Avances en la Ingeniería del Lenguaje y del Conocimiento

Research in Computing Science

Series Editorial Board

Research in Computing Science es una publicación trimestral, de circulación internacional, editada por el Centro de Investigación en Computación del IPN, para dar a conocer los avances de investigación científica y desarrollo tecnológico de la comunidad científica internacional. **Volumen 88,** diciembre de 2014. Tiraje: 500 ejemplares. *Certificado de Reserva de Derechos al Uso Exclusivo del Título* No. : 04-2005-121611550100-102, expedido por el Instituto Nacional de Derecho de Autor. *Certificado de Licitud de Título* No. 12897, *Certificado de licitud de Contenido* No. 10470, expedidos por la Comisión Calificadora de Publicaciones y Revistas Ilustradas. El contenido de los artículos es responsabilidad exclusiva de sus respectivos autores. Queda prohibida la reproducción total o parcial, por cualquier medio, sin el permiso expreso del editor, excepto para uso personal o de estudio haciendo cita explícita en la primera página de cada documento. Impreso en la Ciudad de México, en los Talleres Gráficos del IPN – Dirección de Publicaciones, Tres Guerras 27, Centro Histórico, México, D.F. Distribuida por el Centro de Investigación en Computación, Av. Juan de Dios Bátiz S/N, Esq. Av. Miguel Othón de Mendizábal, Col. Nueva Industrial Vallejo, C.P. 07738, México, D.F. Tel. 57 29 60 00, ext. 56571.

Editor responsable: *Grigori Sidorov, RFC SIGR651028L69*

Research in Computing Science is published by the Center for Computing Research of IPN. **Volume 88**, December 2014. Printing 500. The authors are responsible for the contents of their articles.

Printed in Mexico City, in the IPN Graphic Workshop – Publication Office.

Volume 88

Avances en la Ingeniería del Lenguaje y del Conocimiento

David Pinto
Darnes Vilariño (ed.)

Instituto Politécnico Nacional, Centro de Investigación en Computación
México 2014

ISSN: 1870-4069

Instituto Politécnico Nacional (IPN)
Centro de Investigación en Computación (CIC)
Av. Juan de Dios Bátiz s/n esq. M. Othón de Mendizábal
Unidad Profesional "Adolfo López Mateos", Zacatenco
07738, México D.F., México

http://www.rcs.cic.ipn.mx
http://www.ipn.mx
http://www.cic.ipn.mx

Indexed in LATINDEX and Periodica / Indexada en LATINDEX y Periódica

Printing: 500 / Tiraje: 500

Printed in Mexico / Impreso en México

Prefacio

El simposio en Ingeniería del Lenguaje y del Conocimiento (LKE'2014) es la segunda edición de esta serie de eventos. Esta conferencia ha sido organizada en el seno de la Facultad de Ciencias de la Computación de la Benemérita Universidad Autónoma de Puebla (BUAP) por dos años consecutivos. Nace como una iniciativa del laboratorio de Ingeniería del Lenguaje y del Conocimiento con la finalidad de ofrecer un espacio académico y de investigación en el cual sea posible reportar trabajos relacionados con el área.

Por supuesto que este evento promueve la cooperación entre diferentes grupos de investigación pues permite el intercambio de resultados científicos, prácticas y nuevo conocimiento.

El presente volumen contiene una serie de contribuciones originales que han sido seleccionadas a partir de un proceso de evaluación ciega doble (*double blind*), lo cual significa que los nombres de los autores de los artículos y los nombres de los revisores son ambos desconocidos. Este procedimiento es ejecutado en aras de proveer una evaluación anónima que derive en artículos de mayor calidad para este volumen; particularmente, en esta ocasión la tasa de rechazo fue del 26%, cuidando que en todos los casos, al menos dos especialistas del comité revisor hicieran una evaluación de la pertinencia, originalidad y calidad de cada artículo sometido.

Esperamos que este volumen sea de utilidad para el lector y que este segundo simposio en sí mismo sea un espacio de intercambio científico productivo que enriquezca la colaboración entre estudiantes y académicos en el ámbito de la ingeniería del lenguaje y del conocimiento.

El proceso de revisión y selección de artículos se llevó a cabo usando el sistema libremente disponible llamado EasyChair, http://www.easychair.org.

Diciembre 2014

David Eduardo Pinto Avendaño
Darnes Vilariño Ayala

Table of Contents

Page

Spatial-Temporal Model for Emotion Interpretation

Lucio C. Vázquez S.[1], Ivo H. Pineda T.[2], María J. Somodevilla[2], Concepción Pérez de Celis H.[2], and Mario Rossainz L.[2]

Benemérita Universidad Autónoma de Puebla, Fac. Ciencias de la Computación, Puebla, México.
[1]luciovs@prodigymovil.com,
[2]{ipineda,mariasg,cpelish,mrossainz}@scs.buap.mx

Abstract. Interpreting facial expressions is a task that humans perform everyday automatically. Most of gesture recognition research are focused to discriminate different facial expressions, while interpreting them is relatively a new area. This paper deals with how to recognize an emotional state, since it can be represented as states at a time t that are influenced directly by a previous state at $t-1$ therefore exists an inherent temporarily.

Keywords: Hiden Markov Models,spatial temporal variability, face tracking, emotion identification , emotion interpretation

1 Introduction

From the perspective of computer science the idea that a secuence of facial gestures might represent a complex emotional state which can be identified automatically , this is precisely the motivation of this research work. An example would be analyze a sequence of facial gestures during a job interview to determine if the interviewer is lying or even determine a psychological condition more complex. Another example is the problem of lie detection, facial gestures proven valuable information for determining whether a person is lying or not. Psychologists have concluded that no single gesture represents a lie, it is involved a sequence of gestures which would help to identify when a person lies. Human gestures are a powerful source of communication and represent an unconscious emotional response many times. The human being is capable of creating a number of gestures that often follow patterns given by culture, geographical location, etc. Although a group of people share a set of gestures to communicate subtle differences.

Interpret facial expressions is a task that humans perform every day automatically during the communication process either verbally or not, almost regardless of lighting conditions or perspective that is taken of the face. In contrast, the gesture recognition systems are sensitive to these conditions. Previous work mainly consisted on to discriminate gestures such as anger, joy, sadness, disgust,anger and surprise in a single frame. However some human emotions such as surprise

merges into fear,amusement, relief,anger , disgust depending upon what it was that surprised us. So determine it was a pleasant or an unpleasant surprise could be interesting.This research deals with how to recognize an emotional state, since it can be represented as states at a time t that are influenced directly by a previous state at $t-1$ therefore exists an inherent temporality.

2 Previous Work

Facial gestures are result of evolution and natural selection [2] in response to various situations. According to Ekman [3] the facial gestures are strongly related to emotions that are expressed unconsciously. Ekman [6] differs in functionality of gestures together with facial gestures to Frilund, he considered that gestures are oriented messages it might reveal the intention of adopting a behavior.

Perhaps its origin is not as important as their interpretation; Ekman have developed methodologies for the analysis of gestures from the observation, concluding that a person can be trained and be able to interpret them correctly. Further analysis show that some people can fake a suggested expression: happy, sad or angry, but they do not now how to emerge suddenly, how long to keep it, or make it disappear in that instant. Over a thousand different facial gestures are anatomically possible , but only a few have a real sense according to Ekman.

Ekman's work consisted on to interpret facial gesture correctly, one of his most important contributions [4] addresses the problem of detecting when a person lies. This analysis is very comprehensive because it includes changes in the face, body movements, tone and speed of speech. Ekman found several criteria for determining whether an emotion was being sincere or was real. In one experiment, Ekman found that over a period of 1/25 seconds or so people in some situations show a facial gesture so quickly that an untrained person is almost impossible to see. Ekman called these gestures *micro expressions* that provides leakage of a concealed emotion.*Squelched expressions* are much more common.Sometimes when an expression emerges it is interrupted and also covered with another expression. Smile is the most common cover or mask [4].

Facial Action Coding System (FACS) [5] shows how to classify muscular activity during facial gestures. This changes are called Action Unis (AU).AUs are grouped into AUs in the Upper Face (eyebrows,forehead,eyelids) and Lower Face AUs (up/down,horizontal,oblique,orbital,miscellaneous).
There is an important difference between facial expression recognition and human emotion recognition, while facial recognition classifies into abstract classes or labels of the deformations caused by facial muscle movement.Human emotion recognition are result of a variety of factors such as voice's tone, posture, gestures or even facial expressions. Basically emotion recognition is an attempt to understand a situation including its context. This research works on digital images of a face in order to interpret facial expressions and its temporal context. Figure 1 from B. Fasel and J. Luettin work [7], shows the most widely used architecture in computer science for facial gesture recognition.

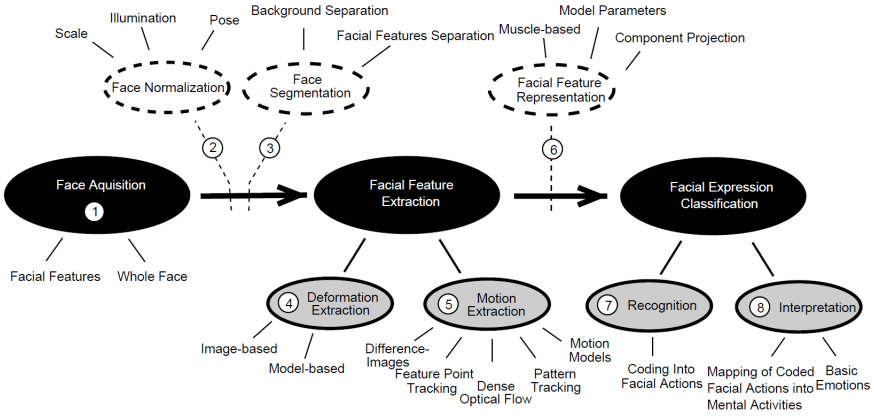


Fig. 1. Generalization of a facial gesture analysis. [7]

There is more of one approach to gesture recognition for instance using mathematical models that incorporate changes in form and lighting. This includes the features geometry of iris or nostrils, the position of these characteristics determine the location of the face. Probabilistic techniques such as the [13] are also mentioned in the literature. Other authors report the use of active contour models [8] , wavelets [10] and rule-based techniques such as FACS [5] among others .

In 2007 S.Mitra and T. Acharya [11] presented a complete revision of the state of the art concerning the recognition of gestures involving hands, arms, face, head and body in general. Based on [11] gesture recognition is divided into three categories such as:

– Hand and arm gestures.
– Head and face gestures.
– Body gestures.

Most of the tools used for gesture recognition, use statistical modeling, computer vision and/or pattern recognition. Most of the problem has been solved using statistical modeling, specially PCA, HMM, Kalman filter and even finite state machines. According to this it is concluded that there are four major approaches:

– HMM
– Filtering particles and condensation algorithm.
– Finite State Machines.
– Soft Computing and connectionist approaches.

3 Methodology

The main goal of this research is to recognize emotions through facial gestures. As it was stated in previous section there are several approaches for facial gesture recognition given a single frame.In contrast emotion recognition could be required a sequence of facial gestures. This methodology considered the following steps :

1. Image Acquisition
2. Feature Extraction
3. Gesture Recognition
4. Emotion Recognition

In general the process begins by capturing a **video sequence**, which is segmented into *frames* ,after video has been segmented most significant features on face are marked. Every frame in the sequence is labeled as one of the following gestures: neutral, anger,surprise, joy, disgust, sadness , fear combined with the intensity of the emotion (low,mid,high). Therefore exists 19 possible labels since neutral have only one intensity. After gesture recognition the labeled sequence is evaluated into a first order HMM. Figure 2 shows the proposed methodology.

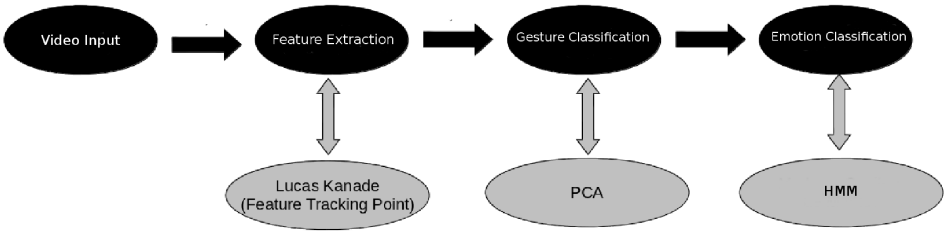


Fig. 2. Proposed Methodology.

Feature Extraction. Let I, J be gray scaled images ,$\boldsymbol{u} = [u_x, u_y]$ where $\boldsymbol{u}$ is an image point on the first image I and (u_x, u_y) are the two pixel coordinates. The goal of feature tracking is to find the location $\boldsymbol{v} = \boldsymbol{u} + \boldsymbol{d} = [u_x + d_x, u_y + dy]$. Lucas-Kanade's algorithm [9] with Bouguet's improvement [1] was used to track and mark must significant features on face , even in areas of low contrast. Since this algorithm is sensitive to changes in lighting face normalization results a reasonable approach for reducing variations.There are twelve facial features without contrast or texture problems as it is showed in figure 3. These points are tracked and marked in every frame when a human face is detected. Since the background is removed in every frame, face detection is used to crop and re size

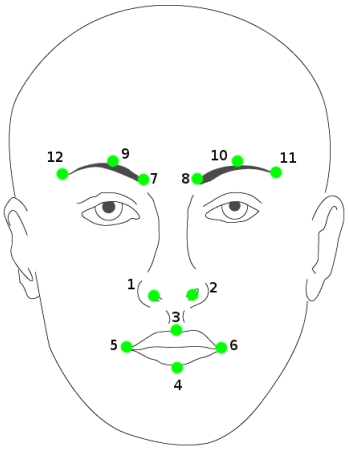


Fig. 3. Feature points. *1.-Left nostril. 2.-Right nostril. 3.-Upper lip. 4.-Lower Lip. 5.-Left corner. 6.-Right corner. 7.-Outer of left eyebrow. 8.-Outer of right eyebrow. 9.-Center of left eyebrow . 10.-Center of right eyebrow. 11.-Inner of left eyebrow. 12.-Inner of right eyebrow.*

the region of interest (ROI). As result of this process every frame contains just a marked face and have the same size.

Gesture Recognition. Let be $A = \{x|x_i$ is a frame of the video sequence to analyze$\}$, and $B = \{$*neutral,low anger,mid anger,high anger, low disgust,mid disgust,high disgust,low fear,mid fear,high fear,low joy,mid joy,high joy,low sadness,mid sadness,high sadness,low surprise,mid surprise,high surprise*$\}$ a set of emotions respectively. Every frame is subject to Principal Component Analysis (PCA) in order to reduce dimensionality of the feature space by considering the first 30 components. Given a training set of N images $C = [I_1, I_2, ..., I_N]$ where each I_J is an image representing one of the six basic emotions, PCA deals with the weights of image's training set then Euclidean and Mahalanobis distances are obtained to determined the best match . Mahalanobis distance is given by $((\boldsymbol{x} - \boldsymbol{y_i})'C^{-1}(\boldsymbol{x} - \boldsymbol{y_i}))^{1/2}$ where $\boldsymbol{x}$ and $\boldsymbol{y_i}$ are elements from the testing and training set respectively. The covariance matrix C is calculated between the tested frame and every frame in the training set.

Emotion Recognition. Let be a set of T emotional hidden states $\omega = \{\omega_1, \omega_2, \omega_3, ..., \omega_T\}$. Given a visible state sequence $V = \{x|x \in B\}$ we determine the probability that this particular sequence was generated by a particular model θ. Determine the most likely sequence of hidden states ω led to those observations that defines which emotional state is. Given the coarse structure of a model *(the number of states and the number of visible states)* but not the probabilities a_{ij} and b_{jk} and a set of training observations of visible symbols *(video examples)*, we determine a_{ij} and b_{jk} probabilities using the Baum-Welch algorithm. For

instance surprise is the briefest emotion and most of the times merges to pleasant or unpleasant gestures depending upon what it was that surprised us. Such an ergodic HMM for the emotion surprise have the states *neutral, pleasant surprise, unpleasant surprise* as shown in Figure 4. The probability that the model θ

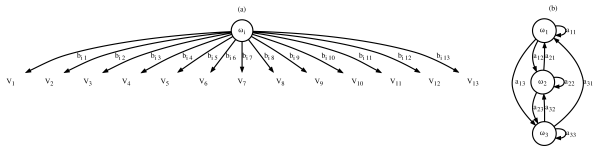


Fig. 4. HMM for the surprise emotion.(a) Every hidden state ω_i emits surprise,joy,disgust or anger with three different intensities.Neutral emission has only one intensity **(b)** Since surprise is never expected, *neutral* as hidden state is included in this model besides pleasant and unpleasant surprise.

produces this sequence V^T of visible states is the probability that this video contains a surprised expression regardless it was pleasant or unpleasant. Given this sequence of visible states V^T , the problem is to find the most probable sequence of hidden states. Therefore it is concluded whether it was a pleasant or unpleasant surprise. Table 1 shows a simple example for classifying an emotional state using the model shown in Figure 4.

States / Emissions	Neutral	Pleasant Surprise	Unpleasant Surprise
Neutral	0.4	0.025	0.025
High Surprise	0.00671875	0.02203125	0.02203125
High Joy	5.1806640625E-4	0.003206542	0.0

Table 1. Forward algorithm and its probabilities for $V^3 = \{V_1, V_4, V_7\}$.This table shows in the last iteration that pleasure surprise is the must probable state for the sequence V^3.

4 Results

The training set for the gesture recognition step has 900 faces depicting a neutral expression and the six basic emotions: anger, disgust, fear, joy, sadness, surprise. Some images are from the University of Stirling face database [14] and the Japanese Female Facial Expression **(JAFFE)** database [12]. Gender and race distribution is shown in Table 2. A classifier for each of the six fundamental facial gestures (joy, sad, angry, fear, surprise, disgust) and one for neutral gesture were

Ethnicity	Men	Women
American (U.S. citizen)	10	13
Scottish	32	34
Japanese	0	10
Iranian	0	35
Mexican	25	15

Table 2. Gender and race distribution. Training set for gesture recognition step

trained to build a facial gesture classifier. The normalized distance is used as a deciding factor.

4.1 Detection of pleasant/unpleasant surprise gestures

A HMM was trained with 20 videos depicting a pleasant or unpleasant surprise using the Baum-Welch algorithm. In order to induce surprise at this experiment several videos showing suddenly a scary images were used. Reaction differs for each person depending its emotional state. The entire set of videos were recorded with a frame rate of 30 frames per second but for this analysis 15 frames per second are only used in order to reduce the length of the emission sequences V^T without losing capability for capture even micro-expressions. It were calculated the most probable sequence of hidden states for 15 testing videos. Table 3 shows the most probable state reached in last observation which is used as a deciding factor where *P=Pleasant,U=Unpleasant,O=Other*. Results are acceptable since the classificator fails only in video 12 and 13. Video 12 shows a surprise expression but it not merges into another emotion, therefore this test shows a desirable response. Otherwise the video 13 starts with an unpleasant surprise expression omitting a neutral expression at the begining.

Repeated states appears in every video sequence since human face holds and expression at least for $\frac{1}{2}$ seconds (micro-expressions) some post-processing may be applied and just get the sequence somewhat independent of variations in rate. Convert the sequence $\{\omega_1, \omega_1, \omega_2, \omega_3, \omega_3, \omega_1\}$ to $\{\omega_1, \omega_2, \omega_3, \omega_1\}$, seems to be appropriate for emotion recognition in order to reduce the length of this sequence that can reach 30 symbols per second.

4.2 Detection of lie gestures

Many factors such a facial expression, tone of voice, slip of the tongue, or certain gestures could leak our true feelings. Exclusively facial expressions will be used for other reasons which are beyond the scope of the present research. Micro and squelched expressions are hints to discover if someone is lying but they are not conclusive. The first issue to consider in estimating whether or not there will be any clues to deceit is whether or not the lie involves sense of emotions when is happening a lie. Based on we have proposed a method for emotion recognition

States / Video	$P(Neutral\|T)$	$P(Pleasant\|T)$	$P(Unpleasant\|T)$	**Test**
1	5.180664E-4	**0.003206**	0.0	P
2	5.180664E-4	0.0	**0.002137**	U
3	5.180664E-4	0.0	**0.001068**	U
4	**0.002781**	4.199218E-4	0.001705	O
5	**0.003882**	4.199218E-4	0.002990	O
6	**0.004984**	0.004275	4.199218E-4	O
7	3.115234E-4	**0.001760**	0.0	P
8	3.115234E-4	0.0	**0.001173**	U
9	3.115234E-4	0.0	**5.869140E-4**	U
10	**0.005011**	0.002767	8.398437E-4	O
11	8.197021E-5	**5.139770E-4**	0.0	P
12	**0.008289**	0.004275	0.004275	P/U
13	**7.189025E-5**	5.123138E-6	6.508712E-5	U
14	2.426757E-4	0.0	**7.475585E-4**	O
15	7.421875E-4	**0.003632**	0.0	P

Table 3. Reached states after decoding algorithm. The decoding algorithm finds at each time step t the state that has the highest probability of having come from the previous step and generated the observed visible state V_k. (P=*Pleasant*, U=*Unpleasant*, O=*Other*)

into a temporal context our approach for evaluate lies is based on micro and squelched expressions. Figure 5 shows a model for representing a moment of *lying* and its transitions between hidden states before any training. Micro expressions provide a full expression of the concealed emotion, but so quickly that they are usually missed. The presence of any micro expression depicting any emotion is a lying sign further a smile is the most common cover or mask.

The experiment for gathering videos is based on Ekman's *nurse experiment*, that offers the chance to test out and practice the ability to control expression of your feelings. It is basically consisted on watch an unpleasant film meanwhile a person describes the film as pleasant. The same video was used to collect testing videos. Finally a HMM was trained with 15 videos using the Baum-Welch algorithm. Reaction differs for each person depending its emotional state. Figure 6 shows the difference between each hidden state probability at the last emission for some video samples. Microexpressions were the most difficult emotion to recognize.Since microexpressions are the most unusual gestures only 5 samples were obtained.

5 Conclusions

This paper describes and tests a probabilistic model for automatic emotion interpretation by using a set of observations obtained from real time situations. The use of a first order HMM is justified by the inherent temporality in human emotions. In this paper was introduced an approach for analyze human emotions

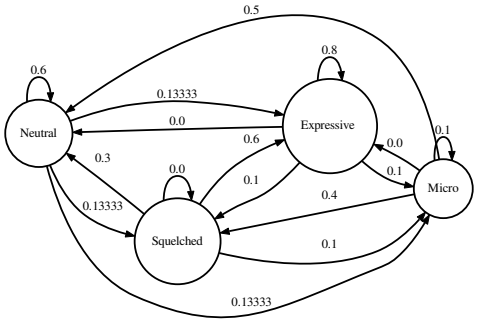


Fig. 5. HMM for lie detection. Each hidden state ω_i can emit one of the 19 available visible states from neutral to any of the six basic emotions and its intensities.

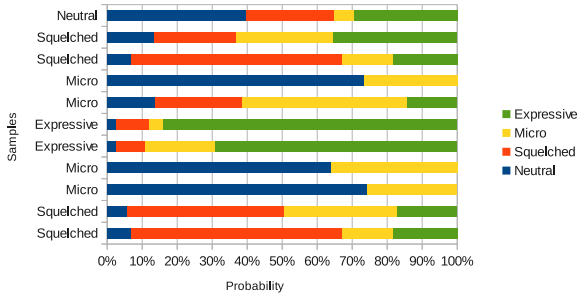


Fig. 6. Recognition results of the proposed lie detector model.

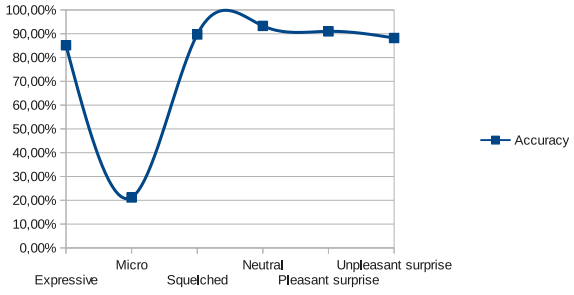


Fig. 7. Average accuracy for the proposed methodology.

in a temporal context. Preliminary results shows that this approach allows to analyze several human emotions in a temporal context, not only the two models proposed. Future work will improve the methodology and its implementation, introducing other variables such as voice tone, arm gestures, heart rate fluctuations and other sources of information like activity in social networks which are used to show several emotions.

I't is considered that this approach can be applied in differents escenarios, such as airport security where a shorts ans specific questions can trigger small change in people's face; other area of interest could be for human resources departments in order to detect when a person is lying , based on the fact that some changes develop slowly, because of these phenomena are rarely perceptible over a short space of time.

References

1. Bouguet, J.Y.: Pyramidal implementation of the lucas kanade feature tracker description of the algorithm. `http://robots.stanford.edu/cs223b04/algo_tracking.pdf` (2000)
2. Darwin, C., Ekman, P., Prodger, P.: The expression of the emotions in man and animals. Oxford University Press, USA (2002)
3. Ekman, P.: Cross-cultural studies of facial expression. Darwin and facial expression: A century of research in review pp. 169–222 (1973)
4. Ekman, P.: Telling lies: Clues to deceit in the marketplace, politics, and marriage. WW Norton & Company (2009)
5. Ekman, P., Friesen, W.: Facial action coding system, chap. IV. Consulting Psychologists Press, Stanford University, Palo Alto (1977)
6. Ekman, P., Friesen, W.: Unmasking the face: A guide to recognizing emotions from facial clues. Ishk (2003)
7. Fasel, B., Luettin, J.: Automatic facial expression analysis: a survey. Pattern Recognition 36(1), 259–275 (2003)
8. Kass, M., Witkin, A., Terzopoulos, D.: Snakes: Active contour models. International journal of computer vision 1(4), 321–331 (1988)
9. Lucas, B.D., Kanade, T.: An iterative image registration technique with an application to stereo vision (ijcai). In: Proceedings of the 7th International Joint Conference on Artificial Intelligence (IJCAI '81). pp. 674–679 (April 1981)
10. Lyons, M., Budynek, J., Akamatsu, S.: Automatic classification of single facial images. Pattern Analysis and Machine Intelligence, IEEE Transactions on 21(12), 1357–1362 (1999)
11. Mitra, S., Acharya, T.: Gesture recognition: A survey. Systems, Man, and Cybernetics, Part C: Applications and Reviews, IEEE Transactions on 37(3), 311–324 (2007)
12. Miyuki Kamachi, M.L., Gyoba, J.: The japanese female facial expression database. `http://www.kasrl.org/jaffe.html` (2012), [Online; accessed 22-November-2012]
13. Samaria, F., Young, S.: Hmm-based architecture for face identification. Image and vision computing 12(8), 537–543 (1994)
14. University, S.: Psychological image collection at stirling. `http://pics.psych.stir.ac.uk/` (2012), [Online; accessed 22-November-2012]

P-Median: A Performance Analysis

María Beatríz Bernábe Loranca[1], Jorge Ruiz Vanoye[2], Rogelio González Velázquez[1], Marco Antonio Rodríguez Flores[1], Martín Estrada Analco[1]

[1]Computer Science Department, Benemérita Universidad Autónoma de Puebla, México
beatriz.bernabe@gmail.com, marco89_rf@hotmail.com,
rogelio.gzzvzz@gmail.com, mestrada@cs.buap.mx
[2]Universidad Autónoma del Carmen
Ciudad del Carmen Campeche, México
jorge@ruizvanoye.com

Abstract. This work approaches the P-median problem with a partitioning around medoids methodology. This problem has been extensively studied because of the multiple applications and its NP-hard nature; therefore several are the efforts to find optimal solutions.
In this paper we consider two case studies: 1) instances from OR-Library and 2) geographical objects from the metropolitan zone of Toluca, Mexico. The methods we use to work with these data are: Partitioning Around Medoids (PAM), Variable Neighborhood Search (VNS), Simulated Annealing (SA), P-Median with MATLAB (PML) and finally Tabu Search (TS).

Keywords: P-median, partitioning, clustering, algorithms.

1. Introduction

In territorial partitioning, two models are the most used: the location-allocation and the set partitioning models. These models seek to group small geographical areas called basic units in a given number of bigger geographical clusters, named territories. The territorial partitioning problem can be modeled like a P-median problem with certain restrictions under the concept of partition, this is, if $\Omega = \{x1, \dots, xn\}$ is a finite set with n objects we wish to classify and let $k < n$ be the number of classes where we want to group the objects; a partition $P = (C, , \dots, Ck)$ from Ω in k ses$C1, \dots, Ck$, is characterized by the following conditions:

$$1.\ \Omega = \bigcup_{i=1}^{K} c_i$$

$$2.\ C_i \cap C_j = \emptyset, \qquad for\ every\ i \neq j$$

The P-median problem consists in finding the best configuration of facilities to attend the population's demand in the best way [1]. Given a set of 1000 nodes (coordinates) in the cartesian plane, where every node possesses a certain demand that must

be fulfilled, 10 service providers must be installed to satisfy this demand at the minimum cost. The cost can be determined in a proportional way to the distance between nodes.

Given the characteristics of this problem, the formulation to solve it was established in [2]. The problem was named p-median. This problem locates p facilities to minimize the total distance between the demand points and their nearest facilities [1]. For this we solve the allocation problem and minimize the distance and demand of the nodes. The computational complexity of this kind of problems requires using approximate methodologies to give a satisfactory response in regard to quality and time. In this point, two kinds of data have been processed with different heuristic variants based on the P-median model.

2. K-medoids and P-Median

The K-medoids and the P-median problems are similar to each other, while the k-medoids problems deals with cluster analysis to form groups or classes of similar objects, the P-median problem is a facility location problem where the goal is to place facilities on a geographical space to satisfy a certain demand in the area while minimizing transport costs and/or other logistical restrictions (like demand values). When K-medoids methodologies like Partitioning Around Medoids (PAM) are applied to group geographical data, they are basically solving a basic P-median problem where the main goal is to minimize distances between objects (similarity) without other logistical restrictions (uncapacitated). In this paper we employ benchmark matrices from OR-Library for uncapacitated p-median problems and real geographical data from the valley of Toluca, Mexico.

In the section below, we present elementary notions about PAM and the P-median problem and the metaheuristics we have employed in our analysis.

2.1. Partitioning Around Medoids

In the partitioning around medoids methods, PAM has a good reputation because it's capable to achieve good results. It was developed by Kaufman and Rousseuw [3]. This method assumes that n objects exist and a representative object is determined for every group to find K-clusters (groups), this representative is a medoid. After selecting the K medoids, the algorithm tries to analyze the possible pairs of objects, such that every object is grouped with the most similar medoid. The cost is calculated for each combination according to the defined clustering measurement to determine the best configuration that will be used as the starting point for the next iteration. The complexity of a single iteration is $O(K(n-K)^2)$, therefore the computing time is inefficient for high values of n and K.

