Advances in Artificial Intellingence

Research in Computing Science

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 20**, Mayo, 2006. Tiraje: 500 ejemplares. *Certificado de Reserva de Derechos al Uso Exclusivo del Título* No. 04-2004-062613250000-102, 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: *Juan Humberto Sossa Azuela,* RFC *SOAJ560723*

Volume 20

Volumen 20

Advances in Artificial Intelligence

Volume Editors:
Editores del Volumen

Alexander Gelbukh
Sulema Torres
Itzamá López

Instituto Politécnico Nacional
Centro de Investigación en Computación
México 2006

ISSN: 1665-9899

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

Printing: 500
Impresiones: 500

Printed in Mexico
Impreso en México

Preface

Modern Artificial Intelligence is a branch of computer science that relates computers with nature in two complementary ways. On the one hand, it aims to provide computers with the intellectual abilities of natural creatures—animals and humans—such as the ability to learn, think, and speak. On the other hand, it studies very powerful ways existing in the nature for solving highly complex search and optimization problems. In many cases the solution is a combination of the two: a naturally-inspired algorithm used to make the computer more human-like.

This volume contains 20 carefully selected internationally peer-reviewed and revised original research papers on both theoretical advances and practical applications of artificial intelligence techniques. The papers are structured into the following six sections:

– Neural Networks and Genetic Algorithms
– Natural Language Processing
– Human-Computer Intrefaces
– Information Retrieval
– Robotics
– Intelligent Tutoring Systems

The volume will be useful for researches and students working in the respective areas of artificial intelligence, as well as for all readers interested in artificial intelligence research and applications.

This volume is a result of work of many people. In the first place we thank the authors of the papers included in this volume, for it is the technical excellence of their papers that gives it value. We thank also the members of the International Editorial Board of the volume and the additional reviewers for their hard work on selecting the best papers out of many submissions we received. We would like to thank Edgar Gatalán Salgado and Alejandro Cuevas Urbina, as well as the personnel of the Center for Computing Research of the National Polytechnic Institute, in the first place Oralia del Carmen Pérez Orozco and Ignacio García Araoz, for their indispensable help in the preparation of the volume, with special thanks to Israel Román.

With this 20th volume and the previous one we commemorate the 10th anniversary of the Center for Computing Research of the National Polytechnic Institute, the 70th anniversary of the National Polytechnic Institute, and the 15th anniversary of CIC, the International Conference on Computing.

May, 2006

Alexander Gelbukh
Sulema Torres
Itzamá López

Table of Contents

Índice

Page/Pág.

Information Retrieval

Robotics

Intelligent Tutoring Systems

Neural Networks
and Genetic Algorithms

Feature Extraction and Automatic Recognition of Plant Leaf Using Artificial Neural Network

Qingfeng Wu, Changle Zhou and Chaonan Wang

Institute of Artificial Intelligence, Computer Science Department,
Xiamen University, 361005, Fujian, P.R. China
qfwu@xmu.edu.cn

Abstract. Plant recognition is an important and challenging task. Leaf recognition plays an important role in plant recognition and its key issue lies in whether selected features are stable and have good ability to discriminate different kinds of leaves. From the view of plant leaf morphology (such as shape, dent, margin, vein and so on), domain–related visual features of plant leaf are analyzed and extracted first. On such a basis, an approach for recognizing plant leaf using artificial neural network is brought forward. The prototype system has been implemented. Experiment results prove the effectiveness and superiority of this method.

1 Introduction

The recognition and identification of plant has great significance to explore genetic relationship of plant and explain the evolution law of plant system. However it is a very time consuming task, which is usually done by botanist currently. When recognizing and identifying plant, people usually observe leaf, flower, stem, fruit and so on to extract discriminating feature [1]. Because such discriminating features can be directly observed and obtained by people when they observing leaf images, people expect to fulfill the recognition and identification of plant automatically or semi-automatically by computers [2].

As an important organ of plants, recognition and identification of leaves is an important step for plant recognition and identification. In addition, leaves of plants are planar and easy to be input into the computer by scanner and digital camera. At present, most of related work focuses on the study of leaf images of plant. Im. [3] recognized maple leaves by the shape. Wang. [4] represented the shape of leaf with a centroid-contour distances curve. The problem of the above methods lies in the simplicity of the description of leaves feature. Namely these methods mostly focused on contour of leaf and neglected other features such as leaf dent, leaf vein and so on. Fu. [5] extracted leaf vein from leaf images by neural network, but did not further present the features of leaf vein. Zhang. [6] retrieved the standard tobacco leaf database by the features of color, texture and shape (mainly perimeter and area). The problem of this method is the lack of representation of domain features of leaves.

We believe that the key issue of leaf recognition, which is the same to that of plant recognition, is whether extracted features are stable and can distinguish individual

© A. Gelbukh, S. Torres, I. López (Eds.)
Advances in Artificial Intelligence
Research in Computing Science 20, 2006, pp. 3-10

leaves. Following this idea, features of shape, margin, dent and vein are extracted first in this paper to represent leaves; On such a basis, an approach for recognizing plant leaf using artificial neural network is brought forward. The prototype system has been implemented and the experiment result proves the effectiveness and superiority of this method.

The rest part of the paper is organized as follows. Visual features of leaf images are described in Section 2. An approach for recognizing plant leaf using artificial neural network is brought forward in Section 3. In Section 4, experimental results and discussions are presented. In Section 5, conclusions and further work are given.

2 Extraction of Leaf Image Features

Visual features of image can be classified as general visual features and domain-related visual features. General visual features, such as color, texture and shape are used to describe common features of all the images and have no relation with specific type and content of images. For leaf recognition, domain-related visual features of leaf image should be extracted. Combined with the morphology characteristic of leaves, several domain-related visual features are extracted with consideration of three aspects of leaf: shape, dent and vein.

2.1 Leaf Shape

Shape of objects is greatly helpful to object recognition and retrieval. In this paper, four visual features are defined to represent the shape of leaf.

Slimness (F1): this feature is defined as the ratio of length to width of leaves as Formula 1,

$$Shape\_slimness = \frac{l_1}{l_2}, \tag{1}$$

where, l_1 is the maximum length between the bonding point of leafstalk with leaf surface and all the points at the leaf margin. l_2 is the maximum length of line between the points on leaf margin which is vertical to l_1.

Roundness (F2): the feature is defined as Formula 2,

$$Shape\_roundness = \frac{4\pi A}{P^2}, \tag{2}$$

where, A is the area of leaf image; P is the perimeter of leaf contour. And roundness expresses the extent to which a shape is a circle [7]. A circle's roundness is 1 and a long bar's roundness is close to 0.

Solidity (F3): this feature is defined as Formula 3,

$$Shape\_solidity = \frac{S_1}{S_2}, \quad (3)$$

where, S_1 is the internal area connecting the valley points; S_2 is the external area connecting the top points(as shown in Fig. 2). Solidity expresses the degree of splitting depth in a leaf [8].

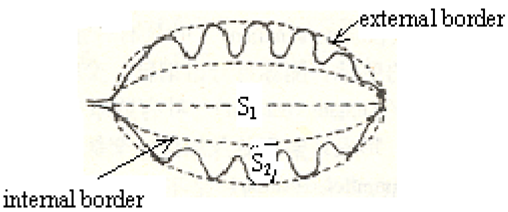


Fig. 1. Internal and External Shape

t_{i+1}^{6}	t_{i+1}^{0}	t_{i+1}^{1}
t_{i+1}^{5}	t_i	t_{i+1}^{2}
t_{i+1}^{4}	t_{i-1}	t_{i+1}^{3}

Fig. 2. Possible Directions of Point t_{i+1}

Moment invariants (F4~F10): slimness and roundness describe the shape feature of leaf to some extent. However, such description is rough. In order to describe the leaf shape in detail, this paper adopts moment invariants as shape describer. Please refer to ref. [9] for the detailed formula.

2.2 Leaf Dent

Leaf margin and leaf dent contain rich information, and they play an important role in leaf image recognition. In the following some visual features are extracted to represent leaf margin and leaf dent.

Coarseness (F11): this feature expresses the coarseness of the leaf margin and is defined as Formula 4,

$$Dent\_margin = \frac{P}{P'}, \quad (4)$$

where, P is the perimeter of leaf contour, and P' is the length of internal border.

Features of leaf dent: leaf dent is regarded as detailed patterns on contour shapes, and wavelet maximum (*i.e.* Wavelet Local Extrema) is used to represent the features of leaf dent (for example the size, sharpness, angle)[10]. The describer can be obtained by the following wavelet transformation as shown in Formula 5.

$$W_s\theta(t) = \theta(t) * \psi_s(t), \quad (5)$$

where, $s = 2^j$, j is the level corresponding to scale s, $j = 0,\ldots,n$. $\theta(t)$ is the tangential orientation change along the contour and t is the arc length from the starting point on the contour shape.

The extrema are extracted and each extremum describes a corner appearing at a certain scale. The following feature vectors are described.

Size (F12): this is the largest scale at which the WLE of a leaf dent appear.

$$Dent\_size = \log_2(s_t), \tag{6}$$

where, s_t is the top scale.

Angle (F13): this is measured by the extremum value at each scale.

$$Dent\_angle_{s_v} = e_{s_v}, \tag{7}$$

where, $Dent\_angle_{s_v}$ is the angle at scale s_v .

Sharpness (F14): This is measured by estimating the Lipschitz exponent of the WLE values.

$$\begin{aligned} &e_s \cong A * exp(-\alpha s) \\ &Dent\_sharpness = \alpha \end{aligned}, \tag{8}$$

where, α is the Lipschitz exponent and A is a constant.

2.3 Leaf Vein

As an inherent trait, the leaf vein definitely contains the important information despite of its complex modality. The venations of main vein and secondary vein are usually similar to the structure of whole plant. By analyzing the venations, more detailed characteristic of leaf, even that of the whole plant can be obtained [11]. In reference [5], modality of leaf venation can be extracted accurately. Based on this work, further features are extracted to present the leaf vein.

Ramification (F15): the number of ramification of the main vein can be used to measure the complexity of venation. By watering the main vein from the end point of leaf stem, the number of the ramification is the diffluent times of the water when it flows along the main vein. It is defined as the following Formula 9[12,13].

$$Vein\_ramification = \frac{fc_i}{l_i}, \tag{9}$$

where, l_i is the length of main vein, fc_i is the number of ramification.

Camber (F16): camber expresses the degree of crook of main vein. $T = \{t_0, t_1, \ldots, t_n\}$ represents a main vein, and t_{i-1}, t_i, t_{i+1} are the three continuity point. Supposing the positions of t_{i-1}, t_i are determined, t_{i+1} may has seven directions as shown in Fig. 3. When the direction of t_{i+1} is t_{i+1}^{0}, t_{i-1}, t_i, t_{i+1} reveal linear relation in this domain, that is, and the main vein has no turning at the point of t_{i+1}. Otherwise, vein has turning at the point of t_{i+1} [12,13].

Camber is defined as:

$$Vein\_camber = \frac{rn_T}{n+1}, \tag{10}$$

where rn_T is the number of turning of main vein.

3 Computational Model for Recognizing Plant Leaf

There are a great many recognition methods. In this paper, we use a feed-forward back-propagation neural network. The selected neural network has 3 layers, input layer with 16 nodes, hidden layer with 32 nodes and output layer with 6 nodes. The number of nodes of input layer is the same as the number of extracted features, and the number of nodes of output layer is the same as the number of plant categories to be recognized. Back propagation algorithm is used to train the neural network [14,15]. And minimize the error between real output and expect output by adjusting the weight of connections.

The trained NN are embodied in the plant leaf recognition system. Figure 3 shows the flow chart of our approach of recognizing plant leaves.

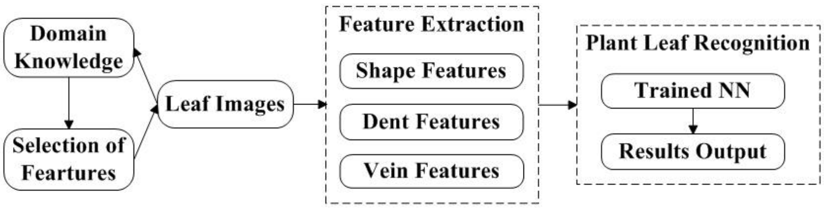


Fig. 3. Block Diagram of Plant Leaf Recognition Approach

4 Experiments and Discussions

4.1 Validity Evaluation of Visual Features

The effectiveness and stability of extracted visual features plays an important role in recognition system. So the first experiments were to examine the validity of leaf visual features discussed in this paper. Each time only one feature was used as the recognition feature. The recognition rate is shown in Fig. 4.

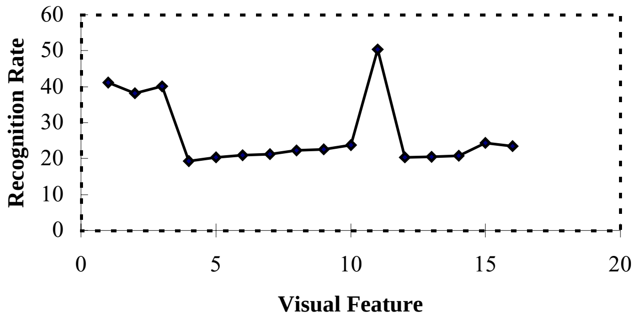


Fig. 4. Recognition Rate of Individual Feature

Based on the experimental result shown in Fig. 4, the following conclusions can be drawn: 1) Different visual feature has different ability of distinguishing leaves. Owing to it, these features can be classified into the global features and the local features. To improve the recognition performance, a hierarchical recognition method for leaf image is to be brought forward in the further work. 2) We should not use exclusive visual feature to represent image because exclusive feature doesn't have strong ability to recognize the species of plant leaf.

4.2 Performance of Leaf Image Recognition

Moreover, the recognition process was carried out with all the visual features of leaf images. We collected six kinds of plant leaf images, and there are thirty images in each category. These images are separated into the training set and the test set respectively with the ratio of 6:4. The experiment result is shown in Table 1.

Table 1. Recognition Performance of Visual Features with Different Size of Training Set

Plant Species	Recognition Accuracy Rate (%)	
	54 (50%)	108 (100%)
P1	91.0%	94.4%
P2	94.6%	95.2%
P3	89.7%	91.1%
P4	92.5%	96.8%
P5	95.2%	98.6%
P6	96.3%	97.9%

From Table 1, we can conclude that the recognition of leaves with all selected visual features is satisfying and the recognition performance is improved after trained with different size of training set. The experiment results validate the high ability of extracted features to distinguish different kinds of leaves.

4.3 Experiment of Leaf Recognition

The recognition system in C++ has been implemented on a PC (CPU: PIV 2.6GHz, RAM: 512M). Based on 1200 leaf images, the average time recognizing one image is about 0.45 seconds and the training time is about 12.3 seconds. Fig. 5 shows the first, second, third and the fourth most similar candidate to the leaf to be recognized.

Fig. 5. Result of Leaf Image Recognition

5 Conclusions

The key issue of leaf recognition lies in whether selected features are stable and have good ability to discriminate individual leaves. In this paper, from the view of plant morphology (such as shape, vein, dent and so on), domain-related visual features of plant leaf are analyzed and extracted. On such a basis, an approach for recognizing plant leaf using artificial neural network is brought forward. To prove the effectiveness of the methods proposed in this paper, a series of experiments are conducted. Experiment results prove the effectiveness and superiority of our methods.

Our future work will focus on: 1) the extraction of plant leaf from the image with background consisting of various objects; 2) the construction of hierarchical recognition model of leaf image.

Acknowledgment

This work was supported by National Science Foundation of P.R. China (60275023).

References

1. T.E. Weier, C. R. Stocking, *et al.* : Botany: An Introduction to Plant Biology. John Wiley and Sons(1982)
2. Hengnian Qi, Tao sh., Shuihu Jin: Leaf Characteristics-based Computer-aided Plant Identification Model. Journal of Zhejiang Forestry College. 20(3), 2003:281~284
3. Im. C., Nishida, H.,anh Kunii T.L.: Recognizing Plant Species by Leaf Shapes- a Case Study of the Acer Family. In Proceedings of IEEE International Con. On Pattern Recognition, 1998:1171~1173
4. Wang Z, Chi Z, Feng D, Wang Q: Leaf Image Retrieval with Shape Features. Lecture Notes in Computer Science 1929. In: Laurini R ed. Advances in Visual Information Systems. Berlin: Spring-Verlag, 477~487
5. Fu H, Chi Z, Chang J, Fu X: Extraction of Leaf Vein Features Based on Artificial Neural Network-Studies on the Living Plant Identification Ⅰ. Chinese Bulletin of Botany. 21(4), 2004:429~436
6. Zhang B, Zhang H: Content Based Image Retrieval of Standard Tobacco Leaf Database. Computer Engineering and Application.2002.07:203~205
7. Jain AK: Fundamentals of Digital Image Processing, Prentice-Hall, London ,UK,1989
8. Takeshi Saitoh, Toyohisa Kaneko: Automatic Recognition of Wild Flowers. In Proceedings of IEEE International Con. On Pattern Recognition 2000, Vol. 2:507~510
9. Ming-Kei Hu: Visual Pattern Recognition by Moment Invariants. IEEE Trans. on Information Theory. 8(2), 1962:179~187
10. Ershad Husseein, Yuichi Nakamura, Yuichi Ohat: Analysis of Detailed Patterns of Contour Shapes Using Wavelet Local Extrema. In Proceedings of International Con. On Pattern Recognition, 1996:335~339
11. Ash A W, Ellis B, Hickey L J, Johnson K R, Wilf P, Wing S L: Manual of Leaf Architecture: Morphological Description and Categorization of Dicotyledonous and Net-Veined Monocotyledonous Angiosperms. Washington D C: Smithsonian Institution, 1999: 26~27
12. X.S. Zhou, T.S. Huang: Edge-based Structural Features for Content-based Image Retrieval. Pattern Recognition Letters, 2001, 22:457~468
13. J.W Han, L. Guo.: A Shape-based Image Retrieval Method Using Salient edges, Signal Processing: Image Communication, 18,2003:141~156
14. B.D.Ripley: Pattern Recognition and Neural Networks. Cambridge University Press,1996
15. K. Mehrotra, C. K. Mohan, and S. Ranka: Elements of Artificial Neural Networks. A Bradford Book, The MIT Press, Cambridge, Massachusetts, London, England, 1997

Genetic Algorithms for the Numerical Solution of Variational Problems Without Analytic Trial Functions

Carlos Toledo

Center for Intelligent Systems, Monterrey Institute of Technology
Campus Monterrey, Monterrey, Nuevo León 64849, México
ctoledo@itesm.mx

Abstract. A coding of functions that allows a genetic algorithm to minimize functionals without analytic trial functions is presented and implemented for solving numerically some instances of variational problems from physics. ...

1 Introduction

The *genetic algorithm* (GA)[**?**,**?**] has become popular as as a reliable computerized method for solving problems from a wide range of domains, such as function optimization, handling them even in nonlinear, multidimensional search spaces. A conventional GA is a stochastic search method inspired in natural selection and the darwinian idea of the survival of the fittest in which the possible solutions of a problem (metaphorically the points in the search space) are coded as fixed-length strings of characters of an alphabet (usually binary) that resemble the chromosomes of alive beings. A GA evolves a population of search "points" chosen randomly appliying iteratively on them operators of *selection*, *crossover* and *mutation* for creating new populations (generations).

Selection consists in giving a proportionally bigger number of offspring to the fitter individuals so the characteristics that make them better prevail. The combination of this characteristics for generating new individuals is achieved through crossover, that is the interchange of portions of the strings of characters of two individuals paired randomly, giving birth to two new individuals for the next generation. In its simplest form in a GA all individuals are removed (die) after reproduction.

The last iterative step consists in making random changes to the strings of individuals chosen with a small probability, which is named mutation after the natural process that it resembles. After some generations the individuals tend to concentrate around the fittest "points" in the search space, so it can be said that all of the process was a way of optimizing the function employed to determine the fitting.

The predominant kind of optimization problems attacked with GAs to date have been those in which the strings of an alphabet that make the evolving population code literally points in a multidimensional space, where each dimension

© A. Gelbukh, S. Torres, I. López (Eds.)
Advances in Artificial Intelligence
Research in Computing Science 20, 2006, pp. 11-16

represents a parameter or variable of interest. When the potential solutions of a problem are *functions* and not points, as it is the case for variational problems [?], the most popular GA approach developed to date has been that of choosing a set of analytic trial functions and combining them in the fittest way. There are two main ways for doing so: weightening them, case where a string of weights is an individual, or using *genetic programming* (GP) [?] where the trial functions and the mathematical operators needed for combining them are the alphabet that gives shape to each member of the population.

In this paper a way to directly represent numerical functions as strings (individuals) of a GA is presented, followed by its successful implementation on some instances of variational problems from physics.

2 Angular Genes are not Real Genes

Lets take $\mathfrak{G}$ as the alphabet chosen to code individuals. In the literature of GAs (without taking GP into account) two main alphabets are usually discussed, the so called *binary genes* $\mathfrak{G} = 0, 1$ and the *real genes* $\mathfrak{G} \in \Re$. Even when there is not any special restriction on $\mathfrak{G}$ in the definition of the GA, only the implicit warning that it must facilitate the heredity of the fittest characteristics for obtaining acceptable results, it is curious how the attention in the field has been biased toward the binary and real alphabets. One of the goals of this paper is to emphazise the importance of focusing attention in other alphabets, in the extra information that is possible to get from them, specifically in an *angular* one, which will be called from now on *angular genes.* Aren't angular genes just real genes? The distinction made is based in the commonly forgotten fact that angles are not numbers [?], they are an entity by themselves.

The angular genes code piecewise functions as a string α of the angles between each consecutive pair of linear segments. For any combination of angles it is possible to scale and rotate the collection to fit the initial and final desired values. Taking $y_1 = y(x_1)$ and $y_N = y(x_N)$ as the initial and final values of the piecewise function $y_k = y(x_k)$ to be represented in the range (x_1, x_N) with $N-1$ linear segments and $k = 2, 3, \ldots, N$, the coding is defined as follows:

$$R = \sqrt{(y_N - y_1)^2 + (x_N - x_1)^2}$$

$$\beta = \tan^{-1}[(y_N - y_1)/(x_N - x_1)]$$

$$r^2 = \left[\sum_{i=2}^{N} s_{i-1} \cos\left(\sum_{j=1}^{i-1} \alpha_j\right)\right]^2 + \left[\sum_{i=2}^{N} s_{i-1} \sin\left(\sum_{j=1}^{i-1} \alpha_j\right)\right]^2$$

$$\gamma = \tan^{-1}\left[\sum_{i=2}^{N} s_{i-1} \cos\left(\sum_{j=1}^{i-1} \alpha_j\right) \Bigg/ \sum_{i=2}^{N} s_{i-1} \sin\left(\sum_{j=1}^{i-1} \alpha_j\right)\right]$$

$$x_k = x_1 + \sum_{i=2}^{k} \left({}^{R}\!/_{r}\right) s_{i-1} \cos\left(\sum_{j=1}^{i-1} \alpha_j - \gamma + \beta\right) \tag{1}$$

$$y_k = y_1 + \sum_{i=2}^{k} \left({}^{R}\!/_{r}\right) s_{i-1} \sin\left(\sum_{j=1}^{i-1} \alpha_j - \gamma + \beta\right) \tag{2}$$

Where $-\sigma \leq \alpha_j \leq \sigma$ and s is a string of real numbers $0 < s_i \leq 1$ that codes the relative length of each linear segment and together with α forms an individual. Having (1) and (2) we can further define:

$$\Delta x_i = \left({}^{R}\!/_{r}\right) s_i \cos\left(\sum_{j=1}^{i} \alpha_j - \gamma + \beta\right)$$

$$\Delta y_i = \left({}^{R}\!/_{r}\right) s_i \sin\left(\sum_{j=1}^{i} \alpha_j - \gamma + \beta\right)$$

$$y'_i = \tan\left(\sum_{j=1}^{i} \alpha_j - \gamma + \beta\right) \tag{3}$$

$$y''_i = \tan(\alpha_{i+1})(1 + y'_i y'_{i+1})/\Delta x_i \tag{4}$$

Taking $\Theta_i = \Theta(x_i)$ as the evaluation in x_i of the function that minimizes the functional, for the best found individual we have:

$$\left|\alpha_{i+1} - \tan^{-1}\left(\frac{\Theta'_{i+1} - \Theta'_i}{1 + \Theta'_i \Theta'_{i+1}}\right)\right| < \varepsilon \tag{5}$$

Equation (5) is a measure of the error of the best approximation found that clarifies the influence of a proper choice of σ according to the problem. If σ is too small the error can be surely surpassed but if it is too big the search space grows.

Ignoring the differences in a tenth of σ and 1 the search space explored has a size of approximately $10^{N-1} \times 20^{N-1}$.

3 Examples

The presented coding was used to solve instances of four well known variational problems from physics. In each case the population used had a size of 100 with $N = 101$, crossover based in interchanging angles with probability of 1 from randomly paired individuals after an angle chosen at chance (one point crossover), changing randomly (mutating) angles in the range $[-\sigma, \sigma]$ with probability of 0.05, and 500 generations. The number of runs made for all cases was ten. For the three first cases it was not needed the help of s, so $s_i = 1$, but not for the last where $0 < s_i \leq 1$. The algorithm was written in MATLAB and implemented in a personal computer with Pentium(R)4 CPU, 2.4GHz, 448 MB RAM.

3.1 Curve of Shortest Distance in the Euclidian Plane

The functional to minimize is

$$J = \int_{x_1,y_1}^{x_N,y_N} \sqrt{(dx)^2 + (dy)^2}$$

whose known solution [?] is the straight line $y = ax + b$. For the case $x_1 = y_1 = y_N = 0$, $x_N = 1$ with minimum $J = 1$, the average solution found by the algorithm with $\sigma = 0.005\pi$ was $J = 1 + 1.66 \times 10^-5$ with standard deviation of 1.2×10^-6.

3.2 Curve of Minimum Revolution Area

Considering two paralel coaxial wire circles to be connected by a surface of minimum area that is generated by revolving a curve $y(x)$ about the x-axis, the functional to minimize is

$$J = \int_{x_1,y_1}^{x_N,y_N} 2\pi y \sqrt{(dx)^2 + (dy)^2}$$

whose known solution [?] is the catenoid $y = \cosh(ax + b)/a$. For the case $-x_1 = x_N = 0.5$, $y_1 = y_N = 1$ with minimum $J = 5.9917$, the average solution found by the algorithm with $\sigma = 0.005\pi$ was $J = 5.9919$ with standard deviation of 1.4×10^-5.

3.3 Fermat's Principle

According to Fermat's principle light will follow the path $y(x)$ for which

$$J = \int_{x_1,y_1}^{x_N,y_N} n(x,y) \sqrt{(dx)^2 + (dy)^2}$$

is minimum when n is the index of refraction. When $n = e^y$ the solution is $y = \ln(a/\cos(x + b))$. For the case $-x_1 = x_N = 1$, $y_1 = y_N = 1$ with minimum $J = 4.5749$, the average solution found by the algorithm with $\sigma = 0.01\pi$ was $J = 4.5752$ with standard deviation of 5.2×10^-5.

3.4 The Energies of the Hydrogen Atom

The hydrogen atom [?] is the quantum system made of a proton and an electron whose energies, without taking into account the degeneracies, can be found minimizing the functional

$$E_n = \frac{1}{c} \int_0^\infty \left[\frac{\hbar^2}{2\mu} (u_n'^2 + \left(\frac{n(n-1)\hbar^2}{2\mu r^2} - \frac{q^2}{4\pi\varepsilon_0 r} \right) u_n^2 \right] dr$$

with $c = \int_0^\infty u_n^2 dr$, $u_n(r) = rR_n$, $u_n(0) = 0$, R_n^2 is the probability distribution for the radial location of the electron, q is its charge, $\mu = m_e m_p/(m_e + m_p)$ the reduced mass of the system, ε_0 the permitivity of free space and $\hbar$ the Planck's constant divided by 2π. The energies of the system are ruled by the equation $E_n = -13.6052\,\mathrm{eV}/n^2$.

The algorithm was used to find the three first energies of the system. In this case the result found by each run depends strongly in a right choice of r_N such that $u_n(r_N)^2 \approx 0$. For the ground state $E_1 = -13.6052\,\mathrm{eV}$ the best found was about eight times the Bohr radius, the average solution found by the algorithm was $E_1 = -13.5987\,\mathrm{eV}$ with a standard deviation of $0.028\,\mathrm{eV}$. For $n = 2$, $E_2 = -3.4014\,\mathrm{eV}$, the best choice for r_N made was about fifteen times the Bohr radius and the average solution found by the algorithm was $E_2 = -3.42467\,\mathrm{eV}$ with standard deviation of $0.005\,\mathrm{eV}$. For $n = 3$, $E_3 = -1.5117\,\mathrm{eV}$, the best choice for r_N made was about twenty five times the Bohr radius, reaching an average solution of $E_3 = -1.5103\,\mathrm{eV}$ with standard deviation of $0.001\,\mathrm{eV}$. In the three cases $\sigma = 0.005\pi$.

4 Conclusions and Future Work

The examples shown were chosen with demostrative purposes. Better aproximations for specific cases can be reached increasing N and improving the choice of σ, with the extra computational effort it implies. Even thought that it was showed the efficiency of the coding to minimize the functionals presented it will be necessary the development of a theory of difficulty to give a more concise explanation of the kind of problems that could be hard to solve using it, like those already existent for binary genes like *deception* [**?**,**?**] and *NK landscapes* [**?**]. Another useful future development will be that of general ways of handling problems with constraints. An important potential application of the kind of genetic algorithm presented would be in those cases where there are not analytic solutions available, like in many quantum systems.

5 Acknowledgments

The author was supported by a research fellowship from Monterrey Institute of Technology and the Mexican Council of Science and Technology (CONACYT) while writing this paper.

