las mismas características como lo hacen Snow *et al.* [1].

Agradecimientos. Este trabajo fue realizado con el apoyo parcial del Gobierno de México (SNI, COFAA-IPN, PIFI-IPN, SIP-IPN 20113295 y 20111146, CONACYT 50206-H), CONACYT-DST India (proyecto "Answer Validation through Textual Entailment"), Gobierno del DF, México (ICYT PICCO10-120), Proyecto Europeo WIQ-EI 269180.

6 References

1. Snow, R., Prakash, S., Jurafsky, D., y Ng, A. Y.: Learning to Merge Word Senses. EMNLP-CoNLL 2007, pp. 1005–1014. ACL (2007)
2. Vickrey, D., Biewald, L., Teyssier, M. y Koller, D.: Word-sense disambiguation for machine translation. EMNLP 2005, pp. 771–778. ACL (2005)
3. Diab, M.: Word sense disambiguation within a multilingual framework. Ph.D. dissertation. University of Maryland, College Park, College of Park, MD (2003)
4. Ng, H. T., Wang, B., y Chan, Y. S.: Exploiting parallel texts for word sense disambiguation: an empirical study. ACL 2003, vol. 1, pp. 455–462. ACL (2003)
5. Ide, N.: Cross-lingual sense determination: Can it work? En: *Computers and the Humanities*, vol. 34, 1–2, pp. 223–234. Springer (2000)
6. Ide, N.: Making senses: Bootstrapping sense-tagged lists of semantically-related words. En: Gelbukh, A. (Ed.). *Computational Linguistics and Intelligent Text Processing*, LNCS, vol. 3878, pp. 13–27. Springer (2006)
7. Ide, N. y Wilks, Y.: Making sense about sense. En: Agirre, E. y Edmonds, P. (Eds.) *Word Sense Disambiguation: Algorithms and Applications*, pp. 47–73. Springer (2006)
8. Specia, L.: A hybrid relational approach for WSD—first results. COLING/ACL 2006 Student Research Workshop, pp. 55–60. ACL (2006)
9. Carpuat, M. y Wu, D.: Improving statistical machine translation using word sense disambiguation. EMNLP-CoNLL 2007, pp. 61–72. ACL (2007)

10. Chan, Y. S. y Ng, H. T.: Word sense disambiguation improves statistical machine translation. *ACL 2007*, pp. 33–40. *ACL (2007)*
11. Stark, M. M. y Richard F. Riesenfeld, R.F.: *WordNet. An electronic lexical database*. En: *Memorias de 11th Eurographics Workshop on Rendering*. MIT Press (1998)
12. Agirre, E. y Edmonds, P.: *Word Sense Disambiguation: Algorithms and Applications*. Springer (2006)
13. Gale, W.A., Church, K., y Yarowsky, D.: Estimating upper and lower bounds on the performance of word-sense disambiguation programs. En: *Memorias de the 30th Annual Meeting of the Association for Computational Linguistics*, pp. 249–256. *ACL (1992)*
14. Kilgariff, A.: Word senses. En: Agirre, E. y Edmonds, P. (Eds.) *Word Sense Disambiguation: Algorithms and Applications*, pp. 29–46. Springer (2006)
15. Navigli, R.: Meaningful clustering of senses helps boost word sense disambiguation performance. En: *Memorias de the 44th Annual Meeting of the Association for Computational Linguistics joint with the 21st International Conference on Computational Linguistics*, pp. 105–112. *ACL (2006)*
16. Navigli, R.: *Word Sense Disambiguation: a Survey*. En: *ACM Computing Surveys*, vol. 41 (2), pp. 1–69. ACM Press (2009)
17. Weaver, W.: *Translation*. En: Locke, W. N. y Booth, A.D. (Eds.) *Machine Translation of Languages: Fourteen Essays*. MIT Press, Cambridge (1949)
18. Ide, N., Erjavec, T. y Tufis, D.: Automatic sense tagging using parallel corpora. En: *Memorias de 6th Natural Language Processing Pacific Rim Symposium*, pp. 212–219 (2001)
19. Ide, N., Erjavec, T. y Tufis, D.: Sense discrimination with parallel corpora. En: *Memorias de ACL Workshop on Word Sense Disambiguation: Recent Successes and Future Directions*, pp. 54–60. *ACL (2002)*
20. Snyder, B. y Palmer, M.: The English all-words task. En: *Memorias de ACL 2004 Senseval-3 Workshop*, pp. 41–43. *ACL (2004)*
21. Decadt, B., Hoste, V., Daelemans, W., Bosch, A.: Gambl, genetic algorithm optimization of memory-based WSD. En: *Memorias de ACL/SIGLEX Senseval-3*, pp. 108–112. (2004)
22. Mihalcea, R. y Faruque, E.: Senselearner: Minimally supervised word sense disambiguation for all words in open text. En: *Memorias de ACL/SIGLEX Senseval-3*, 155–158. (2004)
23. Yuret, D.: Some experiments with a naïve bayes wsd system. En: *Memorias de ACL/SIGLEX Senseval-3*, pp. 265–268. *ACL (2004)*
24. Strapparava, C., Gliozzo, A., Giuliano, C.: Pattern abstraction and term similarity for word sense disambiguation, 229–234. En: *Memorias de ACL/SIGLEX Senseval-3*. *ACL (2004)*
25. Navigli, R. y Velardi, P.: Structural semantic interconnections: a knowledge-based approach to word sense disambiguation. En: *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 27(7), pp. 1075–1086. *IEEE (2005)*
26. Lee, Y. K. y Ng, H. T.: An empirical evaluation of knowledge sources and learning algorithms for word sense disambiguation. En: *Memorias de the 2002 Conference on Empirical Methods in Natural Language Processing*, pp. 41–48. *ACL (2002)*
27. Carpuat, M. y Dekai Wu, D.: Word sense disambiguation vs. statistical machine translation. En: *Memorias de the annual meeting of the association for computational linguistics*, pp. 387–394. *ACL (2005)*
28. Carpuat, M. y Dekai Wu, D.: Evaluating the word sense disambiguation performance of statistical machine translation. En: *Memorias de the Second International Joint Conference on Natural Language Processing (IJCNLP)*, pp. 122–127. *IJCNLP (2005)*
29. Ledeneva, Y. y Sidorov, G. *Recent Advances in Computational Linguistics*. *Informatica. International Journal of Computing and Informatics*, 34 (2010) pp. 3–18 (2010)