PAM Algorithm

```
1:  s ←InitialSolution()
2:  change ← true
3:  While change do
4:  For each medoid m in s do
5:    For each non-medoid o do
6:      s' ←Swap(m, o)
7:      If Cost(s') < Cost(s) then
8:        s ←s'
9:        change ← true
10:     else
11:       change ← false
12:     end if
13:   end for
14: end for
15: end while
16: Return s
```

2.2. P-Median

The territorial and logistical problems have a wide range of applications; from political districting to social services and commercial territories location, etc. In the related literature, geographical compactness and connectivity criteria are employed to measure the quality of the solutions. According to Kalcsics et al. [4], a territory is geographically compact if its shape is round and isn't distorted; however an exact definition doesn't exist. Generally, the sum of the distances between the basic units (objects) and their representative medoid is the compactness measurement. This is how this problem can be modeled like a P-median one [5, 6]. The P-median problem can be expressed as follows:

A finite set of objects must be partitioned in exactly p groups. Each of these groups is characterized by one of its objects that was selected to be the median of the group. A distance is specified for every pair of objects and the sum of the distances between objects and their respective medians must be minimized. Its formulation is the following:

Let $N = \{1, \dots n\}$ be the set of indices for the clients and $J = \{1, \dots m\}$ the set of indices for the potential locations. Typically $N = J$. For every *(i,j)*, *i* element of *N*, *j* element of *J*. Let C_{ij} the cost of assigning client *i* to the median located in *j*. The following decision variables are defined:

$$Y_j = \begin{cases} 1 & If\ the\ median\ is\ located\ in j \in J \\ 0 & otherwise. \end{cases}$$

$$X_{ij} = \begin{cases} 1 \text{ If } the\ client i \in N is\ assigned\ to\ the\ median\ located\ in\ j \in J. \\ 0 \qquad otherwise. \end{cases}$$

Then, the P-median problem is formulated in the following way:

$$Min \sum_{i \in N} \sum_{j \in J} C_{ij} X_{ij} \qquad (1)$$

Subject to:

$$\sum_{j \in J} X_{ij} = 1 \qquad \forall i \in N \qquad (2)$$

$$\sum_{j \in J} Y_{ij} = p \qquad (3)$$

$$X_{ij} \leq Y_j \qquad \forall i \in N, j \in J \qquad (4)$$

$$X_{ij} \in \{0,1\}, Y_j \in \{0,1\} \; \forall \; i \in N, j \in J \qquad (5)$$

Restrictions (2) ensure that each client is assigned to one single median. Restriction (3) guarantees that exactly p locations for the medians are selected. Restrictions (4) establish that the clients are assigned to a median only if this one has been selected. Finally, the set of restrictions (5) specifies that the decision variables are binary.

The classical version of the P-median problem was formulated in 1960 as an extension to the simple facilities location problems proposed, however an algorithm of polynomial order doesn't exist to solve the P-median problem, this is why is considered an NP-hard problem and is still studied.

We employ three approximation methods to solve P-median and K-medoids. The heuristic methods are commonly used to approximate NP problems. We chose three metaheuristics that have been extensively used to solve the P-median problem in particular [7].

2.3. Variable Neighborhood Search

VNS is a metaheuristic technique proposed and described in several works by Mladenovic and Hansen [8, 9, 10]. The basic idea is to combine the application of a local search procedure with a systematic search neighborhood change. The algorithm applies the local search to a solution from the neighborhood of the best solution stored (current solution). If it's impossible to improve this current solution, a bigger neighborhood is considered. When a better solution is obtained the process is restarted. It tries to exploit the idea that the local optimums tend to gather only in certain regions. A more recent tutorial can be found in [11].

The following algorithm shows how the solutions are obtained [12].

VNS Algorithm.

```
Input
  Nk: k=1..kmax, neighborhood structures
```

```
  Sa: current solution
  Sp: neighbor solution of Sa
  Sol: local optimal solution

1:  While Stopping Condition not fulfilled do
2:    k ← 1
3:    While k <kmax do
4:      Sp←GetNeighbor(Sa, Nk)
5:      Sol ←LocalSearch(Sp)
6:      If Cost(Sol) < Cost(Sa) then
7:        Sa← Sol
8:      else
9:        k ←k + 1
10:     end if
11:   end while
12: end while
13: Return Sa
```

2.4. Simulated Annealing

Simulated Annealing is a neighborhood search algorithm with a probability of acceptance criterion based on thermodynamics. It's an optimization method inspired on the metal quenching process used around the 500's B.C. The metal quenching process consists of three phases: a warming phase up to a certain temperature; during the second phase the high temperature is maintained, this allows the molecules to be arranged into minimum energy states, followed by a controlled cooling phase to increase the size of its crystals and reduce its defects. The Metropolis algorithm proposed in 1953 [13] is the pioneer of the simulated annealing methods but Kirkpatrick and Gelatt were the first ones to apply it to optimization problems to find solutions for the travelling salesman problem with several cities [14].

SA Algorithm

```
1: s ←InitialSolution()
2: T ← T0
3: g ← 0
4: While Stopping Conditions not fulfilled (g, T) do
5:    s' ←RandomNeighborFrom(s)
6:    If Cost(s') < Cost(s) then
7:      s ←s'
8:    else if Random(0,1) <exp((Cost(s)-Cost(s'))/T)then
9:      s ←s'
10:   end if
11:   g ←g + 1
12:   T ←Update(g, T)
```

```
13: end while
14: Return s
```

2.5. Tabu Search

Tabu Search emerged from diverse works published in the late 70's. Despite the fact that the main concepts and strategies behind it already existed, it was in 1989 when the name and methodology were formally established by Fred Glover and Manuel Laguna in the homonymous book Tabu Search [15].

TS is a metaheuristic that guides one or more local heuristics to reach zones of the solution space that regular heuristics usually miss. The local heuristic procedure is an operation used to move between solutions within a defined neighborhood, until reaching a local optimum or fulfilling a stopping criterion. A component from TS that distinguishes it from other metaheuristics is the use of adaptive memory that allows a more flexible search behavior because it stores relevant data about the search process, for example, attributes of the current solution to avoid exploring the same zones of the solution space more than once or to guide the search towards unexplored zones. In this way the motor behind TS is an intelligent and strategic guide capable of making decisions instead of relying on randomness and probability like simulated annealing. This is the biggest contrast between TS and most metaheuristics.

TS Algorithm.

```
Input:
  Number of facilities p
  Number of iterations nit
  Number of iterations for second phase nit2
  Number of worse solutions permitted (perturbation)ip
  Tabu Tenurett

1: pc ← 0
2: ic ← 1
3: S ←InitialSolution()
4: S* ← S
5: While ic < nit do
6:    prev_cost← Cost(S)
7:    Move(S)
8:    If Cost(S) > prev_cost then
9:          pc ← pc+1
10:   end if
11:   If Cost(S) < Cost(S*) then
12:         S* ←S
13:   end if
14:   If pc > ip then
15:         PerturbSolution(S)
```

```
16:         pc ← 0
17:   end if
18:   UpdateTabuLists()
19:   ic← ic+1
20: end while
21: S ← S*
22: CleanTabuStates()
23: For i← 0 until nit2 do
24:   Move(S)
25:   If Costo(S) ←Costo(S*) then
26:         S*←S
27:   end if
28: UpdateTabuLists()
29: ic ← ic+1
30: end if
31: Return S*
```

The initial solution is generated with the Stingy Drop method, which has given great results to generate initial solutions [7, 16].

Whereas the Move function is a modified faster version of the interchange or swap method proposed by Withtaker that evaluates the profit of interchanging a facility with a candidate facility, in our case we randomly select a cluster and its objects are evaluated as potential facilities and the most profitable is swapped with the medoid [17].

3. Experimentation

The experiments made were tested in a Samsung RV415 laptop with 2GB of RAM and AMD® E-350 CPU at 1.60GHz.

The instances we present bellow were evaluated with VNS, PAM, SA, PML (MATLAB) and TS. The first instances are available in [18] and are widely used in literature to test P-median algorithms. The second instances belong to a real map of the Toluca valley obtained from the National Institute of Statistics, Geography and Informatics of Mexico (INEGI). We present our results in tables 1, 2 and 3 in section 4.

3.1. OR-Library – Uncapacitated P-Median

OR-Library is a collection of test data sets for a variety of Operations Research (OR) problems. OR-Library was originally described in J.E.Beasley [19]. We have taken the 40 available uncapacitated p-median data files that range from 100 to 900 nodes. See Table 1 and 2.

3.2. The Valley of Toluca

This map represents the basic geostatistical areas defined by a census from the year 2000 by the INEGI. It contains 469 objects and our results can be seen in table 3.

4. Results

In this section we cover the results we obtained in several test runs proposed for our data sets.

4.1. OR-Library Instances

In tables 1 and 2, we see the results for the OR-Library instances. We can observe that PAM and TS (table 2) attain the best results in comparison with VNS, SA and PML (Matlab), however PML has a clear advantage in regard to computing time.

Table 1. Uncapacitated P-median problems from OR-Library (Worst methods)

Pmed	OR-Library			VNS		SA	
	Nodes	P	Best	Cost	Time (sec)	Cost	Time (sec)
1	100	5	5819	5819	103	6209	28
2	100	10	4093	4341	180	4646	70
3	100	10	4250	4467	180	4785	47
4	100	20	3034	3380	250	3693	93
5	100	33	1355	1664	360	1820	119
6	200	5	7824	7917	330	8349	49
7	200	10	5631	5952	540	6446	104
8	200	20	4445	5204	1003	5536	174
9	200	40	2734	3385	1860	3626	286
10	200	67	1255	1700	1980	1850	907
11	300	5	7696	7803	720	8346	149
12	300	10	6634	7200	900	7717	298
13	300	30	4374	5126	1860	5475	692
14	300	60	2968	3823	1440	3992	71
15	300	100	1729	2464	240	2558	1721
16	400	5	8162	8423	300	8958	220
17	400	10	6999	7651	540	8197	322
18	400	40	4809	5821	2700	6038	505
19	400	80	2845	3747	2760	3881	2036
20	400	133	1789	2647	2040	2755	1909
21	500	5	9138	9557	240	10231	102
22	500	10	8579	9433	300	9802	170
23	500	50	4619	5645	3600	5941	839
24	500	100	2961	3974	600	4065	1440
25	500	167	1828	2726	720	2852	240
26	600	5	9917	10312	60	10869	14

27	600	10	8307	9065	120	9511	25
28	600	60	4498	5664	480	5799	141
29	600	120	3033	4114	780	4176	70
30	600	200	1989	2960	1320	3058	47
31	700	5	10086	10528	70	11157	93
32	700	10	9297	10383	120	10818	119
33	700	70	4700	6007	720	6166	49
34	700	140	3013	4193	1500	4286	104
35	800	5	10400	11037	120	11698	174
36	800	10	9934	9994	180	11544	250
37	800	80	5057	6460	1620	6715	50
38	900	5	11060	11725	180	12252	10
39	900	10	9423	10570	300	11017	25
40	900	90	5128	6632	2460	6803	40

Table 2. Uncapacitated P-median problems from OR-Library (Best methods)

pmed	OR-Lib	PAM		PML (Matlab)		TS	
	Best	Cost	Time (sec)	Cost	Time (sec)	Cost	Time (sec)
1	5819	5819	0	5891	0.039791	5819	2.556
2	4093	4105	0	4118	0.049311	4093	1.672
3	4250	4250	0	4399	0.056247	4250	1.604
4	3034	3046	1	3088	0.083438	3041	5.703
5	1355	1355	1	1378	0.13098	1394	5.928
6	7824	7824	0	8027	0.075893	7824	49.28
7	5631	5645	1	5646	0.14841	5631	21.744
8	4445	4457	2	4472	0.25138	4451	19.764
9	2734	2753	8	2841	0.49444	2804	31.729
10	1255	1263	14	1295	0.83192	1318	25.288
11	7696	7696	0	7721	0.15035	7696	145.137
12	6634	6634	1	6651	0.28253	6634	63.67
13	4374	4374	20	4467	0.81831	4388	48.169
14	2968	2974	56	3013	1.5988	3091	37.845
15	1729	1738	82	1761	2.6418	1858	48.857
16	8162	8162	1	8232	0.27005	8162	222.629
17	6999	6999	2	7019	0.47927	6999	97.449
18	4809	4811	67	4873	1.8845	4840	25.538
19	2845	2859	296	2899	3.6658	2927	29.422
20	1789	1805	600	1866	6.0295	1882	36.45
21	9138	9138	0	9138	0.38812	9138	164.141
22	8579	8669	4	8670	0-74461	8579	58.606
23	4619	4619	160	4694	3.5808	4664	58.606
24	2961	2965	938	3009	7.1164	3093	127.046
25	1828	1844	1608	1896	11.9582	1937	132.722
26	9917	9917	2	10093	0.56943	9917	389.316
27	8307	8307	9	8364	1.0522	8307	68.365
28	4498	4515	605	4579	6.207	4551	35.594
29	3033	3039	2101	3104	12.3188	3181	66.12

30	1989	2009	2208	2037	20.3498	2119	105.318
31	10086	10086	2	10086	0.74536	10086	479.083
32	9297	9301	8	9331	1.429	9310	109.158
33	4700	4703	1495	4798	9.7708	4735	47.558
34	3013	3026	4685	3097	19.7808	3168	119.309
35	10400	10400	2	10406	0.96892	10400	413.429
36	9934	9934	10	9954	1.8497	9934	141.098
37	5057	5064	2092	5118	14.4414	5278	68.316
38	11060	11060	8	11153	1.2062	11060	86.544
39	9423	9423	13	9451	2.3816	9423	99.102
40	5128	5138	5076	5190	20.838	5214	76.852

4.2. Toluca Valley

In table 3, we present our results obtained for 15 instances using our second data set. We can see that PAM and TS obtain the best results compared with the rest of our algorithms. However because of its intensification, PAM requires a considerable time, for example when P equals 200 PAM finds a solution in 25200 seconds, PML does it in 14.7711 seconds and TS, 34.504.

With this map TS achieves good computing times and the solutions are better than PML, except for instances 10, 11 and 12 and surpasses PAM in tests 2 and 4. After these, VNS is the next method with good results and at last SA. The nomenclature for table 3 is n: test number, P: medians (number of groups), T: computing time in seconds and C: the cost attained.

Table 3. Toluca Valley (460 nodes)

n	P	VNS		PAM		SA		PML (Matlab)		TS	
		C	T	C	T	C	T	Cost	T	C	T
1	5	24.24951	14	23.9643	7	25.0138	0	24.3004	0.37133	23.9643	12.767
2	10	16.8784	21	15.837	17	17.5422	1	16.8496	0.97753	15.5434	8.594
3	15	13.8743	28	12.3637	56	14.74491	1	13.4858	1.1002	12.3767	15.238
4	20	12.0909	46	10.4553	120	12.6997	2	11.2325	1.4222	10.3671	12.949
5	30	9.682202	60	8.0539	300	10.2202	2	8.5286	2.1689	8.0743	11.361
6	33	9.416298	53	7.496998	420	9.552296	2	7.9745	2.4193	7.5694	11.887
7	40	8.322398	61	6.435499	660	8.834101	3	6.8658	2.9317	6.5874	10.973
8	67	5.985501	62	4.360599	2880	6.3118	5	4.6986	5.8809	4.5531	11.652
9	80	5.3999	108	3.711501	2880	5.554698	5	4.0329	8.1257	3.9411	12.432
10	90	4.7458	120	3.3553	3600	5.7672	5	3.2601	9.8468	3.5681	14.222
11	100	4.494802	182	3.0419	3603	4.561401	5	2.7308	14.0718	3.2501	13.954
12	133	3.419801	241	2.3111	11820	3.6658	6	2.4554	13.7786	2.4568	15.705
13	140	3.360899	256	2.1862	12000	3.4056	6	2.3286	9.5345	2.3209	25.646
14	150	3.113098	257	2.0272	18000	3.1526	6	2.1564	10.1451	2.1443	28.511
15	200	2.221599	579	1.4159	25200	2.2481	8	1.5033	14.7711	1.4999	34.504

5. Conclusions

Our analysis of the PAM, VNS, SA, PML and TS algorithms allows us to see the strategies that achieve the best results.

From tables 1 and 2, we clearly see that PAM is the one that obtains the most optimal solutions but its computing time is excessive for instances with several nodes and high values of P. In table 3, for example, we see that PAM takes 25200 seconds to return a solution with 469 nodes and 200 P. In this matter, is clear that PML (MATLAB) works faster by returning a solution in less than 15 seconds for the biggest instance.

On the other hand, TS reaches the best known solutions for the OR-Library instances in most of the cases. PML gives good results for certain instances (for example test run 31 in 0.74536 seconds) but the cost of the solutions is inferior to the costs achieved by TS in most instances and the computing time for TS is a clear improvement from PAM. This can be observed in table 2, however the computing time for TS can be dramatically reduced according to the number of iterations, for example, test 40 finished in 76 seconds while instance 35, which is smaller, in 413 seconds.

VNS and SA present the same issue, their computing times rely on the parameters passed to the algorithms and could respond in much shorter times with the appropriate values, however for these two we have used first improve strategies (the first improving move is selected). This strategies sacrifice optimality for speed, therefore despite our attempt to give them more time to reach a solution (high parameters) they have given the worst results in both tests, this tells us that completely random strategies are unreliable even with extended execution times.

With TS we have implemented best improve strategies (the best move is chosen) while maintaining a good performance, for example for the smaller instances of OR-Library (100 to 300 nodes) the algorithm handles 40000 to 70000 iterations in less than 5 seconds.

PAM relies on a thorough search until no improvement is made having a big influence on the speed to find a solution for big values of P and more than 400 nodes. This makes it the best choice only for small problems to find optimal solutions in competitive times.

In table 3, the map of Toluca, a small sized problem TS is the best choice in terms of time versus optimality because it surpasses the cost of the solutions obtained by PML in several occasions under 35 seconds and in tests 2 and 4 finds better solutions than PAM.

In conclusion the best improve strategies are the most adequate to reach optimal solutions, however a balance should exist between random and intensification strategies to reduce the effect on the performance like we did it in our TS algorithm. Our current challenge is to improve this algorithm to reach the best known solutions for the hardest p-median problems from OR-Library while maintaining a good performance.

References

1. Daskin, M. S.: Network and Discrete Location: Models, Algorithms and Applications, John Wiley and Sons, Inc., New York (1995)
2. Hakimi, S.: Optimum location of switching centers and the absolute centers and medians of a graph. Operations Research, 12, 450-459 (1964)
3. Kaufman, L., Rousseeuw, P.: Clustering by Means of Medoids. In: Y. Dodge, Editor, Statistical Data Analysis Based on the L1-Norm and Related Methods, Amsterdam: North-Holland, 405-416 (1987)
4. Kalcsics, J., Nickel, S., Schröder, M.: Towards a Unified Territory Design Approach: Applications, Algorithms and GIS Integration. TOP, 13, 1–74 (2005)
5. Hess, S.W., Samuels, S.A.: Experiences with a sales districting model: criteria and implementation. Management Science, 18, 41– 54 (1971)
6. Zoltners, A., Sinha, P.: Sales territory design: thirty years of modeling and implementation. Marketing Science 24, 313–331 (2005)
7. Mladenovic, N., Brimberg, J., Hansen, P., Moreno, J. A.: The p-median problem: A survey of metaheuristic approaches. European J Operational Research, 179 (2007)
8. Hansen P., Mladenovic, N.: Variable neighborhood search, Les Cahiers du GERAD 96-49 (1996)
9. Mladenovic, N., Hansen, P.: Variable Neighborhood Search. Computers & Operations Research. 24, 1097-1100 (1997)
10. Hansen, P., Mladenovic N., Pérez M. J.: Variable Neighbourhood Search. Inteligencia Artificial, Revista Iberoamericana de Inteligencia Artificial. No.19 ISSN 1137-3601, 77-92 (2003)
11. Hansen, P., Mladenovic, N.: Variable Neighborhood Search. In Fred Glover & Gary. A. Kochenberger (Eds.), Handbook of Metaheuristics, Kluwer Academic Publishers, 145-170 (2003)
12. Pelta, A.D.: Algoritmos heurísticos en bioinformática. PhD dissertation, Universidad de Granada, España (2000)
13. Metropolis, N., Rosenbluth, A., Teller, E.: Equation of state Calculations by Fast Computing Machines, J. Chem. Phys., 21, 6, 1087-1092 (1953)
14. Kirkpatrick, S., Gelatt, C. D., Vecchi, M. P.: Optimization by Simulated Annealing. Science, 220, 671-680 (1983)
15. Glover, F., Laguna, M.: Tabu Search. Kluwer Academic Publishers, Norwell, MA, USA (1997)
16. Al-khedhairi, A.: Simulated Annealing Metaheuristic for Solving P-Median Problem. Int. J. Contemp. Math. Sciences, 3, 1357-1365 (2008)
17. Resende, M., Werneck, R.: A fast swap-based local search procedure for location problems. Annals of Operational Research, 150, 205-230 (2007).
18. OR-library http://people.brunel.ac.uk/~mastjjb/jeb/orlib/pmedinfo.html Retrieved on August 25 2014
19. OR-Library: distributing test problems by electronic mail. Journal of the Operational Research Society, 41, 1069-1072 (1990)

Integración de un sistema de información geográfica para algoritmos de particionamiento

María Beatriz Bernábe Loranca, Rogelio González Velázquez
Benemérita Universidad Autónoma de Puebla, Facultad de Ciencias de la Computación, Puebla, Pue. México
beatriz.bernabe@gmail.com

Resumen. Para problemas de Particionamiento Geográfico (PG), se buscan agrupaciones de objetos de acuerdo a las condiciones geográficas. Generalmente, las respuestas del particionamiento geográfico en otros trabajos han sido mostradas textualmente o en un grafo, sin embargo, para problemas donde datos geográficos son los que se agrupan, representar gráficamente las particiones resultantes es un proceso complicado pero necesario. Esto implica que la agrupación use recursos adecuados para este propósito como geometría computacional o herramientas de interfaz con un Sistema de Información Geográfica (SIG).

En este trabajo nos ocupamos de presentar una breve revisión del particionamiento geográfico y el proceso e implementación que hace posible observar resultados de opciones de particionamiento en mapas. Para este propósito se diseñó un conjunto de módulos que se comunican con un SIG. Este proceso suele iniciarse con la selección de datos que continua con la escogencia de un algoritmo de particionamiento geográfico en distintas categorías (compacto, homogéneo para variables poblacionales, P-mediana, multiobjetivo, Relajación Lagrangeana, homogéneo en el número de objetos, etc.). El resultado del particionamiento genera archivos de salida, sin embargo, el sistema acepta un documento de texto compuesto de una lista con los objetos gráficos y la relación al grupo que pertenecen. El procedimiento final está constituido de una interfaz con un SIG con el fin de distinguirlos resultados de las diferentes agrupaciones en mapas. A este sistema le hemos llamado Sistema de Interfaz Gráfico para Particionamiento (SIGP).

Palabras clave: Particionamiento, Sistema de Información Geográfica (SIG).

1. Introducción

Una breve discusión de Particionamiento es necesaria en este trabajo considerando que SIGP resuelve el problema de generar el mapa asociado a distintas opciones de agrupamiento. La importancia del sistema gráfico que se ha desarrollado se sitúa en la explotación de las herramientas para desarrolladores de SIG con el propósito de construir un sistema capaz de mostrar resultados de agrupaciones en modo gráfico.

Los datos que el sistema admite son de naturaleza geográfica y como caso de estudio, se han considerado las Agebs (ÁreasGeoestadísticas Básicas).

Se describen aspectos importantes de las estructuras de datos implícitas en la implementación de cada uno de los módulos que permiten la comunicación entre ellos. El objetivo es lograr un deseable desempeño de todos los componentes del sistema SIGP para conseguir que los resultados de particiones se reflejen en un mapa. Por último, el sistema se integra de 3 pasos que funciona a nivel usuario de la siguiente manera:

1: Se elige la entidad federativa de interés, cada entidad está dividida en un número de zonas geográficas Agebs. En este paso, el usuario puede separar de la entidad un subconjunto de Agebs que satisfagan características deseables para un problema específico. Un subsistema de consulta para selección de variables ha sido implementado para este propósito [1].

2: Particionar las Agebs con: a) Compacidad con Recocido Simulado(RS), b) Compacidad con Búsqueda por Entorno Variable(VNS, por sus siglas en ingles Variable NeighborhoodSearch) [2], c) Homogeneidad en el número de grupos[3], d) Homogeneidad en variables[4], e) Particionamiento multiobjetivo [4], P-mediana[5], f) Particionamiento Alrededor de los Medoides (PAM), g) Relajación Lagrangeana (RL) [5].

3: Mostrar las particiones gráficamente en mapas.

De acuerdo a lo anterior el presente trabajo se encuentra organizado como sigue: Introducción como sección 1. En la sección 2 se presentan algunos puntos de los métodos de agrupamiento. En la sección 3 se expone el desarrollo del sistema gráfico y en la sección 4 se presentan algunos resultados experimentales. Finalmente en la sección 6 se exponen conclusiones y trabajo futuro.

2. Parte experimental

Motivados por su amplia gama de aplicaciones, distintos trabajos han desarrollado técnicas para agrupar datos de diferentes tipos. Especial atención ha merecido el Particionamiento.

2.1. Agregación

Un término significativo para agrupar datos espaciales, es la agregación, la cual es citada cuando a agrupación con restricciones de compacidad geométrica se refiere. La agregación, no es más que un caso particular de cluster donde debe asegurarse la continuidad geográfica entre los elementos agrupados. Este asunto especial de análisis cluster es llamado generalmente análisis cluster con restricción de continuidad espacial.

Los métodos de clasificación usan generalmente una noción de proximidad entre grupos de elementos, para medir la separación entre las clases que se buscan. Se introduce el concepto de agregación, entendida como una disimilitud entre grupos de individuos:

Sean $A, B \subset \Omega$, entonces la agregación entre A y B es $\delta(A,B)$, tal que δ es una disimilitud en el conjunto de partes $\mathrm{P}(\Omega)$:i) $\delta(A,A) = 0$ para todo $A \in \mathrm{P}(\Omega)$

y ii) $\delta(A,B)=\delta(B,A)$ para todo $A,B\in \mathrm{P}(\Omega)$ .Usualmente, la medida de agregación está basada en la disimilitud d medida sobre Ω [6].

2.2. Métodos clásicos de Particionamiento

En los métodos de Particionamiento, se busca una única partición de los objetos en estudio en k clases disjuntas. En la clasificación por Particionamiento se tiene que siendo $\{x_1,x_2,..,x_n\}$ el conjunto finito de n objetos a clasificar y $k<n$ el número de clases en los cuales que se desea clasificar a los objetos. Una partición $P=\{C_1,..,C_k\}$ de Ω en k clases $C_1,..,C_k$ está caracterizada por las siguientes 2 condiciones: 1) $C_i\cap C_j=\varnothing$ 2) $\Omega=\bigcup_{i=1}^{k} C_i$ $i\neq j$. Es posible permitir eventualmente que algunas de las clases C_i sea vacía, de manera que en realidad las particiones $P=\{C_1,..,C_k\}$ que se consideran son particiones Ω en k o menos clases. Sin embargo, las particiones óptimas de acuerdo al criterio de inercia contienen exactamente k clases no vacías [6].Generalmente, la clasificación por particiones es planteada como un problema de optimización. Esto es, dado un conjunto de n objetos denotado por $X=\{x_1,x_2,..,x_n\}$ en que $x_i\in R^D$, sea K un número entero positivo conocido apriori, el problema del clustering consiste en encontrar una partición $P=\{C_1,..,C_k\}$ de X, siendo C_j un conglomerado conformado por objetos similares, satisfaciendo una función objetivo $f:R^D\to R$ y las condiciones: $C_i\cap C_j=\varnothing$ C para i ≠ j, y $C_i\cup C_j=X$.

Para medir la similitud entre dos objetos x_a y x_b se usa una función de distancia denotada por $d(x_a,x_b)$, siendo la distancia euclidiana la más usada para medir la similaridad. Así la distancia entre dos diferentes elementos $x_i=(x_{i1},.,x_{iD})$ y $x_j=(x_{j1},.,x_{jD})$ es $d(x_i,x_j)=\sqrt{\sum_{l-1}^{D}(X_{il}-X_{jl})^2}$.

Los objetos de un conglomerado son similares cuando las distancias entre ellos es mínima; esto permite formular la función objetivo f , como

$$\sum_{j=1}^{k}\sum_{x_i\in C_j} d(x_i,\overline{x_j})^2 \qquad (1),$$

es decir, se desea minimizar (1), donde x_j conocido como elemento representativo del conglomerado, es la media de los elementos del conglomerado C_j ,

$$x_j = \frac{1}{|C_j|} \sum X_i \in C_j \quad (2)$$ y corresponde al centro del conglomerado.

Bajo esas características, el clustering es un problema de optimización combinatoria, y ha sido demostrado que es un NP-difícil [7]. Dada la naturaleza combinatoria de este problema, su resolución requiere del uso de métodos aproximados, lo cual justifica el uso e incorporación de heurísticas [2, 3, 4, 5].

3. Resultados

Algunos problemas de optimización combinatoria, requieren resolver clasificación por particiones. Otras aplicaciones demandan particiones que respeten la compacidad geométrica y/o homogeneidad para variables o balanceo en el número de objetos que conforman los grupos. En estas propuestas, las condiciones espaciales de las variables geográficas en la clasificación favorecen la creación de regiones espacialmente compactas, lo cual se traduce en la satisfacción de la restricción de continuidad geográfica. Por otra parte, la utilización de este tipo de aproximaciones implica, en algunos casos, otorgar gran importancia a las variables geográficas para garantizar la satisfacción de la restricción de continuidad geográfica. Sin embargo, esto supone que el papel de las variables no geográficas (por ejemplo variables socioeconómicas) dentro del proceso de agregación pasaría a ser secundario, aun así, este tipo de variables es muy importantes cuando la agregación resuelve problemas como equilibrio entre variables, homogeneidad o balanceo en el número de grupos.