References

1. Goldberg, D. E. *Genetic Algorithms in Search, Optimization and Machine Learning*, Addison Wesley, Reading, Massachusetts (1989).
2. Goldberg, D. E. *The Design of Innovation. Lessons from and for Competent Genetic Algorithms*, Kluwer Academic Publishers, Boston, Massachusetts (2002).
3. Arfken, G. B. and Weber, H. J. *Mathematical Methods for Physicists*, Hartcourt Academic Press, Burlington, Massachusetts (2001).

4. Koza, J. R. *Genetic Programming: On the Programming of Computers by the Means of Natural Selection*, MIT Press, Cambridge, Massachusetts (1993).
5. Lakoff, G. and Núñez, R. E. *Where Mathematics Comes From: How the Embodied Mind Brings Mathematics into Being*, Basic Books, New York, New York (2000).
6. Cohen-Tannoudji, C. *Quantum Mechanics*, Wiley, New York, New York (1977).
7. Kauffman, S. A. *The Origins of Order. Self-Organization and Selection in Evolution*, Oxford University Press, New York, New York (1993).

Hybrid Evolutionary Methods for the Solution of Complex Scheduling Problems

José Antonio Vázquez Rodríguez* and Abdellah Salhi

Mathematical Sciences Department, University of Essex
Wivenhoe Park, Colchester, U. K.
javazq@gmail.com, as@essex.ac.uk

Abstract. This paper is concerned with the minimisation of makespan and maximum lateness when scheduling Flexible Flow Shops (FFS). Even though, hybrid evolutionary methods have provided with competent solution tools for several hard combinatorial problems, their efficiency for FFS is not known. In this paper, the idea of hybridising Genetic Algorithm (GA) with the Shifting Bottleneck Procedure (SBP) and GA with a Local Search (LS) procedure, is explored. The proposed algorithms, named the Shifting Bottleneck Genetic Algorithm (SBGA) and Memetic Algorithm (MA); and the simple algorithms, GA and SBP, were compared on solving well known benchmarks and two large sets of randomly generated instances. The results revel that both hybrid methods are very successful.

1 Introduction

This paper is concerned with the minimisation of makespan and maximum lateness when scheduling shops with multiple stages and multiple machines per stage. We focus on the particular case in which all the jobs follow the same processing direction. This manufacturing environment is known as Flexible Flow Shop (FFS), [3]. Following the notation in [13], the problems of interest are denoted as: $FFc||C_{\max}$ and $FFc|r_j|L_{\max}$ for makespan and maximum lateness minimisation, respectively.

The FFS scheduling problem is intractable, even the two stages shop version is NP-Hard, [6]. It is still NP-hard if preemptions are allowed, [7]. We refer the reader to [16], [8], [10] and [15] for surveys on approaches to it, including exact methods, deterministic algorithms, heuristics and metaheuristics. The focus here is on metaheuristics and bottleneck exploiting heuristics.

The processing stage with the lowest capacity/work-load ratio, named bottleneck or critical stage, constraints the global performance of the system. Efficient heuristics such as the Shifting Bottleneck Procedure (SBP) exploit this knowledge by maximising the bottleneck utilisation. Examples of successful SBP adaptations to the FFS are described in [19] and [4]. Metaheuristics, on the other

* Corresponding author

© A. Gelbukh, S. Torres, I. López (Eds.)
Advances in Artificial Intelligence
Research in Computing Science 20, 2006, pp. 17-28

hand, are the basis of several state of the art methods for a wide range of combinatorial problems, these include the FFS. For instance in [9], [2] and [12], Genetic Algorithms (GA) were designed to efficiently solve different variants of the problem. In [17] and [11], Tabu Search (TS) was utilised. The combination of heuristics, specially a global search method, such as an evolutionary algorithm, with a problem tailored heuristic, is a good strategy to generate even stronger solution approaches. Memetic Algorithms (MA), which combine GA with an improving procedure, are a good example of them. However, our literature review revels that this idea has not been studied in the case of the FFS problem.

This paper introduces two new algorithms based in the idea of hybridising GA with two specialised methods. The first algorithm uses the SBP to support the GA and was named the Shifting Bottleneck Genetic Algorithm (SBGA). The second algorithm uses a local search method to improve the solution provided by GA. We refer to this method as Memetic Algorithm (MA).

In order to test the "synergy" produced by the combinations of methods in both hybrids, SBGA, MA, GA and SBP were used to solve well known instances of the problem, as well as a large set of randomly generated problems. The reported results show that both of the new methods are effective solution tools for the problem.

The rest of the paper is organised as follows. Section 2 is the formal description of the problem; assumptions notation and a model are presented. Section 3 describes the SBGA. Section 4 presents GA and the MA. Section 5 describes the numerical experiments, including the problems and their origins, the results and a discussion. Section 6 concludes the paper.

2 Problem Description

It is assumed that n jobs are to be processed through m different stages. Each stage has at least one machine, with one or more having at least two parallel identical machines. Any machine can process at most one job at a time and any job is processed on at most one machine at a time. Furthermore, every job is processed on at most one machine in any stage. Preemptions are not allowed, i.e. once the processing of a job has started on a given machine, it can not be stopped until it is finished.

2.1 Notation

- j = job index; k = stage index; l = machine index;
- n = number of jobs; m = number of stages; m_k = number of machines at stage k;
- o_{jk} = operation of job j to be processed on k; $O_k = \cup_j o_{jk}$;
- p_{jk} = processing time of o_{jk}; r_{jk} = release time of o_{jk}; d_{jk} = due date of o_{jk};
- C_j = completion time of job j; $L_j = \max\{C_j - d_{jm}, 0\}$, lateness of job j.

2.2 Problem Formulation

Let A^{kl} be a set of operations $o_{jk} \in O_k$ assigned for processing to machine l at stage k. Let S^{kl} be a permutation of the elements in A^{kl} representing the order in which operations must be processed. Let $S^k = \cup_{l=1}^{m_k} S^{kl}$ and $S = \cup_{k=1}^{m_k} S^k$. Given that the functions approached are regular, our interest is limited to non delay schedules. Because of this, S, being the set of sequences of jobs in all machines, represent a unique schedule. For S, to be feasible, the following must hold: (1) $\cup_{l=1}^{m_k} A^{kl} = O_k \ \forall k$ and (2) $\cap_{l=1}^{m_k} A^{kl} = \emptyset \ \forall k$. These constraints guarantee that all operations to be processed in k are assigned for processing strictly once. Let ψ be a FFS problem instance and Ω^ψ the set of all feasible schedules for ψ. The problem is to find a $S \in \Omega^\psi$ such that its makespan $C_{\max}$,

$$\min_{S \in \Omega^\psi} \max_j C_j(S) \tag{1}$$

or its maximum lateness $L_{\max}$,

$$\min_{S \in \Omega^\psi} \max_j L_j(S) \tag{2}$$

is minimum.

3 The Shifting Bottleneck Genetic Algorithm

The SBP was introduced in [1], and so far, is one of the best established algorithms to approach multiple-stage shops. SBP breaks the overall problem into sub-problems and solves one at a time. In [4], an efficient adaptation of the original SBP is presented (for the rest of the paper SBP refers to this version). SBP uses the reversibility principle of the parallel machines problem (see Section 3.2) to obtain good solutions for it efficiently. The proposed approach, uses GA to prioritise the jobs in the critical stage of the shop. The rest of the stages are scheduled in the order dictated by their criticality with the heuristic employed by SBP.

3.1 Critical Stage

As mentioned, SBGA schedules the stages according to their criticality. The criticality of stage k is the minimum time required so that all the jobs with an operation in k pass through the shop. The one that requires the highest time, restricts the total capacity of the shop and is the critical stage or bottleneck. The criticality of stage k, CS_k, is calculated as follows,

$$CS_k = \frac{1}{m_k} \left(\sum_{y=1}^{m_k} LSA_y^{k-1} + \sum_{j=1}^{n} p_{jk} + \sum_{y=1}^{m_k} RSA_y^k \right) \tag{3}$$

where $LSA_y^k = \{LSA_1^k, LSA_2^k, ..., LSA_n^k\}$ are the sums $\sum_{b=1}^{k} p_{jb}$, $\forall j$ in the ascending order and $RSA_y^k = \{RSA_1^k, RSA_2^k, ..., RSA_n^k\}$ the sums $\sum_{b=k+1}^{m} p_{jb}$, $\forall j$

in the ascending order [4]. In the brackets, the first term is the minimum time elapsed before k becomes fully active, the second is the work load of k (which divided by m_k provides the minimum processing time in k) finally, the third term is the minimum time required by the last jobs processed in k to exit the shop. The critical stage k is the one with the largest CS_k value.

3.2 Heuristic Approach for the Parallel Machines Scheduling Problem

SBP and SBGA decompose the original problem into stages and schedule one at a time fixing the rest. A sub-problem for stage k, can be handled as a parallel machines problem with release dates r_{jk} and due dates d_{jk} ($Pm|r_{jk}, d_{jk}|L_{\max}$, see notation in [13]) and a solution to it can be obtained as follows.

Algorithm $procedureA()$

1. while $O_k \neq \emptyset$ **do**
   - **a.** set t as the maximum between the time that the first machine in k becomes idle and the minimum release time of the operations in O_k;
   - **b.** select the operation $o'_{jk} \in O_k : r_{jk} \leq t$ with the smallest due date (d_{jk}), break ties by preferring longer p_{jk};
   - **c.** assign o'_{jk} to the first idle machine;
   - **d.** $O_k = O_k \backslash o'_{jk}$.

Given a $Pm|r_{jk}, d_{jk}|L_{\max}$ problem, its inverse $P'm|r'_{jk}, d'_{jk}|L_{\max}$, is obtained using the negatives of the due dates and release dates of Pm as release and due dates, respectively, in $P'm$, i.e. $r'_{jk} = -d_{jk}$ and $d'_{jk} = -r_{jk}$. Let $\pi = (\pi(1), \pi(2), ..., \pi(|O_k|))$ be a sequence of tasks in a given machine, then $\pi' = (\pi(|O_k|), \pi(|O_k|-1), ..., \pi(1))$ is the reverse of π. The algorithm presented in [4], for the $Pm|r_j, d_j|L_{\max}$ problem is as follows.

Algorithm $Pm()$

1. obtain $P'm$;
2. solve Pm and $P'm$ using $procedureA()$, reverse the solution obtained for $P'm$, calculate $L_{\max}$ for both solutions.
3. Return the schedule with the minimum maximum lateness.

Note that $procedureA$ is not an exact algorithm, therefore, the solutions provided by it, when applied to the original problem and its inverse, may be different. We are interested in the best of them.

3.3 Representation and Evaluation of Solutions

A classical permutation representation, as explained in [5], was adopted. In this, every individual is a permutation $\pi = (\pi(1), \pi(2), ..., \pi(|O_k|))$ where $\pi(i)$ is a job index. π represents the sequence to prioritise the operations at the critical stage k'. In k', operations are assigned in the order dictated by π to the machine that allows them the fastest completion time. Let K be the set of stage indexes (k) sorted in decreasing order of their CS_k values (see Section 3.1), in such a way that K_1 points to the bottleneck of the shop. The procedure to evaluate an individual is as follows.

Algorithm $CP(\pi, K)$

1. set $S = \emptyset$ (an initial empty schedule).
2. generate S^{K_1} by assigning each o_{jK_1} in the order dictated by π to the machine l in stage K_1 that allows it the fastest completion time, $S = S \cup S^{K_1}$;
3. update release times and due dates as described in Section 3.4;
4. **for** $i = 2 : m$ **do**
  a. generate S^{K_i} using $Pm()$, $S = S \cup S^{K_i}$;
  b. update release times and tails as described in Section 3.4;
5. **return** $\max_j C_j(S)$ and S.

Note that at steps 2 and 4a, S is updated with the information of sub-schedule S^{K_i} obtained for stage K_i.

3.4 Updating Release Times and Due Dates

At the initialisation, the release times of operations to be processed at stages posterior to the first one are calculated as follows $r_{jk} = r_{j1} + \sum_{b=1}^{k-1} p_{jb}$, $k > 1$; and the due dates of previous stages to the last one, as follows $d_{jk} = -d_{j1} + \sum_{b=m}^{k+1} p_{jb}$, $k < m$. The release times at the first stage and due dates at the last one are given as part of the problem, otherwise they are set to 0.

Let us suppose that stage $\hat{k}$ is the one just scheduled at step 2 or 4a of the $CP()$ procedure. The release times must be updated for the operations in every stage $k \geq \hat{k}$ and the due dates for the operations in every stage $k \leq \hat{k}$. Remember that S^{kl} is the sequence of jobs assigned to machine l in k (see Section 2.2), let $q = \left|S^{kl}\right|$. The release times are updated as follows.

Algorithm $URT()$
for $k = \hat{k} : m$ **do**
  a. **if** $S^k = \emptyset$ (if stage k has not been scheduled) **do**
    i. $r_{jk} = r_{jk-1} + p_{jk-1}, \forall j;$
    ii. **end**
  b. **for** $l = 1 : m_k$ **do**
    i. $r_{S_1^{kl}k-1} = r_{S_1^{kl}k-1} + p_{S_1^{kl}k-1};$
    ii. **for** $h = 2 : q$ **do**
      A. $r_{S_h^{kl}k} = \max\left\{r_{S_h^{kl}k-1} + p_{S_h^{kl}k-1}, r_{S_{h-1}^{kl}k} + p_{S_{h-1}^{kl}k}\right\}.$

The due dates are updated with the following procedure.

Algorithm $UDD()$
for $k = \hat{k} : 1$ **do**

a. if $S^k = \emptyset$ (if stage k has not been scheduled) **do**
  i. $d_{jk} = d_{jk+1} + p_{jk+1}, \forall j;$
  ii. end
b. for $l = 1 : m_k$ **do**
  i. $d_{S_q^{kl}k} = d_{S_q^{kl}k+1} + p_{S_q^{kl}k+1};$
  ii. for $h = q - 1 : 1 : -1$ **do**
    A. $d_{S_h^{kl}k} = \max \left\{ d_{S_h^{kl}k+1} + p_{S_h^{kl}k+1}, d_{S_{h+1}^{kl}} + p_{S_{h+1}^{kl}k} \right\}.$

3.5 Genetic Operators

SBGA maintains a population *Pop* of N individuals. At every generation, a new population *Pop*′ is generated sampling in the search space by means of the genetic operators and using the information of the best individuals in *Pop*. Each new individual is created using one of the GA operators: crossover, mutation or direct reproduction. The order crossover (OX) was chosen as recombination method. It showed a better performance in a pre-experimental stage when compared with other crossover methods such as edge, cycle and partially mapped crossover (see [5]). In OX, two individuals (parents) and two crossing points are randomly chosen. The elements of the first parent that are in between the two crossing points are copied in the same positions to the new one. The rest are copied in the same order than in the second parent following a toroidal path (see [5] for more details). The mutation employed is the random generation of a new individual. Let *rep* be N minus the number of individuals not created through crossover or mutation. In order to provide elitism, the *rep* best fitted individuals in *Pop* are copied to complete *Pop*′.

The selection method of individuals, to participate for crossover or mutation, is binary tournament selection. In this, 2 individuals are selected randomly from the population, they compete among them, and the fittest participates in the crossover process, [5].

3.6 SBGA General Framework

At the initialisation N individuals are generated randomly and evaluated using the $CP()$ procedure. The criticality CS_k of every stage k is calculated using formula 3 and the ordered set K' is generated (see Section 3.3). The release times r_{jk} and due dates d_{jk} are calculated as described in Section 3.4.

The general framework of SBGA is as follows.

Algorithm $SBGA()$

1. calculate the ordered set K' (see Section3.3), initialise release times and due dates (see Section 3.4).
2. generate a set *Pop* of N random permutations, $CP_{i=1}^{N}(Pop_i, K')$, i.e. evaluate each solution using the procedure described in Section 3.3, let π^* be the best solution in *Pop*;
3. generate a set *Pop*′ of N new solutions by applying the GA operators (Section 3.5), $CP_{i=1}^{N}(Pop'_i, K')$;

4. let Pop be the best N solutions in $Pop' \bigcup \pi^*$, let π^* be the best solution found so far;
5. if stopping condition not met **go to** 3;
6. return π^*.

4 Alternative Methods

Given that SBGA is a hybrid algorithm, it is important to know if its SBP component provides any added value to the final algorithm. In order to explore this, a GA named Multiple Stages Representation GA (MSRGA), whose individuals represent full schedules, was encoded. MSRGA uses the same genetic operators as SBGA, but it does not use any specialised information. Since SBP has been proved to be an effective scheduling tool, it is expected that it will provide good information to SBGA. However, to evaluate how good this information is, it will be compared with the one provided by a Local Search (LS) method based on an efficient, already tested, neighbourhood function, [11]. A third algorithm, which uses LS to improve the performance of MSRGA, was designed. Given that these sort of hybrid methods are usually referred to as Memetic Algorithms, this third algorithm was named MA. Finally, in order to know to what extent the addition of a stochastic method to SBP, is beneficial, SBP as described in [4], was implemented and tested.

The rest of this section describes the implementation details of MSRGA and MA. We refer the reader to [4], for details on SBP.

4.1 Multiple Stages Representation GA

Representation and Evaluation of Individuals Every individual in MSRGA is a set $\Pi = \{\pi_1, \pi_2, ..., \pi_m\}$ of permutations, one for every stage of the shop, representing the assigning order of jobs at every stage. To evaluate an individual, operations at stage k are scheduled in the order dictated by π_k in the machine l in k that allows them the fastest completion time. This procedure is as follows.

Algorithm $CP_{MSRGA}(\Pi)$
1. set $S = \emptyset$ (an initial empty schedule);
2. for $k = 1 : m$ **do**
- **a.** generate S^k by assigning the operations in the order dictated by π_i to the machine that allows them the fastest completion time;
- **b.** $S = S \cup S^k$, update release times (see Section 3.4);

3. return $\max_j C_j(S)$.

Note that stages are scheduled in the order $1, 2, ..., m$, because of this, the release times need to be updated just for the stage being scheduled (k) and, if $k < m$, for stage $k + 1$ too. The due dates are not updated.

GA General Framework At initialisation, N individuals are generated randomly and evaluated using the $CP_{MSRGA}(\Pi)$ procedure. At every iteration, a new population Pop' is generated using the genetic operators as described in Section 3.5. But in this case, let A and B be the set of m permutations of parents A and B chosen to be recombined through crossover. Every permutation c_k of the new individual C, is the result of applying OX to the permutation $a_k \in A$ and $b_k \in B$.

The general framework of MSRGA is as described in Section 3.6 for the SBGA, but at step 1, there is no need to calculate K' neither the due dates. In steps 2 and 3 $CP(\pi, K')$ is substituted by $CP_{MSRGA}(\Pi)$.

4.2 Memetic Algorithm

In MA, the representation of individuals and the genetic operators are as described for MSRGA. However, in MA the procedure to evaluate individuals requires the use of a Local Search (LS) procedure. We refer the reader to [11] for a description and theoretical basis of the neighbourhood function on which LS is based.

To evaluate an individual, this is decoded by means of $CP_{MSRGA}()$, LS is then used to improve it. In algorithmic form, the evaluation of individuals in MA is as follows.

Algorithm $CP_{MA}(\Pi)$
1. $S = CP_{MSRGA}(\Pi)$, i.e. generate initial schedule;
2. $S' = LS(S)$, i.e. improve S using LS;
3. return $\max_j C_j(S')$.

The general framework of MA is as the one described for MSRGA, but substituting $CP_{MSRGA}(\Pi)$ with $CP_{MA}(\Pi)$.

5 Computational Experience

In order to evaluate the described methods, they were run on three different problem sets. The first one is, perhaps, the best known test-set for the FFS problem, [18]. The second and third are random instances generated in a similar fashion as in [19], [4] and [9].

Being SBP a deterministic heuristic, it does not require input parameters and it was run once on every instance. For the rest of the algorithms, the following parameter settings showed being appropriate.

- population size: SBGA 100, MSRGA 100, MA 30
- crossover rate: 95% for the three algorithms
- mutation rate: 1% for the three algorithms
- stopping condition: MSRGA was run on every problem until it did not show improvement for 100 consecutive generations or it reached a maximum processing time of 60 seconds. This time was recorded and set as the stopping condition for MA and SBGA.

Table 1. performance on IBM Wittrock's instances

Instance	SBP	MSRGA	MA	SBGA
1	761	820	**760**	**760**
2	769	793	763	**755**
3	761	784	767	**759**
4	781	787	772	**761**
5	**961**	**961**	**961**	**961**
6	667	674	665	**659**

SBGA, MSRGA and MA were run 5 times on every instance, the best found solution was reported. The four algorithms were encoded in Java S.E. 5.0. All the experiments were executed on identical PC's (Pentium IV, 3.0GH, 1Gb RAM) running Windows XP.

5.1 Results on IBM Wittrock's Instances

Here we report the performance of the 4 algorithms on Wittrock's test-bed, [18], from an IBM production line. This line, inserts components into printed circuit cards. Every card is transported in a magazine that holds 100 identical cards. The magazines have to go through three different types of machines: two "DIP inserters", three "SIP inserters" and three robots. The required time for each magazine in every stage depends on the type of card that it holds. The problem instances are the production needs for 6 days, for which, a daily schedule with minimum makespan, is required. Any transportation times between stages were neglected. Table 1 presents the results.

In 5 out of 6 instances SBGA outperformed SBP and GA and it was better than MA in 4 out of 6. MA, on the other hand, was superior to MSRGA in 5 out of 6 instances and to SBP in 4 out of 6. These results suggest that SBP and LS provide useful information to GA, being the one by SBP superior. However, conclusions can not be obtained from such a small sample. Next, the results in two large sets of randomly generated instances are reported.

5.2 Randomly Generated Instances

A set of 1080 instances for makespan and 1296 for maximum lateness were generated in a similar fashion as in [9] and [4]. The makespan and maximum lateness obtained by the algorithms, on each of these, were compared with the lower bound (LB) described in [14]. Since our problems are minimisation ones, when a solution reaches a makespan or lateness value equal to the one provided by LB, an optimal solution has been found. Table 2 displays the success rate of each algorithm on both objectives.

The most successful method is SBGA, followed by MA, SBP and finally MSRGA. Enough evidence has been collected to conclude that a generic method such as GA is not competitive. Moreover it is evident that SBP and GA work quite well together, the success rate of SBGA is around twice the one of SBP

Table 2. success rate on randomly generated instances

Opt. criterion	MSRGA	MA	SBP	SBGA
makespan	11.42%	32.72%	24.26%	**54.91%**
lateness	9.26%	46.38%	38.04%	**60.57%**

and around 5 times higher than that of MSRGA. MA, on the other hand, is also more competitive than MSRGA and SBP, but not as SBGA. Next, the mean deviations from the lower bounds obtained by the algorithms are presented. The results are reported for subsets of instances dictated by their characteristics, this will provide an idea of the effects of the instance characteristics on the algorithms performance.

The deviation on $C_{\max}$, $DC_{ai}\max$, of algorithm $a \in \{MSRGA, SBGA, ...\}$ on instance i with respect to the lower bound value LB_i, is calculated as:

$$DC_{ai}\max = \frac{C\max_{ai} - LB_i}{LB_i} * 100. \quad (4)$$

Given a set I of problem instances of interest, the deviation with respect to the lower bound for $L_{\max}$ is calculated as:

$$DL_{aI}\max = \frac{\sum_{i \in I}(L\max_{ai} - LB_i)}{\sum_{i \in I} LB_i} * 100 \quad (5)$$

where $DL_{aI}\max$ is the maximum lateness deviations from LB, obtained by algorithm a on a set of instances I.

Table 3 presents the mean $DC_{ai}\max$ and $DL_{aI}\max$ values of each algorithm on the sets of instances described on the first two columns.

The best performing algorithm, on the whole and on the different instance subsets presented in Table 3, is SBGA. It can be concluded from this, that GA and SBP collaborate positively. On the other hand, MA also reported better results than SBP and GA which shows the benefits of adding LS to MSRGA. The results on Table 3, let no place for discussion the need of using specialised information to approach FFS problems. However, it was necessary its testing to safely conclude that SBP and LS enhance the performance of GA. Moreover, the quality of information provided by SBP is comparable and superior, at least on the testbed presented, to that of a competitive local search method.

Regarding the computational times, SBP is far faster than any of the three other methods. Its required time never exceeded 5 seconds even for the largest instances. On the other hand the rest of the methods run for 1 minute in most of the instances with 6 and 7 stages and 100 jobs. For the rest of the problems this time limit was rarely reached. On the practice 1 minute of computational cost, to schedule a 700 operations shop, is reasonable. For an acceptable extra cost, GA can enhance the performance of SBP from an average deviation of 5.86% to 1.38% and from 3.15% to 0.83% for makespan and maximum lateness, respectively. This could represent important savings on real world shops.

Table 3. deviation from lower bound on randomly generated instances

parameter	value	$\overline{DC}_{ai}$ max				$\overline{DL}_{aI}$ max			
		MSRGA	MA	SBP	SBGA	MSRGA	MA	SBP	SBGA
n									
	10	14.36	4.97	8.99	2.68	112.16	9.56	18.32	5.72
	20	22.38	3.26	7.15	1.58	98.56	4.23	6.05	1.10
	50	88.46	1.65	4.06	0.73	79.87	1.32	1.99	0.39
	100	127.88	1.84	3.21	0.54	65.23	0.79	1.29	0.35
m									
	$[2,3]$	36.45	2.35	4.79	1.29	39.65	0.79	1.35	0.50
	$[4,5]$	61.35	2.59	5.75	1.23	48.11	1.12	1.72	0.37
	$[6,7]$	92.01	3.85	7.03	1.63	59.36	1.37	2.11	0.40
m_k									
	$[1,2]$	70.36	2.87	6.83	0.84	38.37	0.68	2.75	0.31
	$[1,4]$	54.68	2.66	5.42	1.14	59.84	1.09	1.99	0.39
	$[1,8]$	64.77	3.26	5.31	2.17	77.59	0.98	1.90	1.18
p_{jk}									
	$[10,20]$	40.02	2.37	3.87	1.13	65.23	0.69	1.13	0.40
	$[10,100]$	86.52	3.49	7.83	1.64	89.26	0.78	2.60	0.59
total									
		63.27	2.93	5.86	1.38	78.34	2.67	3.15	0.83

6 Conclusion

A new hybrid GA and SBP for the solution of FFS scheduling problems has been presented. This algorithm was compared versus a Multiple Stage Representation GA, a Memetic Algorithm and the Shifting Bottleneck Procedure on the solution of well known benchmarks and two large randomly generated test sets.

The proposed algorithm outperformed its competitors in all tests. It obtained deviations of 1.38% and 0.83% from lower bounds for makespan and lateness minimisation respectively and success rates of 54.9% and 60.57% for the same problems. Given that SBGA outperformed both, SBP and GA, it is reasonable to conclude that there is an important collaboration between the GA component and the SBP one. In other words, the relation GA and SBP, as in SBGA, seems to be highly "synergic". On the other hand, it is also clear that the information provided by SBP to GA is of a higher quality than that of a well established local search method.

To implement SBGA does not represent more challenge than SBP, moreover, the former improved the results of the latter by a 4% and 2%, on average, for makespan and maximum lateness minimisation, respectively. Because of this, and the fact that the CPU cost required by SBGA is reasonable, SBGA is a sensible choice in the practice.

Despite the extent of this investigation, it remains to test SBGA on problems with other optimisation criteria; and investigate its performance on other problems such as job, open and assembly shops.

7 Acknowledgements

It is a pleasure to acknowledge the support from **CONACyT** through grant 178473.

References

1. J. Adams, E. Balas, and D. Zawack. The shifting bottleneck procedure for job shop scheduling. *Management Science*, 34:391–401, 1988.
2. S. Bertel and J. C. Billaut. A genetic algorithm for an industrial multiprocessor flow shop scheduling problem with recirculation. *European Journal of Operational Research*, 159:651–662, 2004.
3. S. A. Brah and J. L. Hunsucker. Branch and bound algorithm for the flow shop with multiple processors. *European Journal or Operational Research*, 51:88–99, 1991.
4. J. Cheng, Y. Karuno, and H. Kise. A shiting bottleneck approach for a parallel-machine flow shop scheduling problem. *Journal of the Operations Research Society of Japan*, 44:140–156, 2001.
5. A. E. Eiben and J. E. Smith. *Introduction to Evolutionary Computing.* Springer, 2003.
6. J. N. D. Gupta. Two-stage hybrid flow shop scheduling problem. *Operational Research Society*, 39:359–364, 1988.
7. J. A. Hoogeveen, J. K. Lenstra, and B. Veltman. Preemptive scheduling in a two-stage multiprocessor flow shop is NP-hard. *European Journal of Operational Research*, 89:172–175, 1996.
8. M. E. Kurz and R. G. Askin. Comparing scheduling rules for flexible flow lines. *International Journal of Production Economics*, 85:371–388, 2003.
9. M. E. Kurz and R. G. Askin. Scheduling flexible flow lines with sequence dependent set-up times. *European Journal of Operational Research*, 159:66–82, 2003.
10. R. Linn and W. Zhang. Hybrid flow shop scheduling: A survey. *Computers & Industrial Engineering*, 37:57–61, 1999.
11. E. Nowicki and C. Smutnicki. The flow shop with parallel machines: A tabu search approach. *European Journal of Operational Research*, 106:226–253, 1998.
12. C. Oguz and M. F. Ercan. A genetic algorithm for hybrid flow shop scheduling with multiprocessor tasks. *Journal of Scheduling*, 8:323–351, 2005.
13. M. Pinedo. *Scheduling Theory, Algorithms and Systems.* Prentice Hall, 2002.
14. D. L. Santos, J. L. Hunsucker, and D. E. Deal. Global lower bounds for flow shops with multiple processors. *European Journal of Operational Research*, 80:112–120, 1995.
15. A. Vignier, J. C. Billaut, and C. Proust. Les problèmes d'ordonnancement de type flow-shop hybride: état de l'art. *Operations Research*, 33:117–183, 1999.
16. H. Wang. Flexible flow shop scheduling: Optimum, heuristics and artifical intelligence solutions. *Expert Systems*, 22:78–85, 2005.
17. B. Wardono and Y. Fathi. A tabu search algorithm for the multi-stage parallel machines problem with limited buffer capacities. *European Journal of Operational Research*, 155:380–401, 2004.
18. R. J. Wittrock. An adaptable scheduling algorithm for flexible flow lines. *Operations Research*, 36:445–453, 1988.
19. Y. Yang. *Optimization and Heuristic Algorithms for Flexible Flow Shop Scheduling.* PhD thesis, Columbia University, 1998.