Indice de Autores

A

Acevedo Ávila, Ricardo	1
Aguilat Lasserre, Alberto Alfonso	18
Alejo, Roberto	213
Ariel Lucien, García Gamboa	11

B

Barceló-Aspeitia, Axel	59
Barrón, Lucía	223
Beltrán, Armando	223
Bertani, David	18
Brena Pinero, Ramón F.	103
Buendía, Barbara	96
Buzón-Cantera, Ivo Erasmo	28

C

Caballero-Caballero, Sergio Alex	28
Calderon, Dario	213
Calderón-Chávez, Alfonso	11
Calderón-Chávez, Francisco Javier	11
Carvajal-Gámez, Blanca E.	40
Castro, Félix	50
Castro-Manzano, José Martín	59
Chirinos, Raúl	50
Cibrian, Franceli	223
Cid, José Alejandro	69
Cota Ortiz, Guadalupe	195
Cruz Reyes, Rafael	123

D

Dominguez Hernandez, Reyna Karem	18
Díaz Zagal, Sergio	123
Díaz, Cristina	79

E

Escobar-Acevedo, Adelina	88
Espíndola Cisneros, Patricia	18

F

Franco, Oscar	96
---------------	----

G

Gallegos-Funes, Francisco J.	40
Gallegos-Funes, Francisco Javier	185
García Ceja, Enrique A.	103
García García, Andrés David	1
García Hernández, René Arnulfo	123
Gelbukh, Alexander	232

Gonzalez Sanchez, Blanca Estela	18
Gonzalez Zeta, Alejandro	123
Gonzalez, David	113
Gonzalez-Mendoza, Miguel	1, 11, 113, 167, 205
Guerra-Hernández, Alejandro	59
Guevara Cruz, María Esther	132
Gutierrez, Susana	142
Gutiérrez Estrada, Citlalilh	123
Gutiérrez-Ayala, Salvador Antonio	113
Guzmán-Cabrera, Rafael	88
H	
Hernandez, Dante	152
Hernandez-Gress, Neil	113, 167
Hernández Gress, Neil	205
Herrera, Oscar	79, 157
I	
Ibarra-Orozco, Rodolfo Ibarra-Orozco	167
J	
Jimenez, Marco	142
L	
Ledeneva, Yulia	123
Lizarraga, Giovanni	142
Lopez-Pimentel, Juan Carlos	167
López, Fernando	69
López-Bonilla, José L.	40
M	
Martínez Trinidad, José Francisco	132
Mata Mata, Victor	177
Medina Celaya, Mario Alberto	195
Montes-Y-Gómez, Manuel	88
Mora, Román	157
Mora-Vargas, Jaime	28
Morales Escobar, Saturnino Job	132
Mujica, Dante	185
P	
Pon-Gomez, Chistian	167
Q	
Quiroz, Marsia	223
R	
Ramos, Victor	167
Rodríguez Pérez, Ivonne	132
Rojas López, Pedro Alfonso	195
Rosales-Silva, Alberto J.	40
S	
Suárez, Joel	50

T	
Tabarez Paz, Israel	
Torres Arellano, Juan Carlos	205
V	213
Valdovinos Rosas, Rosa Maria	
Villaseñor-Pineda, Luis	213
Vázquez-Palco, Hugo	88
Z	113
Zatarain, Ramon	
Zhila, Alisa	223
	232

Creado y producido por la Sociedad
Mexicana de Inteligencia Artificial

Impreso en los talleres gráficos de Ágora
Medios, S.A. de C.V., Priv. de Hidalgo #6,
Col. Santiago Miltepec, Toluca, Estado de
México, C.P. 50020. Tel.: 01(722) 272 74 30.

Octubre de 2011, Edición 500 ejemplares.

SMLA 25 Aniversario



TECNOLÓGICO
DE MONTERREY.



Casa abierta al tiempo
UNIVERSIDAD AUTÓNOMA METROPOLITANA



ISBN: 978-607-95367-3-2



9 786079 536732