Es posible representar un esquema funcional del sistema SIGP en un diagrama. Supóngase que solo 2 opciones de clasificación están disponibles: grupos compactos y grupos homogéneos. En la figura 1, se puede observar que cada uno de los módulos funciona independiente o en combinación con otros módulos. Destaca una propiedad interesante de la estructura del sistema: la posibilidad de que estos módulos sean trasladados a otras aplicaciones o agregar otros módulos. En la figura 2, el módulo de selección de datos dentro del módulo de consultas entrega un subconjunto de Agebs de interés, de tal modo que genera una matriz de distancias Euclídeas ajustada para ser procesada por las disponibles opciones de agrupamiento.

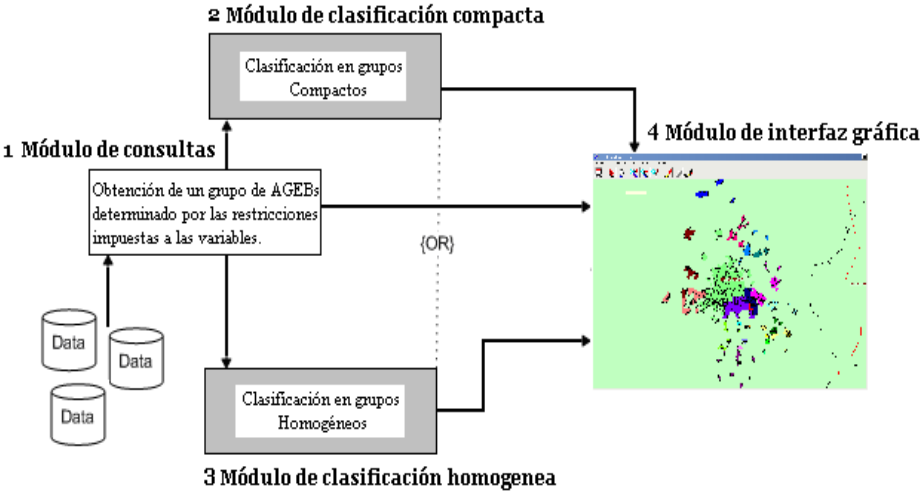
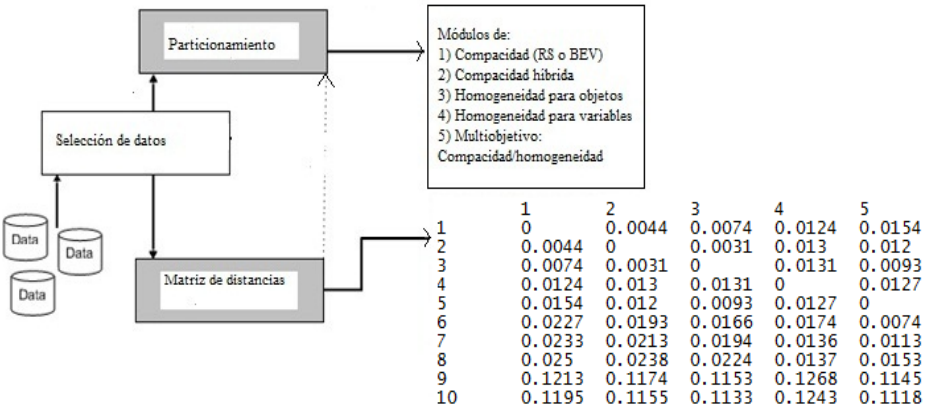


Fig. 1. SIGP básico



	1	2	3	4	5
1	0	0.0044	0.0074	0.0124	0.0154
2	0.0044	0	0.0031	0.013	0.012
3	0.0074	0.0031	0	0.0131	0.0093
4	0.0124	0.013	0.0131	0	0.0127
5	0.0154	0.012	0.0093	0.0127	0
6	0.0227	0.0193	0.0166	0.0174	0.0074
7	0.0233	0.0213	0.0194	0.0136	0.0113
8	0.025	0.0238	0.0224	0.0137	0.0153
9	0.1213	0.1174	0.1153	0.1268	0.1145
10	0.1195	0.1155	0.1133	0.1243	0.1118

Fig. 2. Matriz de distancias*

3.1. Módulo de clasificación compacta

El módulo de clasificación se describe en pseudocódigo para que pueda distinguirse las estructuras de datos de cada componente y el enlace que permite la comunicación entre estos. El módulo de clase *CGroup* es independiente y contiene dos archivos: *modCompactness* y *modVarDt*. El componente dependiente es una instancia de esta clase llamada *compactenessGroup*. El archivo *modVarDt* define dos estructuras de datos utilizadas para la entrada y salida de este módulo:

1) *PublicTypeTItemClusters* *y* *2) PublicTypeTClusters*

```
agebKey As Integer                    n As Integer
cluster As Integer                    nClusters As Integer
End Type                              ítem () As TItemClusters
                                      End Type
```

Como parámetro de entrada, el modulo Clasificación requiere del método *obtain-Compactness*, el cual, almacena los datos que se agruparán de dos maneras: 1)Con una variable de tipo *TCluster* que indica el número de Agebs que se particionan (*n*). El número de grupos a obtener (*nClusters*) se realiza con un arreglo de tipo *TItem-Cluster* para guardar las claves de los Agebs (*agebKey*) además del cluster al que pertenece (*cluster*) y 2) Mediante un apuntador a la base de datos que almacena los Agebs a clasificar y las variables asociadas que denotan el nombre del campo que señalan las Agebs (además de los nombres de los campos con las distancias entre los Agebs y los nombres de las tablas utilizadas). El resultado de la clasificación es recogida en una variable de tipo *TClusters* (ver figura3).

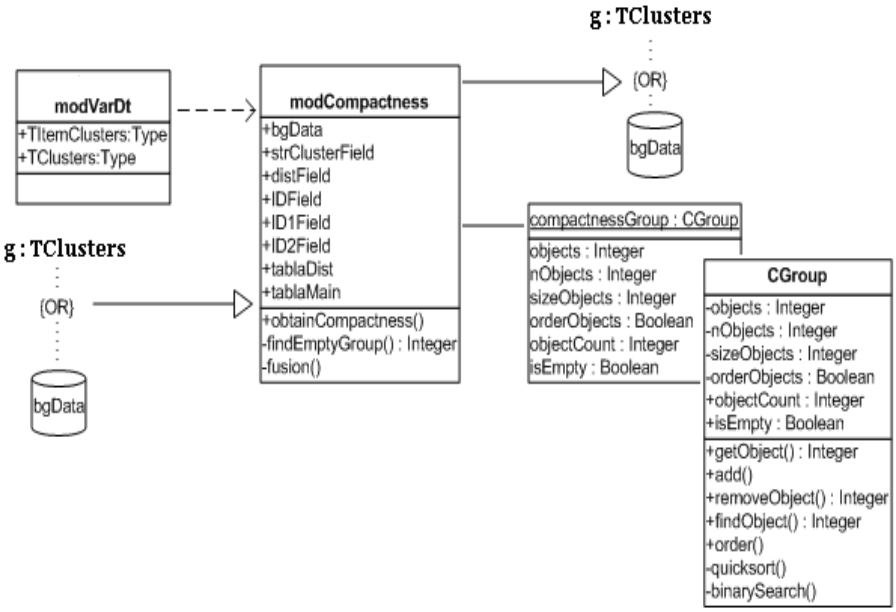


Fig. 3. Módulo de clasificación en grupos compactos

3.2. Módulo de clasificación homogénea

Este módulo reutiliza los componentes *modVarDt* y *CGroup* que maneja el módulo de clasificación compacta y definen una plantilla que puede ser usada en otras rutinas. Un archivo módulo *modHomogeneity* produce una instancia de la clase *CGroup* y utiliza la definición de las estructuras de datos *TItemClusters* y *TClusters* para el control de datos de entrada-salida (ver figura 4).

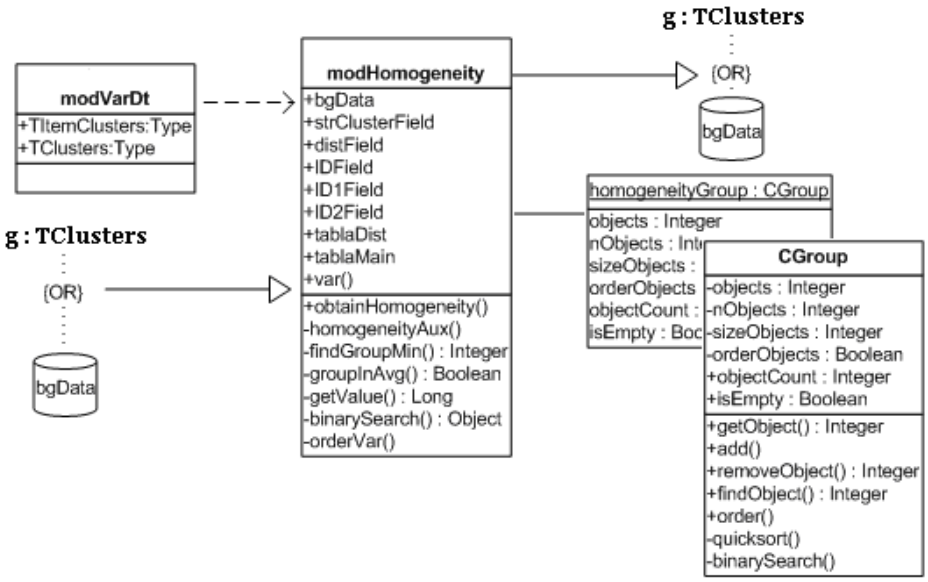


Fig. 4. Módulo de clasificación en grupos homogéneos

3.3. Módulo de interfaz gráfica

En general, los algoritmos de Particionamiento que se han desarrollado, generan tres archivos de resultados [2, 3, 4, 5]. El archivo que nos interesa es el código de Ageb con el número de grupo que le corresponde (formato de archivo para la interacción con MapX) [8]. Las bondades de este formato aseguran que un mapa es generado aun cuando el agrupamiento haya sido implementado con otro lenguaje y el archivo de entrada respete propiedades del formato (ver figura 5).

Los mapas pueden tener diferentes formatos y la extensión *tab* para capas de mapas es aceptada por el SIG MapInfo. Dependiendo de las capas (de población, mares ríos, carreteras etc.), es posible implementar procedimientos que habiliten funciones de componentes del SIG en combinación con el lenguaje visual. La capa que hemos utilizado consiste en la división geográfica de Agebs.

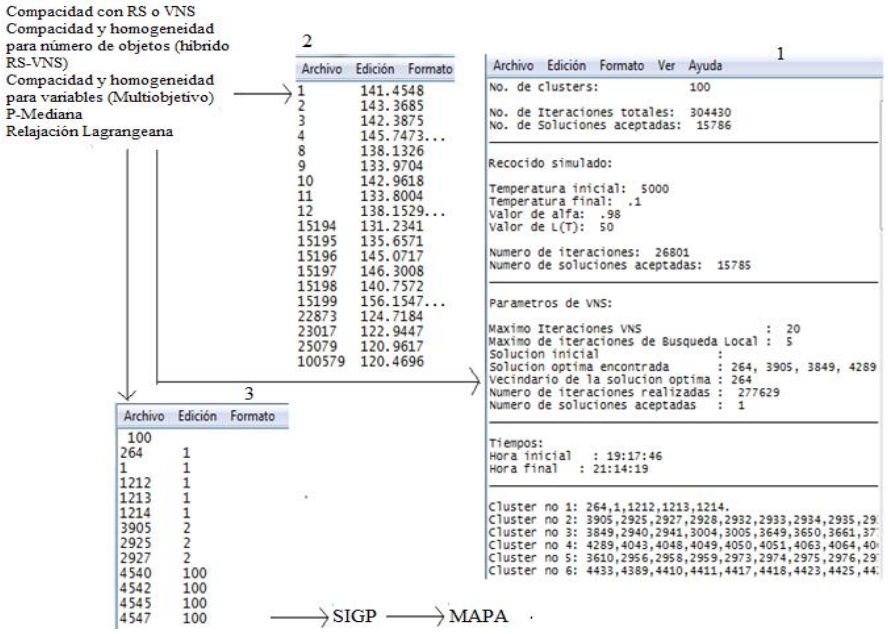


Fig. 5. Diagrama de flujo de datos

En este punto, MapInfo cuenta con una herramienta para el control y administración de las capas (layers). Este control se conoce como MapX. Un análisis exhaustivo de MapX ha sido necesario para explotar sus funciones y propiedades con el fin de construir la interfaz particionamiento-mapa para el SIGP. MapX es un componente ActiveX que permite integrar la funcionalidad de MapInfo en distintas aplicaciones. Se integra usando lenguajes de programación estándar: Visual Basic, Visual C++, Delphi, PowerBuilder y Oracle Express Objects. Desde luego estas características determinaron la elección de MapX.

Los componentes fundamentales de este módulo central son dos "formas" y 3 módulos: 1) *frmMapXInterfaz*, 2) *frmClusterProperties*, 3) un módulo de clase *Color* y 4) dos archivos de modulo llamados *modFuncMapymodMap*. El módulo *Color* y *modFuncMap* son independientes del módulo central, pero la instancia de la clase *Color* si es dependiente del módulo. El módulo de interfaz gráfica, establece un llamado a la forma *frmMapXInterfaz*. Obedeciendo a los valores de las variables que son enviadas como argumentos, se colorean determinadas zonas de un estado para mostrar el mapa correspondiente en una ventana. La figura 6 muestra la interacción entre los componentes de este módulo.

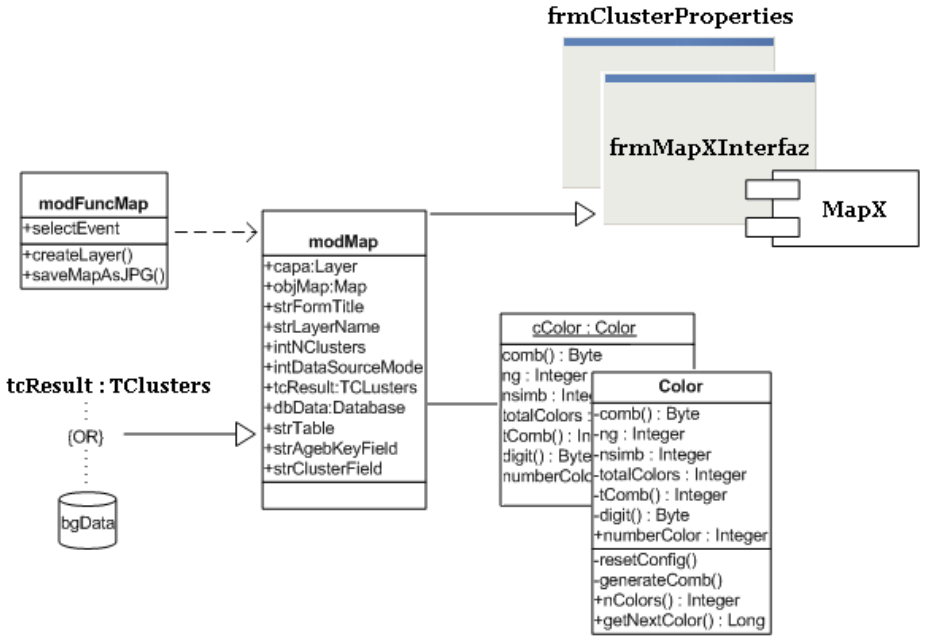


Fig. 6. Módulo de interfaz gráfica.

La forma *frmclusterProperties* ofrece opciones para que el usuario final elija el color de los grupos u ocultar algunos de estos.

El propósito de *modFuncMap* es articular y reunir todas las funciones generales a un control MapX, mientras que *modMap* tiene propiedades y métodos que interactúan sobre la información gráfica mostrada en *frmMapXInterfaz*.

La declaración de los parámetros de entrada, a nivel código, se reduce como sigue:

Public Const DATABASE_MODE As Integer = 1
Public Const TCLUSTER_MODE As Integer = 2
Public formTitle As String
Public layerName As String
Public nClusters As Integer
Public dataSourceMode As Integer
Public tcResult As TClusters
Public dbData As DAO.Database
Public strTable As String
Public strAgebKeyField As String
PublicstrClusterField As String

formTitle. Texto que va a contener la ventana.
layerName. Nombre de la capa (layer) de MapX a ser mostrada.
nClusters. Número de grupos de zonas, por lo que se generaran para los *nCluster*, tonos diferentes para cada grupo utilizando la clase *Color*.

dataSourceMode. Especifica el modo en que se van a introducir los argumentos. Esta estructura cuenta con dos modos y dependiendo del módulo escogido, se especifican los siguientes parámetros:

1. si *datasourceMode* = TCLUSTER_MODE

tcResult. Objeto que contiene los Agebs y el grupo al que pertenece cada uno.

2. si *datasourceMode* = DATABASE_MODE

dbData. Referencia a la base de datos que contiene la información

strTable. Nombre de la tabla en donde se encuentran los datos

strAgebKeyField. Nombre del campo que contiene a las claves de los Agebs

strClusterField. Nombre del campo que contiene la pertenencia de grupo de cada AGEB.

El tipo de dato *TClusters* también es una opción que comunica información de entrada al módulo como se puede apreciar en la figura 6.

Por cada grupo (*nCluster*), un llamado es hecho al método *createLayer* definido en *modFuncMap*. El método *createLayer* elabora una capa (*Layer*) con el color que es dado como argumento (se forma con la clase *Color* para no repetir tonos entre los grupos) y este método la sobrepone en la capa actual (*layerName*). Las operaciones zoom, colocar texto en el mapa, figuras, cambiar el estilo del texto y copiar el mapa al portapapeles son habilitadas por *frmMapXInterfaz,*y se facilitan a través del modelo de objetos de MapX.

3.4. Integración de los módulos

Integrados los módulos principales, la inserción de ventanas (*forms*) es importante para la comunicación con el usuario. Los archivos *modProject* y *modDatos*, han sido implementados para permitir la incorporación de otros módulos de agrupamiento, alcanzando así, una propiedad indispensable de calidad en software: Portabilidad. Por otra parte, *ModProject* y *modDatos* definen variables que registran información en tiempo real de las acciones del usuario y se acompañan de métodos que acceden a las bases de datos. En la figura 7 se observan los llamados entre módulos (línea punteada). La línea sólida especifica entrada y salida, mientras que la línea punteada define las llamadas entre componentes. La palabra "cat." se refiere a la tabla catálogos de la base de datos principal. En esta figura, también se distinguen "módulos separados", los cuales pueden ser reemplazados por otros módulos y generar el mapa correspondiente. Por otra parte, es necesaria la disponibilidad de la capa geográfica de Agebs (censo, ríos, montañas, etc.). En la figura 7 se aprecian distintos módulos de agrupamiento, sin embargo, la primera versión de este sistema disponía de dos opciones: compacidad y homogeneidad. En recientes actualizaciones, se incorporaron al SIGP otros procedimientos de particionamiento con el fin de medir la eficiencia de compatibilidad, flexibilidad, reusabilidad y portabilidad de los procedimientos vistos como módulos. Un reto es reunir todas las opciones de agrupamiento que hemos venido desarrollando en un sistema integrado, además de actualizar la base de datos con todas las entidades federativas de México y ofrecer un sistema robusto capaz de resolver problemas de agrupamiento de muestreo o de zonas para extractos poblacionales e incluso reagrupamiento para fines de distritación básica.

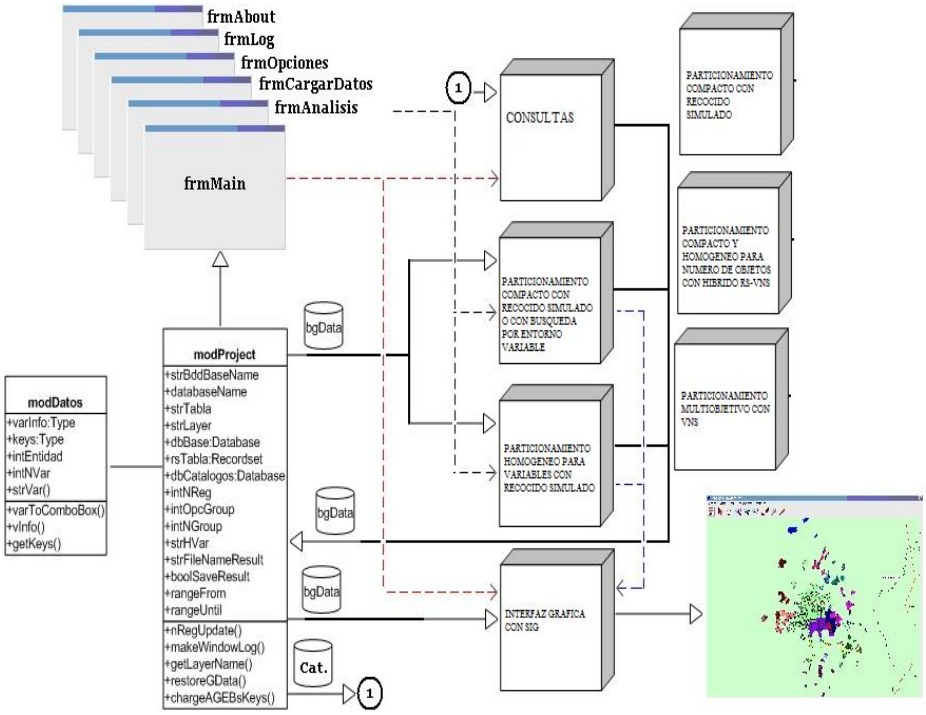


Fig. 7. Integración de todos los módulos en la aplicación final.

4. Discusión de resultados

En esta sección exponemos brevemente 2ejemplos de particionamiento: homogéneo para variables poblacionales y Relajación Lagrangeana RL. Como caso de estudio se ha considerado la Zona Metropolitana del Valle de Toluca (ZMVT), compuesta de 469 Agebs.

Ejemplo 1: Supóngase que serán agrupados los Agebs de la ZMVT, a la consulta *población femenina por encima del promedio* donde la variable *"población total" (Z001)*mantiene homogeneidad. Se requieren 16 grupos. La tabla 1 que muestra el valor que tiene cada grupo para la variable Z001 y el número de elementos que tiene cada grupo y concluimos que la homogeneidad resultante es satisfactoria. La figura 8muestra el mapa para la división de 16 grupos a la consulta población femenina por encima del promedio, manteniendo homogeneidad en la variable *"población total"*.

Grupo	Valor	Elementos
1	62579	14
2	64280	14
3	62842	12
4	62849	15
5	63470	13

6	64880	14
7	65012	12
8	64149	15
9	62547	12
10	63575	13
11	64656	10
12	65166	15
13	64286	15
14	65204	13
15	64250	15
16	64658	14

Tabla 1. Resultados de homogeneidad para el ejemplo 1

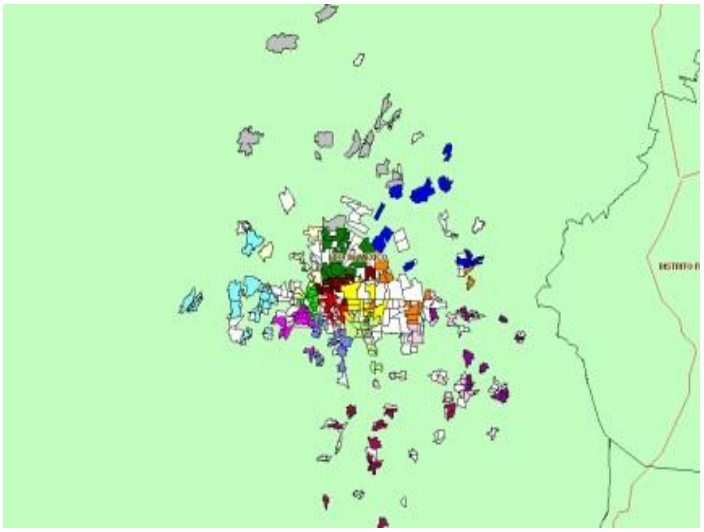


Fig. 8. 16 grupos para *población femenina por encima del promedio*

Ejemplo 2: Se desarrolló un esquema de relajación Lagrangena para el problema de la P-mediana. Para obtener cotas inferiores, se resuelve el dual Lagrangeano utilizando un algoritmo de optimización subgradiente. Este problema se implementó en FicoXpress [5] y las soluciones se han comparado con los resultados de un algoritmo exhaustivo PAM [9]. En la tabla 2 se concentraron resultados para 24, 47 y 94 grupos y en la figura 9 se presenta el mapa para 24 grupos y los resultados del agrupamiento se describieron en el formato que SIGP requiere (ver figura 9).

			FICO XPRESS		PAM
Instancia	Grupos	Solución Óptima	Cota Inferior	Tiempo	Tiempo
			Relajación lineal		
1	24	9.1986	9.1986	42.5	79

2	47	5.7338	5.7338	35.05	431
3	94	3.2089	3.2086	33.26	2188

Tabla 2. Resultados RL para 3 instancias

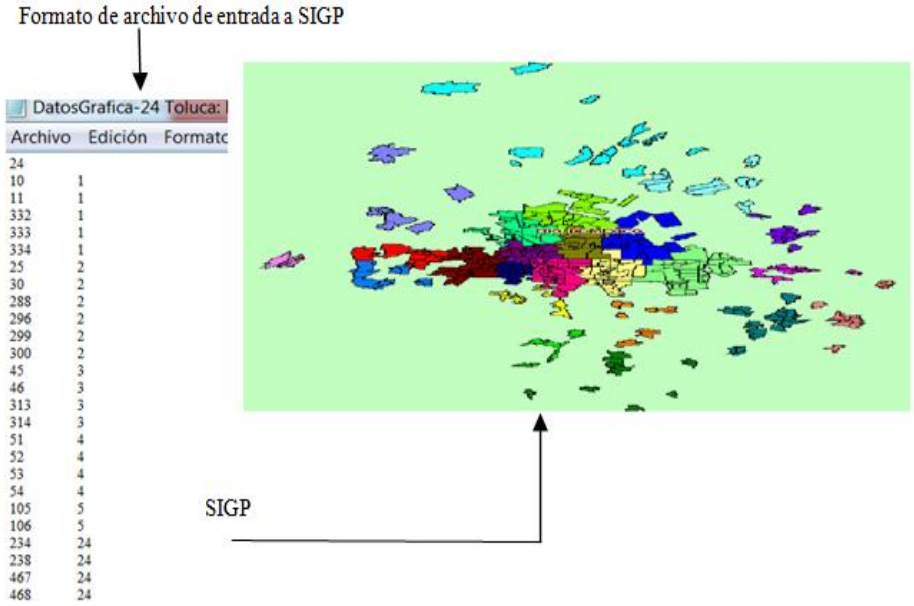


Fig. 9. Resultado de 24 grupos con RL

5. Conclusiones

El trabajo que hemos expuesto significa una importante contribución en el desarrollo de sistemas para problemas de agrupamiento geográfico que requieran resultados en mapas. La mayoría de los trabajos presentan sus resultados concentrados en una tabla con el costo de la función objetivo, el tiempo de cómputo y valor de los parámetros del algoritmo. Sin embargo, las agrupaciones deber ser visibles para asegurarse de que el agrupamiento responde correctamente en cuando a compacidad, conexidad y homogeneidad. El sistema que hemos expuesto en este trabajo responde con claridad los resultados de particiones compactas en un mapa. Subrayamos que el resultado en mapas es posible siempre que se encuentre disponible en Agebs, la capa de una zona a agrupar con formato MapX (extensión tab). La desventaja reside justamente para el caso contrario, cuando la capa de la zona no es extensión tab, el SIGP no puede construir el mapa debido tanto a las especificaciones del sistema como a las propiedades de MapX. Por otra parte, si de Particionamiento compacto se trata, es deseable que las agrupaciones logradas también sean conexas, y se han tenido iniciativas para demostrar analíticamente que los algoritmos que hemos implementado son compactos y

conexos. Atendiendo este aspecto, los resultados vistos en mapas revelan que se cumple conexidad y compacidad gráficamente.

Referencias

1. E. Zamora. Implementación de un Algoritmo Compacto y Homogéneo para la Clasificación de AGEBs bajo una Interfaz Gráfica. Tesis de Ingeniería en Ciencias de la Computación, Benemérita Universidad Autónoma de Puebla, Puebla, México, 18-27, 2006.
2. B. Bernábe, J. Espinosa, J. Ramírez, and M. A. Osorio. Statistical comparative analysis of Simulated Annealing and Variable Neighborhood Search for the Geographical Clustering Problem. *Computación y Sistemas*, vol. 42-3, pp. 295-308, 2009.
3. B. Bernábe, D. Pinto, E. Olivares, J. Vanoye, R. González, J. Martínez. El problema de homogeneidad y compacidad en diseño territorial. XVI CLAIO Congreso Latino-Iberoamericano de Investigación Operativa, 2012.
4. B.Bernábe, C. Coello, M. A. Osorio. A Multiobjective Approach for the Heuristic Optimization of Compactness and Homogeneity in the Optimal Zoning. *JART Journal of Applied Research and Technology*, vol.10-3, pp. 447-457, 2012.
5. J. Díaz., B. Bernábe B., Luna E., Olivares, J. L. Martínez. Relajación Lagrangeana para el problema de particionamiento en datos geográficos. *Revista de Matemática Teoría y Aplicaciones* vol. 19-2, pp. 43-55, 2012.
6. E. Pizza, A. Murillo., &J. Trejos. Nuevas técnicas de particionamiento en clasificación automática. *Revista de Matemática: Teoría y Aplicaciones*, vol. 6-1, pp.1–66, 1999.
7. E. Vicente, L. Rivera, D. Mauricio. Grasp en la resolución del problema de cluster. ISSN: 1815-0268, vol. 2- 2, pp. 16-25, 2005.
8. MapX Developer´s guide. MapInfo Corporation. Troy, NY.
9. L. Kaufman, P. Rousseeuw. Clustering by means of medoids. Statistical Data Analysis based on the L1 Norm, North-Holland, Amsterdam , pp. 405-416, 1987.

Aproximación GRASP-VND para el problema de asignación cuadrática

Rogelio González, Beatriz Bernábe, Martín Estrada,Antonio Alfredo Reyes
Benemérita Universidad Autónoma de Puebla,
Facultad de Ciencias de la Computación, Puebla, Pue. México
{rgonzalez,mestrada}@cs.buap.mx
{beatriz.bernabe,alfred.reymo}@gmail.com

Resumen. En este artículo se presentan soluciones al problema de asignación cuadrática (Quadratic Assignment Problem - QAP), el cual es clásico en optimización combinatorial y está clasificado como un problema NP-Completo. Este problema consiste en encontrar una asignación óptima de *n* instalaciones a *n* ubicaciones, de tal manera que se minimice el costo de transportación de materiales entre las instalaciones y las ubicaciones. Se debe considerar la distancia entre las ubicaciones, así como el flujo de materiales entre las instalaciones. Para buscar soluciones a QAP, hemos implementado un procedimiento que une metaheurística de un procedimiento denominado Randomized Adaptive Search Procedure Greedy (GRASP), con otro llamado Variable Neighborhood Search Descent (VND).