Natural Language Processing

Simple Features to Identify Tags in Named Entity Recognition for Spanish Texts

Marisol Fierro Ayón, Héctor Jiménez-Salazar &
José de Jesús Lavalle Martínez

Facultad de Ciencias de la Computación
B. Universidad Autónoma de Puebla
C.U. 72570, Puebla, México
Tel. (+52-222)2295500 Ext. 7212, Fax (+52-222)2295672
marisolfa@hotmail.com, hgimenezs@gmail.com, jlavalle@aleteya.cs.buap.mx

Abstract. In this work Named Entity Recognition (NER) using Memory-Based Learning (MBL) is presented. This application is based on several works that deal with this topic. Our contribution is the analysis of some feature sets, taken from POS, capitalized, context, and without external information sources, in order to constitute the training set for the learning method. In the experiments the corpus from the CoNLL-02 conference was used, and for tag identification 96.14% of precision was reached using just 14 features.
Keywords: named entity recognition, memory based learning.

1 Introduction

Named entities (NE) are phrases that contain persons (PER), organizations (ORG), locations (LOC), dates and quantity names (MISC)[1]. For example, in the following clause there are tags that identify the NE that occur in it:

> [PER Fernando Lozano], presidente de [LOC Valle Alto], llegó al [MISC XXI Torneo Universitario].
>
> *[PER Fernando Lozano], president of [LOC Valle Alto], arrived at the [MISC XXI Universitary Tournament].*

Named entity recognition enriches text representation, and it could be applied to tasks supporting Natural Language Processing. As an example, in Question-Answering Systems the responses to questions using pronouns such as where, who, etc. could be supported by NE.

In this work we are focussing on NE of the classes PER, ORG, LOC and MISC. A NE tagger using few linguistic resources and tools, but having a high degree of precision is of particular interest. Nevertheless, to recognize the whole NE occurring in a text is not very important to us. Since we need the building blocks to construct a NE database within of a journalistic navigational system; hence our interest on just precision.

© A. Gelbukh, S. Torres, I. López (Eds.)
Advances in Artificial Intelligence
Research in Computing Science 20, 2006, pp. 31-38

There have been many works about NER, mainly in CoNLL meetings. Spedifically for Spanish, we can cite the works submitted to CoNLL-02 [2]. For example, Fien De Meulder & Walter Daelemans [3] analysed the influence in using external information sources for NER. They used the TiMBL [8] system for NER in English and German. Their system uses a training file and NE lists (*gazetteer*). Some of the features taken into account for the English language are: the context, parts of speech (POS), capital letter use, the first and last three letters of each word, and ten more features which indicate if a word belongs to some NE list. In the German language case, they also used the root of each word of the context.

Another work related with this is the one of Tjong Kim Sang [12], whom evaluated his strategies in language-independent NER tasks using the Memory-Based Learning (MBL) method. He considered as features: contextual words, POS, and morphological features (prefixes, suffixes, capital letters, etc.). The tests presented were obtained applying waterfall, feature selection and voting methods to Dutch and Spanish. The global performance for Spanish, measured with respect to F_1, was 75.78%.

Xavier Carreras *et al.* [11], also in CoNLL-02, got the highest efficacy, 79.38% for F_1. They employed many features: contextual (POS included), morphological, patterns of words, and predictor words. Also, they confirmed that external knowledge sources are not essential.

Thamar Solorio & Aurelio López [5] employed support vector machines (SVM). Unlike the last two works, the dimension of their representation space is very high, because of the combination of the labels that they used, speech parts, and label of the kind of NE (PER, LOC, etc.). Reclassifying, entities given by an extractor, using SVM together with the idea of combined attributes made possible an increase of 7.5% in F_1. Other works were also considered because they are Spanish oriented [6] [1].

In the following section the classification method is presented, after that the data and experiments are described, finally the conclusions are stated.

2 Classification method

The Memory-Based Learning method is trained from a set of resolved instances. The training consists of mapping each attribute with its entropy, in order to define a weighted metric. Such a metric allows us to calculate the "distance" between two instances based on the difference of their attributes; the higher difference between attributes with greater information, the biggest distance between the instances. Then, a new instance will have the same solution as the instance in the training set closest to it.

Formally, let S be an instance or training set, $\{A_1, \ldots, A_m\}$ a set of attributes, and $\mathcal{C}$ a set of classes, or solutions of each instance in S, which are represented in the m-th attribute (A_m). Each instance $X = (x_1, \ldots, x_{m-1})$ is assigned to the class of the instance:

$$Y_0 = \text{argmin}_{Y \in S} \Delta(X, Y), \tag{1}$$

where

$$\Delta(X,Y) = \sum_{i=1}^{m-1} p_i \cdot \overline{\delta}(x_i, y_i), \tag{2}$$

$p_i = Gn(A_i)$, and

$$\overline{\delta}(x_i, y_i) = \begin{cases} 0 & \text{if } x_i = y_i, \\ 1 & \text{if } x_i \neq y_i. \end{cases}$$

The entropy of S is

$$H_S = -\sum_{s_i \in S} \Pr(s_i) \log_2(\Pr(s_i))$$

Given an attribute A_i we can partition S in classes $S|_{x_{ij}}$ (instances with value $x_{i,j} \in A_i$). In this way, the entropy of S with respect to the attribute A_i is the weighting of the entropy for each partition done with the values of A_i:

$$H_{S(A_i)} = \sum_{x_{ij} \in A_i} H_{S|_{x_{ij}}} \frac{\#S|_{x_{ij}}}{\#S} \tag{3}$$

With this, the information gain of an attribute A_i is defined by, $G(A_i) = H_S - H_{S(A_i)}$ and the gain ratio [10] by

$$Gn(A_i) = \frac{G(A_i)}{H_{S(A_i)}}. \tag{4}$$

The algorithm requires an exhaustive search in the training set (equation 1), but it is possible to save computational resources (memory and processor time) using a *trie* tree to represent the instances, which makes it possible to prune the tree on visit and distance computing steps. This implementation is known as *IGTree* [13]. Also, the TiMBL [8] system offers alternative metrics to the one in equation 2.

Using MBL in NER takes up again the BIO tag scheme; B if it is the beginning of the NE, I if the NE continues, and O if it is out of the NE. Even more, if it is the case of a NE its classification (PER, LOC, ORG, MISC) is added. As an example:

> Fernando/B-PER Lozano/I-PER ,/O presidente/O de/O Valle/B-LOC Alto/I-LOC ,/O llegó/O al/O XXI/B-MISC Torneo/I-MISC Universitario/I-MISC .
>
> *Fernando/B-PER Lozano/I-PER ,/O president/O of/O Valle/B-LOC Alto/I-LOC ,/O arrived/O at /O the /O XXI/B-MISC Universitary/I-MISC Tournament/I-MISC ./O*

In this work we focused in identifying (BIO) tags for the (PER, LOC, ORG, MISC) classes in NER.

3 Data sets

The training file contains 273,037 lines, one tag and one word per line. The test file contains 53,049 lines. In our experiments we just took 35,000 (*train*) and 11,000 (*test*) lines of the respective files. These files were obtained from the CoNLL-2002 congress competition [7].

The next step was to select the features that provide more information in solving the problem. Some feature combinations used in other works were taken into account, with very good results. The most important features were intuitively chosen before the experiments were done. Moreover, other feature combinations were tried in order to check their efficacy in NER. The basic features are the following:

– The word context, taking into account three words before and after the word to tag [6].
– The part of speech corresponding to each word of the context.
– Capital letters used in context, if a word begins with a capital letter, it is represented by 1, in other cases by 0.

The files were produced tagging each word with the basic features, and the class to which belongs to. The modified value difference metric was used (MVDM), with $k = 3$ nearest neighbors [3] [4].

4 Experiments

An initial experiment considered to measure the gain ratio of each feature (see figure 1). Context (7), POS (7), using capital letters, and (7) features are arranged on the horizontal axis.

We can see on the graph that the features providing more information to identify tags are those that use capital letters and context; particularly, the fifth word (the one following the word to tag) and using a capital letter has the maximum gain ratio. In order to check the information shown on the graph, experiments about tag identification, using different feature combinations, were done. The following combinations were used:

1. 7 features: word context.
2. 14 features: word context, POS of each word in the context.
3. 21 features: word context, POS , and using capital letters.
4. 14 features: word context, and using capital letters.
5. 8 features: word context, and using capital letters at the fifth word of the context.

We use standard measures, i.e. precision P, recall R, and F_1 to evaluate our results:

$$P = \frac{\#\text{right tags gotten by the system}}{\#\text{tags gotten by the system}}, \quad (5)$$

$$R = \frac{\#\text{right tags gotten by the system}}{\#\text{right tags}}, \quad (6)$$

$$F_1 = (2 \cdot P \cdot R)/(P + R). \quad (7)$$

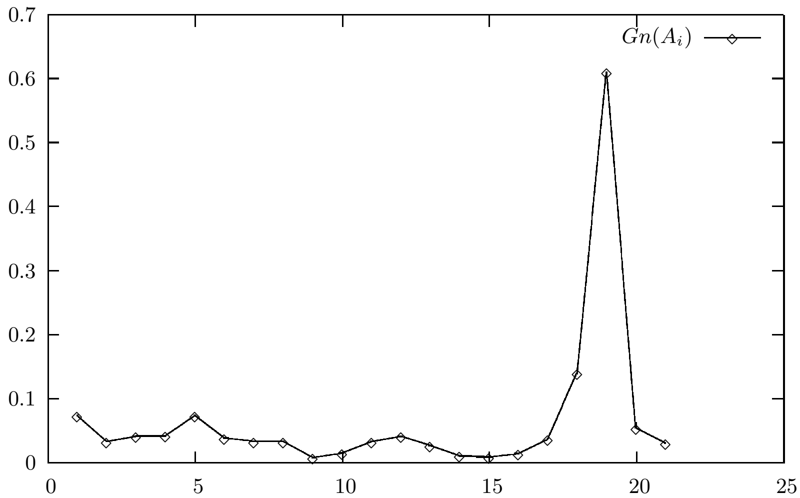


Fig. 1. Information gain of the whole feature set.

As we had said, our interest is on precision, rather than completeness. So, we obtained the recall measure in order to know how many tags will be lost using this method in a text. Recall in each experiment on the CoNLL-02 *test* set is shown on table 1.

Table 1. Recall measure for the different feature combinations.

Tag type	**Experiment** 1	2	3	4	5
O	98.2	97.9	99.6	99.5	98.2
B-ORG	41.3	44.6	66.1	57.5	34.0
B-PER	48.4	51.2	67.2	58.8	43.4
I-PER	49.7	58.7	65.5	67.2	44.9
I-ORG	19.5	26.1	40.9	39.9	16.6
B-LOC	57.4	59.7	68.6	63.2	50.8
I-LOC	34.1	38.9	34.1	34.1	28.0
I-MISC	9.0	15.6	12.0	11.2	8.2
Average	40.6	46.7	**52.5**	**49.7**	37.1

As we can see, the third experiment has the highest recall percentage 52.51%. In this case, the precision is 93.11%. However, this test uses the POS and, in our application, it is not possible to include the use of a POS tagger. So, for this system, precision is preferred rather than recall. We considered in these cases we must adopt the features of test 4, which has recall 49.72% and precision 92.48%. Average standard measurements for the tests are shown in Table 2.

Table 2. Performance for each feature set.

Experiment	P	R	F_1
1	89.53	40.69	55.95
2	90.07	46.77	61.57
3	93.11	52.51	67.15
4	92.48	49.72	64.67
5	88.76	37.15	52.37

In order to situate the results, we did 10 cross validation tests on the development set, and we shall cite the results, about tag identification, presented in the CoNLL-02. The purpose of this comparison is to know how much is lost when some features are omitted (for example, POS), because we are just taking features we are interested in 14 features (experiment 4), and do not need a POS tagger. The averaged measures were $R = 70.71$, $P = 95.89$, and $F_1 = 81.39$. The classification on the development set without cross validating got $R = 69.8$, $P = 96.14$, and $F_1 = 80.87$. Our classification is better than the one presented in [12] ($F_1 = 74.34$). Both took place under the same conditions (test set, learning method, and cross validating). Nevertheless, Carreras *et al.* result [11] ($F_1 = 91.66$) is better than ours.

5 Conclusions

We have shown the results on the efficacy of different sets of features, used in NER by the MBL method, on a collection from the CoNLL-2002 conference. The experiments are based on the combination of basic features: word context (three words before and after), the POS of words in the context, and the presence of a capital letter at the beginning of the words in the context.

We see that contextual and some morphological features are very helpful in classifying tags for NER. Other authors have referred that external information sources are almost useless. In this work we have seen that, omitting the POS of contextual words does not impact the precision in identifying tags for NER: the maximum precision gotten in CoNLL-02 [11] was 92.45 against our result 96.14 reached using just 14 features. Certainly, our recall is very poor (90.88 by them against 69.8 by us), because of the reduced number of features that we used.

References

1. Lluis Padró: Tendencias en el reconocimiento de entidades con nombre propio, en *Tecnologías del Texto y del Habla*, M. Antonia, Martí & Joaquim, Llisterri (Eds.) Edicions Universitat de Barcelona, Fundación Duques de Soria (2004).

2. Eric F. Tjong Kim Sang.: Introduction to the CONLL-2002 Shared Task: Language-Independent Named Entity Recognition, *CNTS - Language Technology Group*, University of Antwerp.
3. Fien, De Meulder & Walter, Daelemans.: Memory-Based Named Entity Recognition using Unannotated Data, *CNTS - Language Technology Group*, University of Antwerp.
4. Walter Daelemans., Jakub Zavrel, Ko van der Sloot & Antal van den Bosch.: TiMBL: Tilburg Memory Based Learner, version 4.3, Reference Guide (2002).
5. Thamar, Solorio & Aurelio, López López.: Adapting a Named Entity Recognition System for Spanish to Portuguese, Instituto Nacional de Astrofísica Optica y Electrónica,2004
6. Alvaro, Balbontín Gutiérrez & José Javier Sánchez Martín.: SPNER-Reconocedor de entidades nombradas para el español, Universidad Europea de Madrid.
7. Language Independent Named Entity Recognition, *Conference on Natural Language Learning 2002*, `http://lcg-www.uia.ac.be/conll2002/ner/`.
8. Machine Learning for Language Engineering and Linguistics, Tilburg Uniersity, `http://pi0657.uvt.nl/`.
9. X. Carreras & L. Padró: A flexible distributed architecture for natural language analyzers, In *Proc. of LREC'02*, Las Palmas de Gran Canaria, España, 2002.
10. C. E. Shannon: Communication Theory, *The Bell System Technical Journal*, (27) 379, p. 19-48, 1950.
11. Xavier Carreras, Lluis Màrquez & Lluis Padró: Named Entity Extraction using AdaBoost, Language Independent Named Entity Recognition, *Conference on Natural Language Learning 2002*.
12. Eric F. Tjong Kim Sang.: Memory-based Named Entity Recognition *Conference on Natural Language Learning 2002*.
13. Walter Daelemans, Antal van den Bosch, Jakub Zavrel, Jorn Veenstra, Sabine Buchholz, and Bertjan Busser: Rapid development of NLP modules with memory-based learning, In *Proceedings of ELSNET in Wonderland*, pp. 105-113. Utrecht: ELSNET, 1998.

Towards a Framework of Verbal Compositionality

Gaspar Ramírez[1], Héctor Jiménez-Salazar[2] & Jim Fidelholtz[1]

[1]Maestría en Ciencias del Lenguaje,
Casa Amarilla, 2 Oriente, CP 72000
[2]Facultad de Ciencias de la Computación,
Ciudad Universitaria, Edif. 135, CP 72570
Benemérita Universidad Autónoma de Puebla
Puebla, Pue., México.
{gasparfirst, hgimenezs}@gmail.com, jfidel@siu.buap.mx

Abstract. Lexical Semantic Theory faces problems of how words acquire different meanings in distinct contexts. In this work we analyze the first steps that should be taken in order to constitute a combinatory dictionary of Spanish verbs. Our proposal is conceived within the Generative Lexicon approach of James Pustejovsky, and we discuss some ideas of how to build a dictionary with such characteristics.

1 Introduction

In modern linguistics and Natural Language Processing (NLP), regular polysemy has recently become a phenomenon of natural languages which people from a wide variety of fields have begun to study. Pustejovsky [5] noticed that the specific polysemy of some aspectual verbs like *terminar, comenzar*, experiencer verbs like *disfrutar*, and many causatives (we have given their Spanish equivalents) depends largely on the particular context in which they occurred. Some examples are

(0)a. Juan terminó/ disfrutó su cigarro.
(0)b. Juan terminó/ disfrutó su café.

In sentence (0a) terminar/disfrutar means "*terminar/disfrutar de fumar*", whereas in (1b) it means "*terminar/disfrutar de beber*". Here *su cigarro* and *su café* 'coerce' the meaning of the verbs. Pustejovsky proposed a Generative Lexicon (GL) where categories such as verbs and nouns are linked by means of a flexible mechanism made up of different levels of semantic representation. This mechanism is able to capture the contribution of verb arguments to the meaning of the verb.

We think that what categories and elements of these categories refer to could be better understood from a co-compositional or co-occurrence perspective where new features of meaning may arise. Thus lexical and syntagmatic features form the core of our proposal for elaboration of a lexicon of verbs in Spanish.

This research is inserted within the field of lexical semantics. The theory of lexical semantics deals with the problem of how words can acquire different

© A. Gelbukh, S. Torres, I. López (Eds.)
Advances in Artificial Intelligence
Research in Computing Science 20, 2006, pp. 39-48

meanings in different contexts, how meanings can arise compositionally, and how semantic types can be mapped into the syntactic forms in a predictable way. To discover what factors in a speech act are responsible for our ability to convey the wealth and diversity of meaning with a quite limited number of linguistic resources is a task that is worthwhile, since empirical research in this area is scarce. The aim of lexical semantics is therefore to provide a detailed description of how expressions in language acquire their content and how this content seems to suffer continued modification and modulation in novel contexts. This research attempts to analyze lexical semantics of verbs both individually and in combination with other lexical items in order to incorporate their linguistically fine-grained description in a combinatory explicative dictionary (cf. Mel'čuk [4]) covering the following important considerations in formal semantic theory.

The methodology that will be employed consist, on the one hand, of grouping the meanings of verbs according to the syntactic frame in which they participate; this is commonly known as verb alternations. On the other hand, it also includes aspect or Aktionsarten [7] of verbs as a way to capture the way verbs are conceptualized. Cognitively, this feature is as important as the difference that exists among countable and uncountable nouns. Finally, verbs will be classified in semantically unique classes. Of course, there are some verbs which may appear in more than one class due to the kind of action which they perform.

In this work two theoretical assumptions are considered to describe in detail the semantics and lexicon of any natural language. First, it is known that without considering the syntactic structure of a language, the study of lexical semantics will not work. In other words, there is no way to separate completely the meaning from the structure that carries it. The second assumption also says that the meanings of words should somehow reflect the deeper conceptual structures in the cognitive system, as well as the domain it operates in.

From a computational lexical semantic perspective the following principles should in some way be considered. First, a clear notion of well-formedness in semantics will be necessary in order to characterize a theory of word meaning. Secondly, lexical semantics should look for representations richer than the descriptions from thematic roles. Thirdly, several levels of semantic interpretation should be used.

Recent works in lexical semantics have been largely focused on clarifying the nature of verb classes and the syntactic structure that each allows (cf. Levin 1985, 1993, taken from [5]). However, we should explain syntagmatically why verb classes behave as they do, and what consequences these distinctions have for the rest of the lexicon and grammar. Thus the aim of this research is to identify the similarities and differences, semantic as well as non-semantic, of verbs considered compositionally, according to the context in which they occur. Following Pustejovsky [5], a lexical semantic theory should not merely map the number of lexical meanings per sentence, on an individual basis. Rather, it should capture the relations between words in a way which facilitates this mapping.

In order to support the lexical representation proposed by Pustejovsky, two basic concepts and their use are introduced in section 2 and 3 of this paper. Section 4 presents a discussion on the development of our combinatory dictionary, referring to some related works.

2 Semantic Classes and Categorial Alternations

In the tradition of formal semantics, perhaps the most relevant aspect of the meaning of a word is its semantic type. Therefore type or categorial information determines not only how a word behaves syntactically, but also what the elements of such categories refer to.

Some examples follow. The verbs *amar* and *odiar* may be considered as relations among individuals in the world, whereas *mujer* would select the set of all individuals that are women. As type distinctions are generally very broad, lexical semantics distinguishes even selectional subsets for members of these categories. A finer lexical semantic representation for the lexical items and its combination with other item is then necessary in order to characterize broadly the expressive power of languages.

2.1 Semantic Classes

This research is based on one of the oldest semantic classifications of verbs, the aspectual class or *Aktionsarten*. This classification considers that verbs and verbal phrases vary according to the types of events that they denote in the world or, in other words, the kind of action they denote. It is usually assumed that there are at least three aspectual types: state, activity, and event, where the last sometimes is divided into accomplishment and achievement events.

Some examples show what we mean by aspectual class. The verb *caminar* in sentence (1) denotes an activity of unbounded duration, that is, the sentence itself does not carry information about the temporal extension of the activity, although deictically it turns out to be an event that did finished in the past.

(1) María caminó ayer.
(2) Maria caminó a su casa ayer.

It is said that sentence (1) denotes an activity. Other examples of this class of verbs are: *dormir, correr, trabajar, beber*, etc. On the other hand, sentence (2) also conveys the same information as the previous one, except that in this case the constraint appears that María finished walking when she arrives to her house. Although there isnt any explicit reference to duration of the activity, this sentence states that the process has a logical culmination, since the activity finishes when María gets home. It is said that this kind of sentence denotes an accomplishment event.

Just as the verb *caminar* seems by default to represent an activity in lexical terms, there are verbs that seem to denote accomplishments lexically. For example, the verbs *construir* and *destruir*, in their typical transitive use, denote accomplishment events since there is a logical culmination to the activity performed.

(3) María construyó una casa.
(4) María destruyó la mesa.

In sentence (3) the coming into being of the house is the culmination of María's act, while in (4) the non-existence of something referred to as a table is the direct culmination or consequence of this act. Verbs of creation are the best examples of accomplishments events. One of the classical diagnostics to probe if a verb (phrasal or not) denotes an accomplishment is its modification by time adverbials like *en una hora*, that is, so-called adverbial frames. Notice that both derived accomplishments (5) and lexical accomplishments (6) permit this modification, while activities (7 and 8) do not.

(5) María caminó a la tienda en una hora.
(6) María construyó la casa en un año.
(7) *Juan se bebió en 20 minutos.
(8) *María se trabajó en una hora.

Apparently, an adverbial frame requires that the verb or phrasal verb make an explicit reference to a change of state, a precondition which is missing in (7) and (8).

An achievement, on the other hand, is an event that undergoes a change of state, similarly to what happens in an accomplishment event, but where the change is thought of as occurring instantaneously. For example, in sentences (9), (10) and (11) the change is not gradual, but something that has a point-like character. Therefore, modification with punctual adverbials such as *a las 3 en punto* suggests that the sentence denotes an achievement event.

(9) Juan murió a las 3 en punto.
(10) Juan encontró su cartera a las 3 en punto.
(11) Maria llegó a la media noche.

Of course, punctual adverbial modification is not restricted just to achievement events, as the following examples show:

(12) Ella nadó el canal a las 10:00 a.m.
(13) El pianista ejecutó la sonata al medio día.
(14) Jaime enseñó su seminario de tres horas a las 2:30.
(15) Él dictó su conferencia a las 4 p.m.

Here the punctual adverbial indicates the beginning of an event with certain duration. It seems that some lexical proprieties of verbs may be affected by the sort of complement with which they interact.

As we can see by the examples given so far, the kind of event that a verb denotes may vary from a compositional perspective. Therefore co-occurrence meaning as well as compositionality should be considered when describing a lexical item. A shift of meaning in the verb arises as a result of the syntagmatic interactions and the semantic and syntactic relationship of the verb with the rest of the items in the sentence.

2.2 Verb Alternations

We also employ a recently developed methodology to group the meanings of verbs in semantic classes through the analysis of the syntactic frames in which they participate; that is, common grammatical verb alternations. Here we can mention as an example of similar work the MIT Lexicon Project, which outlines a large classification of argument verb alternations in English in order to classify verbs into semantically unique classes. Let us consider the following examples, the verbs *hundir, rodar*, and *romper* all have transitive and intransitive forms when their lexical senses are related to the interpretative characteristic of causation.

(16) a. El bote se hundió en un clima tormentoso.
b. El avión hundió el bote en un clima tormentoso.

(17) a. La pelota rodó por la colina.
b. Bill rodó la pelota por la colina.

(18) a. Súbitamente, la botella se rompió.
b. Súbitamente, Maria rompió la botella.

(19) a. La carta llegó a tiempo.
b. *El cartero llegó la carta a tiempo.

(20) a. Mi terminal murió anoche.
b. *La tormenta murió mi terminal anoche.

(21) a. La torre de bloc cayó.
b. *Zacarías cayó la torre de bloc.

Although sentences (19b), (20b), and (21b) are ill-formed, they are certainly understandable. A lexical semantic theory should specify what these two classes share; for example, both have intransitive grammatical forms. Thus it is important to identify similarities among verbs for establishing a domain where lexical items are somehow unified (unification), but equally important is the characterization of how verbs differ (individualization); for example, the latter group does not allow transitive form. The question is whether it is possible to identify the linguistically relevant features that lead us to the distinct behavior of the transitive verbs above. However, as Pustejosvky [5] claimed, we can only explain the behavior of a verb's semantic class can be achieved only by acknowledging that the syntactic patterns in an alternation are dependent on the information carried by the arguments in the patterns themselves. In other words, the diversity of complement types that a verb or other category may take is in large part determined by the semantics of the complements themselves.

There are other alternations of argument change than the ones discussed above, as well as alternations of argument drop.

(22)a. La mujer comió su cena rápidamente.
b. La mujer comió rápidamente.

(23)a. El perro devoró la galleta.
b. *El perro devoró.

(24)a. Juan bebió la cerveza febrilmente.
b. Juan bebió febrilmente.

(25)a. Febrilmente, Juan se hecho de un trago la cerveza.
b. *Juan se hecho de un trago febrilmente.

(26)a. María tarareó una canción mientras caminaba.
b. María tarareó mientras caminaba.

(27)a. María interpretó una canción mientras comía su cena.
b. *María interpretó mientras comía su cena.

Grammatical alternations, along with aspect or *Aktionsarten*, can be used throughout the grammar of a language to make semantic distinctions between verbs on the basis of syntactic behavior, and in the same sense to find similarities. Using categorial selection information as well as the data from grammatical alternations, verbs can be grouped in semantic classes which, at the same time, have predictable syntactic behavior.

3 Levels of Representation

Next, we explain how lexical information is organized within a GL.

Following Pustejovsky [5], a GL is regarded as a computational system that involves at least 4 levels of representation.

1. ARGUMENT STRUCTURE: Specification of number and type of logical arguments, and how they are realized syntactically.
2. EVENT STRUCTURE: definition of the event type of a lexical item and a phrase. Types include STATE, PROCESS, and TRANSITION, and events may have a subevent structure.
3. QUALIA STRUCTURE: Modes of explanation composed of FORMAL, CONSTITUTIVE, TELIC and AGENTIVE roles.
4. LEXICAL INHERITANCE STRUCTURE: Identification of how a lexical structure is related to other structures in the type lattice, and its contribution to the global organization of the lexicon.

Thus he argues that a set of generative devices connects these four levels, providing for the compositional interpretation of words in contexts. These devices are simply semantic transformations, all involving well-formedness conditions on type combinations.

– TYPE COERCION. Where a lexical item or phrase is coerced to a semantic interpretation by a governing item in the phrase, without changing of its syntactic type.
– SELECTIVE BINDING. Where a lexical item or phrase operates specifically on the structure of a phrase, without changing the overall type in the composition.
– CO-COMPOSITION. Where multiples elements within a phrase behave as functors, generating new non-lexicalized senses for the words in composition. This also includes cases of underspecified semantic forms becoming contextually enriched, such as manner co-composition, feature transcription, and light verb specification.

When we define the functional behaviour of lexical items at different levels of semantic representation, we hope to get at a characterization of the lexicon as an active and integral component in analyzing the compositional aspects of sentence meaning.

As we can see by the examples that we have presented so far, new meanings of words seem to emerge if words are regarded in composition rather than considering them as isolated and unrelated lexical items. Therefore, a generative lexicon must be seen as a structured system where different grammatical categories are linked in order to show the semantic relatedness which can arise within a co-occurrence and co-compositional semantic frame.

Next, we shall offer an example of a standard entries under our proposal based on a merger of a Combinatory Explicative Dictionary (CED) with a generative lexicon. In addition to the semantic type system, we also include other items needed in a detailed description of an entry. These items are: the meaning zone, the co-occurrence constraints zone, and the zone of illustrations. Here we offer the meaning of the word in Spanish, with its English translation. However, in this work the zone of co-occurrence constraints is not yet taken up.

Standard definitions

Construir v. tr. (lat. Construere) [29]. Hacer una obra material o inmaterial, juntando los elementos de acuerdo a un plan: *construir un edificio*, *construir una teoría*, 2. LING. Ordenar y enlazar debidamente las palabras en la oración o frase 3. MAT. Trazar o construir un polígono.
[**Construir** v. tr. (lat. Construere) [29]. Make a concrete or abstract work joining the elements according to a plan: *construct a building*, *construct a theory.* 2. LINGUISTICS. Order and correctly connect the words in a sentence or phrase. 3. MATHEMATICS. Plot or construct a polygon.]
Construir: Crear una cosa material o inmaterial, agrupando las partes según un plan trazado.
[**Construir**: Create a concrete or abstract thing, grouping the parts according to a plan.]

Combined CED and GL

$$
\begin{bmatrix}
\text{Construir} & [\mathit{Construct}] \\
\text{EVENTRSTR} = & \begin{bmatrix} \text{E1=} & \textbf{process} \\ \text{E2=} & \textbf{state} \\ \text{REST=} & \leftarrow \\ \text{HEAD=} & e_1 \end{bmatrix} \\
\text{ARGSTR} \quad = & \begin{bmatrix}
\text{ARG1=} & 1 \begin{bmatrix} \textbf{animate-individual} & \\ \text{FORMAL=} & \textbf{physobj} \end{bmatrix} \\
X = 1; & \text{who constructs?} \\
\%\ \mathit{Juan} \sim & [\%\ \mathit{John} \sim] \\
\text{ARG2=} & 2 \begin{bmatrix} \textbf{entity} & \\ \text{CONST=} & 3 \\ \text{FORMAL=} & \textbf{physobj/absobj} \end{bmatrix} \\
Y = 2; & \text{what?} \\
\% \sim \mathit{una\ silla} \simeq & [\% \sim \mathit{a\ chair} \simeq] \\
\% \sim \mathit{una\ teoría} \simeq & [\% \sim \mathit{a\ theory} \simeq \\
\text{D-ARG1} & 3 \begin{bmatrix} \textbf{material} \\ \text{FORMAL=}\ \textbf{mass} \end{bmatrix} \\
Z = 3; & \text{from what?} \\
\% \simeq \mathit{sobre\ el\ clima} & [\% \simeq \mathit{about\ the\ climate}] \\
\% \simeq \mathit{de\ madera} & [\% \simeq \mathit{of\ wood}]
\end{bmatrix} \\
\text{QUALIA} \quad = & \begin{bmatrix} \textbf{create-lcp} & \\ \text{FORMAL=} & \textbf{exist}(e_2, 2) \\ \text{AGENTIVE=} & \textbf{build-act}(e_1, 1, 3) \end{bmatrix}
\end{bmatrix}
$$

4 Discussion

We have established the most important items of GL theory. From a methodological viewpoint it is necessary to build several elements using NLP tools in order to construct a viable lexicon. We think the following steps are indicated:

1. Create a list of lemmas: $L = \{x_1, \ldots, x_N\}$.
2. For each $x_i \in L$:
   (a) Create an initial matrix for x_i, m_i; e.g. using the most frequent sense.
   (b) Obtain all synsets of x_i from EuroWordnet: $S_i = \{s_1, \ldots, s_n\}$.
   (c) Extract a large quantity of sentences from a corpus; e.g. the web: $O_i = \{o_1, \ldots, o_m\}$.
   (d) Assign a sense from S_i to each $o_j \in O_i$ and cluster the o_j according to its sense. Let $C_i = \{c_1, \ldots, c_k\}$ be the clusters obtained.
   (e) Analyze each cluster c_l and update the matrix m_i.

The most difficult task in the above procedure is word sense disambiguation, but the manual work that the above steps imply cannot be ignored. Certainly, there are tools that assign the sense to a word in a context, but they are not very precise. Likewise, the process has to be semiautomatic. It is important to take advantage of other approaches to building lexicons under GL theory.

There have been many projects to create lexicons following the GL proposal. In our work, we try to resolve several problems that have come up in previous works. In the following paragraphs, we give a brief outline of some work related to the construction of lexicons.

Before analyzing the proposals, some important issues must be highlighted. First, a speaker can efficiently create a "new" sense of a word in a given context. Second, if we proceed to build a lexicon based on the GL approach using a corpus, the rules will be able to do better with "new" word uses, insofar as the corpus is larger. Thus a limit to the creative understanding of new uses of words will be the size of the corpus. That is, the challenge is for rules in the GL to provide enough information to proceed when faced with new uses of words, which will obviously be easier with a larger corpus. Kilgarriff [3] refers to the last point.

In [3] Kilgarriff focuses on the power of the GL approach. His evaluation is centered in non-standard word uses, trying to answer whether such uses could be analysed by GL strategies. A non-standard use was defined as not fitting under any dictionary definition of the word in a particular dictionary. He found that from 41 instances of non-standard uses just 5% (two instances) were plausible candidates for the GL treatment. So, without intending to undermine GL analysis, he shows that the GL is suitable for only some lexical phenomena, not all.

Because building a lexical resource is time-consuming and costly, Ruimy et al. [6] report the development and experimental implementation of a combined methodology of knowledge transfer from a monolongual Italian lexical resource to a French semantic lexicon. This work follows the theoretical principles of GL theory. The main idea is from an Italian lexicon to semi-automatically inferr a similar annotated French lexicon. They used translation word pairs provided by bilingual dictionaries in order to assign semantic properties given by the Italian lexicon to word senses of French. Their approach takes as much advantage as possible of similarity between French and Italian; the cognate approach, based on regularities of some suffixes in both Italian and French. On the other hand, in the cases that the cognate-based method was not applicable, they used sense indicators taken from bilingual dictionaries. The success rate for such suffixed words was 95%. Still, the methodology did not prove very efficient in completing the lexicon. However, the authors are hopeful that the methodology used can be applied in similar cases.

Qualia structure is generally understood as a representational tool for expressing the componential aspect of word meaning. While FORMAL, CONSTITUTIVE, TELIC and AGENTIVE qualia roles provide the basic structuring of a semantic type, Busa et al. [2] introduce the notion of Extended Qualia Structure (EQS) in the framework of the development of large-scale lexical resources, the SIMPLE model. EQS is motivated by the fact that lexical items may share the same structural properties. EQS is achieved by decomposing a qualia role into subtypes of the role consistent with its interpretation. There are strong types which create a new type of qualia, and weak types which add information to a type without changing its nature. The authors created a library of templates that provide the constraints and conditions for a lexical to belong to a type. SIMPLE may be viewed as a template-based framework for lexicon development because each type is associated with a template which provides the well-formedness condition for a lexical item to be of a given type.

We have seen that the GL theory continues to develop and has an impact on lexicon building. Furthermore, several strategies for constructing the dictionary of Spanish verbs may be exploited, as the works on this topic suggest.

Of course, all of the above suggestions need to be tried out for Spanish verbs, to see if the theory-based suggestions pan out in practice. In any case, we belive that many of them will prove useful, as has been the case in earlier work on other languages. The master thesis of the first author will include many of the topics mentioned in this paper, and the results will be reported in due course.

References

1. Bouillon, Pierrete & Federica Busa: *The Language of Word Meaning*, Cambridge University Press, 2001.
2. Busa, Federica; Nicoletta Calzolari & Alessandro Lenci: Generative Lexicon and the SIMPLE Model: Developping Semantic Resources for NLP, in [1], pp 333-349, 2001.
3. Adam Kilgarriff: Generative Lexicon Meets Corpus Data: The Case of Nonstandard Word Uses, in [1], pp 312-330, 2001.
4. Mel'čuk, Igor A.: *Dependency Syntax: Theory and Practice*, Albany, State University of New York Press, 1988.
5. Pustejovsky, James: *The Generative Lexicon*, MIT Press, 1995.
6. Ruimy, Nilda; Pierrette Bouillon & Bruno Cartoni: Inferring a Semantically Annotated Generative French Lexicon from an Italian Lexical Resource, in *Third International Workshop on Generative Approaches to the Lexicon*, pp 218-226, 2005.
7. Vendler, Zeno: *Linguistics in Philosophy*, Cornell University Press, Ithaca, 1976.

The Recognition System of CCE (Chinese Conventional Expression) for CSL Learners

Shuang Xiao[1] Hua Xiang[1] Fuji Ren[1,2] and Shingo Kuroiwa[1]

[1] Faculty of Engineer, University of Tokushima 2-1 Minamijosanjima, Tokushima, 770-8506, Japan {sxiao, xianghua, ren, kuroiwa}@is.tokushima-u.ac.jp
[2] School of Information Engineering, Beijing University of Posts and Telecommunications Beijing 100876, China

Abstract. CSL (Chinese as Second Language) learners need relevant reading support because recognition of CCE (Chinese Conventional Expression) is a difficulty for them. In this paper, we mainly propose how to help CSL learners recognize CCE in Chinese text. We have created a CCE data-base with 2,305 conventional expressions of contemporary Chinese. At the same time we have analyzed the basic structures and application forms of these 2,305 conventional expressions. By the analysis we have presented an extraction approach which based on rules and characters of these CCE. In our extraction experiment of CCE, the recall achieved 81.65% and the precision achieved 94.34%.

1 Introduction

As a matter of fact CCE (Chinese Conventional Expression) is a big obstacle for the CSL (Chinese as Second Language) learners, it is obliged to recognize and understand CCE in Chinese way. CCE is a traditional, habitual expression that is widely used in daily life among Chinese. The applications of CCE are very flexible and unique. CCE also is a kind of special phrase which has the fixed structure [1]. Usually, they consist of two or more than two words, and most of them have the strong culture background. Because the natural meaning of CCE is not the simple combination of literal meaning of each word but is the extensional cultural implications of the phrase. In this case the learners maybe can't catch the real meaning of the term even if they can understand the every word of the CCE literally. For example: *'Yin1Wei2 Ta1 Diu1 Le Fan4Wan3, Quan2Jia1 De Sheng1Huo2 Jiu4 Bian4 De2 Fei1Chang2 Kun4Nan4 Le.'* (The living of his family has been very difficult because he lost his job.) In this sentence the underline parts-*'Diu1 Fan4 Wan3'* is a CCE which consist of the word *'Diu1'* (lose) and word *'Fan4 Wan3'* (rice bowl). Here the real meaning of the phrase is 'lose someone's job'. For CSL learners, it is a problem that how to recognize and understand the CCE in this sentence. (The italics are Chinese Pinyin, the Roman transliteration of Chinese characters, which is used throughout this paper for the convenience of English readers. The numbers are tone markers.)

In this paper, we discuss how to help CSL learners to recognize CCE in Chinese text. We have created a CCE database with 2,305 conventional expressions of contemporary Chinese. In section 2, we analyze the basic structures

© A. Gelbukh, S. Torres, I. López (Eds.)
Advances in Artificial Intelligence
Research in Computing Science 20, 2006, pp. 49-62

and application forms of these conventional expressions. In section 3, we present an extraction approach which based on rules and characters of these CCE to recognize them. Section 4 is the description of extraction experiment and some discuss of the result. Finally, we give conclusions and introduce part of our later research work.

2 Analysis of CCE

2.1 Analysis of Chinese Conventional Expressions

The system we presented is based on a database of 2305 contemporary CCE. By the statistical analysis of these 2305 conventional expressions, we have found that contemporary CCE may consist of different character quantity. Generally, contemporary CCE consist of three Chinese characters to twelve Chinese characters. And we have also found the conventional expressions with three Chinese characters are in majority in total quantities of contemporary CCE – they account for 65.2% (1503) of all. The conventional expressions with four Chinese characters take up the second large quantities-about 16.5% (381). Furthermore, the conventional expressions with five, six, seven, eight Chinese characters account for 7.7% (177), 4.9% (113), 3.7% (89) and 1.4% (32) respectively. Other CCE only account for 0.4% (10). Hence, according to situation of quantities, processing on conventional expressions with three, four, and five characters is the most important work for the research.

Simultaneously, we have analyzed the structures and unitary attributes of conventional expressions. We have classified four categories of conventional expressions by various external unitary attributes-verbal phrase expressions, noun phrase expressions, 'clause' expressions and the others expressions. The verbal phrase expressions include: 'predicate-object' structure, 'adverb-headword' structure, 'continuous predicates' structure and 'predicate-complement' structure. The noun phrase expressions include: 'attribute-headword' structure, 'parataxis' structure and 'character De' structure. The 'clause' expression including: 'subject-predicate' structure and 'complicate' structure. 'Complicate' here denotes that the structure is far more complicate than common phrases. It looks like a clause. The other expressions include: 'special' structure and 'comma separate' structure. 'Special' structure indicates the phrases with irrational structure which is absolutely different from general phrase structures. The 'comma separate' structure expression consists of 'in front part' and 'behind part'. Most 'in front part' and 'behind part' of 'comma separate' structure expression have symmetrical structure and equal quantities of characters.

From Table 1 we can learn that contemporary CCE is mainly described by verbal phrase expressions and noun phrase expressions. For the reason they account up to 86.7% of whole conventional expressions, in our research we have focused on these two kinds of conventional expressions.

Table 1. The Structures and Attributes of Chinese Conventional Expressions

Categ.	No.	Structure category	Examples	Q.	Freq.
Verbal phrase	1	'predicate-object' structure	*Chi1 Bai2Fan4* (fathead, to be a 'good-for-nothing')	1056	45.8
	2	'adverb-headword' structure	*Ji1Dan4 Li3 Tiao1 Gu2Tou2* (disposition to find and point out trivial faults)	50	2.2
	3	'continuous predicates' structure	*Jian4 Pian2Yi2 Jiu4 Qiang3* (to gain extra advantage unfairly)	114	4.9
	4	'predicate-complement' struc.	*Huo2 De2 Bu2Nai4Fan2* (the act or process of destroying oneself or itself)	5	0.2
Noun Phrase	5	'attribute-headword' struc.	*Hui1Se4 shou1Ru4* (illegal income)	759	32.9
	6	'parataxis' struc.	*Ban4Jin1 Ba1Liang3* (all the same)	12	0.5
	7	'character De' struc.	*Na2 Bi2Gan3 Zi3 De* (the intellect)	4	0.2
'Clause'	8	'subject-predicate' structure	*Jing3Shui3 Bu2 Fan4 He2Shui3* (none may encroach upon the precincts of another)	172	7.5
	9	'complicate' structure	*Ge1Bo1 She2 Le Wang3 Xiu4Zi3 Li3 Cang2* (to endure an humiliation by one's own self)	35	1.5
Others	10	'special' structure	*San1 Yi1 San1 Shi2 Yi1* (to divide equally, share alike)	18	0.8
	11	'comma separate' structure	*Chai1 Dong1Qiang2, Bu3 Xi1Qiang2* (keep up in one place at the expense of others)	80	3.5
Total	11			2305	100

2.2 Form Analysis of CCE in Applications

We have created a large conventional expressions database of 21,018 example sentences. By statistics and comparing, we have found that there are various expressional situations exist in CCE. The detail analysis is depicted in table 2. We have found that the CCE may remain their primary forms in most of time. These phenomena take up 82.4% in whole quantities of conventional expressions. As the second largest phenomena of conventional expression, the 'inserted words' CCE may take up 14.8% in whole quantities of our database. Another expressions is 'the first Chinese character repeating' phenomena. These phenomena indicate that a 'predicate-object' conventional expression with single verb Chinese character repeats at beginning. They account for 0.6% of the whole quantities. Next expressional situation is 'character replacing'. 'Character replacing' indicates that one certain character of the conventional expression can be replaced by the other characters (usually replaced by single character verb). After replacing, the novel conventional expression will retain the original meaning as before. These kinds of conventional expressions account for 0.5% of whole quantities. Besides, 'order changing' is a frequent expressional phenomenon too. 'Order changing' indicates that the order of conventional expression may be changed with the other inserted words. They account for 1.7% of whole. Because ‘order changing’ expression is extremely complicate, and the database we collected about it is not large enough. Therefore in current work we will not make a detail analysis on these phenomena temporarily.

Table 2. Form Categories of Chinese Conventional Expressions in Application

Form in using	Example	Q.	Freq.
‘unchanged’ form	*Ci3Di4 Wu2 Yin2 San1Bai2 Liang3* (a very poor lie which reveals the truth)	17325	82.4
Form with ‘inserted word’	*Zhua1 Bie2Ren2 De Bian4Zi3* (to seize on other people’s mistake or failure)	3116	14.8
‘the first Chinese character repeating’ form	*Dai4 Dai4 Gao1 Mao4Zi3* (the vain compliments)	123	0.6
Form of ‘character replaced’	*Gan3/Da3/Na2 Ya1Zi3 Shang4 Jia4* (force someone to do something)	102	0.5
‘order changing’ form (with the other insert words)	*Jian3 Zhao2 Pian2Yi2 Le* (to get a bargain, to get an extra advantage) *Pian2Yi2 Dou1 Rang4 Ta1 Jian3 Zhao2 Le* (He has gained all the advantages.)	352	1.7
Total		21018	100

Table 3. Categories of Chinese Conventional Expressions with Inserted Words

Structure categories	Example	Q.	Freq.
‘predicate-object’ structure	*Da3 Le Yi2 Ge4 Piao4Liang4 De Fan1Shen1 Zhang4* (changed completely)	2521	80.9
‘subject-predicate’ structure	*Jia4Zi3 Hen3 Da4* (arrogant, haughty)	327	10.5
‘attribute-headword’ structure	*Zhang3 Shang4 De Ming2Zhu1* (‘a pearl on the palm’, a parent refers affectionately to a beloved daughter)	236	7.6
‘parataxis’ structure	*Ji1Mao2 He2 Suan4Pi2* (tiny things, bits and pieces)	32	1.0
Total		3116	100

From table 3 we can learn that the change of the conventional expressions with ‘predicate-object’ structure is most active in daily applications. They take up 80.9% in entire ‘inserted words’ conventional expressions. The next frequent conventional expression forms are ‘subject-predicate’ and ‘attribute-headword’ structures, they take up 10.5% and 7.6% of whole ‘inserted words’ conventional expressions respectively. Besides, the ‘parataxis structure’ are relative less used, only account for 1.0%.

3 Recognition Processing of CCE

Comparing with English and Japanese, Chinese is a kind of 'isolated language'. It lacks of some characteristics (such as case-auxiliary word and changing of verb forms etc.) which English and Japanese do. In this case Chinese have more difficulties in word segment, lexical analysis and syntactic parsing than others [4][5]. From 90's, many researchers have tried to use shallow parsing technique to Chinese processing. At the same time, statistic method is adopted frequently [6][7][8]. Unfortunately, using the method which is based on frequency or collocation can not succeed fully for CCE. Because of that: many CCE have too small frequency in a corpus and some CCE don't have strong collocation. Thus in this paper, we have proposed a way to extract CCE based on rules and characters. The detail procedure is described as the next five steps.

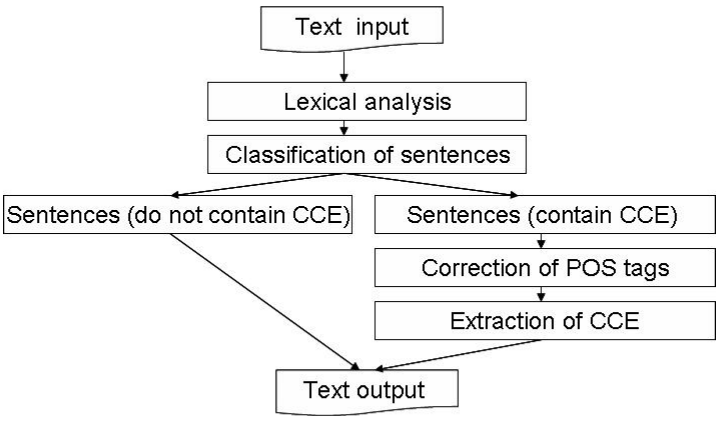


Fig. 1. The Recognition Process of CCE

3.1 Conventional Expression Registration

First of all we have registered almost all the conventional expressions we collected into the system. The registered CCE include not only the CCE themselves but also the structure, POS, and attributes of the CCE. All the CCE we put into register of the system consist of the four kinds of information.

3.2 Lexical Analysis

The lexical analysis system we adopted in our system is ICTCLAS (Institute of Computing Technology, Chinese Lexical Analysis System) from Chinese Academy of Sciences. ICTCLAS have used the approach based on multi-layer HMM. It has a high segmentation precision of 97.58%. And the POS tagging we used is come from POS Tagging Collection from Beijing University.(http://www.nlp.org.cn/)

3.3 Classification of Sentences

After the lexical analysis, we can classify all the sentences of input text into two parts. One part of sentences doesn't contain CCE, the lexical analysis results of them will be output directly. The other part of sentences that contain CCE will be processed further. According to CCE in the sentences, we can divide the sentences with CCE into three types by matching the registration information of the system. The first type of sentences is that with 'continuous words' CCE. We can recognize these 'continuous words' CCE by using Maximum Matching Method. The second type of sentences is that with CCE of 'comma separate' structure. In this case the system will match the 'in front part' of the comma, the comma, and the 'behind part' of the CCE orderly. The third type of sentences is that with 'Inserted Words' CCE. The system will recognize these CCE by matching the every part of them orderly. In match processing, if there are two or more than two word serials can be matched with registration information of CCE, and at the same time if these word serials lap over each other, then the longest words serial will be taken as the candidate CCE in our system.

3.4 Correction of POS Tags

By our test, we have found some POS tags of CCE are incorrect by using ICTCLAS. Therefore, we have corrected wrong POS tags of CCE by TBL (Transformation-Based Error-Driven Learning) method. The TBL method is a kind of statistic method which was proposed by Eric Brill in 1995 [9]. The TBL method can provide high precision of 98% and 95% for POS tags and chunk parsing respectively in English [10][11]. Recently the TBL method is also used in Chinese processing. And the satisfied results of previous works show that the TBL method is effective for Chinese processing [12][13]. There are three necessary requisite of using the TBL method. They are a manually annotated database, an initial annotation program and a template of rule module. Firstly, the unannotated text is passed through an initial-state annotator giving an initial POS tags. Then the initial POS tags result is compared with the truth. A manually annotated corpus is used as our reference for truth. Based on this comparison, we can get a serial of candidature rules. By using the evaluation function, every candidature rule is given a score. We consider that the rule which was given the highest score is the best transformation rule. Next, the best transformation rule is used to correct POS tags of annotated text. Finally, the fore-mentioned processing is repeated until processing meet the finish condition. The detail algorithm will be explained in the following steps.

Initial POS Tags. We delete all the POS tags of CCE in training data-base then annotate them again by using ICTCLAS. The annotation of ICTCLAS is regarded as the basic results of processing.

Generation of Candidature Rules. Candidature rules are generated from the wrong POS tags of CCE. Generation condition of the candidature rules is context environment basically.

Acquisition of Transformation Rules. This is as follows:

(1) Evaluation Function: The evaluation standard for the candidature rule is the improvement of right recognition rate. The unannotated database is processed by the initial-state annotator, and this result in an annotated corpus with errors, determined by comparing the output of the initial-state annotator with the manually derived annotations for this corpus. Next, we apply each of the possible transformations in turn and score the resulting annotated corpus. At each interaction of learning, the transformation is found whose application results in the highest score according to the objective function being used, that transformation is then added to the ordered transformation list and the training corpus is updated by applying the learned transformation. We have provided the following formula of evaluation function: F(r)=C(r)−E(r). In this formula, 'r' is the rule. 'F' is the evaluation function. C(r) is the correct numbers which is obtained by using 'r', E(r) is the error numbers which is obtained by using 'r'.

(2) The End of Learning Process: Learning continues until no transformation can be found whose application results in an improvement to the annotated corpus. In our system, when $F(r) < 1$, learning process will be finished.

(3) Transformation Rules: We have obtained 156 transformation rules by the test. These transformation rules can be classified into three kinds. The first kind of transformation rules is based on POS tags. We use the rule module to describe them. In the rule module, 'P' is POS tag. 'T' is a word. 'PN' is the current POS tag of the word. 'P1' and 'P2'are POS tags of the first word and the second word which are located in the left side of the current word. 'P_1' and 'P_2' are POS tags of the first word and the second word which are located in the right side of the current word. We give a transformation rule and a correlative example as follows:

if {P1P2 is m/q && PN is not n} then { P of T from PN to n};

Ex. 1: '*Ta1/r Xiang4/v Yi4/m Zhi1/q Chu1Shan1/v Hu3/n.*' (He is brave and vigorous like a tiger.) Ex. 1 is the processing result of ICTCLAS. In this example, '*Chu1Shan1/v Hu3/n*' and CCE of '*Chu1Shan1/n Hu3/n* (NP+NP)' have the same word serial, but their POS tag of the word '*Chu1Shan1*' are different. According to fore-mentioned transformation rule, we can correct the POS tag of '*Chu1Shan1*' from 'v'(verb) to 'n'(noun) easily.

If a character serial can be segmented as one word by ICTCLAS, we can transform POS tag of this segmented word from current POS tag to 'cv' (conventional expression). This kind of transformation rule can be described as follows('P' is POS tag. 'K' is a certain characters serial. 'PN' is the current POS tag. 'DC' is characters serial of registration CCE):

if {K can match DC && PN is not cv} then { P of K from PN to cv};

Ex. 2: '*Ta1/r De/u Hu2Li3Wei3Ba1/i Bei4/p Zhua1Zhu4/v Le/y.*' (The despicable fact of his action has been found.) In example 2, '*Hu2Li3Wei3Ba1*' is a characters serial of CCE. But the POS tag is not 'cv'. We will transform its current POS tag from 'i' to 'cv'.

The third kind of transformation rule is based on characters of CCE. For example:

Ex. 3:'*Da4Jia1/r Dou1/d Cheng2/v Lao3Lao3/n Bu4/d Teng2/a, Jiu4 Jiu4/n Bu4/d Ai4/v Le/y.*"(Everyone has been ignored.) Example 3 includes the CCE '*Lao3Lao3/n Bu4/d Teng2/v, Jiu4Jiu4/n Bu4/d Ai4/v*'. But according to analysis result of ICTCLAS, the POS tag of word '*Teng2*' is 'a'(adjective). We can not correct its POS tag according to syntactic environment in this sentence. In this occasion, we can correct its POS tag by using characters of CCE. Its transformation rule module can be described as follow:

if { K can match DC && P1P2PN + comma + P_1P_2P_3 is n/d/a + comma + n/d/v} then { P of T from PN to v};

3.5 Extraction of CCE

The word with POS tag 'cv'. The word what has POS tag 'cv' will be recognized as CCE. We can extract it directly.

Ex. 1: '*Na4/r Jian4/q Shi4Qing2/n Yi3/d [Ba1Jiu3Bu4Li2Shi2/cv] Le/u.*' (That thing is near success.)

The CCE of 'Comma Separate' Structure. For the CCE of 'comma separate' structure, the extraction conditions are 'comma', words serial and POS of every word. If these conditions can be satisfied, we can extract this kind of CCE from sentences.

Ex. 2:'*Ta1/r Yi4 Sheng1/n Ke3Yi3/v Jiao4/v Cheng2/v Ye3/d Xiao1/nr He2/nr,Bai4/v Ye3/d Xiao1/nr He2/nr.*'(Both his success and failure are because of that person.)

'Continuous Words' CCE. The recognition knowledge of 'continuous words' CCE consist of attributes and environments of the CCE. The attribute of CCE include the words, the order and POS tags of the words. Environments of CCE indicate the attributes the words those are located in front or back of the CCE. (1) 'Noun Phrase' CCE: There are three types of 'noun phrase' CCE in our system: 'character De' structure CCE, 'attribute-headword' structure CCE, and 'parataxis' structure CCE. In the types of 'character De' structure only four CCEs have been collected. The structure of them are: 'verb + noun+(noun or proclitic word)+De'. According to semantic relations, the CCE with 'character De' structure can be extracted by its attributes.

Ex. 3: '*Ta1/r Shi4/v [Na2/v Bi2Gan3/n Zi3/k De/b].*'(He is an intellect.)

Comparing with 'character De' structure, the other two types of 'noun-phrase' CCE are far more complex. By analysis of the 'noun-phrase' CCE with 'attribute-headword' and 'parataxis' structure, we have found that there are five detail situations in both of these two types: 'noun + noun' structure, 'quantifier

+ noun' structure, 'noun + noun + noun' structure, 'quantifier + noun + quantifier + noun' structure and 'noun + prclitic' structure. Based on these different structures, we can generate five kinds of POS templates correspondingly. They are 'n-n', '(q)-m-n', 'n-n-n', '(q)-m-n-(q)-m-n', and 'n-k'. Thus the POS expansion templates can be obtained by combination of the interpunction and POS tag of the first forward or backward word next to the original templates. We have obtained 327 POS expansion templates totally. (a part of them is shown in table 4.) Consequently, we can extract 'attribute-headword' structure CCE and 'parataxis' structure CCE by these expansion POS templates. For instance, the CCE '*Hou4Qin2 Bu4Zhang3*' can be extracted by No28 expansion template.

Example 4: '*Ta1/r Zai4/p Jia1/n Zuo4/v [Hou4Qin2/n Bu4Zhang3/n] Yi3 Jing1/d San1/m Nian2/q Le/y.* '(He has been a housekeeper for three years.)

Table 4. Expansion POS templates

No.	P1	POS template	P_1		No.	P1	POS template	P_1
1	m+	n+n	+w		42	a+	n+n	+w
2	m+	n+n	+v		43	a+	n+n	+v
...	...	...	...		44	a+	n+n	+c
26	v+	n+n	+w		...	...	...	...
27	v+	n+n	+v		67	u+	n+n	+w
28	v+	n+n	+d		68	u+	n+n	+v
...	...	...	...		...	...	...	...