1 Introducción

El QAP es uno de los problemas de alta complejidad computacional de Optimización Combinatoria (OC) [1] y consiste en encontrar una permutación de asignación óptima de *n* instalaciones a *n* localidades con el propósito de minimizar el costo de transporte, dadas dos matrices simétricas una de distancias y otra de flujos. El QAP fue propuesto por Koopmans y Beckmann en 1957 [4], en 1976 Shani y González probaron que QAP es un problema NP-completo [5]. Hasta hoy sólo se han encontrado soluciones óptimas usando métodos exactos para instancias de tamaño menores que 30 [6]. El QAP aparece en muchas aplicaciones, tales como el diseño de teclados de computadora, la programación de manufactura, el diseño de terminales en aeropuertos y procesos de comunicaciones, entre otras [7, 8]. El algoritmo exacto más popular para resolver el QAP es el de ramificación y acotamiento (Branch and Baund B&B) con algunas variantes [9]. El primer algoritmo exacto de ramificación y acotamiento en paralelo fue propuesto por Roucairol [9] , sin embargo en los últimos años se ha propuesto métodos de búsqueda se soluciones conocidos como metaheurísticas (MH) que son procedimientos de aproximación de propósito general, tales como los algoritmos genéticos, el recocido simulado, búsqueda tabú y GRASP [6] entre otras. La característica principal de una MH es que producen soluciones muy cercanas a la óptima en un tiempo razonable de cómputo. El modelo matemático del QAP como problema de OC es la siguiente:

Sean N= {1,2,..., n} y F= (f_{ij}) y D= (d_{kl}) dos matrices cuadradas de n x n simétricas se trata de encontrar una permutación $p \in \Pi_N$ que minimice $\sum_{i=1}^{n}\sum_{j=1}^{n} f_{ij}\, d_{P(i)P(j)}$

Donde Π_N es el conjunto de todas las permutaciones del conjunto N. Los datos de entrada son f_{ij} representa el flujo de materiales de la planta i a la planta j y d_{kl} es la distancia de la ubicación ciudad k a la ubicación l.

Supongamos que las matrices de flujo y las distancias F y D respectivamente son simétricas, entonces se tiene que $f_{ij} = f_{ji}$ y $d_{kl} = d_{lk}$, además fij = 0 y d_{kl} = 0, para $i = j$, entonces podemos escribir las instancias de datos en una sola matriz que compacte a F y D como sigue:

$$C = \begin{bmatrix} 0 & d_{12} & d_{13} & d_{14} & \cdots & d_{1n} \\ f_{21} & 0 & d_{23} & d_{24} & \cdots & d_{2n} \\ f_{31} & f_{32} & 0 & d_{34} & \cdots & d_{3n} \\ \vdots & \vdots & \vdots & \ddots & \cdots & \vdots \\ & & & & 0 & d_{n-1\,n} \\ f_{n1} & f_{n2} & \cdots & \cdots & f_{n\,n-1} & 0 \end{bmatrix}$$

Se encuentra disponible una página en Internet llamada QAPLIB [6], en donde podemos hallar las disertaciones, artículos, resultados e instancias de prueba del QAP más recientes.

1.2 GRASP

GRASP es un procedimiento iterativo en donde cada paso consiste en una fase de construcción y una de mejora. En la fase de construcción se aplica un procedimiento heurístico constructivo para obtener una buena solución inicial. Esta solución se mejora en la segunda fase mediante un algoritmo de búsqueda local. La mejor de todas las soluciones examinadas se guarda como resultado final. [1], Festa y Resende muestran diferentes aplicaciones de GRASP [2].

La palabra GRASP proviene de las siglas de *Greedy Randomized Adaptive Search Procedures* que en castellano sería algo así como: Procedimientos de Búsqueda basados en funciones "Greedy" Aleatorizadas Adaptativas. Las principales componentes de la fase de construcción son: una función de evaluación voraz, un procedimiento de elección al azar y un proceso de actualización adaptativo [3].

```
Procedure GRASP (iter, inicio)
1   Input(instancia);
2       While i ≤  iter  do
3            S0← Construcción (inicio);
4            Sol ← Busqueda_Local(S0)
5            ActualizaSolucion(Sol,);
6               end {while}
8   Return (MejorSol);
End {GRASP}
```

Fig. 1. Pseudocódigo genérico para GRASP

El objetivo de la fase de construcción es generar una solución factible de buena calidad, y entre más eficiente sea esta fase en términos de calidad se espera que el trabajo de la fase de postprocesamiento sea menor y así cada iteración GRASP ocurriría más rápido. En la fase de construcción, GRASP genera una lista de candidatos formada por elementos de alta calidad, esta lista es llamada lista restringida de candidatos (LRC). La solución inicial se construye iterativamente considerando un elemento cada vez. En cada iteración del procedimiento constructivo, un elemento es elegido en forma aleatoria de la lista de candidatos para añadirlo a la subestructura como parte de la solución que se construye. La adición de un elemento a la subestructura se determina mediante una función de tipo voraz, esta función mide el beneficio de la selección del cada elemento, mientras la selección de un elemento de esa lista de candidatos depende de los que se hayan elegido previamente.

```
Procedure Búsqueda Local ( p, V(p), s )
1      While s no sea optima local do
2          Encontrar una  mejor solución t ∈ V(s);
3          Sea s = t ;
4      End ;{ While }
5      Return ( s como optima local )
End {local};
```

Fig. 2. Pseudocódigo genérico para la búsqueda local

1.3 Búsqueda de Vecindad Variable Descendente

La VNS también conocida como búsqueda de entorno variable está basada en un principio simple: cambiar sistemáticamente de estructura de entornos dentro de la búsqueda local. Se han realizado muchas extensiones, principalmente para permitir la solución de problemas de gran tamaño, siempre tratando de conservar la simplicidad del esquema básico. Se han implementado extensiones, híbridos y aplicaciones en

inteligencia artificial [12]. La VNS está basada en tres hechos: un mínimo local con una estructura de entorno no lo es necesariamente con otra, un mínimo global es mínimo local con todas las estructuras de entornos y para muchos problemas, los mínimos locales con la misma o distinta estructura de entornos están relativamente cerca. Esta última observación, aunque empírica implica que los óptimos locales proporcionan información acerca del óptimo global. La VND es una de las variantes de la VNS se pueden considerar descendente [12]. La VNS ha sido incorporada con otras metaherísticas que han dado lugar a diversos híbridos con búsqueda tabú (*Tabu Search* TS), GRASP y búsqueda multi-arranque (*MultiStart Search* MS). Se ha aplicado en diversos problemas de inteligencia artificial como satisfactibilidad, aprendizaje en redes bayesianas y la planificación, también en problemas de optimización de empaquetado, localización problemas de rutas así como en el problema del agente viajero [12]. En la Fig. 3 se da el pseudocódigo de VND.

```
Procedure VND ( p, Vk(p), s )
1 Input(V_k , ∀ k=1,…,k_max)
2    While k < k_max l do
3       Encontrar una  mejor solución t ∈ V_k(s);
4         Sea s = t ;
5          k←k+1
4     End ;{ While }
5    Return ( s como optima local )
End {local}:
```

Fig. 3. Pseudocódigo genérico VND

2 GRASP Y VND para el QAP

2.1 GRASP para el QAP

El diseño de este GRASP ha sido empleado por algunos investigadores para resolver el QAP para diferentes instancias Li, Resende y Pardalos [1]. El pseudocódigo se muestra en la Fig. 1. A continuación se describe la fase de construcción en dos etapas.

Primera etapa. Las dos asignaciones iniciales se hacen simultáneamente, específicamente diremos que el recurso i es asignado a la localidad k y el recurso j es asignado a la localidad l, cuando su costo correspondiente a este par de asignaciones es $f_{ij}\ d_{kl}$.

Sean α, β, ($0 < \alpha, \beta < 1$) los parámetros que restringen la lista de candidatos, $F = (f_{ij})$ y $D = (d_{kl})$ las matrices simétricas $n \times n$ con ceros en la diagonal de entrada con las cuales se forma una matriz compacta cuadrada no simétrica.

Sea $[x]$ la parte entera de x y sea $m = n\,(n\text{-}1)/2$ el número de entradas en los triángulos inferior y superior de la matriz compacta. Se procede a listar estas entradas de las distancias y los flujos en orden creciente y decreciente respectivamente, es decir:

$$d_k\,\mathbf{1}_{l^1} \le d_{k\,2\,l^{\varsigma}} \le d_{k\,3\,l^3} \le \ldots \le d_{k\,m\,l^m}$$

$$f_{i1j1} \geq f_{i2j2} \geq f_{i3j3} \geq \ldots \geq f_{imjm}$$

Ahora tenemos dos listas ordenadas; usaremos el parámetro β para restringir ambas listas, por lo cual se cortan hasta el elemento [βm]. Se genera una nueva lista de costos multiplicando las distancias por los flujos en el orden correspondiente, así, se tiene la nueva lista:

$$f_{i1j1} d_{k1l1}, f_{i2j2} d_{k2l2}, f_{i3j3} d_{k3l3}, \ldots, f_{imjm} d_{kmlm}$$

La cual se ordenada en forma creciente, ahora usamos el parámetro α para obtener la lista restringida y definitiva de los candidatos (LRC), de la cual sólo se tomaran los primeros [αβm] elementos y se elegirá aleatoriamente un elemento f_{ij} d_{kl} que representa un costo de hacer un par de asignaciones (i, *k*) y (*j, l)*, es decir, tenemos dos componentes de la solución, que para simplificar será escrita como permutación, donde la componente *k*-ésima y la componente *l*-ésima son colocadas. Aquí se puede apreciar la componente aleatoria del método. Con esto concluye la primera etapa de la fase de construcción.

Segunda etapa. En esta etapa lo que se pretende es completar la solución inicial calculando las *n*-2 asignaciones restantes, mediante un procedimiento ávido que produce una a una las asignaciones que tienen el costo mínimo con respecto a las asignaciones ya existentes y que en caso de empate se romperá aleatoriamente y apoyándose en una componente adaptativa que se encarga de actualizar la solución a medida que esta se va construyendo.
Sea:

$$\Gamma = \{ (j_1, l_1), (j_2, l_2), ..., (j_r, l_r) \}$$

El conjunto de asignaciones que está en construcción. La etapa 2 inicia con $|\Gamma| = 2$ a consecuencia de los resultados de la etapa1.

Sea $C_{ik} = \sum_{(j,l)\in\Gamma} f_{ij} d_{kl}$ el costo de asignar la fábrica *i* a la localidad *k* con respecto a las asignaciones ya existentes. Seleccionamos de las parejas (*i*, *k*) no asignadas la que tenga el costo mínimo C_{ik}, en esto consiste el procedimiento ávido.

En esta parte también hay una lista restringida de candidatos, se ordenan los C_{ik} en forma creciente y se toma aleatoriamente uno de los primeros [αz], donde z es la cantidad de parejas aun no asignadas, nuevamente aparece la componente aleatoria.

La componente adaptativa de GRASP tiene como función actualizar el conjunto Γ adicionando nuevas parejas asignadas, es decir $\Gamma = \Gamma \cup \{(i, k)\}$

Al finalizar esta etapa concluye también la primera fase, se ha construido una solución contenida en el conjunto $\Gamma = \{ (j_1, l_1), (j_2, l_2), ..., (j_n, l_n) \}$ ordenando las primeras componentes de las parejas, tomamos las segundas componentes para formar la permutación equivalente a la solución. En resumen tenemos una solución de buena calidad para arrancar la segunda fase.

2.2 GRASP_VND para QAP

Empezamos la segunda fase o fase de mejoramiento tomando como solución inicial la solución obtenida en la primera fase de GRASP y en la siguiente fase se emplea la metaheurística de VND con tres estructuras de entorno como adyacente, λ-intercambio y 2-intercambio. En cada estructura vecinal encuentra un mínimo local con la clásica búsqueda descendente que consiste en reemplazar iterativamente la solución actual por el resultado de la búsqueda local, siempre que se obtenga una mejor solución. Así la unión de GRASP-VND produce una aproximación a los óptimos como se discute en la siguiente sección.

3 Resultados y experiencia computacional

Se muestran los resultados fueron obtenidos para doce instancias, propuestas por Nugent [10]. Todos los resultados mostrados se obtuvieron restringiendo la lista de candidatos con los parámetros $\alpha = 0.5$ y $\beta = 0.1$ determinados experimentalmente. Las estadísticas de las tabla se obtuvieron de realizar 10 corridas del programa secuencial, para cada instancia, las tablas contienen: **Problema** : nombre de la instancia, ***n***: tamaño de la instancia **MVC**: mejor valor conocido, **MVE**: mejor valor encontrado por GRASP-VND, **CM**: cota mínima, **PE:** porcentaje de error con respecto a la cota mínima, **PSO**: porcentaje en que se obtuvo la solución óptima o el mejor valor conocido, **Iter**: promedio de iteraciones GRASP-VND en que se obtuvo **MVC**. Finalmente la columna correspondiente a **TCPU** contiene el promedio del tiempo de ejecución de las corridas que alcanzaron el mejor valor de la función objetivo.

Tabla 1. Resultados GRASP-VND para las matrices de Nugent.

Problema	*n*	MVC	MVE	CM	PE	PSO	Iter	TCPU
Nug5	5	25	25	25	0	100	2	0
Nug6	6	43	43	43	0	100	3	0
Nug7	7	74	74	74	0	100	5	0
Nug8	8	107	107	97	9.34%	100	4	0
Nug12	12	289	289	264	8.65%	95	65	0.23
Nug15	15	575	575	542	5.73%	90	67	0.79
Nug20	20	1285	1285	1119	12.91%	100	73	3.61
Nug21	21	1219	1219	1004	17.63%	10	485	32.97
Nug22	22	1798	1798	1417	21.19%	85	250	24.05
Nug24	24	1744	1744	1419	18.63%	70	435	63.89
Nug25	25	1872	1872	1532	18.16%	55	420	56.65
Nug30	30	3062	3062	2886	5.75%	5	529	224.78

Los números en la tabla 1 de la columna MVC son los reportados en la literatura como los valores óptimos desde $n = 5$ hasta $n = 30$, se observa que en todas las corridas se obtuvo el óptimo para las instancias de dimensión 5, 6, 7, 8 y 20 en ambas versiones de la búsqueda local. En la tabla 2 se muestran las permutaciones de asignación de los recursos a las localidades cuyo valor asociado es en todos los casos los de la tercera columna.

Tabla 2. Permutaciones de asignación óptimas o con **MVE.**

Problema	Permutación de mejor asignación
Nug5	3,4,5,1,2
Nug6	6,5,4,3,2,1
Nug7	1,2,5,3,4,7,6
Nug8	3,4,8,2,1,5,6,7
Nug12	5,9,1,8,12,11,3,7,2,10,6,4
Nug15	1,2,7,6,14,13,9,4,5,11,10,15,3,8,12
Nug20	19,7,4,6,17,20,18,14,5,3,9,8,15,2,12,10,16,1,11,13,10
Nug21	3,16,19,21,15,9,4,5,18,10,12,2,17,14,8,11,1,13,6,7,20
Nug22	16,22,17,3,11,9,18,14,20,19,6,8,10,2,1,7,12,4,15,13,21,5
Nug24	7,6,17,23,5,3,9,20,10,8,21,2,4,18,13,11,19,16,14,24,12,15,22,1
Nug25	24,4,13,11,5,18,17,14,19,22,10,6,21,12,20,8,1,3,23,15,7,25,16,2,9
Nug30	17,26,3,7,30,28,15,16,21,22,10,29,27,2,6,9,18,5,14,1,25,11,12,23,13,24,4,19,20,8

4 Conclusiones

Se establece de los resultados que es ventajoso utilizar la estrategia hibrida de GRASP con VND que explora parcialmente una vecindad actualizando la búsqueda cuando encuentra una mejor solución que la actual, esto produce que se disminuya el tiempo de ejecución, sin perder diversificación en la búsqueda y obtener con mayor frecuencia el MVC. También se deduce que la estrategia de evaluar todos los vecinos de una solución retarda cada iteración con lo cual se consume mucho tiempo de CPU.

Existen otras estructuras vecinales más elaboradas con las cuales se puede experimentar como son Corrimiento, *N**, 3-intercambio.

La fase de mejoramiento de GRASP, puede complementarse con un esquema de intensificación como reencadenamiento de trayectorias [11].

La metaheurísticas búsqueda dispersa puede ser desarrollada para resolver el QAP utilizando la fase constructiva de GRASP para generar la población inicial y construir el conjunto referencia.

Referencias

1. Y. Li, P.M. Pardalos, M.G.C. Resende. A Greedy Randomized Adaptative Search Procedure for the Quadratic Assignment Problem. In P.M. Pardalos and H. Wolkowicz, editors, *Quadratic assignment and related problems*, vol. 16 of DIMACS *Series on Discrete Mathematics and Theoretical Computer Sience*, pp 237-261. American Mathematical Society, 1994.

2. P. Festa and M.G. C. Resende. GRASP: An annotated Bibliography. To appear in *Ensays and Surveys on Metaheuristics*. P. Hansen and C.C. Riveiro, eds., Kluwer Academic Publihers, 2000.
3. A.D. Dáz, F. Glober, H. M.Ghaziri, J.L. Gonzalez, P. Moscato, F.T. Tseng. Optimización Heurística y Redes Neuronales en Dirección de Operaciones e Ingeniería. Editorial Paraninfo,1996.
4. T.C. Koopmans,and M.J. Beckmann. Assignment problems and the location of economic activities. *Econometrica*, vol 25: pp 53-76,1957.
5. S. Sahni and T. Gonzalez. P-complete aproximations problems. *J. Asssoc. Comp. Machine*. vol. 23 , pp 555-565, 1976.
6. R.E. Burkard, S.E. Karisch ,and F. Rendl, QAPLIB – A Quadratic Assignment Problem, Library, *internet*, http://www.imm.dtu.dk/~sk/qaplib/ins.html.
7. P.M. Pardalos, L.S. Pitssoulis, and M.G.C. Resende. A Parallel GRASP implemetation for the Quadratic Assignment problem. In A. Ferreira and J. Rolim, editors, *Parallel Algorithms for Irregularly Structured Problems – Irregular'94,* pp 111-130. Klower Academic Publishers, 1995.
8. E.L. Lawler. The Quadratic Assignment Problem. *Managnament Sci*., vol. 9, pp 586-599,1963.
9. C. Roucairol. A parallel branch and baund algorithm for the quadratic assignment problem. *Discrete Applied Mathematics*, vol. 18, pp 211-225, 1987.
10. C.E. Nugent, T.E. Vollman, and J. Ruml. An Experimental Comparison of Tecniques for the Assignment of Facilities to locations. *Journal of Operations Research,* vol. 16 : pp 150-173, 1969.
11. M.G.C. Resende, J. L. Gonzalez GRASP: Greedy Randomized Adaptative Search Procedures. *Revista Iberoamericana de Inteligencia Artificial No.19 pp.61-76* (2003).
12. Pierre Hansen, Nenad Mladenovic, Jose Andrés Moreno Pérez. Variable Neighbourhood Search. *Revista Iberoamericana de Inteligencia Artificial* No. 19 pp 79-92. 2003.

Obtención de descripciones significativas para una memoria corporativa

Cristal Karina Galindo Durán[1], R. Carolina Medina-Ramírez[2], y Mihaela Juganaru Mathieu[3]

[1,2] Departamento de Ingeniería Eléctrica, Universidad Autónoma Metropolitana, Unidad Iztapalapa, San Rafael Atlixco 186, Vicentina, 09340 Ciudad de México, D.F., e-mail[1]-cdgalindod@gmail.com, e-mail[2]- cmed@xanum.uam.mx
[3] Institut H. Fayol, École Nationale Supérieure des Mines de Saint Etienne, 158, cours Fauriel, 42023, Saint Etienne Cedex 2, France, e-mail - mathieu@emse.fr.

Resumen. En el presente artículo mostramos una metodología y una serie de pruebas para obtener descripciones significativas para los recursos de tipo documento de una memoria corporativa. Este enfoque se centra en la extracción de las características relevantes relacionadas al contenido de documentos. De este proceso se pueden obtener elementos que pueden servir como parte de índices de información que contribuyan a una recuperación rápida de información; así como la generación de matrices de frecuencias para la aplicación de diversos algoritmos de agrupamiento o minería de texto.

Palabras clave: Memoria corporativa, descripciones significativas, índices, minería de texto.

1. Introducción

El conocimiento dentro de una organización se denomina memoria corporativa (MC) o memoria organizacional (MO) y se define como: la representación explícita, tácita, consistente y persistente del conocimiento en una organización [1]

La memoria de una organización va creciendo y evolucionando cada día, esta memoria se compone de datos provenientes de las bases de datos, documentos textuales (imprimibles), documentos multimedia, personas, por mencionar algunos.

La gestión de los documentos[1] de una organización en el sentido más amplio impone poder guardar de manera permanente los documentos, poder consultarlos fácilmente a cada momento y también poder buscar información en colecciones particulares de documentos, o buscar documentos o personas sobre un tema, un concepto o buscar documentos/personas relacionados (vinculados) entre ellos [2].

[1] Por documento vamos a comprender una unidad conceptual de tratamiento completo; varios documentos van a formar colecciones y las colecciones conforman la memoria de la organización.

Para facilitar las actividades dentro de una organización se hace necesaria la implementación de aplicaciones inteligentes que con la ayuda de recursos semánticos del tipo ontología, descripciones semánticas de los recursos de una memoria [3], así como axiomas y razonadores mejoren los procesos para la gestión de recursos de información dentro de la organización, sin importar si ésta se encuentra centralizada o distribuida (nube corporativa).

Para ello es necesaria la implementación de índices que describan e identifiquen el documento[4], de tal forma que se pueda registrar ordenadamente los temas de los que trata dicho documento con el fin de permitir una clasificación, consulta y recuperación de los mismos.

Por otro lado la indexación semántica [5] implica que adicionalmente se utilicen herramientas de la gestión del conocimiento (técnicas de procesamiento de lenguaje natural, vocabularios controlados, tesauros y ontologías) y herramientas de la Web semántica [6], para encontrar los conceptos que representen con mayor precisión los documentos, de tal forma que su indexación permita consultas con un alto grado de relevancia (utilidad) para el usuario. En nuestro caso proponemos un índice semántico conformado por tres partes importantes: datos bibliográficos, contenido y vínculos de los términos contenidos en un documento en relación con otros términos contenidos en otros documentos, esto realizado con Datos Abiertos Enlazados (DAE). En la figura 1 se presenta la propuesta de nuestro índice semántico, el cual sirve para caracterizar el documento en tres ejes principales.

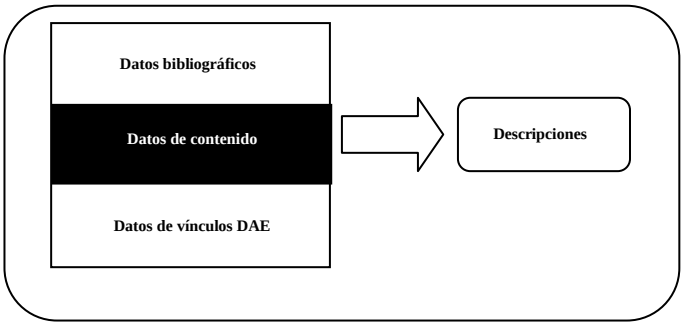


Fig. 1. Elementos que conforman el índice semántico propuesto

El interés de este artículo es centrarse en la obtención de los datos de contenido que describan el recurso de información, la cual se realiza mediante técnicas de procesamiento natural de lenguaje, así como minería de texto, la cual nos permite obtener patrones importantes en el recurso de forma automática y las cuales se pueden utilizar para la caracterización del mismo.

El resto del documento se estructura de la siguiente manera: la sección 2 presenta la metodología utilizada para la obtención de descripciones. En la sección 3 se presenta pruebas realizadas. Finalmente, la sección 4 presenta algunas conclusiones y trabajo futuro.

2. Metodología utilizada para la obtención de descripciones

La metodología aplicada corresponde a una parte del proceso global de la minería de texto, la cual consiste en 4 grandes etapas [8]: 1) Preparación de texto 2) Búsqueda de información, 3) Extracción de información y 4) Minería de texto.

Las etapas no son precisamente secuenciales, el resultado de la minería impone, generalmente, cambios de parámetros (como distancias, criterios) o de métodos dictados por la interpretación de los resultados en la fase de post-minería, pero la fase de preparación de texto es muy cara en términos de tiempo de tratamiento[2]. Porque la fase de preparación de texto es esencial, lo mejor es guardar de manera permanente toda la información que puede servir.

La fase del preparación de texto tiene como objetivo convertir la base de recursos documentales de entrada a un conjunto de palabras (*tokens*) significativas reduciendo así, el número de datos a analizar. El contenido de un documento se transforma en valores, para poder aplicar técnicas de minería de texto o en índices para poder extraer o encontrar fácilmente los documentos relacionada con una búsqueda de información.

La secuencia de los pasos que sigue la aplicación desarrollada para la preparación del texto no es fija, los pasos que se presentan se pueden realizar en orden, sin embargo se pueden regresar algunos pasos anteriores, para realizar los ajustes necesarios. Estos pasos son:

1. Conversión a texto plano: Conversión de los recursos documentales de la memoria corporativa a un formato de texto plano (.txt).
2. Análisis léxico: Basado en herramientas de procesamiento de lenguaje natural como tokenizadores, los cuales separan las palabras. En esta fase se puede eliminan los caracteres especiales como: #, $, %, :-, entre otros.
3. Aplicación de la lematización (*stemming*): Permiten la reducción de las palabras a su forma básica o raíz, por ejemplo, eliminando las partes no esenciales de los términos como prefijos y sufijos o derivando las palabras en plural a su raíz en singular.
4. Eliminación de palabras vacías (*stopword removal*): Busca eliminar los términos con poco significado en la recuperación de información como: pronombres, partículas interrogativas y ciertas preposiciones.
5. Generación de matrices de frecuencia: Se genera un archivo con el tipo de matriz que se desee (frecuencias originales, matriz de términos, matriz inversa de términos, etc.), dependiendo de la métrica requerida.

En la figura 2 se pueden observar de forma gráfica los pasos descritos anteriormente para la preparación de texto de los documentos.

Dichos pasos conforman el proceso de preparación de texto que permite generar datos de calidad y conducir a patrones o reglas de calidad [9].

[2] El tiempo para la preparación de texto es largo en razón del tamaño de las colecciones, del uso de herramienta de análisis gramatical y de la construcción de los índices.

Para los pasos 1, 2,4 y 5 se realizan mediante una implementación en Java y para el paso 3 se hace necesaria la utilización de TreeTagger, la cual es una herramienta de análisis léxico que determina la categoría gramatical de las palabras, así como la forma invariante de dichas palabras [10]. Para obtener la matriz de frecuencia se realiza de igual forma con Java, sobre esta matriz se pueden obtener reglas de agrupamiento y clasificación, árboles de decisión, modelos de regresión o tendencias [11].

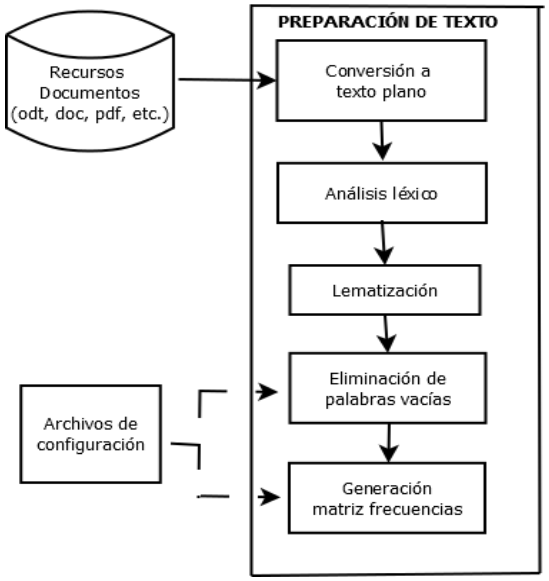


Fig. 2. Etapas de la preparación de texto.

3. Pruebas realizadas

Las pruebas se realizaron sobre dos colecciones de documentos, la primera colección poseía 34 documentos sobre el área de computación específicamente sobre ontologías, la segunda colección posee 150 documento y es más heterogénea abarcado diferentes dominios, ya que posee documentos de minería (texto, datos y XML), Web Semántica y sobre Cursos Masivos en Línea y Abiertos todos los documentos considerados se encuentran en idioma inglés. Se realizaron alrededor de 4 pruebas antes de saber qué *tokens* eran importantes. El tiempo promedio de tratamiento de un documento en formato pdf de 55Kbytes es alrededor de 1.2 minutos para la primera prueba, para la cuarta prueba ese mismo documento es tratado en 50 segundos; aunque la aplicación en la cuarta prueba se realiza una reducción de tiempo esto está relacionado con la reduccion del número de los tokens que se guardan. A continuación se describen dichas pruebas.

3.1.1. Primera prueba

Se consideró utilizar la primera colección, discriminando algunos tipos gramaticales como: preposiciones o conjunción subordinante (*until, before, after, etc.*), conjunción de coordinación (*and, but, nor, or, yet, etc.*), determinador (*a, an, every, not, the,* etc.), determinador wh (*which),* Pronombre wh (*who, what and whom*), adverbío wh (*how, where, why, etc*.) pronombres personales, pronombres posesivos, símbolos (expresiones que no pertenecen al idioma inglés), así como las palabras desconocidas (<*Unknown*>) y la preposición *to.*
En la tabla 1 se presentan las frecuencias más altas obtenidas en la primera prueba, en la primera columna se encuentran los tokens y en la segunda columna las frecuencias.

Tabla 1. Frecuencias más altas de la primera prueba

Tokens	**Frecuencia**
Be	19106
Ontology	4281
Use	3896
Annotation	3046
Have	2888

3.1.2. Segunda prueba

Para esta prueba se considero la segunda colección de 150 documentos, discriminando los mismos tipos gramaticales que en la prueba anterior; sin embargo nos pudimos dar cuenta que aún con la discriminación de esos tipos, obteníamos tipos gramaticales no deseados como: existencial *there*, modales (*can, might, may, should, must,* ought, shall, etc.), terminaciones posesivas, superlativos y comparativos. Por lo que se procedió únicamente a considerar 4 tipos gramaticales: adjetivos, pronombres, adverbios y verbos, ya que de acuerdo al diccionario Británico [12] y la Universidad de Oregon [13] de los ocho tipos gramaticales del idioma inglés, los básicos e importantes son los antes mencionados.