(2) 'Verbal Phrase' CCE: Four types 'verbal phrase' CCE have been collected in our system. There are 'predicate-object' structure CCE, 'adverb-headword' structure CCE, 'continuous predicates' structure CCE and 'predicate-complement' structure CCE. In this paper the 'verbal phrase' CCE has been extracted by their structures, verb attributes or context environment correspondingly.

Some 'verb phrase' CCE, especially CCE with 'adverb-headword' and 'predicate-complement' structure, can be extracted only by their attributes. For the adverbial modifier and complement in them case can be taken as the close adjunctive constituents here. (as ex.5,6). Furthermore, some 'continuous predicates' structure CCE with obvious structural symmetry and semantic correlation can be extracted by their attributes too (as ex.7). Besides, in the case of 'continuous predicates' structure CCE, because the last word of them are intransitive verbs, so the boundary of these CCE can be recognized by attributes of the verbs(as ex. 8).

Ex. 5: '*Ta1/r You3/v Ge4/q [Bu4/d Cheng2Qi4/v] De/b Xiao3Zi3/n.*' (He has a vain son.)

Ex. 6: '*Na4/r Wei4/q Lao3Xiong1/n You3/v Dian3/q [Huo2/v De2/u Bu2Nai4 Fan2/a] Le/y.*' (It is seem that the man do not want to live anymore.)

Ex. 7: '*Li3/nr Xian1Sheng1/n Shi4/v [Chi1/v Ming2/a Bu4/d Chi1/v An4/a] De/u Ren2/n.*' (Mr. Li would rather fight face to face than infighting stealthily.)

Ex. 8: '*Bu4Zhang3/n Bei4/p [Na2/v Xia4Ma3/v] Le/u.*' (The Minister has been dismissed.)

It is very difficult to recognize the linguistic constituent of the last noun in some CCE which with 'continuous predicates' (the last word is noun) and 'predicate-object' structures (as ex.9). According to composing rules of Chinese phrase, generally, whether a linguistic constituent is an object or not can be judged by two necessary conditions [14]. One of condition is that whether the linguistic constituent is an object of action. Based on combinability of the last noun and the words behind this noun, we can give some strict limited conditions to judge the back boundary of the CCE with 'continuous predicates' or 'predicate-object' structures. That is: if a noun and the word behind it can meet the limited conditions we given, the noun can be considered as a part of CCE.

Ex. 9: '*Gong1Si1/n Xu1Yao4/v Yi4/m Qun2/q Neng2/v [Da3/v Tian1Xia4/n] De/u Ren2/n.*' (The Company needs assiduous people.)

(3) 'Clause' CCE: According to structural symmetry and semantic correlation of 'Clause' CCE, the boundary of most 'Clause' CCE can be recognized by their attributes directly (as ex. 10, 11).

Ex. 10: '*She4Hui4/n Li3/f You3/v Bu4Shao3/m [Da4Chong2/n Chi1/v Xiao3 Chong2/n] De1/b Shi4/n.*' (In human society, there are many things operating as the law of jungle.)

Ex. 11: '*Bu4Shao3/m Ren2/n Dou1/d [Shen1/ng Zai4/p Fu2/n Zhong1/f Bu4Zhi1/v Fu2/n].*' (A lot of people neglect the happiness that they have owned.)

(4) 'Special Structure' CCE: 'Special structure' CCE indicate the CCE those disobey the Chinese grammar. In this case, it is impossible to recognize their boundary by context environment they are in. (18 CCE with 'special structure' have been collected in our work.) But the word serials of these CCE are very unique, so the possibility of co-occurrence of these words is very high correspondingly. As ex. 12 shows, the 'special structure' CCE is extracted by their attributes (the words and the order of these words).

Ex. 12: '*Ta1/r [San1/m Xia4/f Wu3/m Chu2/v Er4/m] Jiu4/d Ba3/p Na4/r Jian4/q Shi4/n Gan4/v Wan2/v Le/y.*'(He finished that work very quickly.)

'Inserted Words' CCE. These is as follows:

(1) 'Predicate-Object' Structure CCE: 'predicate-object' structure CCE with inserted words can be divided into four parts. They are verb (DC), verbal complement (DB), DingYu (DY) and object (BY). DingYu indicates the linguistic constituent that can be used to modify the noun object. The structure of this kind of 'Inserted Words' CCE as follows: DC-DB-DY-BY. Firstly the words, words order and POS tag of DC and BY can be confirmed. Then by matching corresponding registration information of the system, the DB and DY can be confirmed. Next a 'predicate-object' structure CCE which was inserted the DB and DY can finally be confirmed.

Based on analysis of Chinese verbal complement, the verbal complement can be classified into four categories. They are possible (or impossible) complement, result complement, direction complement and movement complement. In table 5, some of DB which be inserted in 'predicate-object' CCE are shown.

The composition of DingYu is very complex. The general components of DingYu include quantifier, pronoun, noun, adjective, adverb, conjunction, onomatopoeic word and the auxiliary word 'De' etc. Besides, some complex phrases (such as 'subject-predicate' structure) and sentences can be considered as DingYu to modify noun too. In our database about 97% DingYu have relative simple structures. Thus the majority of DingYu with 'predicate-object' structure CCE can be recognized by their structures. In our work the POS templates were given to judge DingYu. When the POS tag order of DY matches the given POS templates, the DY is recognized successfully. In table 6, some POS templates of DY are described.

Ex. 13: '*Wo3/r Yi3/d [Da3/v Le/u Yi2 Ge4/m Piao4 Liang4/a De/u Fan1 Shen1 Zhang4/n].*'(I have changed completely.)

Table 5. Categories of Chinese verbal complement

No.	category	a part of inserted words
1	possible (or impossible) complement	*De2, De2Le, De2Zhao2, De2Hao3, De2Shang4, De2Xia4, De2Zhu4, De2Chu1, Bu4Le, Bu4Zhao2, Bu4Hao3, Bu4Shang4, Bu4Xia4, Bu4Zhu4, Bu4Chu1, ...*
2	result complement	*Chu1, Ru4, Cheng2, Dao4, Wan2, Hui4, Tong2, Jian4, Si3, Da4, Chang2, Duo1, Hao3, Kuai4, Xiao3, Shao3, Jin3, Huai4, Hei1, Zhong4, Ming2, Man3, ...*
3	direction complement	*Shang4, Xia4, Qi3*
4	movement complement	*Le, Zhe, Guo4*

Table 6. The POS templates of DY

No.	POS template of DY		No.	POS templates of DY
1	(q)+m		10	r+c+r+u+a+u
2	a+(u)		11	n+(u)
3	m+a+(u)		12	n+(u)+(q)+m
4	q+m+a+(u)		13	n+(c)+n+(q)+m
5	r+u		14	n+(u)+a+(u)
6	r+u+(q)+m		15	n+(u)+(q)+m+a+(u)
7	r+u+a+(u)		16	d+(d)+a+(u)
8	d+(d)+a+(u)		17	(q)+m+d+(d)+a+(u)
9	r+c+r+u		...	...

(2) 'Subject-Predicate' Structure CCE: The denial adverb and degree adverb are usually inserted in CCE with 'subject-predicate' structure. They can be

recognized by composition rule of Chinese adverbial modifier. The extraction processing is similar to 'Clause' CCE which we mentioned before.

Ex. 14: '*Ta1/r [Jia4Zi3/n Fei1Chang2/d Da4/a].*'(He is very arrogant.)

(3) 'Attribute-Headword ' Structure CCE (noun+noun): CCE with inserted part are usually made of nouns (noun+noun). Here the back noun is modified by the front noun. And the usual inserted part are the auxiliary words '*De*' or '*Zhi1*'. In this case, we can confirm the attributes of two parts which be separated and the inserted auxiliary word firstly. The boundary of the CCE can be recognized by the expansion template (same as expansion template of 'noun phrase' CCE.)

Ex. 15: '*Ta1/r Ru2Tong2/v Yi4/m Zhi1/q [Guo4Jie1/n De/u Lao3Shu3/n].*' (He is a universally condemned person.)

(4) 'Parataxis' Structure CCE: The 'parataxis' structure CCE with inserted part are usually made of nouns ('noun + noun'). The usual inserted words are paratactic conjunctions such as *'He2', 'Yu3', 'You4', 'Gen1', 'Jia1', 'Tong2'* and *'Dui4'*etc. In this case, we can confirm the attributes of two parts which be separated and the inserted conjunction word firstly. The boundary of the CCE can be recognized by the expansion template (same as expansion template of 'noun phrase' CCE.).

Ex. 16:'*[Ji1Mao2/n He2/c Suan4Pi2/n] De/u Xiao3Shi4/n.*'(tiny things, bits and pieces)

4 Experiment

In current research 1,200 hypertext files (about 13,000 sentences and 2,000 CCE) have been tested by our methods. All these test data are collected from internet, correlative books and papers. And the experiment evaluation was carried out by approaches of Recall, Precision and F-measure.

$$Recall = \frac{No.\ of\ extracted\ CCE\ correctly(C)}{No.\ of\ CCE\ in\ the\ test\ sentences(A)} \tag{1}$$

$$Precision = \frac{No.\ of\ extracted\ CCE\ correctly(C)}{No.\ of\ extracted\ as\ CCE(B)} \tag{2}$$

$$F - measure = \frac{Precision\ \times\ Recall\ \times 2}{Precision\ +\ Recall} \tag{3}$$

In our experiment 1,731 multi-word units were extracted as CCE. Among them 1,633 were correct. We have achieved 81.65% in Recall, 94.34% Precision and 87.54% in F-measure. As Table 7 shows. By the experiment results, we have found our extraction approach for CCE are successful. In this work the structural and semantic characters of CCE is taken as a key point of our research. We have also found that the error of word segment and POS tag are the main causes which produce failing Recall in a small part of CCE. The longer CCE is the more POS tag errors will be. So the extraction of long CCE is far more difficult

Table 7. Result of the Experiment

categories of CCE	A	B	C	Recall	Precision	F-measure
Noun Phrase	496	429	419	84.48%	97.67%	90.59%
Verbal Phrase	862	732	699	81.09%	95.49%	87.70%
'Clause'	169	146	125	73.96%	85.62%	79.37%
'special'	23	21	21	91.30%	100%	95.45%
'comma separate'	65	56	56	86.15%	100%	92.56%
'Predicate-Object'	273	248	227	83.15%	91.53%	87.14%
'Subject-Predicate'	57	51	41	71.93%	80.39%	75.93%
'Attribute-Headword'	32	28	26	81.25%	92.86%	86.67%
'Parataxis'	23	20	19	82.61%	95%	88.37%
Total	2,000	1,731	1,633	81.65%	94.34%	87.54%

than short ones. Thus the length of CCE must be taken in consideration in the future work. Besides, there were two causes leading the wrong extraction. One is boundary recognition of CCE. In this experiment, we successfully recognized the boundaries of 'noun phrase' CCE and 'verbal phrase' CCE. But the boundaries of many 'Clause' CCE were failed in recognition. The other cause is semantic judgment of CCE. In the system, the meanings of CCE still can not be judged very well.

5 Conclusion

In this paper we have discussed how to help CSL learners to recognize CCE in Chinese text. And an extraction approach based on the rules and character of CCE has been presented. By the extraction experiment, the results of both Recall and Precision are over 80%. In the future, we will enlarge the analytical quantity of CCE. For improving extraction result, the length information, semantic analysis and the Particularity Words (name, address and thing etc.) of the CCE will be taken into consideration. Furthermore we will advanced the reading support function by relevant technique.

References

1. Li Xinjian, Issues of Research and Standardization of Chinese Idiomatic Expressions, Applied Linguistics, 2002, 55-60
2. Mitsuko, Yamura Takei, Teruaki Aizawa, Miho FUJIWARA, Cognitive and SLA approaches to Computer-Assisted Reading, Making the Invisible Visible, Transaction of Japanese Society for information and System in Education, 2004
3. Susan M. Gass, Larry Selinker, Second Language Acquisition (An Introductory Course), Second Edition, Mahwah, N.J.: Lawrence Erlbaum Associates, 2001

4. F. Ren, Y. Miyazaki, and K. Tochinai: "An Algorithm for Estimating Chinese Supplementary Words in Japanese-Chinese Machine Translation System", Trans. Information Proce. Society of Japan, Vol.32, No.11, (1991-11), pp.1374-1382
5. C.Zong and F.Ren: "Chinese Utterance Segmentation in Spoken Language Translation", Computational Linguistics and Intelligent Text Processing, Ed. Alexander Gelbukh, Springer, LNCS2588, (2003),pp.516-525
6. Patrick Pantel and Dekang Lin: "A Statistic Corpus-Based Term Extractor", Canadian Conference on AI2001, 36-46
7. Tingting He, Jianzhou Liu, Donghong Ji: "A Statistical of Extracting Chinese Multi-Word Units", Journal of Chinese Language and Computing, 2002, 239-247
8. Hideki Isozaki and Hideto Kazawa: "Speeding up Support Vector Machines for Named Entity Recognition", Information Processing Society of Japan, Vol.44, No.3, 2003
9. Eric Brill: Transformation-Based Error-Driven Learning and Nature Language Processing: A Case Study in Part of Speech tagging, CLV21, 1995
10. Eric Brill, Some Advances in Transformation-based Part of Speech Tagging, In: Proceedings of the Twelfth National Conference on Artificial Intelligence, 722-727, 1994
11. Voutilainen. Atro: "NLPTool, a Detector of English Noun Phrases", In Proceedings of the Workshop on Very Large Corpora, ACL, 48-57, 1993
12. Sujian Li, Qun Liu, Chunk Parsing Based on Hybrid Model, in Maosong Sun, Tianshun Yao, Chunfa Yuan, eds., Advances in Computation of Oriental Languages, Proceedings of 20th International Conference on Computer Processing of Oriental Languages, Tsinghua University Press, pp.118-124, 2003
13. Jun Zhao, Changning Huang: "A model based on transformation to recognize Chinese base-NP", Journal of Chinese Information Processing, Vol.13, No.2, 1999
14. Yuehua Liu, Wenyu Pan and Wei Gu: "Chinese Grammer", Foreign Language Teaching and Research Press, 1986
15. Endong Xun, Changning Huang, and Ming Zhou: "A unified statistical model for the identification of English base-NP", ACL-2000: The 38th Annual Meeting of the Association for Computational Linguistics, Hong Kong, 3 - 6 October 2000
16. Shiwen Yu: "specification for the Prase Structure Knowledge-based of Contemporary Chinese", Journal of Chinese Language and Computing, 13(2), 215-226, 2003
17. Maria Chiara Levorate, Barbara Nesi, and Cristina Cacciari: "Reading comprehension and understanding idiomatic expressions: A developmental study", M.C. Levorato et al. l Brian and Language 91 303-314, (2004)
18. Hiroyuki Shinnou and Hitoshi Isanaha: "Automatic Acquisition of Idioms on Lexical Peculiarity", Trans. Information Proce. Society of Japan, Vol.36, No.8, pp.1845-1854, 1995
19. Andrew Hippisley, David Cheng and Khurshid Ahmad: "The head-modifier principle and multilingual term extraction", Cambridge University Press, Nature Language Engineering 11 (2): 129-157. 2005
20. Danny MINN, SANO Hiroshi: "A Study of Japanese Idioms for Learners of Japanese-A Statistic Approach", IPSJ SIJ Technical Report, 55-62, 2001

Speech Repairs in the DIME Corpus

Iván Moreno[a,b], Luis A. Pineda[a]

[a] Instituto de Investigaciones en Matemáticas Aplicadas y en Sistemas (IIMAS, UNAM)
[b] Facultad de Ingeniería, UNAM
imoreno@turing.iimas.unam.mx, luis@leibniz.iimas.unam.mx

Abstract. In this paper the analysis of speech disfluencies and repairs in a task oriented spoken corpus in Spanish, the DIME Corpus DIME 11, and its relation to dialogue segmentation and discourse markers is presented. A method to detect and correct the most common speech disfluencies for speech recognition in this corpus is also presented.

1 Introduction

One of the main aims of natural language processing and speech recognition is to develop computational systems able to engage in task oriented natural language conversations. At the current state of the technology it is possible to construct dialogue systems focused on domain specific tasks in which simple and grammatical language is used. Current speech recognition systems provide a set of weighed hypothesis of what the speaker is supposed to have said and, usually, the hypothesis with the highest weigh is taken for further processing steps; in particular, the parser finds the syntactic structure and semantic representation of this textual input. In the ideal case, sentences are meaningful and well-formed and the parsing process can proceed in the standard pipe-line architecture; however, spontaneous language, commonly used in conversations, exhibits interjections, pauses, repetitions, etc., and also ungrammatical language, that must be dealt with in order to construct useful systems. These spontaneous speech phenomena are called *speech disfluencies*. In order to process the spoken input disfluencies must be *corrected* and this is usually done within the same elocution, as exemplified by the following elocution taken from corpus.

<sil> *la estufa pegar vamos a quitar* <sil> a <sil> *a intercambiar vamos a poner este fregadero esto* <sil> *de este lado de acá y la estufa de este lado de acá* <sil>

<sil> *the stove place let´s take off* <sil> to <sil> *to interchange let´s put this sink this* <sil> *at this side of here and the stove at this side of here* <sil>

The process of obtaining the intended elocution from the elocution containing disfluencies and correct is also called as elocution's correction process or simply, correction process. Speech disfluencies and repairs do not obey grammatical rules and robust speech recognition systems must have a model to detect and correct them in

© A. Gelbukh, S. Torres, I. López (Eds.)
Advances in Artificial Intelligence
Research in Computing Science 20, 2006, pp. 63-74

order to facilitate or even make possible the parsing process; in this paper a case study of disfluencies and repairs appearing in the corpus DIME is presented; also the construction of decision trees to detect disfluencies and repairs is presented, and a simple algorithm to correct speech repetitions, the most common kind of disfluency, is presented too.

2 Phenomena of the Spontaneous Speech

Speech disfluencies and repairs are related to dialogue segmentation and the presence of discourse markers. Unlike written language where the sentence is a well-defined and understood notion, there is not a natural unit of speech. In order to understand and analyze spoken language, the continuous flux of discourse needs to be divided in manageable units that express basic intentions and correspond roughly to units of intonation and meaning (i.e. speech acts). These units are commonly referred to as *utterances* and the process of dividing the discourse into utterances is referred to as segmentation. The segmentation process is aided by words that mark the boundaries and relations between utterances and these words are referred to as *discourse markers*. Discourse markers also help to identify and correct speech disfluencies. In the rest of this section the notions of segmentation, repairs and discourse markers, as well as their relation, are further elaborated.

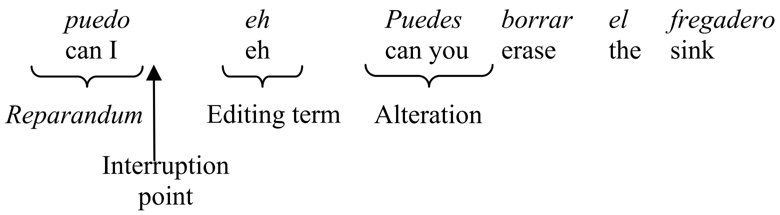
2.1 Segmentation

Human conversation is structured in terms of turns. In each turn a conversational participant expresses one or more intentions, or makes contributions to the effect that his or her beliefs and intentions are understood as intended, maintaining in this way the conversational agreement or common ground; however, the information expressed in each turn may be too large to be considered a natural unit of understanding, and each turn may be divided into one or more basic units or utterances. The proper segmentation of the hearer is fundamental to understand and proceed with the conversation, and also for discourse analysis, as will be seen below.

2.2 Speech Repairs

Speakers make conversational contributions incrementally with the purpose to express intentions; however, it often happens that the expression of an intention process at the same time that the corresponding planning process and speakers may need to review and correct what they have already said. This kind of disfluencies interrupts the normal intention of utterances, and may contain pauses or discourse markers that signal the disfluency and the corresponding repair. Repairs can have different structures; for instance, what has been said can be abandoned completely, to start the idea afresh; speakers can also repeat some words to repair the utterance, or simply introduce a pause, perhaps filled with a word, like an interjection, to have some time to plan how to express the idea. All this forms of disfluenicies and their corresponding correction are referred to as *speech repairs*.

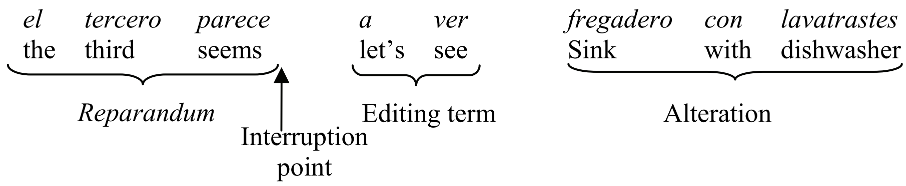
Speech repairs have a standard structure 10; this structure contains three spoken components and one pitch element. The spoken components are called *reparandum*, *editing term* and *alteration*, and the pitch element is the time at which the disfluency is realized, and it is known as *interruption point*. Next, the standard structure of a repair is illustrated:



- *Reparandum*: the part of the utterance which the speaker wants to correct.
- *Interruption point*: the time point at which the disfluency is realizaed, with the corresponding distortion of the normal intonation pattern of the utterance, and it appears at the end of the *reparandum*.
- *Editing term*: a word or a phrase, with a predictable meaning, that is used to fill the pause needed to plan what will be said next; examples editing terms are interjections like *ah*, *mm* or *eh* and also some idiomatic phrases such as *es decir*/this is, *bueno*/well or *perdón*/pardon me
- *Alteration*: the part of the utterance that expresses the right idea and replaces the *reparandum*.

According to the relation between the *reparandum* and the alteration Speech repairs can be classified in three types 10: *fresh starts, modification repairs* and *abridge repairs*, as follows:

- *Fresh Start*: the speaker simply abandons what he just said; there is not relation between *reparandum* and alteration.



- *Modification Repair*: repairs with a non – empty *reparandum*.; most repairs are of this kind. A particular type of modification is the repetition repair, in which one or more words are shared by the *reparandum* and the alteration.

me	*muestras*	*las*	*sillas*	*y*	*bancos*	*que*	*hay*	*que*	*tengas*
show	me	the	chairs	and	banks	that	there are	that	you have

Reparandum (que hay) — Interruption point (after hay) — Alteration (que tengas)

- *Abridged Repair*: repairs without a *reparandum* which only present one or more editing terms.

ahora	*necesito*	*eh*	*una*	*alacena*
now	I need	eh	a	cupboard

Interruption point (after necesito) — Editing term (eh)

2.3 Discourse Markers

Discourse markers are words that have a discourse function instead of their usual lexical category in well-formed sentences, if there is such; words such as *bueno*/well, when it marks that a proposition has been accepted, or as when one answers the telephone; *entonces*/then, when this words establishes a causal relation between two propositions, and *ahora*/now, when the intentions is to mark that a new topic will be addressed, are examples of discourse markers. In such contexts these words do not function as adjectives or adverbs, as they become markers precisely when they abandon such a syntactic function; their purpose is to provide the listener with enough information about the structure of the dialogue and to guide the inference that has to be made to make the communication successful (4, 5, 8, 9).

2.4 Relations Between Phenomena

The phenomena of segmentation, repairs and discourse markers are highly inter-related. Discourse markers together with intonation are important cues for segmentation; discourse markers can also help to detect repairs, because the editing term is generally formed by this kind of markers. In particular, the abridge repair is characterized precisely by the appearance of a discourse marker, normally an interjection. The relation between segmentation and repairs is also complex, as the presence of an interruption point may be confused with an intonation boundary; this problem appears with fresh starts where the alteration can often be taken as a full utterance in itself.

3 The Empirical Study

In the present investigation the DIME corpus 11 was used as empirical base to study disfluencies and repairs. In addition to the tagging levels of this corpus, three extra levels were tagged: (1) speech repairs, (2) Part–Of–Speech (POS) and (3) identification of discourse markers. The word tagging level of the DIME corpus was used as a reference for these three new tagging levels; in addition the *break indices* level of the DIME Corpus, based in the ToBI 1 intonation tagging scheme was used.

3.1 Speech Repairs Level

This tagging level is formed by three sub–levels:

- Structure: A time aligned tagging of *reparadum*, editing term and alteration are marked in this level.
- Type: The type of speech repair (e.g. fresh start, modification or abridge)
- Repair's relations: This level codifies the relations between the words in the different parts of the repair's structure. This level is based on Heeman 9. The tag set for this level is shown in Table 1.

Table 1. Speech repair tag set

Tag	Description
m*i*	Marks that two words are the same
r*i*	Marks that a word replaces another
x*r*	Marks that a word is deleted or inserted
p*i*	Marks a multi-word correspondence, such as the replacement of a pronoun by a longer description
srr<	Marks the onset of the *reparandum* of a fresh start
et	Marks an editing term

3.2 Part–Of–Speech (POS) Level

In this level the lexical category of all words in the utterance are stated. The tag set for this level is based on the analysis of one dialogue of the DIME corpus, and also on using different proposals previously made in the literature both for English and Spanish (2, 6, 7, 9). The final tag set for this level is shown in Table 2.

For instance: *Así/R está*/V *bien*/A? (Is this okay?)

3.3 Discourse Markers Level

As was mentioned, discourse markers are words in which a discursive function predominates over their usual syntactic function; on the basis of this consideration the tag of a discourse marker is formed with the tag of the normal lexical category of the word prefixed with MD (***M**arcador del **D**iscurso*). For instane: *ahora*/MDR *ponme la estufa* (Now, put the stove (for me)).

In addition, three new tags for words that do not have a lexical category were also included as shown in Table 3

Table 2. Part-Of-Speech tag set

Tag	Description
N	Noun
V	Verb
VAM	Auxiliary – Modal Verb
VC	Clitic Verb
A	Adjective
AD	Demonstrative Adjective
TD	Definite Article
TI	Indefinite Article
R	Adverb
RI	Interrogative Adverb
RR	Relative Adverb
RN	Negation Adverb
RA	Affirmation Adverb
P	Pronoun
PD	Demonstrative Pronoun
PR	Relative Pronoun
PI	Interrogative Pronoun
PC	Clitic Pronoun
S	Preposition
C	Conjunction

Table 3. Extra Tags for the discourse markers level

Tags	Description
MDI	Interjection
MDK	Acknowledgment
MDeste	*este*

3.4 The Tagging Task

In the present investigation 8 dialogues of the DIME Corpus were tagged in these three levels; in this exercise 1105 utterance were tagged, out of which 105 presented a repair. Although the speech disfluencies are less than 10% of the data, repairs present characteristic patterns that can be used for the detection and correction task.

4 Detection of Repairs

The empirical data was analyzed in order to find useful variables for the detection and correction of repairs. From this exercise four useful variables were found; these are utterances' duration, number of words, presence of a silence and the type of the dialogue act expressed by the utterance. These variables permitted to identify a detection strategy based on the construction of decision trees.

4.1 Detection Variables

A basic intuition is that utterance with repairs should last longer that the corresponding utterance without a repair. This is corroborated in Figure 1 were it can seen that 77% of the tagged utterances last between 0 and 2000 milliseconds and only 1% of these utterance have a repair. On the other hand, a very large percentage of the remaining 23% presents one or more repairs.

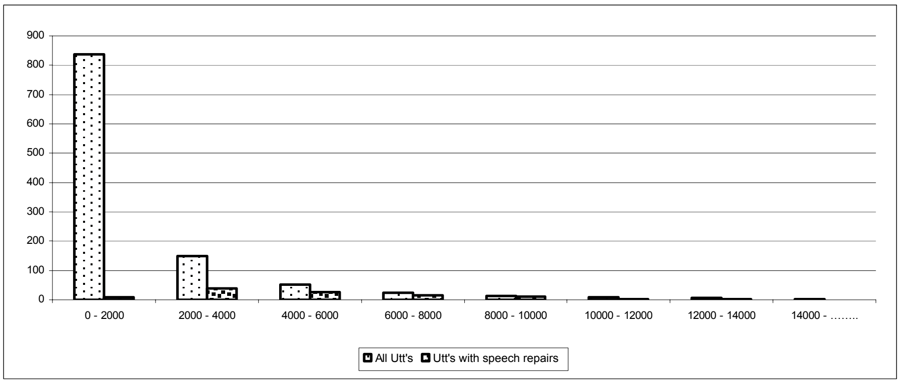


Figure 1. Speech repairs according to duration of the utterance

The second intuition is that an utterance with a repair has a larger number of words than the corresponding utterance without the repairs. This is verified in Figure 2, where utterances are classified in three classes according to their number of words. Region R1 contains utterances with 6 or less words; utterances in R2 contain between 7 and 15 words, and utterances in R3 have more than 15 words. As expected, 79% of all utterances are in R1, but only 2.34% of these have a repair; the critical region is R2 as it has 18.52 % of the utterances, and 30.69% of these have one or more repairs.

Finally, R3 has 3.02% of the utterances and 70% of these have a repair. A further analysis showed that the media of the time duration of utterances with repairs in R2 is longer than the media of the time duration of utterances without repairs in this region for utterances with the same number of words.

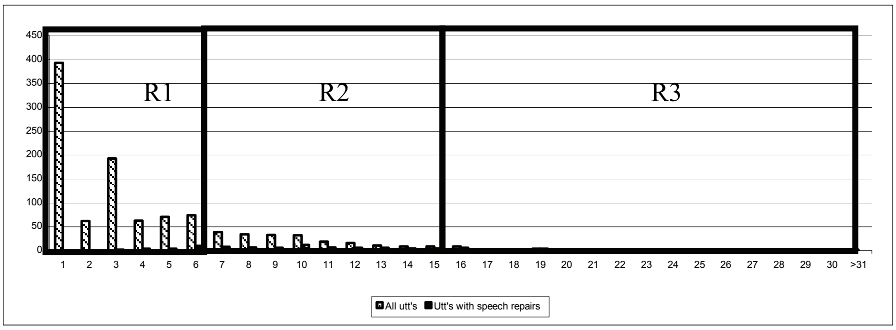


Figure 2. Speech repairs according the number of words in the utterance

The third intuition is that utterances with repairs should have a silence, as it is common that after the interruption point the speaker needs some time to re-elaborate the utterance. This is also verified by the data as 86% of utterances with at least one speech repair present a silence; in addition, a silence usually increases the utterance duration.

Finally, it was observed that speech disfluencies are related to the dialogue act expressed by the utterance; in particular, 64% of the repairs are action directives and 30% are affirms; the intuition behind this observation is that in the case of these two dialogue act types, the speaker is planning along the elocution of the utterance, while other dialogue acts may have a more reactive character.

4.2 Speech Repairs Detection

The four variables identified above suggested a detection strategy based on the construction of a decision tree. For this purpose utterance were classified using CART[1] style decision trees generated with the WEKA[2] tool.

For the construction of the decision tree 105 utterances with and 105 without repairs were taken. The same number of utterances with and without repairs was taken from each dialogue. This strategy helped to balance the process. The resulting decision tree is shown in Figure 3 and the statistics can be observed in Table 4.

---

[1] http://www.salford-systems.com/112.php

[2] http://www.cs.waikato.ac.nz/ml/weka/

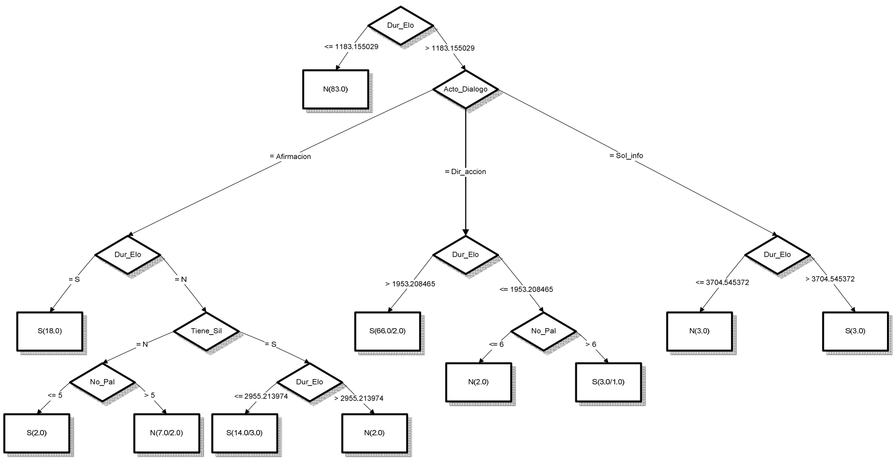


Figure 3: Decision tree to detect a speech repair

Correctly Classified Instances: 86.73%
Kappa: 0.73

	Precision	Recall
Don't have repair	0.94	0.76
Have repair	0.82	0.95

Table 4: Speech repairs detections statistics

As expected, the main classifying attribute was utterance's duration and the second most prominent one was dialogue act type. As the figures in Table 4 show, the classification task is very satisfactory both in terms of precision and recall.

5 Correction Strategy

On the basis of the data analysis a simple strategy for the correction of the most common type of repair in the DIME Corpus, the repetition repair, was identified. In the current tagged data almost 79% of the utterances with one or more repairs are modification repairs; also, almost 77 % of these repairs are repetition repairs. This kind of repairs present, in addition, a simple and regular structure, and a simple heuristics to correct this kind of repairs was identified.

For the definition of the heuristics correction rule two variables were considered: the actual repetition of words and the distance between repeated words, measured in the number of words in between. The intuition is that when a word is repeated in a repair the repeated word appears immediately or in a close proximity of the repeated one. This was verified in the data as about 71% of the utterances with a modification repair present the repetition immediately (i.e. distance = 0) or with only one word in between (i.e. distance = 1); in addition, about 90% of the utterances present a repetition of word with a distance between 0 and 2. This is, most repetitions have a distance of less than 2. It is important to consider that common types of words, like articles and prepositions, are repeated in most utterance regardless whether there is a repair, but repetitions in utterances without repair usually belong to different constituents (e.g. subject and object) and the distance between the repeated words is almost all the time larger than two.

5.1 Correction Algorithm

For the implementation of the detection and correction phase a pipe-line architecture was used; first, all utterances are classified through the decision tree, and those classified as positive are passed through the correction algorithm. The correction algorithm is illustrated next:

1. The input is the utterance produced by the Speech Recognition System; the words are indexed from 1 to N:

eh	*igual*	*con*	*la*	*con*	*la*	*estufa*
eh	same	with	the	with	the	stove

2. The repeated words are identified, and the distance between them is associated to the first instance of the repeated word:

eh	*igual*	*con*	*la*	*con*	*la*	*estufa*
eh	same	with	the	with	the	stove
		1	1			

3. If there are repeated sequences are identified and abstracted as units (i.e. the repetition chunk). In addition, distance between chunks is computed, and the value is associated to the first instance of the chunk.

eh	*igual*	*con*	*la*	*con*	*la*	*estufa*
eh	same	with	the	with	the	stove
		0				

The units (i.e. words or chunks) with distance less or equals than 2 are removed and the remaining units are attached to the remaining instance of the repeated unit as shown below:

eh	*igual*	*con*	*la*	*con*	*la*	*estufa*
eh	same	with	the	with	the	stove
		0				
eh	*igual*	*con*	*la*	*estufa*		
eh	same	with	the	stove		

4. Else (i.e. there are no repetition sequences) if distance is less or equal than 2 remove the words within the distance from the first instance of the repeated word, and also remove the second instance of the repeated word.

entonces	*el*	*primero*	*el*	*tercero*
then	the	first	the	third
	1			
	1			
entonces	*el*	*tercero*		
the	the	third		

The algorithm was tested with the available data and the results are shown in Table 5, as follows:

	Was corrected	Was not
Should be corrected	55%	18%
Should not	4%	23%

Table 5: Speech repairs correction statistics

Table 5 shows that 78% of the utterances were correctly processed and only 22% were handled inappropriately by the heuristics. In particular, the decision classifies all type of repairs, and most fresh starts and abridge repairs where handle correctly by the method. On the other hand, out all repetition repairs 75% were handled correctly by the heuristics and only 25% of these were ignored or badly handle by the method. This can be considered a very promising result.

6 Conclusions

The phenomenon of speech disfluencies is very complex but it has to be faced directly in the construction of speech recognition systems. Heeman 9 provides a very complex method to handle this phenomenon through the definition of multidimensional language models; however, the present study shows that a simple detection strategy paired with a heuristics to correct the most frequent kind of repair can be very effective in the solution of this problem. The present is a preliminary experiment, and we hope that a larger amount of data may be useful to improve the classification rate, to distinguish different kinds of repairs, and to identify specific heuristics to deal with correct other kinds of disfluencies.

In addition additional tagging levels currently available in the DIME Corpus, such as a tonal analysis using INSINT 3 model, or the duration final vowel or consonant extensions can indicate the end of the *reparandum*, and this information can also be very useful for the detection of repairs and their kinds. It is also possible to consider specific discourse markers for the identification and correction of abridged repairs, and it may be possible to do this correction on the fly.

Acknowledgment

We would like to thank support of all members of the DIME project; their tagging work made possible the accomplishment of this work; we would like to mention Ivonne López and Hayde Castellanos for the great help during the POS tag set level definition and Sergio Coria during the decision trees' construction. Also, we thank useful comment and suggestins from James Allen at Rochester University and Joaquím Llisterrí at Universidad Autónoma de Barcelona have given us. The theory and experiments reported in this paper were developed in the context of the DIME-II project, with support of NSF/CONACyT grant 39380-A.

References

1. Beckman, M., Diaz Campos, M., Tevis and J, Morgan, T. (2000). Intonation across Spanish, in the Tones and Break Indices framework, Forbus (14), pp. 9 – 36.
2. D. Farwell, S Helmreich & M. Casper. SPOST: a Spanish Part-of-Speech Targger. http://crl.nmsu.edu/Publications/farwell/far_etal95.html
3. Hirst, Daniel., Di Cristo, A., and Espesser, R. Levels of representation and levels of analysis for the description of intonation systems, In M. Horne (ed) Prosody: Theory and Experiment, Kluwer-Dordrecht, 2000
4. http://es.wikipedia.org/wiki/Marcadores_del_discurso
5. Julian Hirschberg y Diane Litman, Empirical Studies on the Disambiguation of Cue Phrases. Computational Linguistics, 19(3): pp. 501 – 530, 1993.
6. M. Civit & M. A. Martí. Design Principles for Spanish Treebank. En proceedings of the First Workshop on Treebanks and Linguistics Theories (TLT2002), pp. 61 – 77, September 2002.
7. M. Civit. Criterios de Etiquetación y Desambiguación Morfosintáctica de Corpus en Español. PhD thesis. Universidad de Barcelona, 2003.
8. Maria Vittoria Calvi y Giovanna Mapelli, Los marcadores bueno, pues, en fin, en los diccionarios de español e italiano. Artifara, n. 4, sezione Monographica, http://www.artifara.com/rivista4/testi/marcadores.asp
9. P. A. Heeman. Speech Repairs, Intonational Boundaries and Discourse Markers: Modeling Speakers' Utterances in Spoken Dialog. PhD thesis. Rochester University, 1997.
10. Peter A. Heeman, Kyung-ho Loken-Kim, y James F. Allen. Combining the Detection and Correction of Speech Repairs. Proceedings of 4rd International Conference on Spoken Language Processing (ICSLP-96), pp. 358–361, Philadephia, October 1996, also appeared in International Symposium on Spoken Dialogue, 1996, pp. 133-136.
11. Villaseñor, L., Massé, A. & Pineda, L. A. (2001). The DIME Corpus, Memories 3º. Encuentro Internacional de Ciencias de la Computación ENC01, Volume II, C. Zozaya, M. Mejía, P. Noriega y A. Sánchez (eds.), SMCC, Aguascalientes, Ags. México, September, 2001.

Human-Computer Interfaces

Selected Problems of Recognition and Evaluation of Natural Language Commands

Maciej Majewski[1], Wojciech Kacalak[1], Keith Douglas Stuart[2] and David Cuesta Frau[3]

[1] Technical University of Koszalin, Department of Mechanical Engineering
Raclawicka 15-17, 75-620 Koszalin, Poland
{maciej.majewski, wojciech.kacalak}@tu.koszalin.pl
[2] Polytechnic University of Valencia, Department of Applied Linguistics
Camino de Vera, s/n, 46022 Valencia, Spain
kstuart@idm.upv.es
[3] Technological Institute of Informatics, Polytechnic University of Valencia
Plaza Ferrandiz i Carbonell, 2, 03801 Alcoy, Alicante, Spain
dcuesta@disca.upv.es

Abstract. New applications of artificial neural networks are capable of recognition and verification of effects and the safety of commands given by the operator of the technological device. In this paper, a review of selected issues is carried out in relation to estimation of results and safety of the operator's commands as well as the supervision of the process. A view is offered of the complexity of effect analysis and safety assessment of commands given by the operator using neural networks. The first part of the paper introduces a new concept of modern supervising systems of the process using a natural language human-machine interface and discusses general topics and issues. The second part is devoted to a discussion of more specific topics of automatic command verification that has led to interesting new approaches and techniques.

1 Intelligent Two-Way Communication by Voice

The advantages of intelligent two-way voice communication between the technological devices and the operator in Fig. 1 include the following [1,3]:

- More resistance from the operator's errors and more efficient supervising of the process with the chosen level of supervision automation.
- Elimination of scarcities of the typical co-operation between the operator and the technological device.
- Achieving a higher level of organization realization of the technological process equipped with the intelligent two-way voice communication system, which is relevant for its efficiency and production humanization.
- No need of an operator being present at the work stand by the technological device (any distance from the technological device) [7].

© A. Gelbukh, S. Torres, I. López (Eds.)
Advances in Artificial Intelligence
Research in Computing Science 20, 2006, pp. 77-82

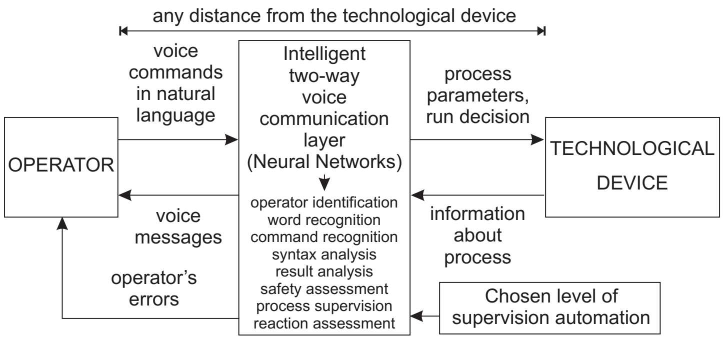


Fig. 1. General scheme of intelligent two-way voice communication between the technological device and the operator

The intelligent two-way voice communication layer in Fig. 2 is equipped with the following intelligent mechanisms: operator identification, recognition of words and commands, command syntax and result analysis, command safety assessment, process supervision, and also reaction assessment [2].

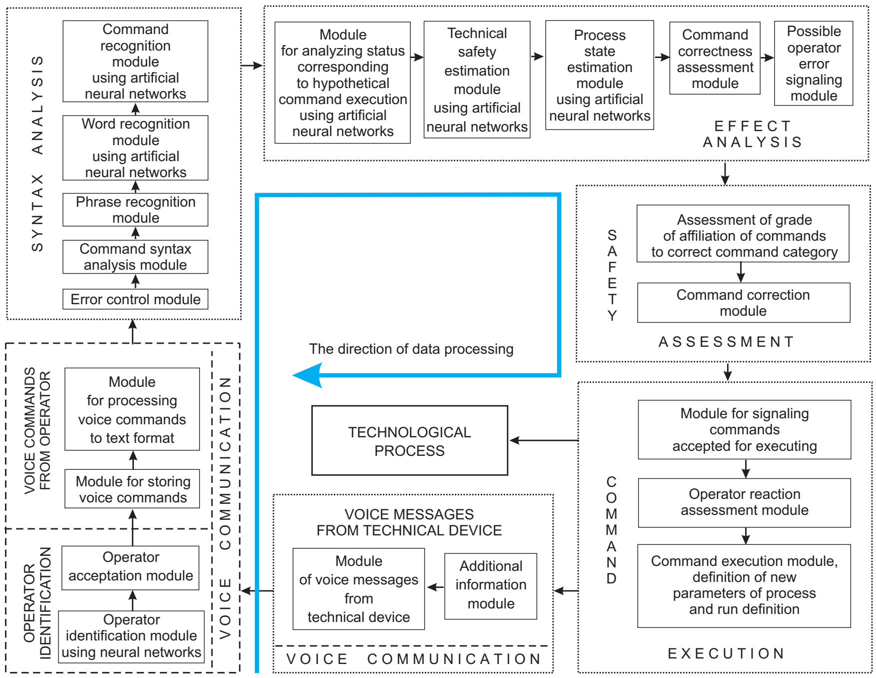


Fig. 2. Scheme of the intelligent layer of two-way voice communication

2 Command Safety Estimation

The effect analysis module, shown in Fig. 3a, makes analysis of the recognised command. The technical safety of the technological device is checked by analysing the state of execution of the commands required to have been done as well as the commands to execute in subsequent decisions. The process parameters to be modified by executing the command are checked and the allowable changes of the parameter values are determined. The analysis of the parameter values is based on the technological process features. The values of the parameter changes are the input signals of the neural network of the process state assessment system. The neurons of the neural network represent solutions to the diagnostics problem. The neural network also makes an estimation of the level of safety of the recognised command. The system for checking the state of the automatic device for grinding small ceramic elements is shown in Fig. 3c, before executing the subsequent commands presented in Fig. 3d. The technological safety assessment system, shown in Fig. 3b, is based on a neural network which is trained with the model of work of the technological device. New values of the process parameters are the input signals of the neural network [6]. As the work result of the system, voice messages from the technological device to the operator about the possibility of executing the command are produced [4,5].

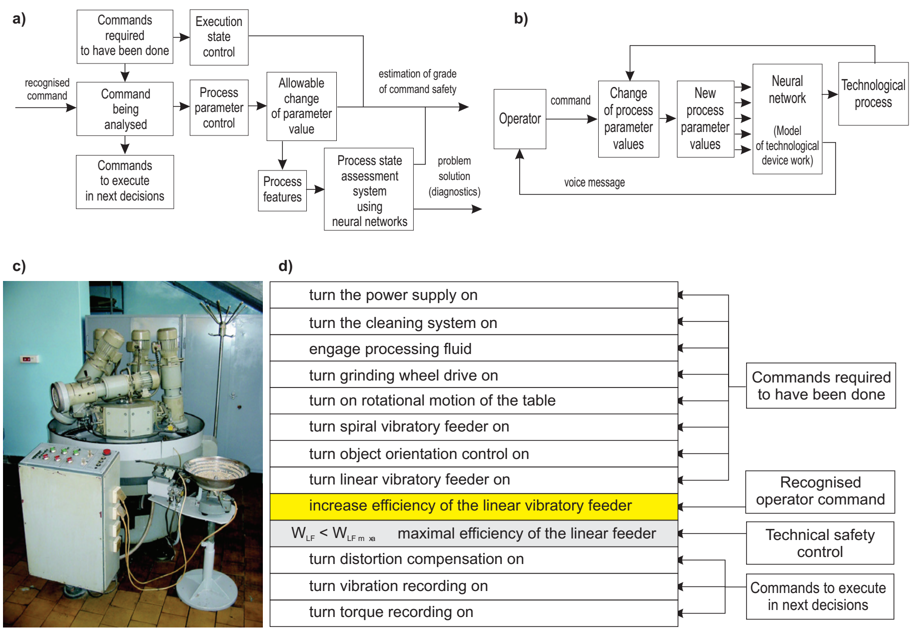


Fig. 3. Scheme of the command effect analysis and safety assessment system

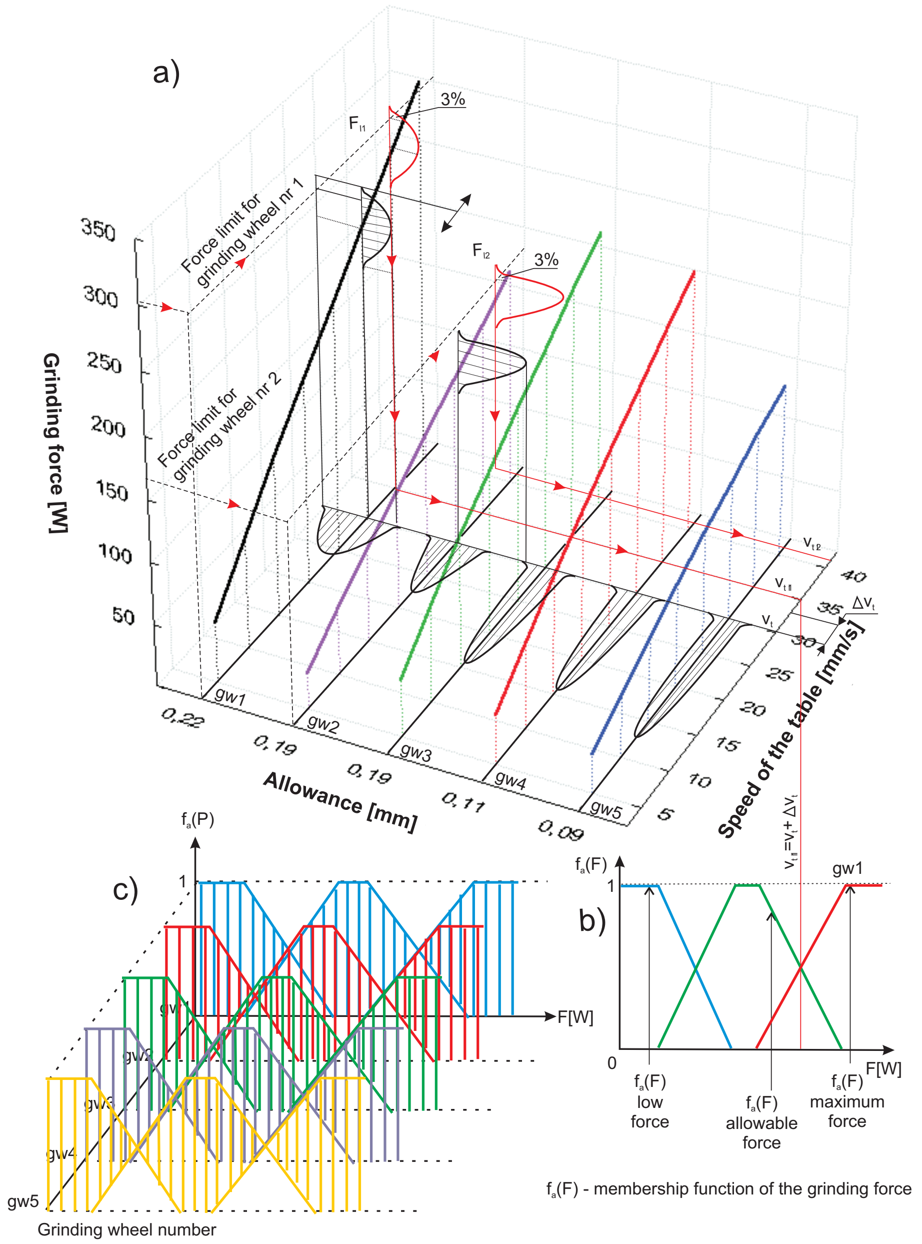


Fig. 4. Algorithm for assessing the technological safety of commands based on the real technological process

An algorithm was created for assessing the technological safety of commands. In Fig. 4, the lines represent force dependence on the grinding process parameters for particular grinding wheels. Based on the specified criteria, the grinding force limit is determined for each grinding wheel. Based on the grinding force limit, the table speed limit is assigned. According to the operator's command, if the increase in speed makes the speed of the table smaller than the smallest speed determined from the force limit for all the grinding wheels, then the command is safe to be executed.

3 Research Results

The simulation set of the technological device diagnostics and the process state assessment, built for creating and training artificial neural networks is shown in Fig. 5a. The neural networks are trained with the model of the technological

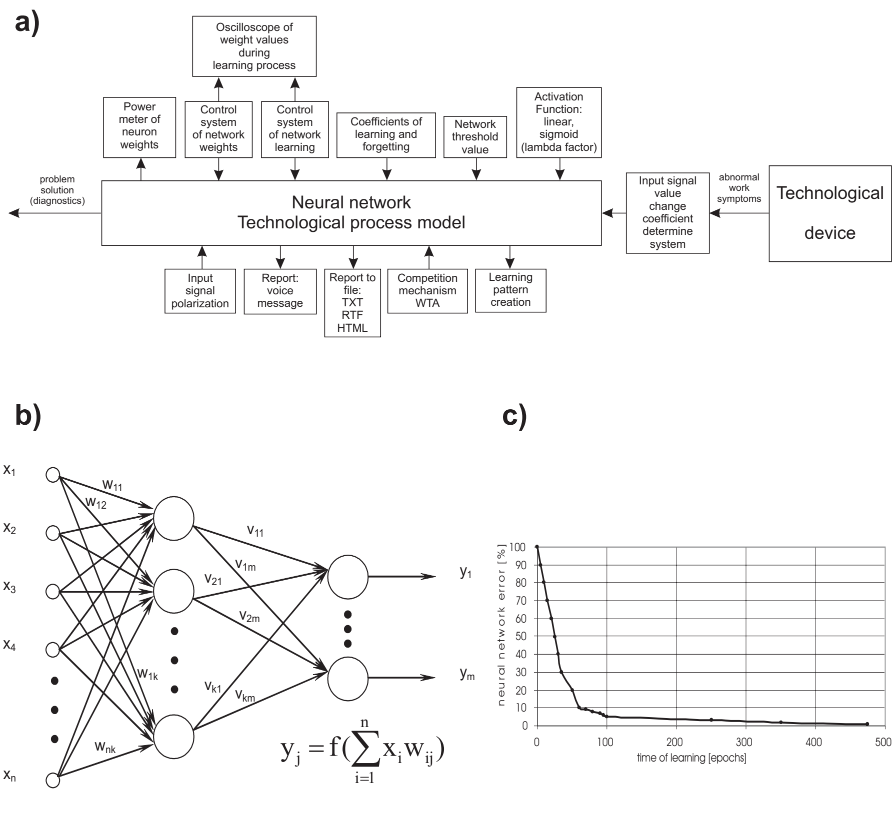


Fig. 5. Neural network simulations of the technological process models, neural network architecture and error rate

process. The applied neural network architecture is presented in Fig. 5b. The networks consist of two layers of neurons with the competitive mechanism.

The ability of the neural network to learn to recognise specific process states depends on the number of learning epochs. The specified time of learning enables the network to minimize the error so that it could work more efficiently. Based on the research, the following conclusion has been reached as shown in Fig. 5c.

Error rate is about 20% at learning time equals 50 epochs and 5% at 100 epochs. The error rate dropped by about 90% after training with 60 series of all patterns.

4 Conclusions and Perspectives

In the automated processes of production, the condition for safe communication between the operator and the technological device is analyzing the state of the technological device and the process before the command is given and using artificial intelligence for assessment of the technological effects and safety of the command. In operations of the automated technological processes, many process states and various commands from the operator to the technological device can be distinguished. A large number of combined technological systems characterize the realization of that process. In complex technological processes, if many parameters are controlled, the operator is not able to analyse a sufficient number of signals and react by manual operations on control buttons. The aim of this research to develop an intelligent layer of two-way voice communication is difficult, but the prognosis of the technology development and its first use shows a significant efficiency in supervision and production humanisation.

References

1. Kacalak, W., Majewski, M.: Recognition and Safety Estimation of Voice Commands in Natural Language Using Artificial Neural Networks. Artificial Neural Networks in Engineering ANNIE 2004 St. Louis. ASME Press New York (2004) 831-836
2. Kacalak, W., Majewski, M.: Selected Problems of Automatic Recognition and Evaluation of Voice Commands in Natural Language Given by the Operator. World Multi-Conference on Systemics, Cybernetics and Informatics, Orlando, Florida (2005)
3. Kacalak, W., Majewski, M.: Intelligent Layer of Two-Way Speech Communication of the Technological Device with the Operator. International Conference on Artificial Intelligence ICAI 2005, Las Vegas, Nevada. CSREA Press (2005) 779-785
4. Majewski, M., Kacalak, W.: Selected Problems of Automatic Evaluation of Commands Using Artificial Neural Networks. International Conference on Intelligent Computing ICIC2005, Hefei Anhui, China (2005) 3605-3614
5. Majewski, M., Kacalak, W., Selected Problems of Automatic Evaluation of Commands Using Artificial Neural Networks. Engineering Mechanics International Journal for Theoretical and Applied Mechanics 12(3) (2005) 185-192
6. Principe, J. C., Euliano, N. R., and Lefebvre, W. C.: Neural and Adaptive Systems: Fundamentals through Simulations. John Wiley and Sons Inc., New York (2000)
7. O'Shaughnessy, D.: Speech Communications: Human and Machine. IEEE Press, New York (2000)

Selected Problems of Intelligent Natural Language Processing

Maciej Majewski[1], Wojciech Kacalak[1], and Keith Douglas Stuart[2]

[1] Technical University of Koszalin, Department of Mechanical Engineering
Raclawicka 15-17, 75-620 Koszalin, Poland
{maciej.majewski, wojciech.kacalak}@tu.koszalin.pl
[2] Polytechnic University of Valencia, Department of Applied Linguistics
Camino de Vera, s/n, 46022 Valencia, Spain
kstuart@idm.upv.es

Abstract. In this paper, a natural language interface is presented which consists of the intelligent mechanisms of human identification, speech recognition, word and command recognition, command syntax and result analysis, command safety assessment, technological process supervision as well as human reaction assessment. In this paper, a review is carried out of selected issues with regards to recognition of speech commands in natural language given by the operator of the technological device. A view is offered of the complexity of the recognition process of the operator's words and commands using neural networks made up of a few layers of neurons. The paper presents research results of speech recognition and automatic recognition of commands in natural language using artificial neural networks.

1 Intelligent Two-Way Speech Communication

If the operator is identified and authorized by the natural language interface in Fig. 1, a command produced in continuous speech is recognized by the speech recognition module and processed in to a text format. Then the recognised text is analysed by the syntax analysis subsystem. The processed command is sent to the word and command recognition modules using artificial neural networks to recognise the command, which is sent to the effect analysis subsystem for analysing the status corresponding to the hypothetical command execution, consecutively assessing the command correctness, estimating the process state and the technical safety, and also possibly signalling the error caused by the operator. The command is also sent to the safety assessment subsystem for assessing the grade of affiliation of the command to the correct command category and making corrections. The command execution subsystem signalises commands accepted for executing, assessing reactions of the operator, defining new parameters of the process and run directives [4]. The subsystem for voice communication produces voice commands to the operator [5].