3.1.3. Tercera prueba

Durante las pruebas anteriores nos dimos cuenta que cuando TreeTagger procesaba los sustantivos, en muchos de los casos la forma base era desconocida, es decir, del 100% de los sustantivos el 40% eran desconocidos, por lo que resultaban frecuencias muy altas con el lema *Unknown*, ya que no eran reconocidos. Para solucionar este problema procedimos a indicarle en la aplicación en Java, que cuando fuese un sustantivo tomara la palabra, tal cual.

3.1.4. Cuarta prueba

En esta prueba procedimos a discriminar los correos electrónicos, las referencias y las citas en estilo Chicago y APA, ya que muchos de los documentos tenían esta variedad de citas. También se procedió a discriminar el tipo gramatical Adverbio, por considerar que no aporta ninguna información para obtener las palabras importantes, por lo que los tipos gramaticales que se están considerando son 3: sustantivos, verbos y adjetivos. Por otra parte se validó el caso particular cuando en las líneas se presenta una palabra segmentada por un guión, por lo que se procedió a eliminar el guión y concatenar las siguientes sílabas para formar así una sola palabra.

En la tabla 2 se presentan los resultados obtenidos de las frecuencias más altas, en la primera columna se encuntran los tokens y en la segunda la frecuencia

Tabla 2. Frecuencias más altas de la cuarta prueba

Tokens	**Frecuencia**
Data	2765
Ontology	1775
Web	1614
Information	1533
MOOC's	1344

4. Conclusiones y trabajos futuros

En este documento se describe una propuesta para la obtención de descripciones significativas de los recursos de información (documentos y personas) de una memoria corporativa. Se enfatiza en el pre-tratamiento de las colecciones de una memoria corporativa la cual contiene documentos e informaciones en varios dominios. Este particular, pone de manifiesto la importancia de extraer las características más relevantes del contenido de un documento, ya que sin la discriminación adecuada se corre el riesgo de obtener resultados erróneos para las etapas futuras (conformación de descripciones, aplicación de algoritmos de agrupamiento, obtención de conceptos, etc.).

Como trabajo futuro se considera realizar pruebas sobre una memoria educativa cuyos recursos documentales estan en diferentes idiomas: español, inglés y francés. Por otro lado, aplicar la metodología descrita en este documento para la obtención automática de conceptos representativos de un dominio y compararlos con vocabularios conceptuales (ontologías) existentes. Lo anterior con el fin de poner a punto la generación de índices semánticos en donde las descripciones significativas son una parte medular de los mismos.

Referencias

1. Gandon L., Fabien. "Ontology Engineering: a Survey and a Return on Experience". Technical Report RR-4396, INRIA, Marzo 2002.
2. Gilli, Juan J. "Diseño organizativo: estructura y procesos". Ed. Granica, 2007.
3. Erik Alarcón Zamora. "Integración Semántica de los Recursos de una Memoria Corporativa". Tesis de maestría en Tecnologías de la Información, UAM-Iztapalapa.
4. Métodos Para el Análisis de Documentos, determinación de su Contenido y Selección de los Términos de Indización NC- ISO 5963: 2000. 2014. URL: http://www.sld.cu/galerias/pdf/sitios/centromed/nc_iso_5963_metodos_para_el_analisis_de_documentos,_determinacion_de_sucontenido_y_seleccion_de_terminos_de_indizacion.pdf.
5. Niño Zambrano, M.A., Jimena, Pérez, D., Pezo, D.M. "Procesamiento para la Construcción de Índices Semánticos Basados en Ontologías de Dominio Específico". Entramado. Vol. 9, No.1, pp. 262-287. Enero 2013.
6. Galindo Durán, C. K., Medina-Ramírez, R. C., Juganaru-Mathieu, M. "Using Linked Open Data to Enrich a Corporate Memory of Universities". Proceedings of the International Conference on e-Learning, e-Business, Enterprise Information Systems, and e-Government EEE'14.Hamid R. Arabnia Azita Bahrami Fernando G. Tinetti. ISBN 1-60132-268-2 2014.
7. Metadata Standards. 2014. URL: http://www.metadataetc.org/bookwebsite/readings/appendixaschemas.htm
8. Montes y Gómez, Manuel. "Minería de texto: un nuevo reto computacional", 2014. URL: http://ccc.inaoep.mx/~mmontesg/publicaciones/2001/MineriaTexto-md01.pdf
9. Liu, H., Motoda, H., "Feature Extraction, Construction and Selection: A Data Mining Perspective". Kluwer Academic, 1998.
10. Sitio oficial TreeTagger, 2014. URL: http://www.cis.uni-muenchen.de/~schmid/tools/TreeTagger/
11. Srivastava, A. N. y Mehran, S. "Text Mining Classification, Clustering" and Applications. CRC Press Taylor & Francis Group, 2009.
12. Sitio oficial Cambridge Dictionaries, 2014. URL: http://dictionary.cambridge.org/dictionary/british/part-of-speech
13. A brief grammatical sketch of English, 2014. URL: http://pages.uoregon.edu/tpayne/engram/Engramsection03.pdf

Identificación probabilística de interacciones medicamentosas

Luis Enrique Colmenares Guillén[1], Luis Daniel Oidor Juárez[2], José Gustavo López y López[3]
[1, 2]Facultad de Ciencias de la Computación
[3]Facultad de Ciencias Químicas
Benemérita Universidad Autónoma de Puebla
Puebla, México
[1]lecolme@gmail.com
[2]daniel_oidor@hotmail.com
[3]jose.lopez@correo.buap.mx

Resumen. La fabricación de productos farmacéuticos representa económicamente, una de las industrias más importantes en todo el mundo. Actualmente, existen tratamientos para casi cualquier enfermedad conocida y con más de un fármaco para cada una de estas. Esto da como resultado que un paciente deba administrarse varios medicamentos durante la terapia farmacológica, con el riesgo consecuente de que existan interacciones negativas para el organismo entre los mismos, denominadas interacciones medicamentosas. El presente trabajo pretende diseñar un método basado en un algoritmo clasificador para identificar las interacciones medicamentosas, mediante el procesamiento de grandes cantidades de datos de forma automatizada. El algoritmo diseñado ayudará a la implementación de sistemas de apoyo para la actividad relacionada con la idoneidad de la prescripción médica. El método que se propone tiene posibilidades de incrementar su eficiencia de forma gradual, a través de conjuntos de entrenamiento, lo que lo convierte en una solución rápida, flexible y adaptable.

Palabras Clave: algoritmo; automatización; clasificador; cómputo; corpus; idoneidad; interacción; farmacología; medicamentos; prescripción.

1. Introducción

Desde tiempos ancestrales, el ser humano se encuentra en una búsqueda permanente de la curación de sus enfermedades, desde el inicio de la civilización, en que sus males eran atribuidos a seres malignos y hechos mágicos. En nuestros días, utilizamos las herramientas que la ciencia y la tecnología para la cura de enfermedades.

Hasta la primera mitad del siglo pasado, el hombre utilizó remedios que, en su gran mayoría, no alteraban de forma importante los mecanismos fisiológicos. Posteriormente la medicina cambió, introduciendo una inmensa gama de medicamentos, capaces de modificar de manera favorable el curso de las enfermedades y la aparición de síntomas y signos. El papel que juegan los

medicamentos en las sociedades actuales es tan relevante que hoy en día, la industria farmacéutica es una de las más dinámicas e importantes para la economía mundial.

Ante este panorama, cada medicamento desarrollado debe seguir un minucioso proceso de pruebas para el aseguramiento de la eficacia del mismo, no sólo en términos de su calidad farmacéutica, sino también en función de la gravedad de los efectos secundarios y de las reacciones que estos puedan provocar en el ser humano.

Sin embargo, en la prescripción médica la indicación de uso de un solo medicamento es inusual, por lo general un paciente debe administrarse dos o más. Es por ello que los desarrolladores de medicamentos deben realizar también pruebas de combinaciones de estos, de modo que sea posible evitar una interacción medicamentosa que afecte de manera negativa al paciente, ya sea por inhibición de los efectos de uno de ellos, la generación de efectos adversos o el aumento de toxicidad de alguna de las sustancias activas.

Para ello, se han realizado múltiples estudios por parte de laboratorios, centros de investigación y empresas privadas en las áreas de química y medicina.

Pero, como se ha dicho antes, esta industria es una de las más grandes de la economía mundial. ¿Se podría aseverar que el médico conoce (y recuerda) todas y cada una de las posibles interacciones entre medicamentos que pudieran afectar nuestra salud? Definitivamente no. A pesar de los grandes avances tecnológicos, la prescripción médica es una actividad que sigue siendo aplicada por el profesional de la salud de forma "manual", a través del uso de los conocimientos generados por la investigación científica en materia farmacéutica, sin asistencia de dispositivos o mecanismos automáticos, como ya sucede en otras áreas de la medicina como los análisis clínicos y el diagnóstico médico. Además actualmente se han desarrollado procesos que implican evaluación del perfil farmacoterapéutico con el fin de identificar problemas relacionados con los medicamentos (Idoneidad de la prescripción y Conciliación de la medicación), que mediante un proceso de intervención antes de la aplicación se evitan errores de medicación en el paciente hospitalizado.

Es necesario, entonces, diseñar métodos basados en el cómputo automático de grandes cantidades de datos, que sirvan como base para el desarrollo de herramientas que provean a los profesionales de la salud y a los desarrolladores de fármacos, apoyo en la identificación de las interacciones medicamentosas. Estos métodos deberán ser diseñados para explotar la información generada por los estudios e investigaciones realizados por los especialistas en la materia, siendo su principal aporte la automatización del procesamiento de información y la aplicación de la ciencia computacional en el desarrollo de soluciones que potencien la actividad humana.

Identificar correctamente las interacciones medicamentosas es un proceso complejo, dado que el resultado de calificación de la interacción dado un par de medicamentos específico generalmente es un conjunto de posibles efectos adversos provocados por ese par de medicamentos. Cada uno de estos efectos tendrá una probabilidad de suceder en un paciente específico.

Dada esta complejidad, el algoritmo que se diseñará será un clasificador, con la finalidad de que nos proporcione un conjunto de resultados que puedan ser ponderados conforme a su probabilidad.

El objetivo principal de este trabajo es proponer una aproximación a un algoritmo clasificador que permita la identificación y ponderación probabilística de

interacciones medicamentosas. El diseño del algoritmo estará orientado a que la identificación se lleve a cabo de forma previa a la prescripción médica, o bien como una herramienta para el proceso de la Idoneidad de la prescripción, a fin de detectar problemas relacionados con los medicamentos y evitar resultados negativos de la medicación.

Para alcanzar este objetivo será necesario llevar a cabo algunas tareas que conforman los objetivos específicos del presente trabajo, las cuales se enumeran a continuación.

- Realizar una investigación de los métodos actuales para la identificación de interacciones medicamentosas.
- Identificar las fuentes confiables de información sobre las características, efectos y contraindicaciones de medicamentos.
- Analizar las ventajas y características generales de los diferentes tipos de clasificadores.
- Determinar el tipo de clasificador en que se basará el diseño del algoritmo que será producto del proyecto.

Para alcanzar este objetivo será necesario llevar a cabo algunas tareas que conforman los objetivos específicos del presente trabajo, las cuales se enumeran a continuación.

2. Fuentes de información sobre medicamentos

La información es un elemento poderoso y fundamental en el desarrollo de cualquier actividad humana de cualquier índole. En el caso de la medicina, y en específico de la prescripción médica, contar con la información correcta y oportuna es un requisito indispensable.

Para que la ciencia computacional pueda ofrecer herramientas que coadyuven con el objetivo de la medicina es necesario que ésta última le proporcione fuentes de información que sirvan como base de dichas herramientas.

En la actualidad, el principal problema es seleccionar la información más relevante y de mayor calidad [1]. Toda la información relacionada con los medicamentos está incluida en un campo más amplio, conocido como información biomédica.

2.1. Pirámide de Hynes

Las fuentes de información biomédicas (incluyendo lo referente a medicamentos) pueden clasificarse mediante el modelo piramidal propuesto por **R. Brian Haynes**, mejor conocido como *Pirámide de las 5s*, que puede observarse en la Fig. 1.

El modelo de Haynes es el más aceptado para la clasificación de fuentes de información relacionadas con la biomedicina. En él se pueden diferenciar cinco niveles que a continuación se describen brevemente.

- Estudios. Representan las fuentes primarias, incluyen los artículos biomédicos originales.
- Síntesis. Incluyen recursos que indexan y publican revisiones sistemáticas y metaanálisis.
- Sinopsis. Comprenden resúmenes estructurados de artículos originales, así como boletines e informes de evaluación de medicamentos elaborados por comunidades autónomas, hospitales y agencias reguladoras.
- Compendios. Incluyen revisiones sistemáticas y resúmenes sobre patologías o tratamientos determinados, constituyen las fuentes que eran clasificadas en la forma clásica como secundarias.
- Sistemas. Aquí se incluyen aplicaciones de cómputo auxiliares en la toma de decisiones, como bases de datos y sistemas de evaluación automatizados.

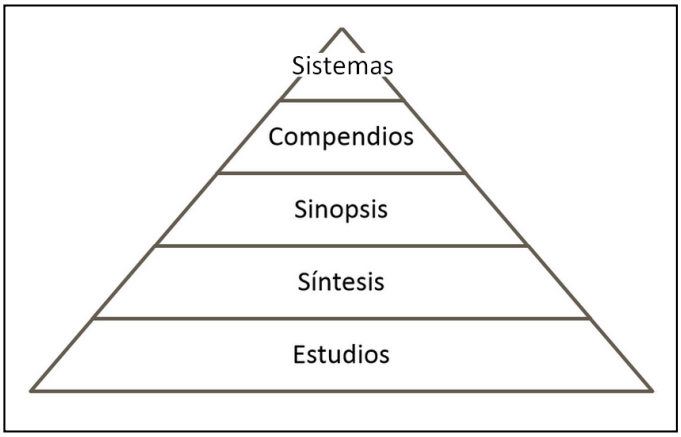


Fig. 1. Modelo piramidal de las 5s de Haynes.

2.1 Principales fuentes de información sobre medicamentos

Como parte del proceso de investigación comprendido en el presente trabajo, se identificaron tres principales fuentes de información referentes a medicamentos. Estas fuentes fueron elegidas por ser las más aceptadas en el campo de la investigación biomédica.

Diccionario de Especialidades Farmacéuticas de Thomson. Mejor conocido como PLM, es una de las fuentes más utilizadas por los médicos y farmacias en México. Incluye una lista completa de todos los fármacos aprobados por la Secretaría de Salud en México, lo que conforma un total de más de 3,500 productos

farmacéuticos. Se encuentra disponible en su versión impresa y, gracias a los avances tecnológicos actuales, en versiones digitales en Internet y en forma de aplicaciones para dispositivos móviles. PLM México tiene a disposición de cualquier persona con acceso a Internet una versión web disponible en la dirección electrónica http://www.medicamentosplm.com/ [2].

Micromedex. Es una base de datos norteamericana que contiene amplia información de medicamentos y sustancias relacionadas, así como de pruebas de laboratorio e interacciones medicamentosas. Su contenido es actualizado constantemente mediante la revisión sistemática de estudios médicos, siendo así una fuente confiable de información. Consta de diversos productos, entre los cuales se encuentra DRUGDEX, un sistema de información específico sobre medicamentos, su administración, efectos adversos e interacciones [3].

Vademécum. Es un catálogo de especialidades, entre las cuales se encuentra la información relacionada a los medicamentos. Puede encontrarse en versión impresa o en versión digital, que es distribuida por medio de un CD-ROM que contiene todos los archivos necesarios para ser instalado en un equipo de cómputo. Contiene información relevante relacionada con productos farmacéuticos, y está destinado a los profesionales de la salud, tales como médicos y farmacéuticos [4].

Existen otras muchas fuentes de información, sin embargo se mencionan solamente las tres anteriores dada su alta aceptación y amplio uso por parte de los profesionales de la salud. La identificación de estas fuentes como fiables y generalmente aceptadas es un paso fundamental en la elaboración de un método para la identificación de interacciones medicamentosas. Un método basado en información de calidad estará en condiciones de proporcionarnos resultados de calidad, como los que son requeridos en la práctica de la prescripción médica.

3. Estado del arte

Durante las últimas décadas, la ciencia computacional se ha desarrollado rápida y continuamente, lo que ha producido computadoras de propósito general con mayor procesamiento de cómputo y movilidad. Así, la ciencia computacional se ha convertido en una herramienta de apoyo en la resolución de problemas.

Como parte de esta incorporación de las nuevas tecnologías a ámbitos de la actividad humana, la medicina y la farmacia han encontrado en la computación una herramienta poderosa para la agilización y mejoramiento de los procesos de apoyo biomédico.

Con relación a las interacciones medicamentosas, con ayuda de la ciencia computacional se han obtenido grandes avances, entre los que destacan algunos métodos diseñados con la finalidad de ayudar a la identificación de las causas de un efecto adverso, entre las cuales se pueden encontrar interacciones de medicamentos. A continuación, se mencionan algunos algoritmos más importantes.

Algoritmo de Karch y Lasagna. Es un algoritmo que, aunque ya han pasado más de 35 años de su publicación, sigue siendo un estándar para la identificación de los efectos adversos de medicamentos, entre los cuales existen combinaciones de dos o más fármacos. Contempla la secuencia temporal entre el cuadro clínico que presenta un paciente y los fármacos presuntos responsables de dicha sintomatología, mediante la evaluación de la relación causa-efecto. Esta relación puede clasificarse como Definida, Probable, Posible o Condicional [5].

Algoritmo de Kramer. Consiste en una secuencia de preguntas y una escala de calificación que permite, al final de la aplicación del cuestionario, establecer la causalidad por categorías. Consta de 56 preguntas dicotómicas (sí/no). Es también un algoritmo diseñado para determinar si una reacción fue generada por un medicamento o un conjunto de medicamentos en específico [6].

Algoritmo de Naranjo y colaboradores. Es un algoritmo basado en el de Karch y Lasagna y que consta de un cuestionario, como el algoritmo de Kramer, pero de menor cantidad de cuestiones (solamente 10 preguntas dicotómicas). Al igual que los algoritmos anteriores, no fue diseñado específicamente para determinar la interacción entre fármacos, sino de relacionar un efecto adverso con su causal. Resulta eficaz dada su simplicidad y su corta extensión [7].

La particularidad que comparten los tres métodos mencionados es que su objetivo es identificar de manera general la causa de un efecto adverso en un paciente, lo que nos hace pensar en ellos como métodos *a posteriori* con relación a la prescripción médica. Esto implica que una interacción medicamentosa deberá presentarse al menos una vez para poder ser evaluada e identificada como potencialmente negativa.

Un esfuerzo importante lo representa el **Corpus Drug-Drug Interactions**. Fue desarrollado a partir de una colección de textos médicos en los cuales figuran las diversas interacciones medicamentosas, incluso las que no han sido probadas por la ciencia médica, con su catalogación de certeza. Entre las fuentes de información biomédica utilizadas en el desarrollo de dicho corpus, se encuentra la base de datos **Micromedex**. Este trabajo genera las bases para realizar métodos de explotación de dicha información mediante la aplicación de técnicas de tratamiento de texto.

Un método de identificación de interacciones mediante la utilización de un corpus permitirá establecer la probabilidad de que esta se presente en un paciente, antes de que los medicamentos implicados sean prescritos, evitando así incomodidades y reacciones no deseadas en seres humanos.

4. Técnicas de clasificación

Para que la información contenida en un corpus sea procesada y clasificada es necesario diseñar un método basado en técnicas de clasificación de grandes cantidades de texto.

Se puede formalizar la *clasificación* como una aproximación de una función objetivo no conocida que describe la forma en la que instancias del problema deben

ser clasificadas, mediante otra función, denominada *clasificador*. La función objetivo se puede representar como en (1), mientras que el clasificador se encuentra representado en la forma mostrada en (2).

$$\Phi: I \times C \rightarrow \{T, F\} \tag{1}$$

$$\Theta: I \times C \rightarrow \{T, F\} \tag{2}$$

C es un conjunto predefinido de categorías, en tanto que *I* es un conjunto de instancias del problema. Es común representar cada instancia $i_j \in I$ como una lista $A = \{a_1, a_2, \ldots, a_{|A|}\}$ de valores característicos, denominados atributos. Si $\phi\ i_j \times c_i \rightarrow T$, entonces i_j es un ejemplo positivo de la categoría c_i. Si, por el contrario, $\phi\ i_j \times c_i \rightarrow F$, entonces i_j es un ejemplo negativo de c_i.

Es posible generar de forma automática el clasificador mediante el proceso llamado *aprendizaje supervisado*. Este proceso implica la observación de los atributos de un conjunto de instancias ya clasificadas, de modo que sea posible asignar una instancia no clasificada en una determinada categoría. Un requisito para la construcción del clasificador es contar con una colección Ω, denominada *conjunto de entrenamiento*, de ejemplos tales que el valor de la función $\phi(i_j, c_i)$ sea conocido para cada $(i_j, c_i) \in \Omega \times C$.

Existen diversas técnicas de clasificación, con diferentes fortalezas y características. A continuación, se revisan cuatro técnicas cuyos resultados en la clasificación de texto destacan sobre otras.

4.1. Vecinos más cercanos

Es un clasificador supervisado, también conocido *como k-NN*. Para su aplicación se exploran los atributos de los elementos del conjunto de entrenamiento para determinar la categoría a la que pertenecerá una nueva instancia no clasificada. Esta técnica utiliza la información suministrada por las k instancias del conjunto de entrenamiento más cercanas a la instancia que se desea clasificar.

En la Fig. 2 están representadas doce elementos pertenecientes a dos categorías diferentes, la *Categoría A*, conformada por seis cuadros de color verde, y la *Categoría B*, que la forman seis círculos de color naranja. La instancia x es la que deseamos clasificar. Dentro del círculo mayor podemos observar los tres vecinos más cercanos, es decir, que k = 3.

Dado que, de los tres vecinos más cercanos, uno pertenece a la *Categoría A* y dos a la *Categoría B*, después de aplicar la técnica, la instancia x será asignada a la *Categoría B*.

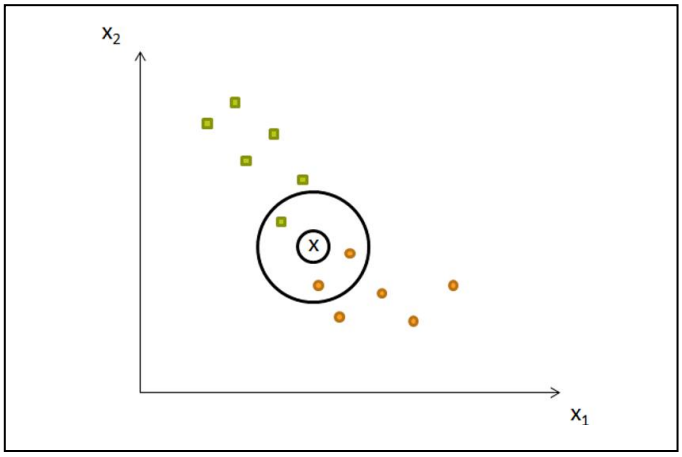


Fig. 2. Técnica de clasificación de los vecinos más cercanos.

4.2. Naïve Bayes

Es un clasificador probabilístico basado en el **Teorema de Bayes**, enunciado por **Thomas Bayes**. Básicamente, este teorema relaciona la probabilidad de un evento *A* dado *B* con la probabilidad del evento *B* dado *A*.

La fórmula del **Teorema de Bayes** es:

$$P(A_i|B) = \frac{P(A_i)P(B|A_i)}{\sum_{i=1}^{n} P(A_i)P(B|A_i)}$$

Es importante recordar la siguiente definición de la probabilidad condicional:

$$P(BA_i) = P(B|A_i)P(A_i) \quad (3)$$

El método de **Naïve Bayes** usa frecuencias para calcular probabilidades condicionales con el fin de realizar predicciones sobre nuevas instancias del problema. Un clasificador de este tipo puede ser tanto descriptivo como predictivo. A continuación se formalizará la definición del clasificador Naïve Bayes.

Dados los eventos *E* y *F*, se sabe que:

$$E = EF \cup EF^c$$

EF y EF^C son mutuamente excluyentes, por lo tanto:

$$P(E) = P(EF) \cup P(E\,F^C)$$

$$P(E) = P(E|F)P(F) + P(E|F^C)P(F^C)$$

$$P(E) = P(E|F)P(F) + P(E|F^C)\big(1 - P(F)\big) \tag{4}$$

Lo que la ecuación (4) pone de manifiesto es que la probabilidad de E es una ponderación de su probabilidad condicional dada la ocurrencia y la no ocurrencia de F. Ahora, se supone que los eventos $F_1, F_2, \ldots, F_n$ son mutuamente excluyentes, lo que se puede expresar como:

$$E = \bigcup_{i=1}^{n} E_i$$

A partir de la definición de probabilidad condicional, expresada en (3) y dado que los eventos EF_i, para todo i de 1 a n, son mutuamente excluyentes, se infiere que:

$$P(E) = \sum_{i=1}^{n} P(E\,F_i)$$

$$P(E) = \sum_{i=1}^{n} P(E|F_i)P(F_i) \tag{5}$$

De esta forma, la ecuación (5) muestra que, para los eventos $F_1, F_2, \ldots, F_n$, de los cuales puede darse la ocurrencia de uno y solamente uno, se establece que la probabilidad de E es igual al promedio de las ponderaciones de $P(E|F_i)$.

La suposición de que los eventos son mutuamente excluyentes es la que da origen al nombre de **Naïve** (ingenuo), ya que esto no siempre sucede. A pesar de ello, el método ha sido implementado con buenos resultados, por lo que es uno de los clasificadores más aceptados.

4.3. Support vector machines

Las **Máquinas de Soporte Vectorial** o **Support Vector Machines** (SVM) son un conjunto de algoritmos cuya técnica se basa en el aprendizaje de dos categorías distintas de entrada. Dado que el objetivo es clasificar instancias en una sola categoría, con la descripción proporcionada por los datos elabora una frontera de decisión alrededor de los datos de aprendizaje, para luego buscar la separación máxima entre categorías. De esta forma se divide al espacio muestral en categorías distintas.

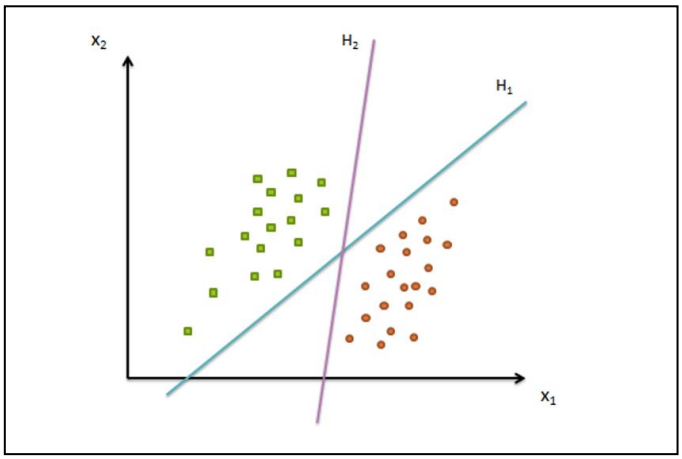


Fig. 3. Funcionamiento del algoritmo SVM.

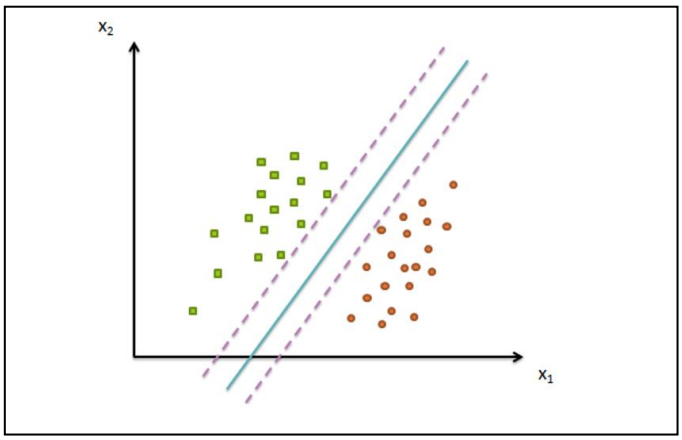


Fig. 4. Hiperplano y vectores de soporte para SVM.

En la Fig. 3, se muestra un ejemplo del funcionamiento de SVM. En él, se representan los datos en el plano xy. El algoritmo intentará encontrar un hiperplano de dimensión 1 que constituye el límite que separa las dos categorías existentes. La posición de una nueva instancia a un lado u otro de este hiperplano determinará la pertenencia de dicha instancia a la categoría específica correspondiente.

Como se puede observar, existe un número infinito de hiperplanos posibles que dividan las instancias en dos categorías, por lo que debe encontrarse el mejor de ellos. Para esto, el algoritmo elige el hiperplano cuyo margen de separación entre las instancias de ambas categorías sea el máximo, es decir, existirán dos líneas paralelas al hiperplano que indicarán la distancia con las instancias de cada categoría. A estas líneas se les llama *vectores de soporte*. En la Fig. 4 los vectores de soporte aparecen como líneas punteadas

5. Búsquedas mediante corpus

La definición más simple de **corpus** nos refiere a éste como una colección, generalmente amplia, de textos. Sin embargo, cuando el término es usado en el ámbito de la lingüística computacional, éste tiene más implicaciones.

La primera implicación se refiere al lugar donde este conjunto de textos está almacenado. Leech [8] introduce el concepto de corpus como un *emocionante fenómeno, una magnífica gran cantidad de texto, almacenada en una computadora*.

Por su parte, Francis [9] agrega a su definición de corpus el que esta colección *se asume como representativa de un determinado idioma, dialecto o subconjunto de un idioma para ser usado en análisis lingüístico*.

Sin embargo, tal vez la mejor definición la proporciona el grupo de trabajo que está dedicado a los corpus de texto. Denominado como EAGLES (Expert Advisory Group on Language Engineering Standards) [10], este grupo define un corpus como *una colección de piezas de un idioma seleccionadas y ordenadas de acuerdo a criterios lingüísticos explícitos con el fin de ser usados como ejemplo de un idioma*.

En el campo de la ciencia biomédica, se han desarrollado diversos corpus con el objeto de analizar lingüísticamente la información contenida relacionada con los aspectos biomédicos. Uno de los objetivos del presente trabajo es desarrollar un corpus que contenga información específicamente relacionada con las interacciones entre medicamentos.

La finalidad es generar un corpus de dominio biomédico orientado a la identificación de interacciones medicamentosas, que sea base para el desarrollo de un algoritmo clasificador que permita identificar interacciones entre medicamentos.

6. Aproximación a un algoritmo clasificador de interacciones medicamentosas

Hasta ahora, se ha proporcionado la base teórica sobre la cual se desarrollará una aproximación a un algoritmo clasificador de interacciones medicamentosas, basado en búsquedas a través de corpus.

Se sabe que los efectos de una interacción medicamentosa negativa pueden presentarse para ciertos pacientes, en tanto que en otros, éstos podrían ser imperceptibles o, incluso, nulos. El conocimiento o ignorancia sobre todas las causas que se relacionan con un efecto adverso hacen que la clasificación de éstos sea una función probabilística que pondere su ocurrencia para un paciente cualquiera.

Evidentemente, se intenta resolver un problema para el que se cuenta con datos parciales, solamente considerando la presencia de medicamentos, sin contemplar el resto de variables relacionadas con el paciente, como su peso, masa muscular, enfermedades crónicas o antecedentes familiares, entre muchos otros.

Por otro lado, es importante destacar que uno de los principales aportes de este trabajo es proporcionar las bases para el desarrollo de sistemas y herramientas de apoyo biomédico, en específico, para la prescripción médica. Actualmente, la identificación de una interacción medicamentosa por parte de un profesional de la salud, en el supuesto de que se desconozca o exista incertidumbre sobre la misma, comprende la búsqueda de la información de cada medicamento involucrado, proceso que se realiza sin ayuda de sistemas expertos.

Se busca, mediante el desarrollo de este método de clasificación, proporcionar una respuesta rápida de la probabilidad de interacciones entre medicamentos, lo que significará una ventaja sobre el proceso realizado de forma manual.

Dado lo anterior, se eligió el clasificador de **Naïve Bayes** como el tipo de algoritmo que se desarrollará, debido a la fortaleza de la técnica para realizar predicciones a partir de datos parciales y por su rapidez.

El desarrollo del presente trabajo generará beneficios a mediano y largo plazo, posibilitando el desarrollo de sistemas comerciales que puedan ser implementados en hospitales, farmacias y centros de salud, en los cuales se implemente el algoritmo desarrollado.

El algoritmo clasificador diseñado podrá operar bajo un esquema funcional como el que se muestra en la Fig. 5. Los datos de entrada, lo conforman un par de medicamentos, cuya posible interacción será evaluada por el algoritmo clasificador. Estos datos son ingresados mediante una terminal y llegarán a un servidor de aplicaciones, donde residirá la implementación del algoritmo. Desde ahí, se realiza la búsqueda y extracción de los datos relacionados en el corpus que contiene la información de cada medicamento para ser clasificados por el algoritmo. El resultado se envía a la terminal, que muestra si existe una posible interacción entre los medicamentos proporcionados.

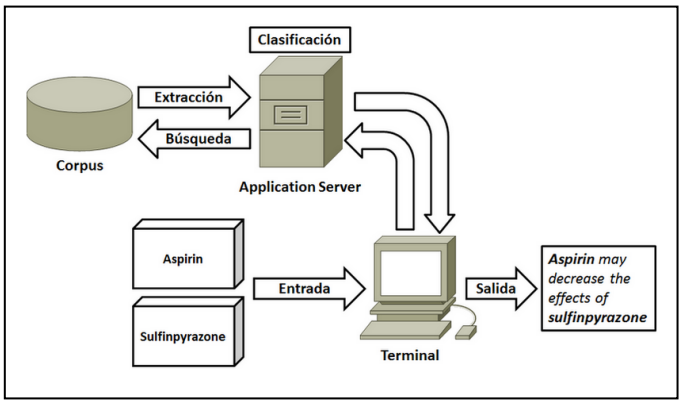


Fig. 5. Esquema funcional de una implementación del algoritmo clasificador.

Una característica importante de un clasificador es que su salida depende únicamente de la entrada de datos, en este caso, del corpus suministrado conteniendo la información sobre interacciones medicamentosas. Es decir, se trata de un método

incremental, que mediante la aplicación constante del algoritmo, a través de sistemas que lo implementen, generará más datos que pueden ser fácilmente incorporados al método original como entradas, con lo cual se expande el dominio de aplicación del algoritmo a mayores sectores poblacionales diferenciados por condiciones patológicas, de raza o con determinadas condiciones de salubridad.

7. Pruebas

Para comprobar la efectividad del método propuesto y, eventualmente, realizar adecuaciones y ajustes para obtener mejores resultados, se deberán realizar una serie de pruebas que nos ayuden al aseguramiento de la calidad del algoritmo clasificador.

Para tal efecto, se realizarán pruebas con cincuenta medicamentos de los más utilizados en México. El Laboratorio de Farmacia Clínica de la Benemérita Universidad Autónoma de Puebla ha facilitado una lista con medicamentos propuestos para ser considerados en las pruebas una vez terminado el desarrollo del algoritmo propuesto.

Una de las principales características de estas pruebas es que deberán realizarse en un ambiente real controlado, es decir, se buscará una unidad de servicios médicos para incorporar el uso del algoritmo en las prescripciones médicas que se realicen en dicha unidad. En esta etapa, los resultados que arroje el algoritmo conformarán un conjunto de casos que deberán ser evaluados por los profesionales de la salud encargados del control, administración y almacenamiento de medicamentos, para corroborar la efectividad del algoritmo.

El término controlado hace referencia al hecho de que la implementación que se desarrollará para efecto de las pruebas estará disponible solamente para llevar a cabo la evaluación de resultados, sin incorporarse por completo a la operación habitual de la unidad de servicios médicos donde se efectuarán las pruebas.

8. Conclusiones

La identificación de interacciones medicamentosas es uno de los problemas, dentro de la actividad de la prescripción médica, que requieren la aplicación de la ciencia computacional para desarrollar herramientas automatizadas que proporcionen apoyo a los profesionales de la salud.

Mediante el análisis del problema y el diseño de una propuesta de solución, se concluye que el presente trabajo es factible, dadas las condiciones actuales de las actividades relacionadas con la prescripción de medicamentos.

Como ya se explicó, la solución propuesta es un algoritmo clasificador basado en el de **Naïve Bayes** cuyos resultados serán ponderaciones de probabilidad de la posible interacción entre dos medicamentos.

El desarrollo del algoritmo permitirá su implementación en herramientas y sistemas para ser integrados a soluciones que permitan a los profesionales de la salud contar con información suficiente de apoyo en la toma de decisiones. Sin embargo, el principal impacto del presente trabajo es desarrollar un método para identificar interacciones medicamentosas previas a la prescripción médica, con la finalidad de evitar en el paciente efectos no deseados y afectaciones a su salud.

Referencias

1. Rancaño, I.; Rodrigo, J. A.; Villa, R.; Abdelsater, M.; Díaz, R.; Álvarez, D.: Evaluación de las páginas web en lengua española útiles para el médico de atención primaria, Aten Primaria, vol. 31, no. 6, pp. 575-584 (2003)
2. Diccionario de Especialidades Farmacéuticas 2012. http://www.medicamentosplm.com. Accedido el 25 de junio de 2013
3. Micromedex. http://www.micromedex.com/. Accedido el 26 de junio de 2013
4. PR Vademécum México.http://mx.prvademecum.com/. Accedido el 26 de junio de 2013
5. Armijo, J.; González, M.: Estudios de seguridad de medicamentos: Métodos para detectar las reacciones adversas y la valoración de la relación causa-efecto, El ensayo clínico en España, pp. 161-190 (2001)
6. Kramer, M. S.; Levental, J. M.; Hutchinson, T. A.; Feinstein, A. R.: An algorithm for the operational assessment of adverse drug reactions, I: background, description, and instructions for use, vol. 242, pp. 623-632 (1979)
7. Naranjo, C. A.; Busto, U.; Sellers, E. M.: A method for estimating the probability of adverse drug reactions, Clin Pharmacol Ther, vol. 30, pp. 239-245 (1981)
8. Leech, G.: Corpora theories of linguistic performance. J. Svartvik (Ed.): Directions in Corpus Linguistics, pp. 105-122 (1992)
9. Francis, W. N.: Problems Assembling and Computerizing Large Corpora. Johansson, S (Ed.), pp. 124-136 (1982)
10. Expert Advisory Group on Language Engineering, Text Corpora Working Group Reading Guide. EAG-TCWG-FR-2. (1996)
11. Freer, E. B.; Chavarria, J. C.: El desarrollo de la computación y su influencia en la medicina, Revista Costarrisence de Ciencias Médicas, vol. 13, pp. 59-70 (1992)
12. Martín, H.; Martín, S.: Cómo localizar la mejor evidencia científica, Recursos de información para la MBE (2007)
13. Calderón, C. A.; Urbina, A. P.: La Farmacovigilancia en los últimos 10 años: actualización de conceptos y clasificaciones. Logros y retos para el futuro en Colombia, Revista de los estudiantes de medicina de la Universidad Industrial de Santander, vol. 24, pp. 57-73 (2011)
14. Mitchell, T.: Machine learning. Ed. Mc Graw-Hill (1997)
15. Mladenić, D.; Grobelnik, M.: Feature selection for unbalanced class distribution and Naïve Bayes. Department of Intelligent Systems, J. Stefan Insitute (1999)
16. Pacheco, S. D.; Díaz, L. G.: El clasificador Naïve Bayes en la extracción de conocimiento de bases de datos, Ingenierías, Abril-Junio, vol. 27, no. 8 (2005)
17. Cortés, C.; Vapnik, V.: Support-Vector Networks, Machine Learning, vol. 20, pp. 273-297 (1995)
18. Aas, K.; Eikvil, L.: Text categorization: A survey, Norwegian Computing Center, *Technical Report* (1999)
19. Segura, I.: Application of Information Extraction techniques to pharmacological domain: Extracting drug-drug interactions, Tesis doctoral, Universidad Carlos III, España (2010)
20. Téllez, A.: Extracción de información con algoritmos de clasificación, Tesis, Instituto Nacional de Astrofísica Óptica y Electrónica, México (2005)

Sistema hipermedia basado en competencias para el diagnóstico del aprendizaje de fracciones matemáticas (SMCDAFRAC)

E. Erica Vera Cervantes [1,2], Carmen Cerón Garnica[1], Yadira Navarro[1] y María Magdalena Ortiz Funez[1]

[1]Benemérita Universidad Autónoma de Puebla
Facultad de Ciencias de la Computación
Avenida San Claudio, 14 Sur, Ciudad Universitaria
[2]Centro de Estudios Superiores en Educación (CESE)
Puebla, México
{eevclibra,academicaceron,ynavarro44}@gmail.com

Resumen. A partir de la Reforma Integral de Educación Básica, se estableció el Plan de estudios 2011, en el cual se definen las competencias para la vida, el perfil de egreso, los estándares curriculares y los aprendizajes esperados en los alumnos de este nivel. El propósito de este artículo es presentar el diseño y desarrollo del Sistema Hipermedia Adaptativo (SHA) el cual considera las características del estudiante (perfil), para adaptar y presentar los contenidos basados en competencias para apoyar a desarrollar las habilidades en las matemáticas, en el tema de fracciones de acuerdo a los programas de estudio vigentes. Finalmente se presentan los resultados obtenidos del sistema al realizar una prueba piloto con una muestra de alumnos de tercer grado de primaria.

Palabras clave: Competencias, sistema hipermedia adaptativo, multimedia y fracciones.

1. Introducción

Las instituciones educativas han incorporado el uso de las Tecnologías de Información y Comunicación (TIC) con una tendencia en el enfoque constructivista donde lo importante es aprender a aprender, para usar el objeto de conocimiento y saber dónde encontrar la información adecuada para la solución de problemas determinados [1]. Las TIC han proporcionado mayor velocidad, alcance y acceso a distintos recursos de información. Actualmente se requiere sistemas de información tecnológicos que permitan una personalización y adaptación de la información a las necesidades específicas de aprendizaje de cada alumno de tal forma que en el campo de la educación el sistema permita ayudar a planificar el proceso educativo de acuerdo a los contenidos para tomar decisiones para mejorar el rendimiento académico de los alumnos. La mayoría de los diseños de los sistemas interactivos permite a los alumnos mayor interactividad, concentrarse en el razonamiento y en la solución de problemas [2]. Es

decir, que los entornos de aprendizaje posibiliten actividades reales y contextualizadas para los estudiantes.

Actualmente los profesores han identificado una problemática en la enseñanza de las matemáticas al ocupar los métodos tradicionalistas que fomentan un aprendizaje pasivo, memorístico y no reflexivo. A partir de lo anterior existe el interés de incorporar las Tecnologías de Información y Comunicación (TIC) que contribuyan a la generación de ambientes de aprendizaje para propiciar mediante el uso de diversos recursos tecno-pedagógicos el desarrollo de habilidades en alumno.

Este proyecto tiene como propósito el diseño y desarrollo de un Sistema Hipermedia Adaptativo (SHA) denominado SMCDAFRAC (Sistema Multimedia basado en Competencias para el diagnóstico del aprendizaje de Fracciones), el cual surge de la necesidad de involucrar el uso de las TIC y apoyar el proceso de enseñanza-aprendizaje, donde los nuevos planes de estudio de acuerdo a la Reforma Integral de Educación Básica [3], los cuales promueven un aprendizaje por competencias y estrategias de aprendizaje de acuerdo a las necesidades de los alumnos para activar el pensamiento e integrar esos saberes a su desempeño cotidiano. El sistema es un medio para apoyar a los profesores y los alumnos en el campo de las matemáticas, el cual contempla la hipermedia, la adaptabilidad y el contenido educativo mostrando los distintos niveles de saberes/aprendizajes que el alumno debe adquirir de acuerdo a los planes y programas de estudio.

El objetivo del sistema es apoyar a propiciar el pensamiento matemático para el desarrollo de la imaginación, creatividad y el razonamiento lógico. El SHA permite identificar las necesidades de aprendizaje y proponer actividades de aprendizaje que conlleven al desarrollo de competencias en el alumno para la solución de problemas en el tema de fracciones. Esto le ayudará aprender a aprender y reflexionar sobre sus competencias adquiridas para mejorar su desarrollo intelectual y autonomía.

Por lo cual este documento se organiza de la siguiente manera: la fundamentación teórica de los sistemas hipermedias adaptativos, la estructura de los sistemas web y las competencias en las matemáticas se revisan en la sección 2; el análisis y diseño del sistema en UML, la arquitectura cliente-servidor, el diseño de la base de datos, el desarrollo del sistema con tecnología Active Server Pages (ASP) y HTML se presentan en la sección 3; el desarrollo y las pruebas del sistema se muestran en la sección 4 y finalmente en la sección 5 se presentan las conclusiones y el trabajo a futuro de esta investigación.

2. Marco teórico

2.1. Sistema hipermedia adaptativo (SHA)

La educación ha sido una de las áreas de aplicación más populares en el área de los Sistemas Hipermedia Adaptativos. Las investigaciones que se han realizado en este campo, se han dirigido sobre todo a técnicas y métodos de adaptación para apoyar las necesidades de aprendizaje del usuario, sus intereses, conocimientos previos y brindar

un gran volumen de información. Según Brusilovsky un Sistema Hipermedia Adaptativo es un sistema basado en hipertexto e hipermedia que refleja algunas características del usuario en el modelo de usuario y aplica este modelo para adaptar varios aspectos visibles del sistema al usuario [4]. Para Gaudioso define los SHA son aquellos sistemas de hipermedia capaces de ajustar su presentación y navegación a las diferencias de los usuarios, reduciendo así los problemas de desorientación y falta de comprensión, propios de los sistemas hipermedia no adaptativos [5]. Un sistema se considera adaptativo cuando se adapta de forma automática y personalizada a las necesidades del usuario [4]. Por lo que el SHA permite personalizar la información almacenada y la presenta a los usuarios según sus preferencias, conocimientos e intereses. El proceso de personalización permite mostrar la información que es apropiada para tipo de conocimiento y aprendizaje de cada usuario [6]. El modelo de adaptación posee un conjunto de reglas que permiten adaptar los contenidos al perfil del usuario. El uso de reglas para establecer la adaptación está inspirada por varias investigaciones sobre los SHA, entre ellas la propuesta por Raad y Causse llamada "Modelización de la adaptación del Sistema Hipermedia basado en reglas Activas" que propone una separación entre la parte del comportamiento y las entidades del modelo que permiten añadir nuevas técnicas [7]. En esta investigación se utilizan reglas de la forma: "***Con la ocurrencia de un evento E, si la condición C se cumple, entonces el sistema ejecuta una acción A***." En general, las reglas permiten al sistema seleccionar adaptativamente, considerando las características del usuario y el tipo de contenido que debe aprender para cumplir con los objetivos, donde se logran manifestar relaciones interesantes a partir de la información existente.

2.2. Multimedia

En el ámbito de la computación el término multimedia designa el uso de varios recursos o medios, como audio, video, animaciones, texto e imágenes en una computadora, sin quedarse, sólo, en un collage de medios, al integrar los datos que puede manejar la computadora. La multimedia ofrece posibilidades de creatividad mediante los sistemas de computación. Según Bartolomé, el objetivo de la multimedia aplicada a la educación es alcanzar a desarrollar destrezas y actitudes necesarias en los estudiantes de modo que ellos se puedan comunicar con distinto lenguajes y medios, que además puedan desarrollar autonomía personal y espíritu crítico para formar una sociedad justa y multicultural [8]. La necesidad de incorporar los materiales multimedia en la educación se hace cada vez más latente ya que nos encontramos inmersos en una sociedad del conocimiento y la información; los alumnos demandan, cambios en los procesos de enseñanza para que el aprendizaje sea significativo y resulte motivador para ellos asistir a clases dinámicas, entretenidas y contextualizadas.

2.3. Competencias matemáticas y resolución de problemas

Se entienden así las matemáticas como un conjunto de cuestiones y problemas, de ideas y formas de actuar y de tecnologías simbólicas y organizativas que conllevan no

sólo utilizar cantidades y formas geométricas, sino también hacerse preguntas y resolver problemas, obtener modelos e identificar relaciones y estructuras, de modo que, al analizar los fenómenos y situaciones que se presentan en la realidad, se puedan obtener informaciones y conclusiones que inicialmente no estaban explícitas [8]. Las competencias en matemáticas enfatizan en que el alumno debe ser competente para argumentar, cuantificar, analizar críticamente la información, representar y comunicar, resolver y enfrentarse a problemas, usar técnicas e instrumentos matemáticos e integrar los conocimientos adquiridos [9]. La Reforma Integral de Educación Básica en su enfoque por competencias [3] afirma que "La resolución de problemas es el mejor camino para desarrollar estas competencias ya que es capaz de activar las capacidades básicas del individuo, como son: leer comprensivamente, reflexionar, establecer un plan de trabajo, revisarlo, adaptarlo, generar una hipótesis, verificar el ámbito de validez de las soluciones, etc. A su vez, posibilita experimentar, particularizar, conjeturar, elegir un lenguaje apropiado, probar una conjetura, generalizar, utilizar distintas partes de las matemáticas, verificar una solución, etc."[9].

Las competencias enfatizan en la habilidad para seguir determinados procesos de pensamiento (como la inducción y la deducción, entre otros) y aplicar algunos algoritmos de cálculo o elementos de la lógica, lo que conduce a identificar la validez de los razonamientos y a valorar el grado de certeza asociado a los resultados derivados de los razonamientos válidos. Disposición favorable y de progresiva seguridad y confianza hacia la información y las situaciones que contienen elementos o soportes matemáticos, así como hacia su utilización cuando la situación lo aconseja, basadas en el respeto y el gusto por la certeza y en su búsqueda a través del razonamiento (1ª fase: comprender). Utilizar los elementos y razonamientos matemáticos para enfrentarse a aquellas situaciones cotidianas que los precisan. Por tanto, la identificación de tales situaciones, la aplicación de estrategias de resolución de problemas, y la selección de las técnicas adecuadas para calcular, representar e interpretar la realidad a partir de la información disponible están incluidas en ella (2ª fase: pensar) [9].

3. Análisis y diseño del sistema

Para el análisis y el diseño del sistema se determinaron los Casos de Uso del sistema en UML [10]. Para Grimon un SHA debe modelar tres partes: el subsistema adaptativo, el subsistema de hipermedia y el modelo de tareas [11].
El modelo del análisis del Sistema Hipermedia Adaptativo contiene los siguientes componentes para su funcionalidad: Modelo del Usuario, Modelo de Adaptación, Modelo de Contenidos y el Modelo de Diagnóstico como se muestra en la Figura 1.

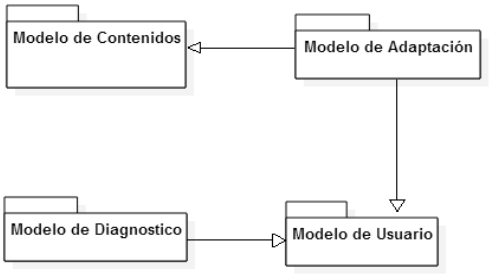


Fig. 1. Modelo de Componentes del Sistema Hipermedia

El sistema permite identificar tres usuarios para el acceso y manipulación del sistema:

- Usuario Administrador: Realizar todos los procesos del sistema como es eliminar, modificar la información y consultas en general.
- Usuario Docente: Puede realizar consultas generales, alta de materiales y modificación con respecto a los contenidos, actividades y evaluación del tema.
- Usuario Alumno: Consulta los contenidos y materiales de información, al realizar el diagnóstico, el sistema adapta y determina los tipos de aprendizajes que requiere dominar y le propone una serie de actividades para el desarrollo de las competencias.

El sistema permite presentar una adaptación para los perfiles de los distintos usuarios, como se muestra en la Figura 2, en el caso del alumno.

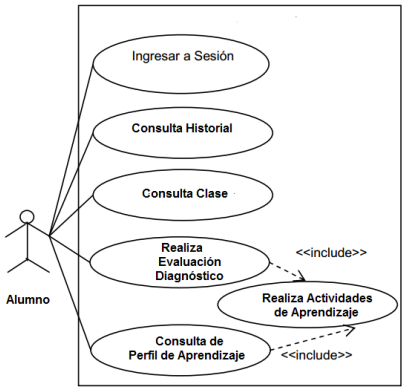


Fig. 2. Diagrama de Caso de Uso del Alumno

3.1. Diagnóstico académico para la evaluación de los aprendizajes de fracciones

El sistema presenta una adaptación para los perfiles de los distintos usuarios de acuerdo al Diagnóstico Académico. El procedimiento que se propuso para lograr en el Sistema Hipermedia Adaptativo realiza varias tareas, como son:

a) Diagnosticar los conceptos o procedimientos que no se aprendieron de Fracciones.
b) Emitir un nivel de evaluación (puntuación) del aprendizaje mostrado por el alumno con respecto al tema de Fracciones.
c) Recomendar medidas para superar problemas y mejorar los aprendizajes.
d) Incorpora fragmentos de autoinstrucción para remediar las deficiencias de aprendizaje diagnosticadas.

3.1.1. Selección de las unidades y aprendizajes relevantes

El proceso de evaluación del alumno inicia con la determinación del perfil deseable. En él se incluyen todos los conocimientos y habilidades que se esperan del estudiante, como resultado de haber intervenido en un proceso de enseñanza- aprendizaje y de acuerdo a sus necesidades.
Por lo que la adaptación de las unidades o temas seleccionados se realiza mediante los siguientes tres pasos:
a) Seleccionar las ideas principales o esenciales, o temas relacionados a los números fraccionarios.
b) Determinar los procesos cognitivos deseables para cada idea esencial.
c) Señalar el tipo de aprendizaje.
El aprendizaje se puede clasificar de diferentes formas:

- **Indispensable.** Es el aprendizaje referido a cuestiones que son básicas para el tema, unidad o área, es decir, conceptos, hechos y procedimientos que el alumno debe tener presentes toda su vida. Las ideas indispensables corresponden a un pequeño grupo dentro de las esenciales.
- **Esencial.** Es el aprendizaje que abarca todas las ideas principales extraídas del tema, unidad o área.
- **Antecedente o componente.** Son ideas que corresponden a complementos o antecedentes de la idea esencial para poder responder a los otros aprendizajes.

3.1.2. Elaboración del diagnóstico para los aprendizajes del tema de los números fraccionarios.

El método consta de tres pasos:
a) Seleccionar las ideas principales o esenciales.
b) Determinar los procesos cognitivos deseables para cada idea esencial.
c) Señalar el tipo de aprendizaje: indispensable o esencial.

Para cada idea esencial o indispensable se separa en sus componentes o antecedentes. La forma más elemental para definir las ideas esenciales consiste en plantearse la pregunta: *¿Qué debe saber o saber hacer el alumno de este nivel que aprendió el tema en cuestión?*, el Sistema está diseñado para evaluar las preguntas de acuerdo a los aprendizajes esperados de las ideas principales [12].
El sistema maneja tres niveles de desempeño de las competencias que son: 1) Inicial, 2) Regular y 3) Excelente. A los usuarios los clasifica de acuerdo a su Perfil en: Experto, Medio y Principiante. Para evaluar los contenidos se maneja en tres puntajes: 1) Bajo, menor a la media, 2) Regular, dentro de la media y 3) Alto, superior a la media. Tal como se muestra en la Figura 3.

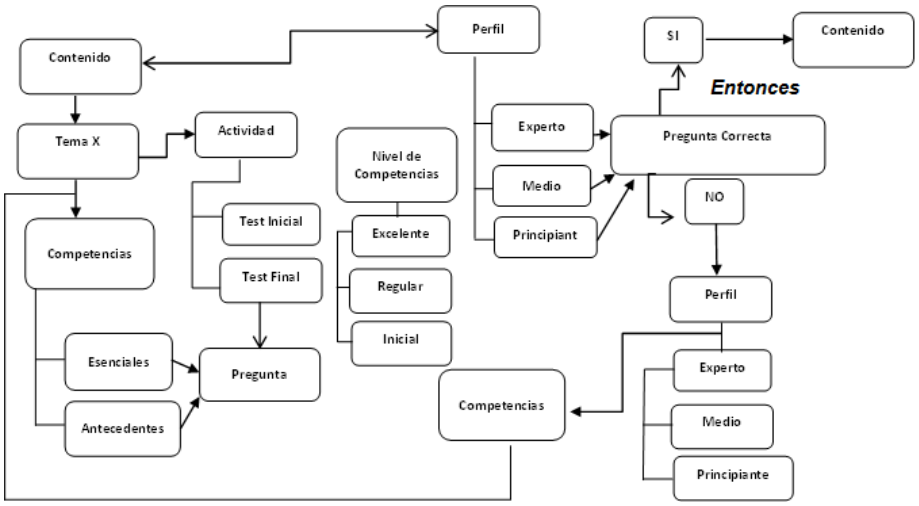


Fig. 3. Diagrama de Reglas para el Diagnóstico del SHA

Para la información del seguimiento académico se utilizó una base de datos en SQL Server, como se muestra en la Figura 4.

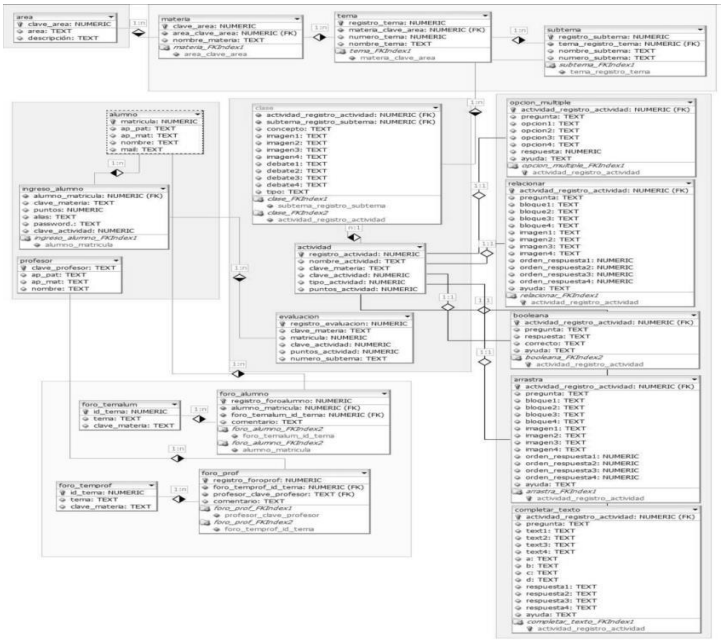


Fig. 4. Modelo de la Base de Datos del Sistema

De acuerdo a lo anterior el sistema se ha adaptado el contenido del tema de acuerdo al perfil de evaluación diagnosticado con lo cual se definieron:
-Modelo de Usuario: Representar la relación de los usuarios (alumnos) con los contenidos del tema, para lo cual fueron almacenados en una Base de Datos con aspectos relevantes de éste, como: datos personales, clases, actividades, puntuación y el diagnóstico del avance entre otra información. Como se puede ver en la Figura 4.
-Modelo de Dominio: Para obtener los contenidos de aprendizaje se utilizó una de las estructuras hipermediales que permite ir paso a paso los niveles necesarios de aprendizaje para que el alumno revise las clases, actividades, recursos y revisar las deficiencias de su aprendizaje.
-Modelo de Adaptación: Esta parte contiene la descripción de la forma en que se adapta el contenido, los recursos y las actualizaciones de su aprendizaje, para lo cual se basó en reglas Activas, que proponen una separación entre la parte del comportamiento y las entidades del modelo En esta investigación se utilizan reglas de la forma: "*Con la ocurrencia de un evento E, si la condición C se cumple, entonces el sistema ejecuta una acción A."*
Lo más importante es obtener estas reglas que aporten un conjunto de reglas para el manejo los contenidos y la evaluación de las competencias.

4. Desarrollo y pruebas del sistema hipermedia basado en competencias para el diagnóstico del aprendizaje de fracciones

Para desarrollo del sistema educativo SEMATFRAC se usó la herramienta denominada Active Server Pages (ASP) siendo un entorno de secuencias de comandos del servidor que se puede utilizar para crear páginas Web dinámicas o para generar robustas aplicaciones web. Las páginas ASP son archivos de texto con la extensión .asp que contienen etiquetas HTML, texto y comandos de secuencias. Pueden llamar a componentes ActiveX para que realicen tareas, como la conexión con bases de datos o cálculos comerciales.

El SHA se implementó en Windows Server 2003 con IIS ver6 y el uso de páginas activas que permite el uso de diferentes scripts y componentes en conjunto con HTML, para mostrar páginas generadas dinámicamente, la tecnología Active Server Pages son un ambiente de aplicación abierto y gratuito en el que se puede combinar código HTML, scripts y componentes ActiveX del servidor para crear soluciones dinámicas y poderosas para la Web [13]. El sistema SMCDAFRAC, inicia entrando a la escuela virtual. Al elegir la opción AULA, el alumno podrá seleccionar la materia de Matemáticas con el tema de FRACCIONES y comenzar su recorrido por el sistema.