© A. Gelbukh, S. Torres, I. López (Eds.)
Advances in Artificial Intelligence
Research in Computing Science 20, 2006, pp. 83-88

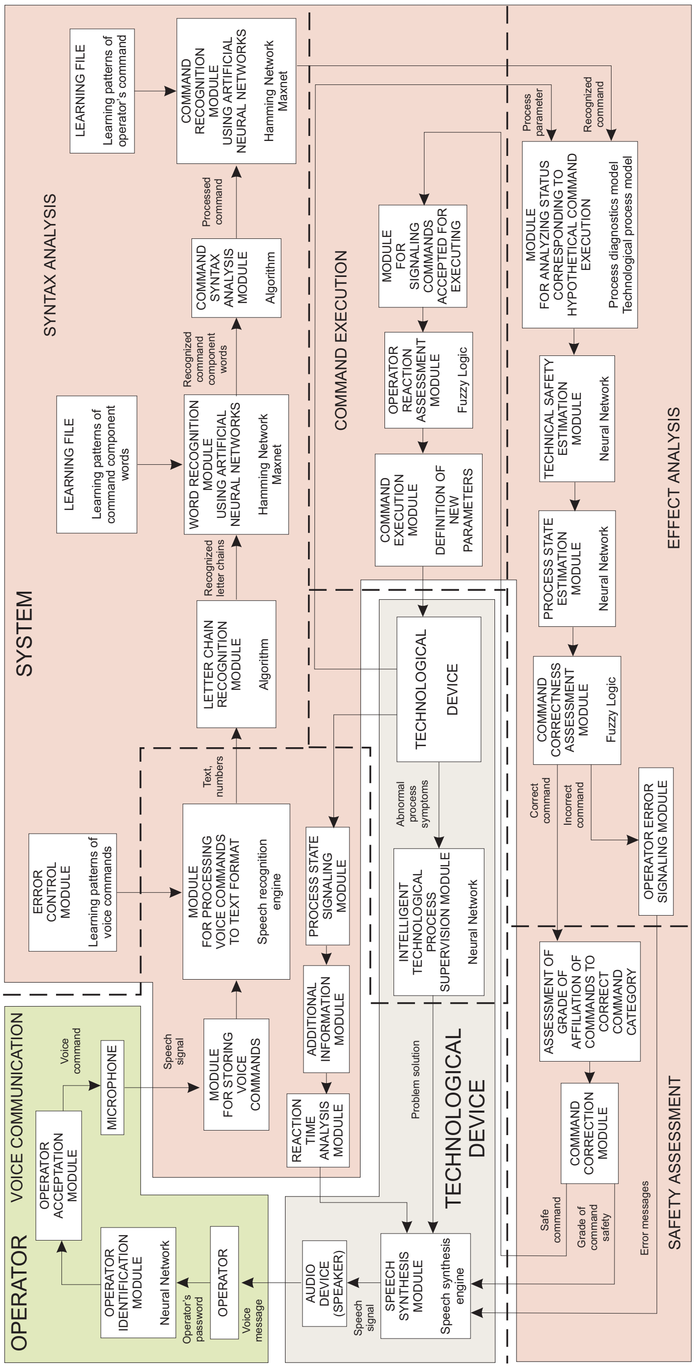


Fig. 1. Architecture of the natural language human-machine interface

2 Recognition of Commands in Natural Language

In the automatic command recognition system shown in Fig. 2, the speech signal is processed to text and numeric values with the module for processing voice commands to text format. The speech recognition engine is a continuous density mixture Gaussian Hidden Markov Model system which uses vector quantization for speeding up the Euclidean distance calculation for probability estimation [1,2]. The system uses context dependent triphonic cross word acoustic models with speaker normalization based on vocal tract length normalization, channel adaptation using mean Cepstral subtraction and speaker adaptation using Maximum Likelihood Linear Regression. The separated words of the text are

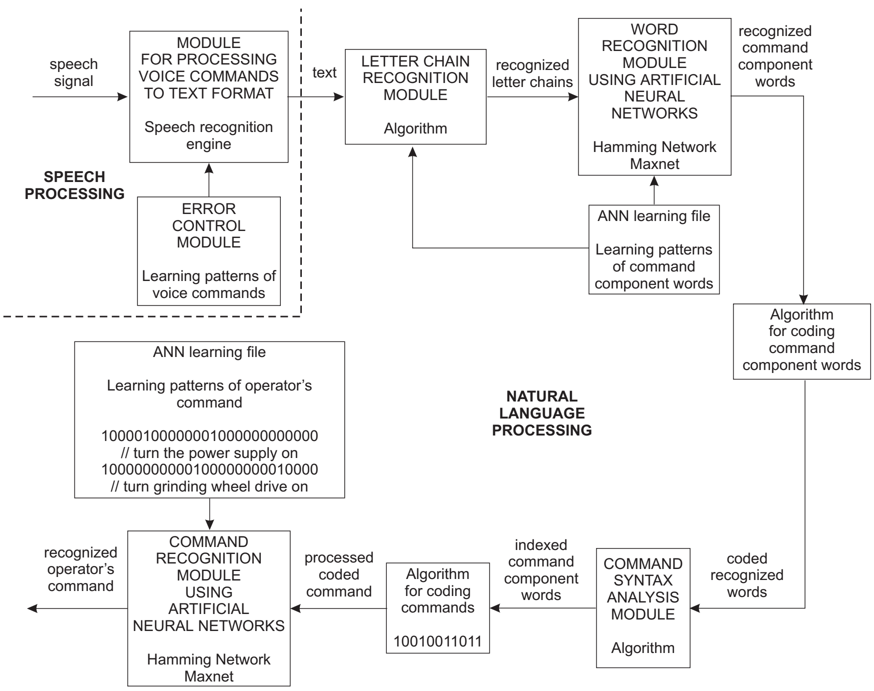


Fig. 2. Scheme of the automatic command recognition system

the input signals of the neural network for recognizing words. The network has a training file containing word patterns. The network recognizes words as the operator's command components, which are represented by its neurons. The recognized words are sent to the algorithm for coding words. Then, the coded words are transferred to the command syntax analysis module. It is equipped with the algorithm for analysing and indexing words. The module indexes words properly and then they are sent to the algorithm for coding commands. The commands are coded as vectors and they are input signals of the command recognition module using neural network. The module uses the 3-layer Hamming neural network in Fig. 3, either to recognize the operator's command or to produce the information that the command is not recognized. The neural network is equipped with a training file containing patterns of possible operator commands [3].

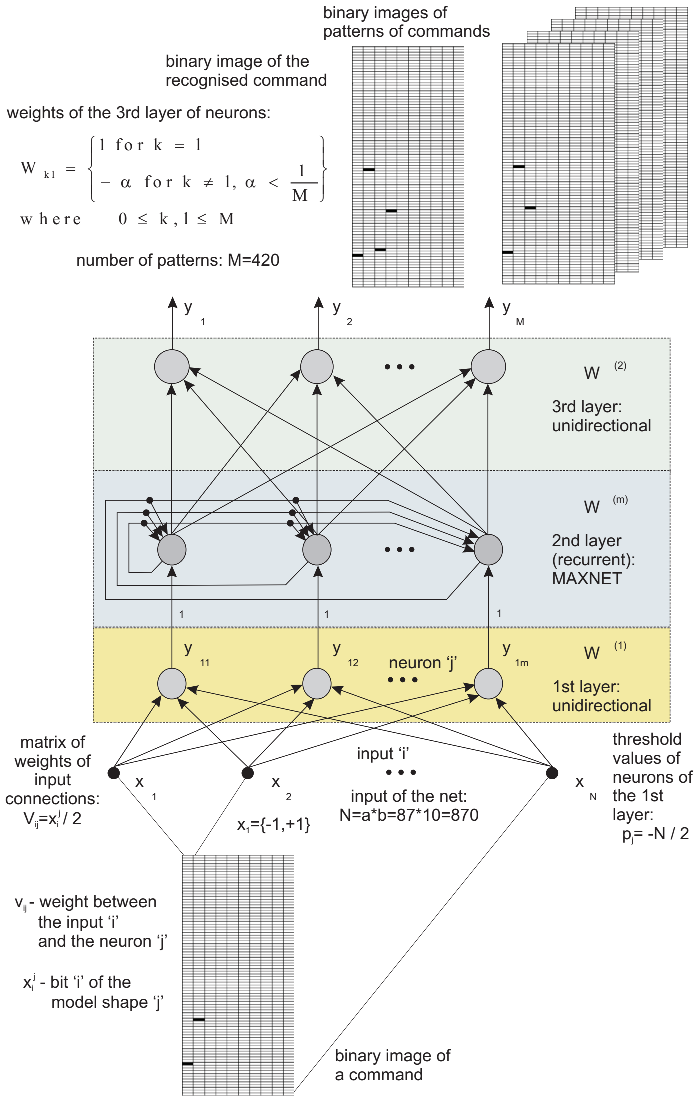


Fig. 3. Scheme of the 3-layer neural network for automatic command recognition

3 Research Results of Automatic Command Recognition

As shown in Fig. 4a, the speech recognition module recognizes 85-90% of the operator's words correctly. As more training of the neural networks is done, accuracy rises to around 95%. For the research on command recognition at different

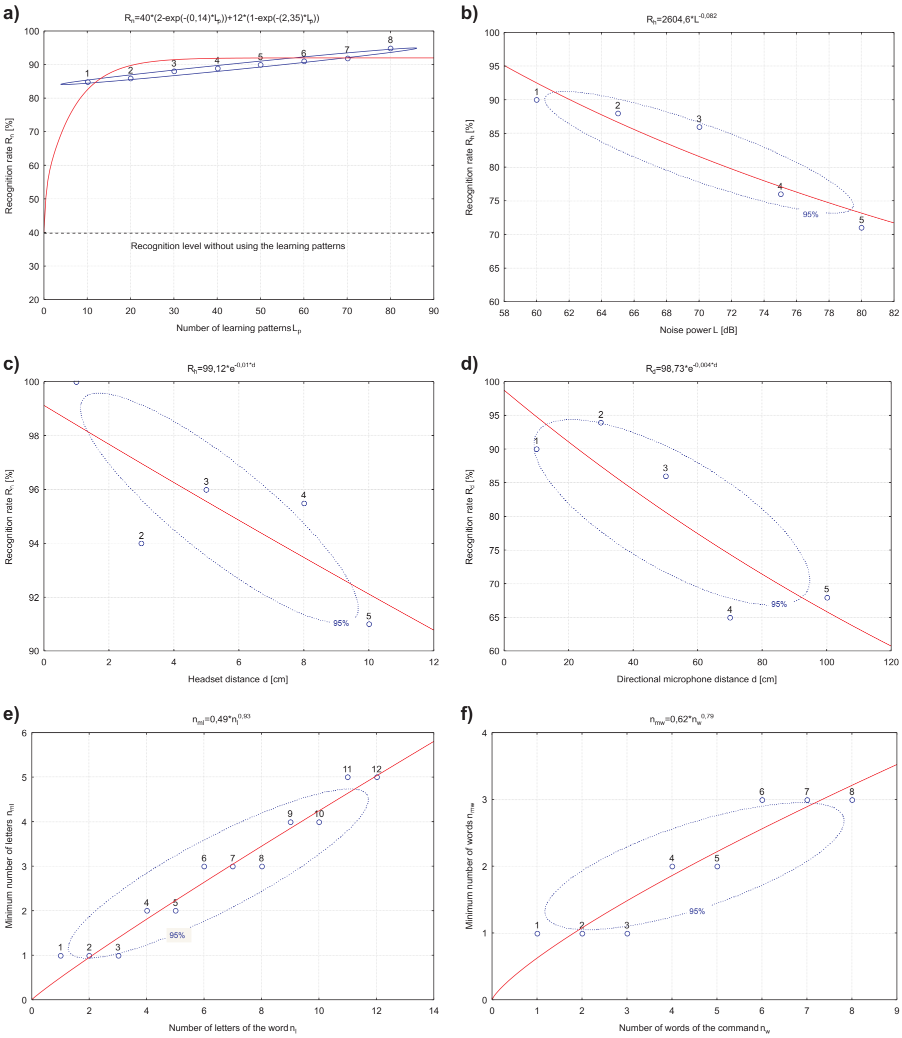


Fig. 4. Speech and command recognition rate

noise power, the microphone used by the operator is the headset microphone. As shown in Fig. 4b, the recognition performance is sensitive to background noise. The recognition rate is about 86% at 70 dB and 71% at 80 dB. Therefore, background noise must be limited while giving the commands. For research on command recognition at different microphone distances, the microphone used by the operator is the headset microphone. As shown in Fig. 4c, the recognition rate decreases when the headset distance increases. The recognition rate dropped by 9% after the headset distance is changed from 1 to 10 cm. Likewise, the research on command recognition at different microphone distances, the microphone used by the operator is the directional microphone. As shown in Fig. 4d, the recognition rate after 50 cm decreases reaching a rate of about 65%. As shown in Fig. 4e, the ability of the neural network to recognise the word depends on the number of letters. The neural network requires the minimal number of letters of the word being recognized as its input signals. As shown in Fig. 4f, the ability of the neural network to recognise the command depends on the number of command component words. Depending on the number of component words of the command, the neural network requires the minimal number of words of the given command as its input signals.

4 Conclusions and Perspectives

The condition of the effectiveness of the presented system is to equip it with mechanisms of command verification and correctness. In operations of the automated technological processes, many process states and various commands from the operator to the technological device can be distinguished. A large number of combined technological systems characterize the realization of that process. In complex technological processes, if many parameters are controlled, the operator is not able to analyze a sufficient number of signals and react by manual operations on control buttons. The aim of this research to develop an intelligent layer of two-way voice communication is difficult, but the prognosis of the technology development and its first use shows a significant efficiency in supervision and production humanization.

References

1. Kacalak, W., Majewski, M.: Automatic Recognition of Voice Commands in Natural Language Using Artificial Neural Networks. Artificial Neural Networks in Engineering ANNIE 2004 Conference, St. Louis. ASME Press, New York (2004) 831-836
2. Majewski, M., Kacalak, W.: Intelligent Human-Machine Voice Communication System. Engineering Mechanics International Journal 12(3) (2005) 193-200
3. Majewski, M., Kacalak, W.: Intelligent Layer of Human-Machine Voice Communication. 9th International Conference KES2005, Melbourne, Australia (2005), Lecture Notes in Computer Science 3683. Springer-Verlag (2005) 930-936
4. Majewski, M., Kacalak, W.: Intelligent Human-Machine Speech Communication System. International Conference on Intelligent Computing ICIC2005, Hefei Anhui, China (2005) 3441-3450
5. O'Shaughnessy, D.: Speech Communications: Human and Machine. IEEE Press, New York (2000)

Automatic Emotional Personality Description using Linguistic Data*

Grigori Sidorov and Noé Alejandro Castro-Sánchez

Natural Language and Text Processing Laboratory,
Center for Research in Computer Science,
National Polytechnic Institute,
Av. Juan Dios Batiz, s/n, Zacatenco, 07738, Mexico City,
Mexico
sidorov@cic.ipn.mx

Abstract. In the paper, we present the system designed for a usage of a psychologist during analysis of a special type of texts – texts of emotional autoreflexive writing. On the basis of linguistic analysis, the psychologist can conclude about emotional state of a person or about a type of his personality. The system is aimed to assist the psychologist. The system has the following features: automatic morphological analysis, calculation of various statistical parameters (frequencies, lexical richness, etc.). The data about words with emotional connotations are given apart because these words represent person's current state. We implemented the mechanism for synchronization of measuring of temperature during text writing and the resulting text. Also, we describe the application of the system in the other field – analysis of political discourse in Mexico.

1 Introduction

One of the main tasks of computer linguistics is providing models for development of applied systems with various kinds of automatic linguistic analysis. Such systems can be applied in diverse areas for solving the problems specific for these areas.

One of the possible areas of application of linguistic data is psychology because it also treats human beings, as well as linguistics. Thus, it is possible to make conclusions about psychological state or about a personality type. For example, in [6] it is claimed that there is a relation between the frequency of usage of auxiliary words, like prepositions, pronouns, articles, etc., and several demographic parameters or personality features. Another example is correlation of suicide tendencies of poets, namely, those, who committed suicide, had more self references in the texts than those who did not [10].

---

* The work was done under partial support of Mexican Government (CONACyT, SNI) and National Polytechnic Institute, Mexico (CGPI, COFAA, PIFI).

One more interesting tendency is related with the detection of intentions to lie or hide information [8]. It is shown that less self-evaluating phrases are used; words with negative emotional evaluation are more frequent; cognitive complex markers are less used.

In the paper, we describe linguistic parameters that are used for automatic emotional personality description. The system that implements this evaluation is described for Spanish language. The proposed approach is rather universal and can be applied in other areas. We applied it for analysis of political discourse of presidential candidates in Mexico.

2 Linguistic Parameters

There are several linguistic parameters for various levels. We calculate standard statistics for text –number of types, number of tokens, number of sentences, medium length of sentence and of paragraph, percentage of vulgarisms, lexical richness (we used two different formulae to calculate it, see below); also, some features of morphological and syntactic structure; and a kind of semantic analysis related with the usage of words from previously prepared lists, like negative, positive, etc.

Let us explain a little bit a concept of lexical richness. Note that it is incorrect to simply calculate the number of lemmas in different texts because it depends non-linearly from the text length, according to the Heaps law [1].

We use two formulae for lexical richness calculation. The first one, index *Brunét* is calculated according to the formula:

$$W = N^{V^{(-0.165)}}$$

where N is text length taken in words, V is the number of different lexemes. Usually, the obtained values belong to the interval from ten to twelve. The lower is the value of this parameter, the greater is the lexical richness.

The other parameter is *Honoré* statistics. It is based on the idea that lexical richness in general is proportional to the number of lexemes used exactly once in the text. The following formula is used:

$$R = \frac{100 * \log N}{1 - (V_1 / V)}$$

where N is the length of the text calculated in words, V is the number of all lexemes used, and V_1 is the number of lexemes with frequency one. In this case, the greater is the resulting value, the greater is the lexical richness.

For calculation of these statistics or for any further analysis, it is important to perform morphological normalization. In our case, we used morphological analyzer for Spanish described in [2].

Apart from lemmatizing, this kind of morphological analyzers allows for calculation of frequencies of grammar forms, for example, verbs in first person, etc.

For resolving the homonymy of parts of speech, we used part of the SVMTool library that has such functionality. The package was trained for Spanish data. It uses the model of Support Vector Machines. The POS tagging accuracy of 96% is claimed.

As far as statistics related to syntactic structures is concerned, we calculate the number of various types of subordinate and coordinate constructions.

For semantic analysis, we take into account the scale related with positive and negative evaluations [9] Corresponding words were chosen in experiments by psychologists.

For example, the following words and their derivates are used:

Table 1. Fragment of the table of emotional words

Words with positive evaluation	Words with negative evaluation	
	Physical threat	Social threat
sincere	asphyxiate	shyness
honest	suffocate	failure
joy	faint away	rejection
kind	infarct	insult
inspired	assault	arrogance
pleasure	suicide	uselessness
calm	illness	awkwardness
contain...	heart...	shame...

3 Application in Psychology

Writing of the texts, where the stress events are modeled in positive perspective, helps to overcome the negative experience. This a special type of text that is called texts of emotional autoreflexive writing ([4], [5], [6], [7]).

The technique of this writing and its further analysis is developed at the psychological faculty of the National University of Mexico in collaboration with psychologists from the United States. The technique implies further thorough analysis of the text by a psychologist. The system allow for automatizing of this analysis, while before it had to be performed manually.

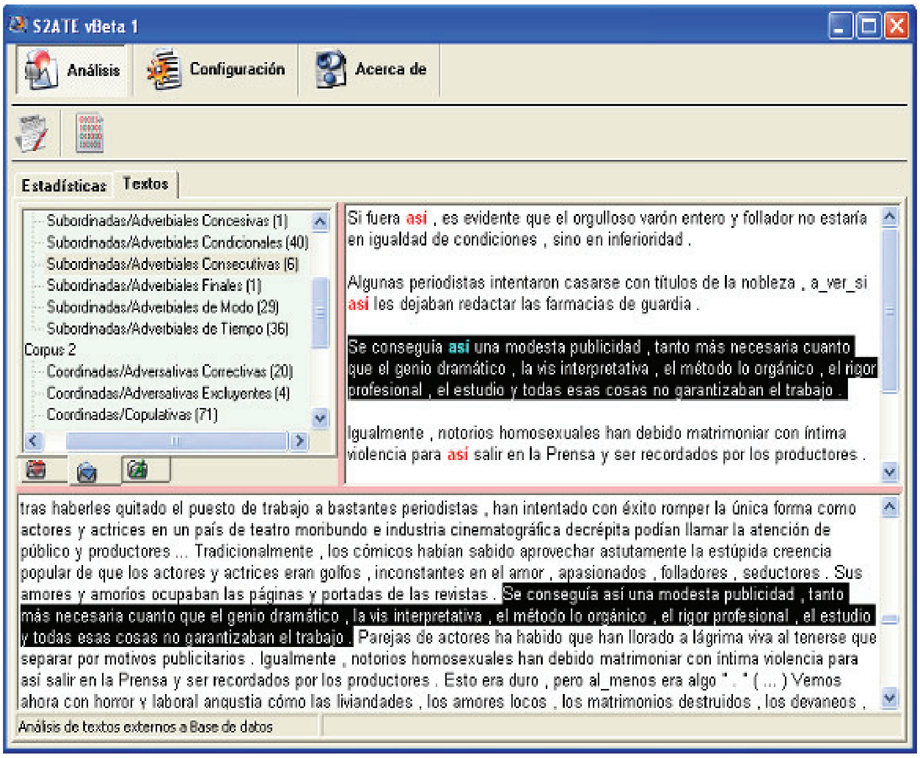
The system contains a database with the information about the persons (patients), their visits, and the corresponding texts. The texts can be grouped into the user-defined corpora or processed independently.

Results of automatic processing are presented for further manual analysis. Besides, there is a possibility to see contexts of usage of a given word, as it is shown in Fig. 1. One context is selected in the list of all contexts. The position of this context in the whole text is selected also, as it is shown at the bottom part of the figure.

There is a possibility of editing of the lists of vulgarisms, positive and negative words.

The system has the possibility of synchronizing with the text a special file that contains the measurements of the temperature taken during text writing for each phrase. Thus, a psychologist can choose a textual fragment and verify at the temperature diagram the corresponding values.

Fig. 1. Contexts of the word "*así*"



4 Another Application: Political Discourse

The developed system is rather universal instrument. We also applied it to analysis of the political discourse of the Mexican presidential candidates. Obviously, the data about the temperature were not available.

For the time being, there are three parties that have chances to win the elections. We can denote them as "party of the right" (PAN), "party of the left" (PRD) and centrist party (PRI). We had access to discourses of candidates of PAN and PRI during their campaign of the 2000, Fox and Labastida; and to the discourses of Lopez-Obrador, the candidate of PRD for the 2006.

Totally, we analyzed 73 texts (41, 16 and 16 correspondingly).

The resulting statistical data are given in Table 2.

Table 2. Statistics of pre-election discourses

	Lopez-Obrador	Labastida	Fox
Totally words	29,720	65,000	53,571
Tokens	9,956	16,434	20,926
Types	7,956	12,213	17,478
Lexical richness (*Honoré*)	447.3	481.3	472.9
Lexical richness (*Brunét*)	9.535	9.334	8.24

As can be seen, according to both statistics of lexical richness, it has the lowest values for the candidate of the "party of the left" (note that for the lexical richness *Brunét*, the lower is the value, the greater is the richness). It can be explained by his orientation to the poorest strata of population. Statistics of lexical richness have controversial values for the candidates of the "party of the right" and centrist party, though the differences are not large. We explain it by the fact that one of candidates touched more themes in his discourses, and, thus, had a chance to use more words with frequency one, which is the crucial factor in one of the statistics (*Honoré*).

Table 3. Usage of some emotional words

Evaluation	Lopez-Obrador	Labastida	Fox
Positive	thank (3) trust (3) security (3)	security (9) honesty (8) calm (4)	sure (6) security (4) thank (1)
Social threat	reject (6) criticism (1) failure (1)	shame (1) fury (1) criticism (1)	insult (4) criticism (1) despite (1)
Physical threat	attack (1)	attack (6) illness (3) wound (3)	accident (1) attack (1) illness (1)

As can be seen, the candidate of the "party of the left" uses less words with positive evaluation, more words with negative social threat and avoids words with negative physical threat.

On the other hand, the candidate of the centrist party uses more words with positive evaluation and with negative physical threat.

This data is preliminary and deserves more detailed processing and analysis. We give it here as an example of the application of the system and its possibilities.

5 Conclusions

We described the system that is designed for helping the psychologist during analysis of a special type of texts – texts of emotional autoreflexive writing. These texts represent stressing situations of life of a person and allow him to blow off; usually they are written by victims of crimes. On the basis of the linguistic analysis of the texts, the psychologist can make conclusions about his emotional state. The system is aimed to help the psychologist, because before the system was developed, this kind of analysis was done manually and took a lot of time.

The system is implemented for Spanish language. It performs automatic morphological analysis, calculates various statistics (frequencies, lexical richness, etc.), calculates apart data for words with emotional connotations, because these words are especially important for psychological analysis. The system has patients' database and a mechanism of synchronization of temperature measurements during text writing with text writing process.

The system is rather universal instrument for linguistic analysis oriented to psychological or social tasks. We applied it without any changes to analysis of political discourse, namely, to the pre-election discourses of Mexican presidential candidates.

References

1. Gelbukh, A. and G. Sidorov. Zipf and Heaps Laws' Coefficients Depend on Language. Lecture Notes in Computer Science N 2004, 2001, Springer-Verlag, pp. 330–333.
2. Gelbukh, A. and G. Sidorov. Approach to construction of automatic morphological analysis systems for inflective languages with little effort. Lecture Notes in Computer Science, N 2588, Springer-Verlag, 2003, pp. 215–220.
3. Gelbukh, A, G. Sidorov, SangYong Han. On Some Optimization Heuristics for Lesk-Like WSD Algorithms. Lecture Notes in Computer Science, N 3513, Springer-Verlag, 2005, pp. 402–405.
4. Domínguez, B., J. Pennebaker, y Y. Olvera. Procedimientos no invasivos para la revelación emocional. Diseño, ejecución y evaluación de la escritura emocional autorreflexiva. 2003.
5. Baños, R. M., S. Quero y C. Botella. Sesgos atencionales en la fobia social medidos mediante dos formatos de la tarea de Stroop emocional (de tarjetas y computarizado) y papel mediador de distintas variables clínicas. International Journal of Clinical and Health Psychology. ISSN 1697-2600. Vol. 5, no. 1, pp. 23-42. 2005.
6. Pennebaker, J., M. Mehl and K. Niederhoffer. Psychological Aspects of Natural Language Use: Our Words, Our Selves. Annual Reviews Psychology, 2003.
7. González, L., *et al*. El impacto psicofisiológico y cognoscitivo de la expresión emocional autorreflexiva sobre la salud. UNAM, México, 2004
8. Newman, M., *et al*. Lying Words: Predicting Deception From Linguistic Styles. Society for Personality and Social Psychology, vol. 29, No. 5, 2003, pp. 665-675.
9. Turney, P. D. and M. L. Littman. Measuring Praise and Criticism: Inference of Semantic Orientation from Association. ACM Transactions on Information Systems, vol. 21, no. 4, pp. 315-346. 2003.
10. Stirman, W. and J. Pennebaker. Word use in the poetry of suicidal and non-suicidal poets. Psychosomatic Medicine, No. 63, 2001, pp. 517-522

A Preliminary Research of Chinese Emotion Classification Model

Yu Zhang[1], Zhuoming Li[1], Fuji Ren[1,2], Shingo Kuroiwa[1],

[1] Faculty of Engineering, The University of Tokushima, 770-8506,
2-1, Minami-josanjima, Tokushima, Japan
{zhangyu, ren, Kuroiwa}@ is.Tokushima-u.ac.jp
lzm@ee.tokushima-u.ac.jp
[2] School of Information Engineering, Beijing University of Posts and Telecommunications, 100876, Beijing, China

Abstract. There have been some studies about spoken natural language dialog, and most of them have successfully been developed within the specified domains. However, current human-computer interfaces only get the data to process their programs. Aiming at developing an affective dialog system, we have been exploring how to incorporate emotional aspects of dialog into existing dialog processing techniques. As a preliminary step toward this goal, we work on making a Chinese emotion classification model which is used to recognize the main affective attribute from a sentence or a text. Finally we have done experiments to evaluate our model.

1 Introduction

The latest scientific findings have indicated that emotions lead an important role in human intelligence, such as decision-making, inter-communication and more. Researchers like Picard have recognized the potential and importance of affect to human-computer interaction, dubbing work in this field as “affective computing” [1].Recent research has placed more emphasis on the recognition of nonverbal information, and has especially focused on emotion reaction. Many kinds of physiological characteristics are used to extract emotions, such as voice, facial expressions, hand gestures, body movements, heartbeat and blood pressure. In order for intelligent user interfaces to make use of user affect, the user’s affective state must invariably first be recognized or sensed. Especially, affective information is pervasive in electronic documents, such as digital news reports, economic reports, e-mail, etc. With the help of natural language processing techniques, emotions can be extracted from textual input by analyzing punctuation, emotional keywords, syntactic structure, and semantic information. [2] It follows that the development of robust textual affect sensing technologies can have a substantial impact in trans-forming today’s socially unkind text-based user interfaces into socially intelligent one.

In this paper, we use verbal information to make a model to acquire emotional information from text with the constructed thesaurus and to recognize the textual sensing of Chinese in a semi-automatic way. We classify the emotion of vocabulary into

© A. Gelbukh, S. Torres, I. López (Eds.)
Advances in Artificial Intelligence
Research in Computing Science 20, 2006, pp. 95-104

12 basic emotion categories, then we give an affect sensing model and give a detailed introduction of all the parts in this model.

This paper will continue as follows, in section 2 we show how emotion words are classified, in section 3 there is an emotion classification model made and we experimented on our system, and gave an evaluation in section 4 and finally we go to the part of conclusion. We believe this research not only can be valuable for Chinese deeper understanding but also can do some help to our Chinese teaching for foreigners.

1.2 Chinese Natural Language Processing

NLP is a subfield of artificial intelligence and linguistics. It studies the problems inherent in the processing and manipulation of natural language, and, natural language understanding devoted to making computers "understand" statements written in human languages. The major tasks of NLP are text to speech, speech recognition natural language generation, machine translation, question answering, information retrieval, information extraction, text-proofing, automatic summarization etc.. In the last few years some researchers begin to pay attention to the emotion recognition in NLP. In American and Japan some researchers have begun to do some work about affect and they have gotten great achievements in natural language such as the famous MIT Media laboratory they work on with English, Tottori University some people work on with Japanese. There is also some progress in Chinese.