El sistema educativo presenta un menú de opciones en el que se determina el rol que le corresponde al usuario en el sistema (Ver figura 5) y de acuerdo al subtema se presentan las actividades y recursos para la clase.

Fig. 5. Menú del Sistema

El nivel de competencias se revisa mediante el diagnostico iniciando con una actividad lo cual permite determinar los aprendizajes y verificar el perfil del alumno (ver Figura 6 y 7).

Fig. 6. Actividad de Diagnóstico

El alumno utiliza diferentes recursos donde se le explican conceptos y/o procedimientos que necesita aprender de acuerdo a las estrategias de aprendizaje con las que fueron diseñadas cada actividad [14]. El sistema tiene videos y animaciones para motivar a realizar las actividades y las evaluaciones.

Fig. 7. Actividad de Aprendizaje Esencial

4.1. Prueba Piloto del Sistema Hipermedia basado en Competencias para el diagnóstico del aprendizaje de Fracciones.

El sistema fue piloteado con alumnos de tercero de primaria a una muestra n=40 cuyo objetivo fue obtener el diagnóstico inicial y final. Las evaluaciones permitieron revisar el nivel de logro del desarrollo de las competencias en matemáticas que adquirieron por los alumnos durante el ciclo escolar 2012-2013, siendo el sistema una herramienta de apoyo utilizada con los alumnos de forma presencial durante las sesiones de clases. Las cuatro competencias que se midieron son: Resolver problemas de manera autónoma, Comunicar información matemática, Validar procedimientos y resultados y Manejo de técnicas eficientes donde cada una representa un 25% del total de las competencias y a partir de esta información es como se realiza la evaluación.

La competencia que más se desarrolló fue "Resolver problemas de manera autónoma" con un 75% de desarrollo, de 10% a un 85% y la segunda fue "El manejo de técnicas eficientes" con un 65%, logrando alcanzar de un 25% a un 90% esto conlleva que el uso de un SHA propicia un aprendizaje más significativo en el tema de fracciones, como se muestra en la Tabla1.

Tabla 1. Resultados del Test Inicial, Test Final y Aportación del desarrollo para el Desarrollo de Competencias mediante el SHA

Competencias Matemáticas	Test Inicial	Test Final	Uso del Sistema
Resolver problemas de manera autónoma	10%/	85%	75%
Comunicar información matemática	45%	90%	45%
Validar procedimientos y resultados	20%	80%	60%
Manejo de técnicas eficientes	25%	90%	65%

A partir de lo anterior se identificó que el promedio de evaluación fue de 86.25%, y se clasificaron los niveles de aprendizaje de acuerdo a los puntajes del Test final.

El nivel de desarrollo de competencias que los alumnos adquirieron con respecto al tema de fracciones, donde la mayoría se encuentra en el nivel "Regular", que demuestra que adquirieron las habilidades esenciales e indispensables para la vida.

Por otra parte el nivel de excelente identifica a los alumnos con mayor dominio de las competencias siendo solo el 10% de la muestra. Tal como se presenta en la Tabla 2.

Tabla 2. Resultados obtenidos del Nivel de Competencias de los Alumnos

Competencias Matemáticas	Nivel de Competencia		
	Inicial	Regular	Excelente
Resolver problemas de manera autónoma	15%	60%	10%
Comunicar información matemática	10%	70%	10%
Validar procedimientos y resultados	30%	40%	10%
Manejo de técnicas eficientes	10%	70%	10%

5. Conclusiones

Una de las principales contribuciones del Sistema Hipermedia basado en Competencias para el diagnóstico del aprendizaje de Fracciones Matemáticas (SMCDAFRAC) es la adaptación de distintos perfiles de usuarios al aplicar el diagnóstico y evaluación de aprendizajes para establecer recorrido de contenidos y recursos de acuerdo a las necesidades de aprendizaje del alumno.

El SMCDAFRAC promueve en los estudiantes un aprendizaje contextual, experimental, participativo y de autoaprendizaje. El historial del alumno permite identificar sus deficiencias en el tema para establecer estrategias para mejorar su desempeño académico y desarrollar las competencias matemáticas.

El Sistema Hipermedia permite que el estudiante desarrolle las competencias disciplinares en las matemáticas de acuerdo a los nuevos programas educativos del 2011 y a apoyar a los docentes, ya que motiva las competencias tecno-pedagógicas al incluir el uso de las TIC en su práctica docente.

Con base a la información obtenida durante la interacción del alumno con el sistema, es importante observar que se fomentó el trabajo en equipo, la participación y el aprendizaje autónomo.

Una de las principales perspectivas de este trabajo es elaborar Sistemas Hipermedias adaptativos en otros niveles educativos: educación media superior y superior, integrando nuevas tecnologías como son los agentes inteligentes que logren tareas más

específicas y puedan establecer relaciones con otros aspectos del aprendizaje (estilos de aprendizaje, estrategias de estudio, etc.).

Así también se propone diseñar y desarrollar sistemas hipermedias adaptativos en diferentes asignaturas de acuerdo a los planes y programas de estudios vigentes, para apoyar el desarrollo de las competencias que son evaluadas por distintas pruebas internacionales, para contribuir a mejorar los resultados y la calidad de la educación en México.

Referencias

1. Coll, C.: Psicología de la educación y prácticas educativas mediadas por las tecnologías de la información y la comunicación: una mirada constructivista. Sinéctica, http://virtualeduca.org/ifdve/pdf/cesar-coll-separata.pdf
2. Navales, O.: Las tecnologías de la información y la comunicación y su impacto en la educación, Universidad Autónoma del Estado de Hidalgo, México (2005).
3. Subsecretaría de Educación Básica. *Reforma de la Educación Básica en México*. SEP, México (2011).
4. Brusilovsky, P.: Methods and techniques of adaptive hypermedia. User Modeling and User Adapted Interaction, v 6, 2-3, 87-129, Netherlands (1996).
5. Brusilovsky, P., Millan, P.: User Models for Adaptive Hypermedia y Adaptive Educational Systems. The Adaptive Web, Lecture Notes in Computer Science 4321, Berlin: Springer-Verlag, pp. 3-53 (2007).
6. Gaudioso, E.: Contribuciones al Modelado del Usuario en Entornos Adaptativos de Aprendizaje y Colaboración a través de Internet mediante técnicas de Aprendizaje Automático. Tesis Doctoral. Madrid (2002).
7. Raad, H., Causse, B.: "Modeling of an Adaptative Hypermedia System Based on Active Ru-les Springer-Verlag, 2002, pp. 149-157 (2002).
8. Bartolomé, A.: Multimedia interactivo y sus posibilidades en educación superior. Pixel-Bit 1994, no. 1, http://www.sav.us.es/pixelbit/pixelbit/articulos/n1/n1art/art11.htm
9. Gutierrez, L. Martinez, E., Nebreda, T.: Las competencias básicas en el área de Matemáticas, 2008
10. Larman, C.: UML y patrones Introducción al Análisis y Diseño Orientado a Objetos. Pearson Education, España. (2002)
11. F. Grimón, "Tesis doctoral: Modelo para la gestión de dominios de contenido en sistemas hipermedia adaptativos aplicados a entornos de educación superior semipresencial", Universidad de Barcelona, España, 2008, pp. 25-35.
12. Quesada, C, Sánchez, J.: Calificación y Diagnóstico del Aprendizaje por Computadora. México: Noriega Editores (LIMUSA).1996
13. Web Design and Application, http://www.w3.org/standards/webdesign W3C (2013).
14. Peñalosa, E.: Estrategias docentes con tecnologías, Pearson Educación de México, México (2013).

Aplicación móvil para mostrar sitios turísticos empleando realidad aumentada y geolocalización

Jonathan García Rosas, Rafael de la Rosa Flores, Hilda Castillo Zacatelco, Ana Patricia Cervantes Márquez

Facultad de Ciencias de la Computación Benemérita Universidad Autónoma de Puebla.
Av. San Claudio y 14 Sur, Ciudad Universitaria, Puebla, Puebla, México.
jesyehil@hotmail.com, {rafael, patty, hilda}@cs.buap.mx

Resumen. Las aplicaciones de realidad aumentada en móviles están enfocadas al uso de marcadores artificiales que consisten en un método invasivo en exteriores para que el usuario los enfoque y despliegue la información que dichos marcadores contengan. Se usa localización espacial por geoposicionamiento para mostrar realidad aumentada sobre determinadas áreas de la ciudad de Puebla, así como tecnologías de reconocimiento espacial usando visión artificial sin marcadores para mostrar objetos en 3 dimensiones sobre la fachada de la Catedral de Puebla

Palabras Clave: Realidad aumentada, geolocalización, marcadores

1. Introducción

La realidad aumentada es una técnica usada para combinar el mundo real con elementos virtuales en un dispositivo donde actúa primordialmente una cámara [1]. Además de estar ligada a dispositivos especiales que simulan objetos virtuales, la realidad aumentada se puede ver en aplicaciones de computadora [3] y, gracias a la proliferación de dispositivos móviles, se puede ver también en aplicaciones que hacen uso de sensores del dispositivo para dar una mayor muestra de convencimiento a los usuarios que hacen uso de ella. Gran parte de las aplicaciones de realidad aumentada están enfocadas al uso de marcadores artificiales, siendo pocas las que hacen usos de servicios de geolocalización y reconocimiento de patrones sobre marcadores naturales.

En este artículo se presentan dos formas para mostrar realidad aumentada. La primera de ellas consiste en geoposicionar marcadores sobre un navegador de realidad aumentada para mostrar su ubicación en tiempo real, mientras que la segunda forma emplea librerías y algoritmos de Vuforia para ubicar la fachada de la catedral de Puebla y superponer un objeto tridimensional al enfocar con la cámara [2]. Ambos desarrollos son empleados en un dispositivo de la línea Smartphone para el sistema operativo Android. Estas tecnologías fueron aplicadas en el sector turístico haciendo énfasis en exhibir los principales sitios de interés del centro histórico de la ciudad de Puebla. Se usan conexiones a un servicio web [4] que devuelve una lista de marcadores y

también se usa la API de Google Maps para descargar los mapas donde se muestran dichos marcadores.

2. Proceso de interacción

Esta sección describe la interacción que existe entre la aplicación ejecutada en el dispositivo móvil, los servicios de geolocalización usados y los servicios web para la obtención de información.

Al ejecutarse la aplicación se solicita la posición mediante GPS, GLONASS o A-GPS de acuerdo a las características del móvil, una vez obtenida, se envía al servidor web junto con el radio en kilómetros para verificar los puntos cercanos a la posición y devuelve los marcadores en formato JSON al cliente. Cuando el cliente recibe los datos correspondientes, se almacenan en el móvil y se muestran en pantalla auxiliados de la cámara del dispositivo, esto es, sobreponiendo la información sobre la pantalla e interactuando cada vez que el móvil cambia de dirección de enfoque. Se hace una solicitud al servidor de Google Maps para ubicar en un mapa los marcadores recibidos. La aplicación puede mostrar el mapa de forma normal o auxiliada con la cámara trasera del móvil mostrar los marcadores recibidos con realidad aumentada. Se despliegan información almacenada en el móvil de cada sitio y animaciones y objetos 3D en determinados lugares importantes (Figura 1).

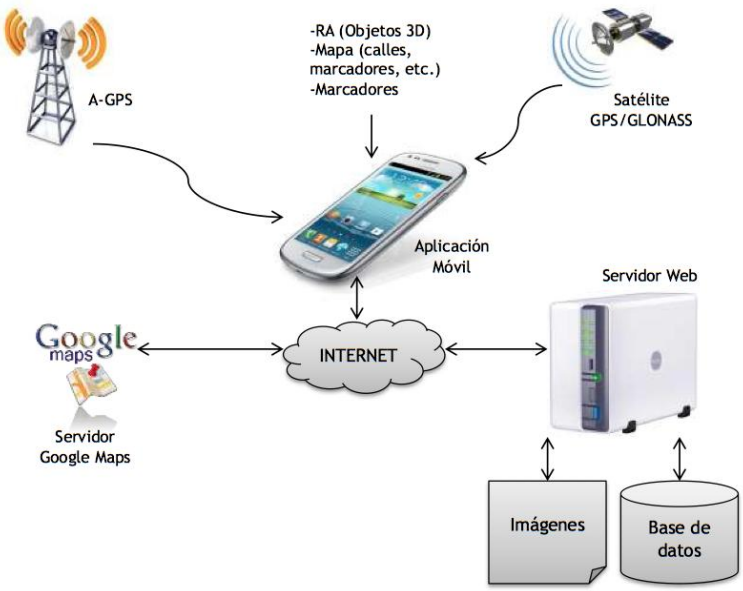


Fig. 1. Interacción del sistema.

El sistema requiere como entrada la ubicación del móvil con tres parámetros: latitud y longitud, la cual es devuelta por el GPS, GLONASS o A-GPS, y un radio de

alcance en kilómetros. Después de obtener la ubicación obtendrá la lista de lugares o marcadores cercanos (de acuerdo al radio deseado) desde el servidor en formato JSON. Un ejemplo de petición hacia el servidor sería "19.00504,-98.204773,2", donde los parámetros corresponden a la latitud, longitud y el radio de alcance respectivamente. Los marcadores recibidos se muestran en tiempo real en la pantalla del dispositivo tomando en cuenta su ubicación geográfica y son almacenados en caché mientras la aplicación esté activa.

2.1. Formatos del navegador RA

La ubicación del sitio será enviada desde el dispositivo móvil al servicio web REST de la siguiente forma:

http://148.228.xx.xx/ServicioWebRest/Api/Sitios/Sitio/latitud,longitud,radio

Donde latitud y longitud corresponden a la ubicación actual del usuario, y radio corresponde a la distancia en kilómetros a la redonda a la que se quiere obtener marcadores.

Ejemplo de envío:

http://148.228.xx.xx/ServicioWebRest/Api/Sitios/Sitio/19.00504,-98.204773,2

El servidor devuelve una lista de sitios turísticos (denominada geoCTI) calculados de acuerdo al radio que se envió como parámetro. Cada elemento de la lista tiene los siguientes parámetros: id, id_categoria, summary, tittle, elevation, lng, lat, url_imagen

Donde id es el identificador del marcador, id_categoria es el identificador para la categoría de dicho marcador, summary tiene un pequeño resumen del marcador, tittle contiene el nombre del marcador, elevation tiene la altitud a la que se encuentra geográficamente el marcador, lng es la longitud de acuerdo a la posición geográfica del marcador, lat es la latitud de acuerdo a la posición geográfica del marcador y url_imagen contiene un vínculo de una imagen del sitio que almacena el marcador.

Un ejemplo del marcador devuelto por el servicio web es el siguiente:

```
"Id":2,"id_categoria":1,"summary":"Fuente de la facultad de ciencias de la computación","tittle":"Fuente FCC","elevation":2100,"lng":-98.20451,"lat":19.005008,"url_imagen":http://cs.buap.mx
```

3. Diseño del sistema

En esta sección se describe el sistema a partir de bloques que fueron generados para dividir las tareas que se desarrollan en el sistema.

El sistema se compone de 5 bloques principales: Presentación, Navegación RA, Mapas, Visualización RA y, Servicios web. Cada bloque cuenta con distintas capas que ejecutan diversas funciones. Para el bloque de presentación, las capas se encargan de la comunicación con el servidor web y la interacción con el usuario. En el caso del bloque de Navegación RA, contiene las capas encargadas de la manipulación con el hardware del dispositivo y las operaciones que se realizan para mostrar la realidad aumentada. El bloque de Mapas es el encargado de mostrar los mapas. El bloque de

visualización RA es el encargado de gestionar las librerías de Vuforia para mostrar la realidad aumentada con vistas en 3D. Por último, el bloque de Servicio Web [4] contiene las capas encargadas de responder a peticiones por parte de los usuarios, así como interactuar con la base de datos de marcadores. En la Figura 2 se observa el diagrama por bloques del sistema y la interacción entre bloques.

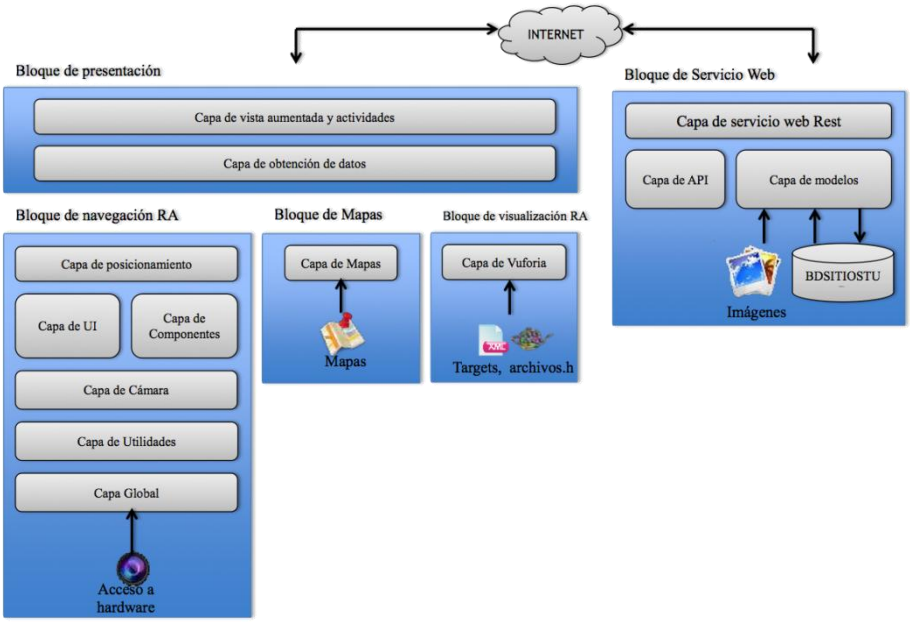


Fig. 2. Diagrama de bloques del sistema.

A continuación se describe cada bloque con sus respectivas capas.

Bloque de presentación. Este bloque es el encargado de la comunicación con el servidor web Rest así como la interacción entre el sistema y el usuario. Contiene las siguientes capas:

- *Capa de vista aumentada y actividades: Esta capa será la encargada de mostrar todas las interfaces al usuario.* Para la visualización del navegador de realidad aumentada debe tener presente el uso del bloque de navegación RA. Para la visualización de objetos en 3D se debe comunicar con el bloque de visualización RA. La visualización de mapas se realiza en conjunto con el bloque de Mapas.
- *Capa de obtención de datos: Esta capa es la encargada de la comunicación con el servidor web para obtener los sitios turísticos.*

Bloque de Navegación RA. Este bloque se encarga de ejecutar el motor de navegación de realidad aumentada.

- *Capa de posicionamiento*. Representa la ubicación física del dispositivo móvil y la disposición de los elementos en pantalla.
- *Capa de UI.* Es la encargada de dibujar los elementos como líneas, puntos, círculos, textos e imágenes.
- *Capa de componentes.* Gestiona los elementos que se muestran en pantalla, dichos elementos son: Marcadores, radar y barra de zoom.
- *Capa de cámara.* Obtiene el acceso a la cámara del dispositivo y contiene las clases para mostrar la imagen capturada en pantalla en tiempo real.
- *Capa de utilidades.* Contiene las clases que realizan las operaciones matemáticas necesarias para mostrar marcadores y calcular la inclinación del dispositivo.
- *Capa global.* Esta capa se encarga del almacenamiento general de los sitios mientras corre la aplicación.

Bloque de Mapas. Este bloque es el encargado de controlar la comunicación con el API de Google Maps [8] para mostrar mapas.

- *Capa de mapas.* Esta clase muestra los mapas y los sitios turísticos.

Bloque de visualización RA. Este bloque es el encargado de mostrar la realidad aumentada a partir de marcadores naturales usando las librerías de Vuforia.

- *Capa de Vuforia.* Esta capa se encarga de renderizar y mostrar un objeto 3D, así como del preprocesamiento digital de una imagen en tiempo real.

Bloque de Servicio Web. Este bloque contiene todas las capas para el manejo de solicitudes en un servicio web. Cuenta con las siguientes capas:

- *Capa de Servicio Web Rest.* Esta capa será la encargada de recibir solicitudes desde el equipo móvil. Es capaz de devolver la información de los sitios cercanos a la posición desde donde se hizo la solicitud.
- *Capa de modelo.* Esta capa contiene las operaciones y consultas que se realizan con la base de datos y se encarga de procesar la información que se reciba para devolverla en el formato correcto.
- *Capa de API.* Esta capa redirige las acciones hacia el Controlador del servicio web a partir de la solicitud url.

4. Implementación

En esta sección se muestra la implementación de la aplicación Android y el servicio web, ambos casos de acuerdo a sus bloques correspondientes. Se tomaron en cuenta distintas tecnologías de desarrollo. En el caso del desarrollo de la aplicación en Android, se programaron las clases en Java y las interfaces en XML con el IDE Eclipse siguiendo el modelo vista controlador [5]. Se usó el lenguaje de programación C++ para los objetos 3D que se despliegan mientras se observa la realidad aumentada,

además de usar las librerías de Vuforia. En el caso del servicio web, se implementó un Servicio web Rest con las tecnologías de ASP.NET MVC para la programación [6] y SQL Server para el almacenamiento de los sitios turísticos.

De acuerdo al diagrama de bloques del sistema que se muestra en la sección de Diseño, los cuatro primeros bloques contienen las clases para la aplicación móvil, mientras que el último bloque contiene las clases para el servidor web Rest.

4.1. Bloque de presentación

La capa de vista aumentada y actividades contiene las vistas desarrolladas en XML y las que se crean en tiempo de ejecución, las cuales muestran tres opciones de navegación: navegador RA, catedral y Acerca de (Figura 3).

Fig. 3. Interfaz de la actividad inicios.

Para la capa de obtención de datos, se tienen las clases que realizan las conexiones para traer los marcadores desde el servicio web REST usando una dirección URL como se muestra en el siguiente ejemplo:

http://148.228.xx.xx/ServicioWebRest/Api/Sitios/Sitio/

4.2. Bloque de navegación RA

En este bloque se crean las clases para el motor de navegación de realidad aumentada. Este contiene la capa de posicionamiento, que se usa para determinar la posición geográfica del dispositivo (con los parámetros latitud, longitud y altitud), su posición física y la posición de los objetos en pantalla (atributos x y y).

La capa de Interfaz de usuario (UI) Es la encargada de dibujar todos los componentes que se muestran en la interfaz. Los objetos que se dibujan en tiempo de ejecución son los siguientes: Cajas (Figura 4), Textos (Figura 4), Imágenes, Círculos (Figura 5), Íconos (Figura 6), Líneas (Figura 5), Puntos (Figura 5).

Las siguientes Figuras ilustran los objetos que pueden ser creados en la capa de Interfaz de usuario.

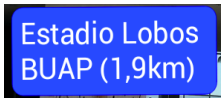


Fig. 4. Caja con texto con referencia de un marcador

Fig. 5. Circulo dibujado para simular un radar con dos líneas, además de puntos de posiciones.

Fig. 6. Ejemplo de ícono de un marcador de la categoría monumento.

La capa de componentes se encarga de fusionar los dibujos realizados en la capa de interfaz de usuario. Se muestran tres componentes principales en pantalla: Radar (Figura 7), Marcador (Figura 8) y Barra vertical (Figura 9).

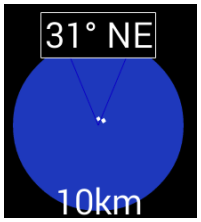


Fig. 7. Radar en la interfaz de navegación RA.

Fig. 8. Marcador en la interfaz de navegación RA.

Fig. 9. Barra vertical de zoom (SeekBar).

La capa de cámara es la encargada del acceso a la cámara en el dispositivo Android. La capa de Utilidades se encarga de las operaciones matemáticas necesarias para posicionar los objetos en la pantalla mediante el uso de matrices, y además se encarga de la obtención de los ángulos de inclinación del dispositivo.

La capa global es la encargada de almacenar los marcadores e imágenes que se usan mientras la aplicación esté corriendo.

4.3. Bloque de mapas

En este bloque se gestionan las conexiones con el API de Google Maps [8] para mostrar los mapas en una actividad y para mostrar rutas entre la ubicación del dispositivo, y un marcador en concreto.

La capa de mapas muestra el mapa con la ubicación del dispositivo móvil además de los marcadores con los sitios turísticos en un radio cercano a la ubicación del dispositivo. Tiene la particularidad de trazar rutas desde el punto en que se encuentra el dispositivo y otro marcador seleccionado mediante el API de rutas Google Maps [8].

4.4. Bloque de visualización de RA

Este bloque contiene las clases de Vuforia para el procesamiento de imágenes y la renderización de un objeto en tres dimensiones para superponerlo sobre una escena, en este caso, la fachada de la catedral de Puebla.

La capa de Vuforia se encarga del procesamiento de imágenes en tiempo real desde la cámara del dispositivo. Su función es encontrar el marcador en una escena y sobreponer un objeto renderizado en 3D para su visualización [7]. La creación de marcadores se realiza desde la página de Vuforia (Target Manager Vuforia). Para ello se necesitan fotografías del lugar en el que deseamos mostrar la realidad aumentada. Este sistema nos devuelve un archivo binario .dat y otro XML para referenciar el marcador cuando se ejecute la aplicación. En la Figura 10 se muestra uno de los marcadores usados para la catedral de la ciudad de Puebla.

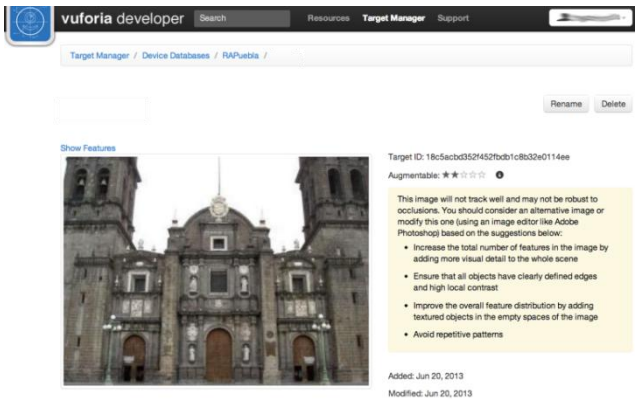


Fig. 10. Marcador implementado para la catedral de Puebla. Está almacenado en un archivo binario .dat al que se accede desde el tracking generado en XML.

Para la creación del objeto en 3D que será visualizado sobre la fachada, se usa Blender para modelar el objeto, determinar su textura y su posición inicial. En la Figura 11 se observa su perspectiva de diseño.

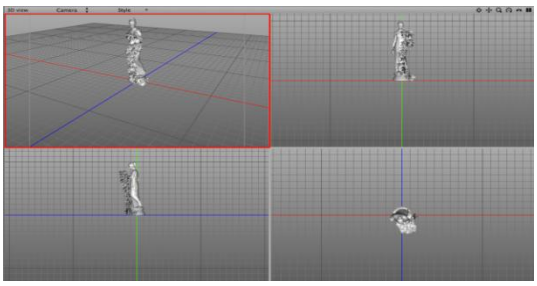


Fig. 11. Perspectivas el diseño de un objeto en 3D con el software Blender. Este objeto es usado para mostrarlo sobre el marcador al que se apunte con la cámara.

4.5. Bloque de servicio web

Este bloque maneja las peticiones que se realizan por medio del método GET de http, gestiona las solicitudes con la base de datos, además de realizar los cálculos para determinar la distancia a la que se encuentran los marcadores de la posición enviada y devolver los marcadores que correspondan al radio solicitado.

La capa de servicio web Rest contiene los controladores para recibir peticiones desde el dispositivo móvil o desde cualquier navegador web, con la particularidad de serializar una lista de marcadores que se devuelven en formato JSON. Esta capa, fue concebida como una API para facilitar el acceso a los marcadores y su uso a la conveniencia que se disponga. Un ejemplo de objeto (denominado geoCTI) serializado que se envía de retorno es el siguiente:

```
{
  "geoCTI":
  [
    {
      "Id":2,
      "id_categoria":1,
"summary":"Fuente de la facultad de ciencias de la computación",
      "tittle":"Fuente FCC",
      "elevation":2029,
      "lng":-98.20451,
      "lat":19.005008,
      "url_imagen":"http://cs.buap.mx"
    }
  ]
}
```

5. Implementación

La aplicación puede ser instalada al generar un archivo .apk desde el entorno de desarrollo (para este caso, se uso el IDE Eclipse). Al arrancar la aplicación en un dispositivo con Android, se muestra la interfaz de inicio con una actividad por dos segundos. Destacan el título con el nombre la aplicación, el ícono de la aplicación así como el logotipo del CTI y de Vuforia (Figura 12).

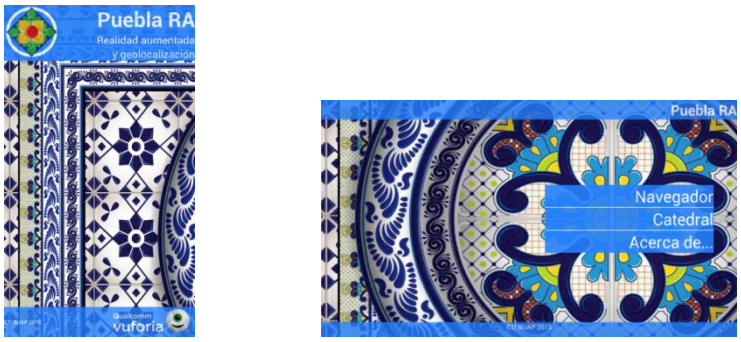


Fig. 12. Arranque de la aplicación (izquierda) y menú (derecha)

5.1. Navegador RA

El navegador de realidad aumentada se muestra accediendo desde el menú en la opción Navegador. Esto mostrará una actividad como se observa en la Figura 13.

Fig. 13. Vista principal del navegador de realidad aumentada.

El navegador RA contiene tres componentes que se describen a continuación:

- *Radar*. Muestra todos los marcadores que se encuentren dentro del límite elegido con la barra de zoom. Cada marcador o sitio turístico es mostrado como un punto de color blanco y cambian de acuerdo a la orientación del dispositivo. En la parte superior del Radar se observa la orientación actual del dispositivo, mientras que en la parte inferior se muestra el radio en metros o kilómetros a los que se desea ver marcadores (Figura 14).

Fig. 14. Vista del Radar en el navegador con orientación hacia el sur y un radio de 1 km.

- *Barra de zoom.* Se encarga de incrementar o disminuir el radio al que se desean ver los marcadores. El radio máximo de visualización es de 10 kilómetros. Al incrementar o disminuir el valor de la barra de zoom, el valor del Radio también cambia. La Figura 15 muestra la barra de zoom de la aplicación.

Fig. 15. Barra de zoom para cambiar el radio.

- *Marcadores.* Los marcadores se muestran sobre la vista de la pantalla de acuerdo a su enfoque en determinada orientación. Todos los marcadores se muestran al enfocar hacia un lugar con el dispositivo móvil en posición horizontal. Estos son mostrados con el nombre del marcador y un ícono de acuerdo a su categoría. En la Figura 16 se observan 2 marcadores al enfocar al sureste.

Fig. 16. Marcadores o sitios enfocados al sureste.

Al pulsar sobre un marcador, nos muestra un cuadro de diálogo en el cual vemos el nombre del marcador, una fotografía de referencia del sitio y la descripción del sitio turístico. También tiene la opción “Como llegar” para visualizar la ruta más cercana a pie para llegar al sitio (Figura 17).

Fig. 17. Al pulsar un marcador, se muestra información del mismo con la opción de trazar una ruta para llegar a él.

5.2. Mapa

Cuando giramos nuestro dispositivo de forma que quede en paralelo al suelo, se muestra el mapa con nuestra ubicación y todos los marcadores que se encuentren dentro del radio deseado. Cada marcador es representado con el ícono de su categoría (Figura 18).

Fig. 18. Mapa con marcadores.

Al elegir la opción “Como llegar” sobre un marcador, se muestra la ruta más corta en el mapa para llegar a dicho sitio (Figura 19).

Fig. 19. Ruta a pie para llegar desde la ubicación actual hacia el marcador elegido.

5.3. Catedral

Al elegir la opción Catedral desde el menú principal, nos muestra una vista con las indicaciones para visualizar la realidad aumentada en la catedral de la ciudad de Puebla. Para ello, se presiona el botón Iniciar.

Al iniciar esta actividad, se muestra una vista de la cámara del dispositivo. La principal diferencia entre la vista de esta cámara con la vista del navegador RA es la precisión para auto enfocar auxiliado del hardware del dispositivo (en el caso en que la cámara del dispositivo cuente con autoenfoque). Para mostrar la realidad aumentada es necesario seguir las siguientes indicaciones:

1. Ubíquese frente a la fachada principal de la catedral.
2. Apunte con el dispositivo con dirección a la fachada.
3. Visualizar la realidad aumentada que se despliega.

En la Figura 18 se observa el objeto estatua mostrado al enfocar la catedral de la ciudad de Puebla. En la Figura 19 se observa el mismo objeto en modo apaisado del dispositivo.

Fig. 20. Objeto estatua superpuesto sobre la fachada de la catedral.

Fig. 21. Objeto estatua superpuesto sobre la fachada de la catedral en modo apaisado.

6. Conclusiones y perspectivas

En este trabajo se presentan dos alternativas de realidad aumentada empleada como método de divulgación de sitios de interés, además de métodos de geolocalización para ubicar cada sitio. Mediante el empleo de estas tecnologías se creó una aplicación para Android que aprovecha los recursos y sensores con los que cuentan los dispositivos móviles hoy en día. Al ser los recursos de hardware con los que cuente el móvil uno de los aspectos más importantes a la hora de desarrollar aplicaciones, estos se aprovechan al máximo puesto que son los encargados de realizar tareas de reconocimiento de imágenes en tiempo real, renderización de objetos en tercera dimensión, así como posicionamiento en pantalla de objetos en 2D, tareas que ocupan un gran porcentaje de la memoria y el poder de procesamiento del procesador del dispositivo. El consumo de recursos de memoria es muy alto para este tipo de tareas y la aplicación desarrollada solo está disponible para dispositivos que cuenten con ciertas características de hardware. El sistema cuenta con un límite de veinte marcadores que se muestran en tiempo real, esto debido a la carga de objetos que se muestran puede resultar muy grande en cuanto al consumo de memoria.

Las tecnologías usadas en este proyecto son tanto de uso libre como privativas. Es necesario mencionar que se usaron marcas registradas como es el caso de Vuforia, que exigen que un logotipo de dicha marca sea mostrada cuando se usa su tecnología para desarrollar y mostrar realidad aumentada. Otro caso es el uso de la API de Google Maps, la cual tiene un límite de 20,000 consultas diarias desde la aplicación y además se exige que los datos que sean consultados a sus servidores (como el uso de las rutas) sean expresamente utilizados para visualizarlos en un mapa.

Otro aspecto a tomar en cuenta es que la aplicación desarrollada necesita de una constante conexión a internet para descargar la información desde un servidor. Cuando se eligió la forma de transmisión de datos, se determinó usar el formato JSON, el cual es un formato muy popular de transmisión así como uno de los que menos tráfico genera a la hora de realizar las consultas a un servidor. Aunque el sistema está preparado para funcionar con redes WiFi, se debe contemplar el uso de datos por parte de un proveedor de internet para móviles que garantice una conexión a internet en todo momento. Otro aspecto a tomar en cuenta es el uso de un chip GPS instalado en el dispositivo, puesto que la aplicación está preparada para funcionar sin la necesidad del mismo, el uso de este servicio mejorará la precisión de la ubicación del dispositivo y los sitios turísticos a su alrededor.

El servicio web se desarrolló con la finalidad de que cualquier persona pueda consultar los lugares que se encuentran a su alrededor, pero con la limitación de obtener solo los 20 lugares más cercanos de acuerdo a su posición o la posición que ellos especifiquen, pero existe la opción de consultar todos los lugares que se encuentren en la base de datos, con la limitantes de que cada persona debe ocupar y manipular los datos de acuerdo a su conveniencia. Dicho servicio web fue probado en el laboratorio del Centro de Tecnologías de la Información de la Facultad de Ciencias de la Computación de la BUAP.

Referencias

1. Raghav Sood. Pro Android Augmented Reality (2012) Chapter 1, Applications of Augmented Reality. Pages 1-12. Chapter 2, Basics of Augmented Reality on the Android Platform. Pages 13-31.
2. Lee, T., Höllerer, T. (2007) Handy AR: Markerless Inspection of Augmented Reality Objects Using Fingertip Tracking. IEEE International Symposium on Wearable Computers (ISWC '07).
3. Kato, H., Billinghurst, M.(1999) Marker tracking and HMD calibration for a video-based augmented reality conferencing system, Proceedings of the 2nd IEEE and ACM International Workshop on Augmented Reality (IWAR 99).
4. Alfonso Felipe Lima Cortés (2012): Desarrollo de aplicación en Android con acceso a Web Service. Introducción al desarrollo de aplicaciones móviles en Android. Tercer congreso regional en TIC.
5. Trygve Reenskaug and James Coplien, (2009). The DCI Architecture: A New Vision of Object-Oriented Programming, www.artima.com/articles/dci_vision.html
6. Simple Example of MVC (Model View Controller) Design Pattern for Abstraction. http://www.codeproject.com/Articles/25057/Simple-Example-of-MVC-Model-View-Controller-Design Consultado Abril de 2013.
7. Vuforia developer Resources. Vuforia SDK Architecture. https://developer.vuforia.com/resources/dev-guide/vuforia-ar-architecture Consultado Agosto de 2013.
8. API de Google Maps, https://developers.google.com/maps/?hl=es, Consultado Noviembre del 2014.

Atribución de autoría combinando información léxico-sintáctica mediante representaciones holográficas reducidas

Jovany Marcos Ramírez, Maya Carillo Ruíz, María Josefa Somodevilla

Benemérita Universidad Autónoma de Puebla
jo.va.ny@hotmail.com
{crrllrzmy, mariajsomodevilla}@gmail.com

Resumen: En este artículo se propone la utilización de la representación holográfica reducida (HRR) en la tarea de Atribución de Autoría (AA). Dicha representación permite combinar información léxica y sintáctica de los textos. En vectores de dimensión manejable. Para contar con vectores de dimensión apropiada se aplica la metodología de Indexación Aleatoria (RI). Los experimentos realizados muestran que la HRR genera resultados equiparables a los reportados en la bibliografía.

Palabras clave: Atribución de autoría, Representaciones holográficas reducidas, Indexación aleatoria.

1. Introducción

La Atribución de Autoría (AA) es una tarea que busca caracterizar el estilo de escritura de autores, con el fin de asignar de forma automática textos de autoría desconocida al autor correspondiente Es decir busca identificar características textuales que al compararlas, permitan discriminar entre documentos escritos por distintos autores [1].

Los métodos generales de AA extraen marcadores de estilo, que se consideran atributos de los textos y se utiliza para entrenar clasificadores [2]. Estas marcas de estilo, incluyen: frecuencia de caracteres, palabras, frase, n-gramas a nivel carácter, combinaciones de palabras o n-gramas de palabras, por mencionar algunos. Es importante señalar que la AA no debe ser abordada de forma temática, ya que las características textuales más importantes no son de tipo temático, pues el objetivo es modelar el tipo de escritura de cada autor con el fin de discriminarlos, incluso en el mismo contexto.

Hoy en día, la cantidad de información disponible es abrumadora y gran parte de ella está en texto plano (e-mails, blogs, foros en línea). En este contexto, han surgido diversos temas que involucran a la AA, por ejemplo: ciber-bullying, detección de plagio, correo no deseado, informática forense, detección de fraude, autenticidad de documentos [3].

En el presente trabajo se propone un método para combinar características léxicas y sintácticas empleando una representación novedosa conocida como representación holográfica reducidas (HRR). La HRR fue propuesta por Plate [6] como mecanismo para representar estructuras complejas y jerárquicas, que no se limitan al lenguaje, pues este tipo de estructuras se encuentran en otras áreas como el análisis de imágenes.

Por otra parte cuando los documentos se representan utilizando la aproximación de bolsa de palabras, sabemos que la dimensión vectorial es igual al tamaño del vocabulario de la colección. Para optimizar el procesamiento existen métodos que buscan reducir dicha dimensión, uno de los más utilizados en recuperación de información es la indexación semántica latente. Sin embargo éste método hace uso de la descomposición en valores singulares que es un proceso computacionalmente caro. Como alternativa Salgren [9] proponen un método conocido como indexación aleatoria (Random Indexing RI por sus siglas en inglés). En el presente trabajo se utiliza la RI para reducir la dimensión vectorial y optimizar el tiempo de procesamiento. La aportación principal de la presente investigación es la utilización de la HRR a la tarea de AA, donde no ha sido utilizada, de acuerdo a la información que se tiene hasta el momento.

Este artículo está organizado de la siguiente manera. En la sección 2 se presentan algunos trabajos relacionados, la sección 3 explica lo que es la representación holográfica reducida. La sección 4 introduce la metodología de indexación aleatoria (RI).La metodología propuesta se describe en la sección 5. Los corpus utilizados así como algunas características de los mismos y el tratamiento previo que se dio a los documentos obtenidos se presentan en la sección 6, en la sección 7 se describen los experimentos realizados, así como los resultados obtenidos. En la sección 8 están las conclusiones y trabajo futuro.

2. Trabajo relacionado

Existen dos formas para realizar AA, 1) el enfoque basado en el perfil del autor, en este se concatenan todos los documentos de un autor presentes en el conjunto de entrenamiento, para crear su perfil. Esto se realiza extrayendo varias características principalmente de bajo nivel, tales como n-gramas de caracteres. Para predecir la autoría de un documento nuevo, se debe calcular la similitud entre los perfiles de autor generados y las características del nuevo documento. Posteriormente el documento de autoría desconocida, se asignará al autor, cuyo perfil tuvo la mayor similitud con él [3], su estructura típica es mostrada en la Figura 1.

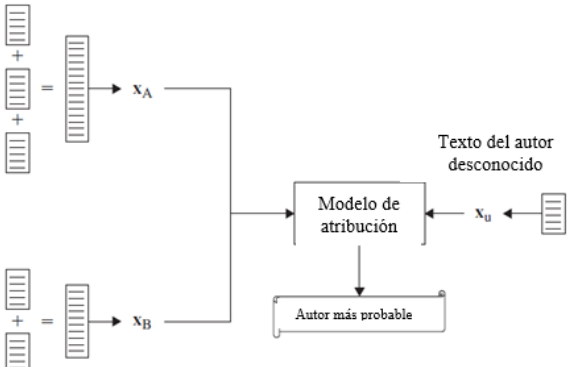


Fig. 1.- Enfoque basado en perfil del autor

2) En contraste el enfoque basado en máquina de aprendizaje, utilizan una representación vectorial, donde cada documento se representa de forma individual por un conjunto de características. Dichos vectores serán utilizados para entrenar un algoritmo de aprendizaje automático. Estos vectores suelen contener características variadas, desde caracteres, longitud de palabras, n-gramas de caracteres, n-gramas de palabras, y partes de la oración (POS). [3], su estructura típica podremos observarla en la Figura 2.

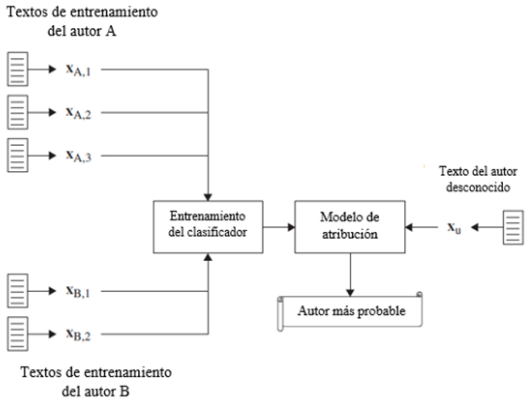


Fig 2. Enfoque basado en instancias

Existen diversas investigaciones en las cuales se han buscado métodos eficientes para la AA, entre ellos podemos mencionar: *Authorship Attribution Using Probabilistic Context-Free Grammars* de Raghavan et al. [2]. Raghavan plantea construir una gramática libre de contexto probabilística para cada autor y el uso de esta gramática como un modelo de lenguaje para la clasificación. Grigori Sidorov et al [4] propone en su investigación el uso de n-gramas, pero no de la manera tradicional, si no obteniendo los n-gramas en función del orden como se presentan en los árboles sintácti-

cos, es decir, seguir el camino del árbol sintáctico para crear los n-gramas, dando a estos el nombre de n-gramas sintácticos (sn-gramas) [4], entre otros.

3. Representación holográfica reducida

Las representaciones holográficas reducidas (HRR´s), son vectores cuyos elementos siguen una distribución normal, con media = 0 (μ=0) y desviación estándar = 1 (σ = 1). Hace uso dl operador de convolución circular ($\otimes$), para combinar la información léxica $x=(x_o, x_1,\ldots,x_{n-1})$ y sintáctica $y=(y_o,y_1,\ldots,y_{n-1})$ en $z = (z_0,z_1,\ldots,z_{n-1})$, . Así z se define como $z = x \otimes y$ [5]

$$z_i = \sum_{k=0}^{n-1} x_k y_{i-k} \qquad i = 0 \text{ a } n\text{-}1 \text{ (subíndices son módulo } n) \textbf{ (1)}$$

4. Indexación aleatoria

Con el uso de la indexación aleatoria (RI) se pretende reducir el espacio vectorial con el cual se trabajará, es decir, cada contexto (documento o palabra) se representa como un vector de tamaño fijo conocido como vector índice (VI). Las entradas de estos vectores serán ceros, con un pequeño número de elementos diferentes de cero, que serán 1's y -1's, en igual proporción. Por ejemplo si los vectores tienen veinte elementos distintos de ceros en un espacio vectorial de 1024, éste tendrá diez 1´s y diez -1´s, estos vectores servirán como etiquetas para las palabras o documentos [7].

La indexación aleatoria se lleva a cabo de acuerdo a los siguientes pasos:

1.- En primer lugar cada contexto (por ejemplo, cada documento o cada palabra) se le asigna una representación única y generada aleatoriamente, se le llama vector índice. Estos vectores de índice sin escasos, de alta dimensión y ternarios, lo que significa que su dimensionalidad (d) es del orden de miles, y que consiste en un pequeño número de 1´s y -1´s distribuidos al azar, el resto de los elementos se establecen en 0's.

2.- A continuación, los vectores de contexto se producen mediante el escaneo a través del texto, y cada vez que una palabra se produce en el contexto (por ejemplo, en un documento, o dentro de una ventana de contexto por deslizamiento), se añade el vector de contexto para la palabra en cuestión. Las palabras son así representadas por vectores de dimensión d que son efectivamente la suma de los contextos de las palabras.

5. Metodología propuesta

La metodología seguida se describe a continuación:

1. Primeramente se preprocesaron los documentos del conjunto de entrenamiento y pruebas, eliminando símbolos no alfanuméricos, así como los valores numéricos.

2. Una vez preprocesados los documentos, con ayuda del etiquetador de partes de la oración de Stanford [8], se procedió a etiquetar cada uno de los documento del conjunto de entrenamiento., Etiquetados todos los documentos se identificaron las etiquetas sintácticas únicas contenidas en estos documentos. El número total de etiquetas para los corpus utilizados se presentan en la Tabla 1.

3. Se utilizó RI para reducir el espacio vectorial, representando todo el vocabulario como vectores de ceros, unos y menos unos. Las etiquetas sintácticas se representaron como HRR y se asociaron mediante a convolución circular a los vectores generados por RI. La dimensión del espacio vectorial para todos los experimentos fue de 2048.

4. Se crearon las representaciones para la aproximación basada en instancia y en perfil

5. Obtenidos los vectores tanto para el conjunto de entrenamiento como el de prueba, para el enfoque basado en instancia se experimentó con los clasificadores J48, Naive Bayes, SVM. Para los enfoques basados en perfil del autor se hizo uso de la distancia euclidiana.

A continuación se ejemplifican los pasos para representar a los documentos. La salida del etiquetador Stanford, está en la Figura 3, en la cual se muestra una fracción de uno de los documentos etiquetados.

```
the_DT one_CD highly_RB visible_JJ success_NN of_IN the_DT stimulus_NN
program_NN has_VBZ been_VBN the_DT cash_NN for_IN clunkers_NNS
```

Fig 3. Etiquetado del texto

En la Figura 4 se ilustra la representación de los documentos. Las etiquetas sintácticas, se representaron como HRRs y las palabras como VI. Estos se combinaron con

la convolución circular. Finalmente los documentos se representaron como la suma de sus palabras representadas como HRRs.

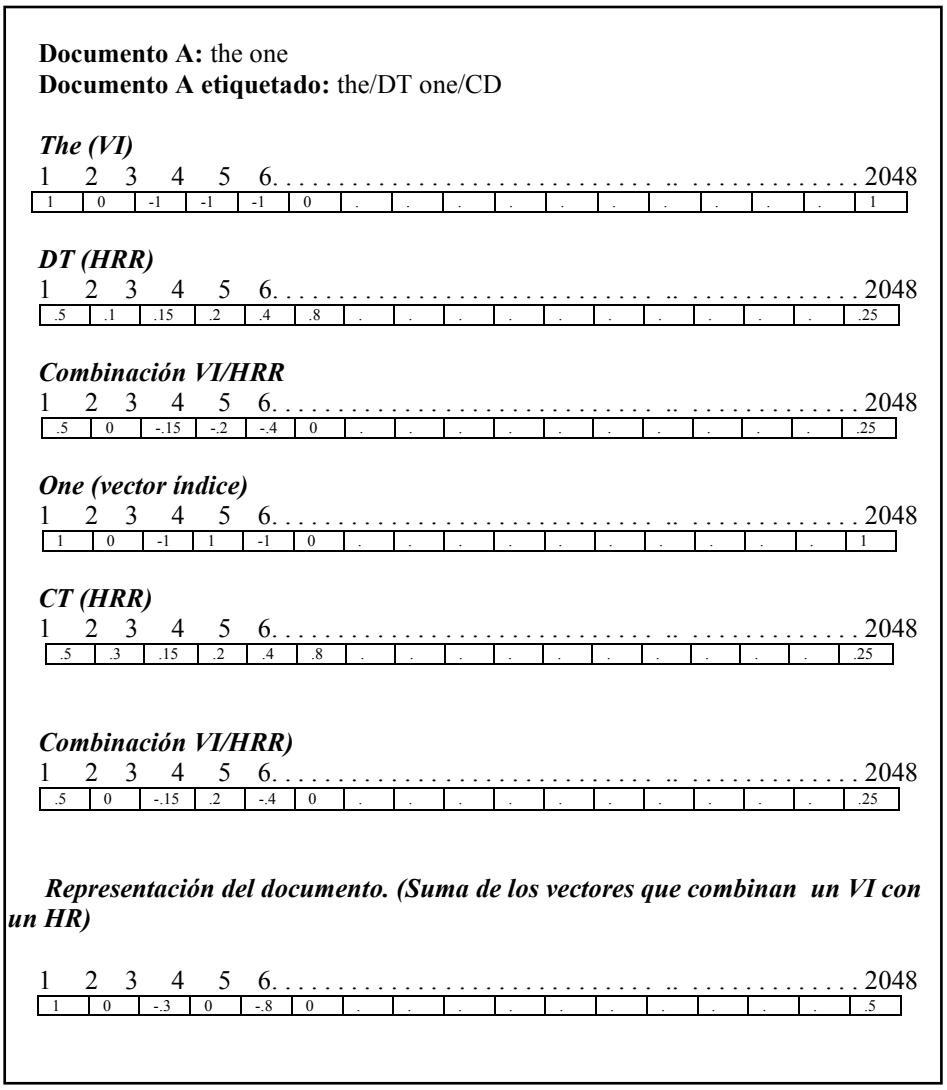


Fig 4. Representación de un documento como HRR

6. Conjunto de datos utilizados

Para los experimentos reportados a continuación, se utilizó un conjunto de 3 colecciones (Poetry, Business, NFL), con un total de 15 autores (6 para Poetry, 6 para Business y 3 para NFL), cabe destacar y hacer énfasis en que el conjunto de documentos de cada autor es pequeño. La Tabla 1, muestra información relevante de dichos corpus.

Tabla 1: Estadísticas sobre los corpus utilizados en los experimentos.

	Vocabulario	Etiquetas Sintácticas	Documentos Train	Documentos Test	Autores
Poetas	6940	34	146	55	6
Negocios	8492	34	85	90	6
NFL	4982	34	48	45	3

7. Experimentos y resultados

Se experimentó con los dos tipos de enfoques mencionados anteriormente: basados en instancias y los basados en perfil. Cabe destacar que para estos experimentos las colecciones fueron normalizadas por número de documentos y por el tamaño del vocabulario.

7.1. Resultados

A continuación se reportan los resultados obtenidos. La aproximación basada en instancias como podrá observase generó resultados muy pobres por lo que se descartó y únicamente se experimentó con la aproximación de perfil de autor.

7.1.1. Resultados del enfoque basado en instancias.

Como se aprecia en la Tabla 2, los resultados obtenidos en el enfoque basado en instancias son en su mayoría bajos y en ocasiones no se obtiene ningún resultado favorable, es decir, no se logra identificar o predecir de forma correcta los documentos que pertenecen al autor en específico, como es el caso del Autor 3 y el autor 6.

Tabla 2: Corpus Poetry, utilizando: J48, Naive Bayes, SVM

	Precisión	Recuerdo	Medida F
Autor 1	0.40	**0.90**	0.56
Autor 2	0.28	0.40	0.33
Autor 3	0.00	0.00	0.00
Autor 4	**0.66**	0.40	0.50
Autor 5	0.54	0.60	**0.57**
Autor 6	0.00	0.00	0.00

7.1.2. Resultados del enfoque basado en perfil del autor.

Los resultados obtenidos del enfoque basado en perfil del autor, fueron más favorables como podrá observarse.

Para el corpus Poetry los resultados se muestran en la Tabla 3.

Tabla 3: Métricas obtenidas para el corpus Poetry

	Precisión	Recuerdo	Medida F	Exactitud
Autor 1	0.70	0.70	0.70	**0.89**
Autor 2	0.50	0.40	0.44	**0.81**
Autor 3	0.55	0.50	0.52	**0.83**
Autor 4	0.36	0.40	0.38	**0.76**
Autor 5	0.45	0.50	0.47	**0.80**
Autor 6	0.33	0.40	0.36	**0.87**

En la Tabla 4 se muestran los resultados para el corpus NFL.

Tabla 4. Métricas obtenidas para el corpus NFL.

	Precisión	Recuerdo	Medida F	Exactitud
Autor 1	0.88	0.90	0.93	**0.95**
Autor 2	0.66	**0.80**	0.72	**0.80**
Autor 3	**0.90**	0.60	0.72	0.84

La tabla 5 muestra los resultados obtenidos del corpus de Business, con mejores resultados que los anteriores. Esto probablemente a que en este corpus los documentos de entrenamiento y pruebas están balanceados, es decir, la misma cantidad de documentos para los autores.

Tabla 5. Métricas para el corpus Business

	Precisión	Recuerdo	Medida F	Exactitud
Autor 1	0.86	0.86	0.86	**0.95**
Autor 2	0.68	0.86	0.76	**0.91**
Autor 3	0.81	0.86	0.83	**0.94**
Autor 4	0.78	0.73	0.75	**0.92**
Autor 5	0.90	0.66	0.76	**0.93**
Autor 6	0.86	0.86	0.86	**0.95**

Se compararon nuestros resultados con resultados tomados del *artículo Authorship Attribution Using Probabilistic Context-Free Grammars [4]*. Se debe tener en cuenta que los corpus utilizados en este artículo fueron completados con secciones de Penn Treebank [11].

La comparación de nuestros resultados para el corpus Poetry, con los reportados en [4] se presenta en la tabla 6, donde: **MaxEnt** y **NB** corresponden a los clasificadores de Máxima Entropía y el clasificador Naive Bayes, **Bigram-I** se refiere al modelo de lenguaje de bigramas con suavizado, **PCFG** es el método propuesto en [4] que utiliza gramática libre de contexto probabilística para modelar el estilo de cada autor, **PCFG-I** corresponde al modelo mencionado anteriormente con interpolación, **PCFG-E,** corresponde a la combinación de máxima entropía y PCFG y finalmente **MaxEnt + Bigram-I**, corresponde a la combinación del clasificador de máxima entropía y bigramas con interpolación. La última columna corresponde a la exactitud obtenida con los HRR.

Corpus Poetry

Tabla 6. Exactitud obtenida para el corpus Poetry

	Artículo							
	MaxEnt	NB	BIgram -I	PCFG	PCFG-I	PCFG-E	MaxEnt+ Brigram-I	HRR
Poetry	56.36	78.18	70.90	78.18	83.63	87.27	76.36	82.00

Se puede observar que el método de PFG no supera a los HRR. Quienes son superados únicamente por la combinación de más de uno de los métodos reportados en [4].

Corpus NFL

Tabla 7. Exactitud obtenida para el corpus NFL

	Artículo							
	MaxEnt	NB	BIgram -I	PCFG	PCFG-I	PCFG-E	MaxEnt+ Brigram-I	HRR
NFL	84.45	86.67	86.67	93.34	80.00	91.11	86.67	88.10

Para NFL, PCFG supera a los HRR, Tabla 7.

La Tabla 8 muestra los resultados paras Business, donde los HRR claramente superan al método de PCFG y aun a la combinación de éste con otros métodos.

Corpus Business

Tabla 8. Exactitud obtenida para el corpus Business

	Artículo							
	MaxEnt	NB	BIgram -I	PCFG	PCFG-I	PCFG-E	MaxEnt+ Brigram-I	HRR
Business	83.34	77.78	90.00	77.78	85.56	91.11	92.22	93.30

8. Conclusiones y trabajo a futuro

En base a los resultados obtenidos de los experimentos realizados con anterioridad podemos concluir lo siguiente:

Los enfoques basados en instancias combinando información léxica y sintáctica, con el uso de los HRR a través del operador de convolución circular, produce resultados poco favorables.

En corpus equilibrados mayores de 50 documentos los HRRs paracen producir resultados adecuados.

Los resultados generados por los HRRs son equiparables a los reportados en la bibliografía.

Como trabajo futuro se experimentará con corpus balanceados y de mayor tamaño para validar que el método probado se comporta de forma favorable. Así mismo se pretende representar como HRRs estructuras lingüísticas mayores que palabras aisladas.

Agradecimientos

Agradecemos a la Vicerrectoría de Investigación y Estudios de Posgrado por el soporte ofrecido para la realización de este trabajo a través del proyecto "Utilización de expresiones lingüísticas para el análisis de sentimientos".

Referencias

1. Stamatatos, E.: A survey of modern authorship attribution methods. Journal of the AmericanSociety for information Science and Technology 60(3), pp. 538–405 (2009)
2. Sindhu Raghavan Adriana Kovashka Raymond Mooney: Authorship Attribution Using Probabilistic Context-Free Grammars. In: Proceedings of the ACL 2010 Conference Short Papers, ACLShort'10. pp.1-3 (2010)
3. Adrian Pastor et al. A New Document Author Representation for Authorship Attribution. Pattern Recognition, Springer Berlin Heidelberg pp. 283-292 (2012).
4. Grigori Sidorov, Francisco Velasquez, Efstathios Stamatatos, Alexander Gelbukh, and Liliana Chanona-Hernández. Syntactic Dependency-Based N-grams as Classification Features. In: Mendoza, M.G. (ed.)MICAI 2012, Part II. LNCS (LNAI), vol. 7630, pp. 1–11. Springer, Heidelberg (2013)
5. Maya Carrillo. Representando Estructura y Significado en Procesamiento de Lenguaje Natural. , Tratamiento del Lenguaje y del Conocimiento, BUBOK PUBLISHING S.L.,(2013)
6. Tony Plate. Holographic Reduced Representations: Convolution Algebra for Compositional Distributed Representations. In John Mylopoulos and Ray Reiter, editors, Proceedings of the 12th International Joint Conference on Artificial Intelligence (IJCAI), pp30-35, Morgan Kaufmann, San Mateo, CA, 1991, 6 pages.
7. Maya Carrillo Ruiz et al. Exploring the Use of Random Indexing for Retrieving Information (2009)
8. Etiquetador Stanford. http://nlp.stanford.edu/software/pos-tagger-faq.shtml (2014)
9. Magnus Sahlgren. An Introduction to Random Indexing. In Methods and Applications of Semantic Indexing Workshop at the 7th International Conference on Terminology and Knowledge Engineering, TKE (2005).

Reviewing Committee

David Pinto
Ivo Humberto Pineda
Beatriz Beltran
Darnes Vilariño Ayala
Mireya Tovar
Manuel Martin
Josefina Guerrero Garcia
Juan Manuel González
Meliza Contreras González
Claudia Zepeda
Ivan Olmos
Mario Rossainz
María Josefa Somodevilla-Garcia

Impreso en los Talleres Gráficos
de la Dirección de Publicaciones
del Instituto Politécnico Nacional
Tresguerras 27, Centro Histórico, México, D.F.
Diciembre de 2014
Printing 500 / Edición 500 ejemplares