Computer application on Chinese NLP is still on the starting stage. The main difficulty is the lack of a comprehensive electronic Chinese thesaurus as a tool to help for analysis. The fundamental element of constructing a system which has the ability to sense the emotional information is the vocabulary that make up a sentence. We consulted many dictionaries for the information of affect such as xiandaihanyu dictionary, hanyuxingrongci dictionary, etc. In these dictionary resources we find the embodied knowledge is almost the same including phonetic, part of speech, semantic, examples and sentential component etc.

2 Emotion Word Classifying

Research on emotion is dogged by ad hoc selections of emotions to work with. There is no agreed benchmark, in the form of a range of emotion terms that a competent system should be able to apply. Without that, it is impossible to assess the performance of emotion detection systems in a meaningful way. In the past emotion has been divided into two categories by some people: pleasure / displeasure. But the classified pleasure/displeasure is too ambiguous to consider the user’s emotion.

In psychology and common use, emotion is an aspect of a person's mental state of being, normally based in or tied to the person's internal (physical) and external (social) sensory feeling. Love, hate, courage, fear, joy, sadness, pleasure and disgust can all be described in both psychological and physiological terms.

In contemporary Chinese the emotion word based on psychology and susceptibility can be divided into 39 kinds [3]. But in all these kinds of emotions there are only part of them being used in our everyday life. We are trying to identify the main compounds (if such they are) that actually occur in everyday life. We have done that by trying to identify a relatively small vocabulary of words that people regard as sufficient to describe most emotional states and events that are likely to occur in everyday life. Using the 7 universal emotion categories defined by Ekman[4], and plus some categories which we have identified important in our research, such as nervous, regretful, love, for the reason of their frequency of being used, and strong sensation. Here we classified emotion into 12 categories. When there is no emotion we call this state equable. The kinds of emotion are shown in the Table 1.

Table 1. 13 kinds of emotion.

Emotion						
Happy Love	Sad Expectant	Fearful Nervous	Disgusted Regretful	Angry Praiseful	Surprised Shy	Equable

3 Emotion Classification Model

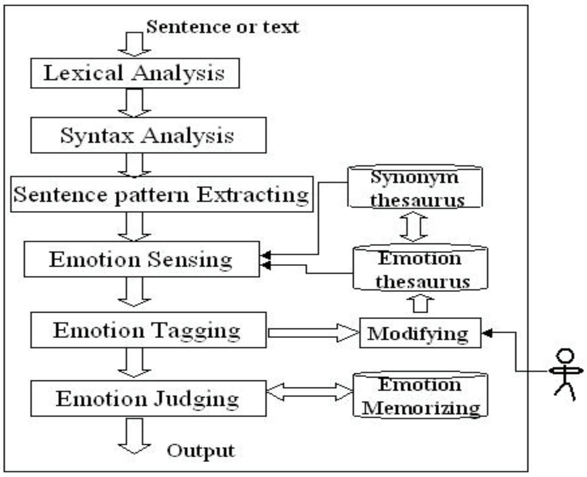


Fig.1 Flow chart of the emotion classification model

The most popular method for performing emotion recognition from text is to detect the appearance of emotional keywords. Generally, not only the word level but also the syntactic and semantic levels may contain emotional information. We make a model of sentence analyzing system. The flow chart is as (Fig.1). In our model there are five parts. They are: Lexical analysis, Syntax analysis, Emotion sensing, Emotion tagging, and Emotion computing. In this model two kinds of database are included in, because they are the basic elements of our research.

3.1 Lexical Analysis

In this part we select the ICTCLAS system for our research. Because it is considered to be one of the best lexical analyzers in the Chinese natural language with the high segmentation accuracy reaching to 97.58%. Basing on the Chinese Lexical Analyzer named ICTCLAS (Institute of Computing Technology, Chinese Lexical Analysis System) of using hierarchical hidden Markov model (HHMM). [5] As everyone know, Hidden Markov model (HMM, L.R. Rabiner, 1989)[6] has become the method of choice for modeling stochastic processes and sequence in natural language processing, because HMM is very rich in mathematical structure and hence can form theoretical basis for use. However, compared with the sophisticated phenomena in natural language, traditional HMM seems hard to use due to the multiplicity of length scales and recursive nature of the sequences. Therefore Shai Fine et al (1998)[7] proposed hierarchical hidden Markov model, which is a recursive and generalized HMM. They apply to word segmentation class-based HMM, which is a generalized approach covering both common words and unknown words. Given a word wi class ci is defined in Fig. 2. Suppose |LEX| to be the lexicon size, then the total number of word classes is |LEX|+9.

This Chinese lexical analysis is based on Shai's work given a formal description of HHMM. For convenience, they also use the negative log probability instead of the proper form. That is:

$$W^{\#} = \underset{W}{\arg\min} \sum_{i=1}^{m} [-\ln p(y_i \mid x_i) - \ln p(x_i \mid x_{i-1})]$$

According to the word class definition, if y_i is listed in lexicon, then x_i is y_i, and p($y_i \mid x_i$) is equal to 1.0. Otherwise, p($y_i \mid x_i$) is probability that class x_i initially activates y_i, and it could be estimated in its child HMM for unknown words recognition.[8]

3.2 Syntax Analysis

This part is inspired by the model of the structural context conditioned probabilistic parsing put forward by the Institute of Computing Technology. There are three probabilistic parsing models, which are successive augmentations of the conventional PCFG(probabilistic context-free grammar) . In this sequence of models outlined, wider and wider structural context is taken as the conditioning events to condition the derivations.

Before emotion classification, there is another important point of our system we have to mention in advance. It is how to construct a database to ensure our system runs smoothly.

3.3 Database Construction

Computer application on Chinese NLP is still in the starting stage. The main difficulty is the lack of a comprehensive electronic Chinese thesaurus as a tool to help for analysis. Thus, to overcome the problem we mentioned above, we intended to construct a machine tractable and readable Chinese emotion thesaurus to help the analysis of Chinese NLP.

Here we constructed a database filled with emotional information of Chinese words in order to help the next part of emotion sensing. On the other hand, due to needing a thesaurus to acquire directly the emotional information of a word we also make this database able to be used singly as a thesaurus. There are about 4800 words collected from various dictionaries such as “ChangYongBaoBianYiXiangJie dictionary”[6] and People’s Daily tagging corpus of the Institute of Computational Linguistics of Peking University. For each emotional word, the corresponding emotion descriptor is manually defined. The emotion descriptor is a set of descriptions of the emotion reactions corresponding to the keywords. Basically, it contains an emotional state label and an image value, which ranges from –2 to 2.

Database Structure. Base on the statistics result in People’s Daily tagging corpus of the Institute of Computational Linguistics of Peking University, the word emotional trends are described and formulized in our dictionary. When constructing the database of emotional information we select the phonetic, part of speech, the image value, and the category of emotion is called emotional attribute. Sometimes the word is not only in one of the 12 categories of emotion, it is in two or three kinds of those emotions. In such an instance every emotion is recorded in our dictionary. Especially in our research, we define the concept of “image value” that is used to express the affective intensity of the emotional words. Many authors agree that emotions can be organized roughly into a two-dimensional space whose axes are evaluation (i.e. how positive or negative the emotion is) and activation (i.e. the level of energy a person experiencing the emotion is likely to display) [9]. That provides a useful basic continuum in which to embed emotion words. Worse than negative or better than positive we defined it as derogatory or commendatory and between them we use neutral as a median . Here are the five levels we have defined:

derogatory-> negative-> neutral-> positive-> commendatory

3.4 Emotion Classification

Emotion classification has four processes consisting of sentence pattern extracting, emotion sensing, emotion tagging and emotion judging. Firstly, the sentence pattern will be extracted from the sentence which had been syntactically analyzed. Then how many emotion words in the sentence will be calculated and with this result the selected words will be tagged with the emotion attribute. At last, there will be a judgment of the sentence, which will express the central emotion of the whole sentence.

Sentence Pattern Extracting. From the results shown in 4.2, we found it is difficult to recognize emotion if the Sentence pattern is not examined. We also found in different sentence patterns, headwords to be extracted are different. For instance, S+V+P pattern “V” is used as copular, such as “shi(是) ” , “kanqilai(看起来) ”, “haoxiang(好像) ”,and so on. So the “P” composition of the sentence will be the main part expressing emotion. We look at this “P” part as our studying point. That is to say the final emotional state is determined based on all emotional keywords which are estimated based on the emotion thesaurus which is a collected emotion corpus we have made. There is also another element in syntax analysis. Conjunction plays a very important role in analyzing emotion sentences. From the grammatical knowledge-base of contemporary Chinese, we classify the connectives into six categories from 97 conjunctions. They are: parataxis, transition, concession, alternative, hypothesis and causation.

Emotion Sensing. Basing the part of “P” mentioned in the section above we can use our database to recognize the emotion category of the sentence in the text. Sometimes there are more than one main word. All these main words are evaluated by their image values, and the average will determine the characteristic of the sentence. The accuracy of this type of sentence can reach 90%. The result of our first step proves our research is effective and feasible. On the other hand, in analyzing complicated sentences relying on The Grammatical Knowledge-base of Contemporary Chinese published by Peking University is also considered to be a good way. By using the grammatical knowledge-base we can get to know which word can be used in which sentence pattern basing on the component of the word.

One more thing we have done is use the synonym thesaurus made by the Information Retrieval Laboratory of HIT (Harbin Institute of Technology University) in order to reduce the query of words, because words can be well classified.

Emotion Tagging. For every possible emotional word and its related intensity, the system also requires particular marks. Unless there is an emotional word in the sentence, all of the sentences will be disregarded. If the emotional word refers to the person himself / herself and it is referring to present continuous, then the parser might generate the output.

After the part of emotion sensing having been analyzed, this part of emotion tagging will apply the result of the Lexical Analysis part which splits sentence into words and detects through the emotion thesaurus to find the corresponding tag category of each word. In the following, tag all the emotional keywords by the image values and the features of emotion. There are five levels in that value form –2 to +2.

In a general way there is more than one emotional keyword in the sentence and all the keywords we have selected have an image value and are arranged in succession according to the order of appearance by the reference frame graph.

Emotion Judging. To extract the emotional state from the text, we assume that every input sentence includes one or more emotional keywords and emotion modification words. The final emotional state is the combination of the three outputs: the emotion descriptors of the emotional keywords, the image values of the emotion keyword and the memory graph of the context sensitivity. Using the image value and memory graph of the keywords which have been tagged over, we can consider the average value of sentences inputted as the sentence's image value. By considering the memory curve of context sensitivity we can also get the emotion features of the sentences. First, we gather all the kinds of the emotion appeared in sentences according to the result of Emotion Tagging. Using the context-sensitive knowledge we can eliminate the most impossible emotion and sort out several kinds of most primary emotions, then finish the part of emotion computing and get the result of computing. Commonly the result is not only the singleness of one kind of emotion, but the admixture of several kinds of emotion.

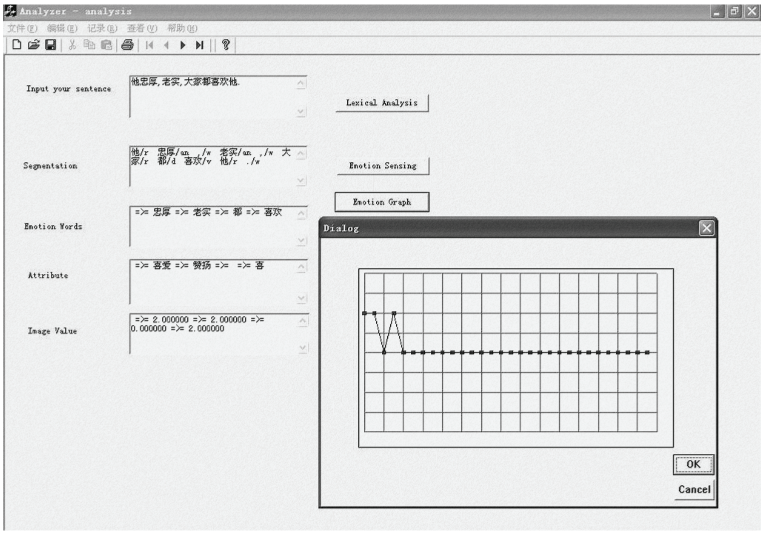


Fig.2. Interface of emotion classification model

The interface of our model is shown in Fig.2 above. People can catch an intuitionistic image from the figure.

4 Experiments and Evaluation

In order to indagate the correct extent of our system we have done two kinds of experiments. One is about the emotion definition extracted from the sentences. The other one investigated the definition of the emotion extracted from the text.

4.1. Emotion classification from sentence

News (http://www.people.com.cn/) (about 410 sentences) pulled out from web is as testing data used to the system which we have constructed, and the experiment that classify emotions from every one sentence was done. Whether the output of our system is correct is checked by person manually. Here is the table:

Table 2. The accuracy of our model on sentence.

	Sentence pattern	Sentences
Sentences (Emotional expression is contained.)	"…shi(是)…" : 120 "…you(有)…" : 30 Others : 50	200
Number of correct answers	"…shi(是)…" : 110 "…you(有)…" : 25 Others : 25	160
Number of incorrect answers	"…shi(是)…" : 10 "…you(有)…" : 5 Others : 15	30
Number of indetermination	"…shi(是)…" : 0 "…you(有)…" : 0 Others : 10	10

From Table 2. we can see the accuracy is divided into three parts according to the sentence pattern. In Chinese sentence pattern "…you…"is like the "…is…"sentence pattern of English. This pattern is used to show speaker's view, opinion, and attitude chiefly. The predicate usually used to express the explanation and the explanation to the subject. When emphasizing or concluding, a hard will is expressed, sometime a soft tone or a euphemistic tone is also shown. The result of the experiment obtained by "…shi…" sentence pattern was very high, and the accuracy can reach to 92%(110/120).

Successful examples:

1. 今天的报告真是太精彩了.(Today's lecture was indeed wonderful.) output-->praiseful
2. 这真是件好事. (This is really a good thing.) output-->happy

Faulty example:

1.这孩子淘气是淘气,可并不是一点儿也不听话. (This child is persuasible although he is naughty.) output-->praiseful correct answer-->love

The meaning of sentence pattern "…you…"in Chinese is same like the pattern of "there is/are "or "sb./sth. has/have…. ". This is also a sentence pattern which expresses the attribute of the subject and has the secondary accuracy reaching to 83.3%.

Successful examples:

1. 教书这个工作很有意义 (Teaching is a meaningful work). output-->praiseful
2. 这家伙有一肚子坏水.(He has lots of evil plans in his brain) output-->disgust

Faulty example:

1. 去年国民收入有了增长. (National income has increased last year.) output-->equable correct answer-->happy

There is a possibility that the judgment is not accurate when the particle is modified by the adverb and put in the last.

From this experiment we can find that the system can not interpret from the character when vague information came out, so there are often sentence that was not able to be understood. The ability of man "Common sense" is necessary.

4.2. Emotion Classification from Text.

Various kinds of information were selected from the internet (http://www.people.com.cn/、http://www.sina.com.cn/) for our experiment. In this experiment we will test the accuracy of emotion classification when the resource are the texts including more sentences than one. In those 250 texts we find three kinds of information stand out from the others.

Table 3. Accuracy of our model on text

Sentences	Kind of text	Number
Randomly selected	News	100
	Business	80
	Story	70
Emotional expression is contained.	News	78
	Business	45
	Story	57
Number of correct answers	News	69
	Business	37
	Story	31
Number of incorrect answers	News	9
	Business	8
	Story	26
Accuracy	News	88.5%
	Business	82.2%
	Story	54.4%

The result of this experiment is also divided into three on the type of the text. From the experiment the highest accuracy is from the news type of the text that can reach to 88.5%. In the successful results, the received accuracy when emotional words in the text which have the same emotion attribute is higher than that emotion caused phenomenon was included in the text. The most difficulty in emotion analyzing text is when there are several kinds of emotion in judging which ones play the strong role but in news and business such trouble like above is few and the alteration of emotion is also few, so the result is not confused by them. It is why the accuracies of them are higher.

5 Conclusion

In this paper we have firstly talked about our emotion thesaurus and how we constructed it, then we using the already constructed thesaurus as our emotion database constructed the emotion classification model which was used to conjecture the emotion one or more in the sentence or text and whether is positive or negative the text or sentence is used to express. At last we have done two experiments and from the results we work on evaluation.

So far we have outlined is in order to prove the model we have tried constructing is feasible and useful although now is on the preliminary stage. This research can be used in special domain such as E-mail quick look user can chose which mail he needs to read firstly. We also think it could be used for the language applications someday in the future.

References

1. Picard, R. W.: Affective Computing, The MIT Press, Mass., 1997.
2. Subasic, P. and A. Huettner: "Affect Analysis of Text Using Fuzzy Semantic Typing," IEEE Transactions on Fussy System, vol. 9, no. 4, 2001, pp.483-496
3. Xiaoying Xu , Jianhua Tao: "Emotion dividing in Chinese emotion system" The 1st Chinese Conference on Affective Computing and Intelligent Interaction(ACII'03) , 8-9 December, 2003 Beijing China pp.199-205, 2003
4. Ekman, P., Friesen, W.V., and Elsworth, P.: (1982) Research foundations In P. Ekman (Ed.), Emotions in the Human Face. London: Cambridge University Press
5. HuaPing Zhang et al "Chinese lexical analysis using Hierarchical Hidden Markov Model" Second SIGHAN workshop affiliated with 41th ACL; Sapporo Japan, July, 2003,pp.63-70
6. L.R. Rabiner. "A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition". Proceedings of IEEE 77(2), pp.257-286
7. Shai Fine, Yoram Singer and Naftali Tishby. "The hierarchical Hidden Markov Model: Analysis and applications. Machine Learning, 32(1), July 1998 "
8. Xianzhen Guo, Chang Yong: BaoBianYiXiangJie dictionary 1996 The Commercial Press
9. Plutchik, R.: The psychology and biology of emotion. Harper Collins, New York, p.58, 1994
10. Changle Zhou , Mental Computing, pp147-176, 2003.3
11. Shiwen Yu, Xuefeng Zhu et al: The Grammatical Knowledge-base of Contemporary Chi nese – A Complete Specification Tsinghua University Press Apr 1998
12. Hugo Liu et al "A Model of Textual Affect Sensing using Real-World Knowledge" Pro ceedings of the 2003 International Conference on Intelligent User Interfaces, IUI 2003, January 12-15, 2003, Miami, FL, USA. ACM 2003, ISBN 1-58113-586-6, pp. 125-132.
13. Cowie, R. Douglas-Cowie, et al "Emotion recognition in human-computer interaction" Signal Processing Magazine, IEEE Jan 2001 Volume 18, pp. 32-80

Information Retrieval

Dimensionality Reduction for Information Retrieval

Franco Rojas López[1], Héctor Jiménez-Salazar[1]
David Pinto[1,2], A. López-López[3]

[1]Facultad de Ciencias de la Computación,
Benemérita Universidad Autónoma de Puebla,
14 sur y Av. San Claudio, Ciudad Universitaria, Edif. 135
Puebla, Pue., México, 72570
{frl99, hgimenezs}@gmail.com
[2]Departamento de Sistemas Informáticos y Computacionales,
Universidad Politécnica de Valencia,
Camino de Vera s/n,
Valencia, España, 46006
davideduardopinto@gmail.com
[3]Laboratorio de Tecnologías del Lenguaje
Instituto Nacional de Astrofísica Óptica y Electrónica
allopez@inaoep.mx

Abstract. This work presents a variation of the traditional text representation based on the vector space model, used in Informational Retrieval. In particular, a representation is proposed, intended to select terms for indexing and weighting them according to their importance. These two tasks are performed taking into account the terms with medium frequency, that have shown an advantage to reveal keywords. The results of experiments using an information retrieval system on the TREC-5 collection show that the proposed representation outperforms term weighting using $tf \cdot idf$, reducing simultaneously the dimensionality of terms to less than 12%.

1 Introduction

Vector Space Model (VSM) was proposed by Salton [10] in the 1970's. This model states a simple way to represent documents of a collection; using vectors with weights according to the terms appearing in each document. Even though several other approaches have been tried, such as the use of representative pairs [7] or the tokens of documents, vector representation based on terms remains a topic of interest, since some other applications of Natural Language Processing (NLP) use it, for instance, text categorization, clustering, summarization and so on.

In Information Retrieval (IR), a commonly used representation is the vector space model. In this model, each document is represented as a vector whose entries are terms of the vocabulary obtained from the text collection. Specifically,

© A. Gelbukh, S. Torres, I. López (Eds.)
Advances in Artificial Intelligence
Research in Computing Science 20, 2006, pp. 107-112

given a text collection $\{D_1, \ldots, D_M\}$ with vocabulary $V = \{t_1, \ldots, t_n\}$, the vector $\overrightarrow{D_i}$ of dimension n, corresponding to document D_i, has entries d_{ij}, where the value of an entry d_{ij} is the weight of term t_j in D_i:

$$d_{ij} = tf_{ij} \cdot idf_j, \tag{1}$$

where tf_{ij} is the frequency of term t_j in document D_i, $idf_j = \log_2(\frac{2M}{df_j})$, and df_j is the number of documents using term t_j. In collections of hundreds of documents, the dimension of the vector space can be of tens of thousands.

A key element in text representation is basically the adequate election of important terms, i.e. those that do not affect the process of retrieval, clustering, and categorization, implicit in the application. Besides, they have to reduce the dimensionality without affecting the effectiveness. It is important, from the reason just explained, to explore new mechanisms to represent text, based on terms appearing in the text. There are several methods to select terms or keywords from a text, many of them affordable in terms of efficiency but not very effective. R. Urbizagástegui [12] used the *Transition Point* (TP) to show its usefulness in text indexing. Moreover, the transition point has shown to work properly in term selection for text categorization [4] [5] [6]. TP is the frequency of a term that divides a text vocabulary in terms of high and low frequency. This means that terms close to the TP, of both high and low frequency, can be used as keywords that represent the text content. A formula to calculate TP is:

$$TP = \frac{\sqrt{1 + 8 \cdot I_1} - 1}{2}, \tag{2}$$

where I_1 represents the number of words having frequency 1. Alternatively, TP can be found as the lowest frequency, from the highest, that does not repeat, since a feature of low frequencies is that they tend to repeat.

This work explores an alternative to the classic representation based on the vector space model for IR. Basically, the proposed representation is the result of doing a term selection, oriented to index the document collection and, in addition, a weighting scheme according to the term importance. Both tasks are based on terms allegedly having a high semantic content, and their frequencies are within a neighborhood of the transition point.

Following sections present the term weighting scheme, experiments done using TREC5 collection, results, and a discussion with conclusions.

2 Term Selection and Weighting

The central idea behind the weighting scheme proposed here is that important terms are those whose frequencies are close to the TP. Accordingly, term with frequency very "close" to TP get a high weight, and those "far" from TP get a weight close to zero. To determine the nearness to TP, we proceed empirically: selecting terms with frequency within a neighborhood of TP; where each neighborhood was defined by a threshold u.

Given a document D_i, we build its vocabulary from the frequency, fr, of each word: $V_i = \{(x,y)|x \in D_i, y = fr(x)\}$. From the vocabulary, we can calculate $I_1 = \#\{(x,y) \in V_i|y = 1\}$ for D_i. So, using equation 2, TP of D_i is determined (denoted as PT_i), as well as a neighborhood of important terms selected to represent document D_i:

$$R_i = \{x|(x,y) \in V_i, TP_i \cdot (1-u) \leq y \leq TP_i \cdot (1+u)\}, \quad (3)$$

where u is a value in $[0,1]$.

The important terms of document D_i are weighted in the following way. For each term $t_{ij} \in R_i$, its weight, given by equation 1, is altered according to the distance between its frequency and the transition point:

$$tf'_{ij} = \#R_i - |TP_i - tf_{ij}|. \quad (4)$$

3 Data Description

TREC-5 collection consists of 57,868 documents in Spanish, and 50 topics (queries). The average size of vocabulary of each document is 191.94 terms. Each of the topics has associated its set of relevant documents. On average, the number of relevant documents per topic is 139.36. The documents, queries and relevance judgements used in the experiments were taken from TREC-5.

4 Experiments

Two experiments were performed, the first aimed to determine the size of the neighborhood u (eq. 3) and, the second was oriented to measure the effectiveness of the proposed scheme on the whole collection TREC-5. In these experiments, we applied standard measures; *i. e.*, precision (P), recall (R), and F_1 measure [13] defined as follow.

$$P = \frac{\#\text{relevant docs. obtained by the system}}{\#\text{docs. obtained by the system}}, \quad (5)$$

$$R = \frac{\#\text{relevant docs. obtained by the system}}{\#\text{ relevant documents}}, \quad (6)$$

$$F_1 = \frac{2 \cdot P \cdot R}{P+R}. \quad (7)$$

4.1 Neighborhood Determination

Two subsets of TREC-5 were extracted, S_1 and S_2 sub-collections, with 933 and 817 documents, respectively. Each one contains documents relevant to two topics, in addition to non relevant documents selected randomly, in a double rate to relevant documents. Several threshold values were tested, whose results are displayed in Figure 1.

Fig. 1. Values of F_1 using three thresholds in two sub-collections of TREC-5.

Sub-collection	u 0.3	0.4	0.5
S_1	0.34	0.37	0.39
S_2	0.28	0.34	0.38

Other values of u led to F_1 values less or equal to those showed in the table of Figure 1. We picked $u = 0.4$, even though this does not produce the maximum F_1, but allows to determine a lower bound of the performance of the proposed term selection.

4.2 Term Selection and Weighting Performance

Document indexing was done using formulas 3 and 4, in addition to classic term weighting (eq. 1) in the whole TREC-5 collection, and submitting the 50 queries. Retrieved documents were sorted according to their assessed similarity to the query (*ranking*). For a vector query $\overrightarrow{q}$, and a document $\overrightarrow{D_i}$, its similarity was calculated using the cosine formula. Finally, to assess the effectiveness, we calculate average precision at standard recall levels, as shown in (fig. 3) [1] for classical and proposed (using TP) weighting.

Figure 2 summarizes the number of terms in the vocabulary for the whole collection, average number of terms per document, and the percentage of terms generated by the proposed indexing with respect to those produced by the classical representation.

Fig. 2. Vocabulary for the Two Representations.

Total/partial	**Classic**	**TP**	**%**
TREC-5	235,808	28,111	**11.92**
Average × doc.	191.94	6.86	**3.57**

5 Discussion

G. P. Luhn based on the argument that high frequency terms are very general and can lead to low precision, while those of low frequency result in low recall, proposed with insight what has been confirmed empirically. As stated above, the problem of determining adequate words to index documents is of interest in several tasks.

The use of transition points for the problem of term selection has shown effectiveness in some contexts [6] [9] [8].

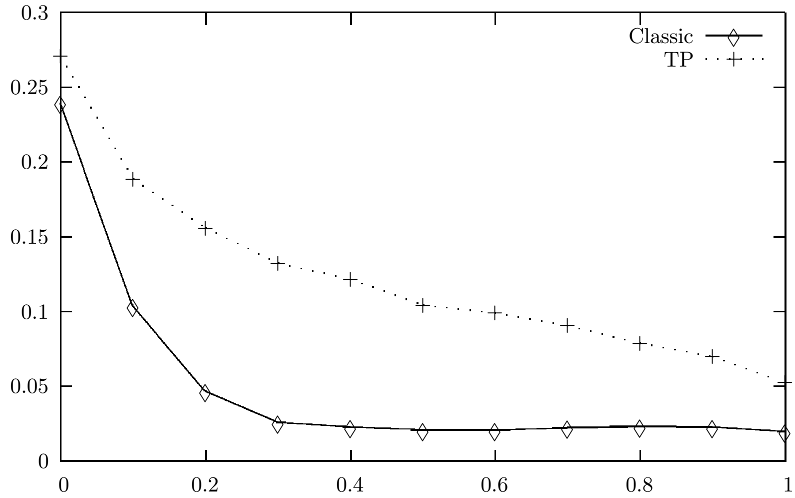


Fig. 3. Average Precision at standard Recall levels, using classical and proposed weighting.

The base on which lies the identification of medium frequencies has been taken from the formulae presented in 1967 by Booth [2], intended to determine a value that it was not high or low. From this formulae, the TP began to be used in the identification of keywords in a text [12]. In the present work, TP was used in a particular way: term weighting considering a neighborhood of frequencies around TP. The formula that determines such neighborhood (eq. 3) comes from the characteristics assumed for the TP [3]. It is not the same case for the weighting equation (4) which modifies the classical weighting (eq. 1). In the former, it is implicit the fact of repeating the terms that occur in the neighborhood as many times as their complementary distance to TP. That is the rationale of the replacement of tf_{ij} in eq. 1 by the proposed weighting (eq. 4). This repetition is a simple way to reinforce the importance of a term whose frequency is in the TP neighborhood.

The text representation problem, using the VSM, implies the selection of index terms and their weighting. Despite the fact that VSM and the classical weighting have several decades of existence, nowadays they are in essence being used in a diversity of NLP tasks; e.g. text categorization, text clustering, and summarization. It is a well known empirical fact that using all terms of a text commonly produces a noisy effect in the representation [11]. The high dimensionality of the term space has led to a index term analysis. For instance, Salton et al. [10] proposed a measurement of discrimination for index terms, i.e terms defining vectors in the space that better discerned what documents answer a particular query. They concluded that, given a collection of M documents, the "more discriminant" terms have a frequency in the range $[\frac{M}{100}, \frac{M}{10}]$. This result suggests to analyze the discriminant value of terms in a neighborhood of TP.

The diversity of proposals on feature selection are conceived into supervised and unsupervised methods. An advantage of the method here presented is its unsupervised nature, so it is possible to use it in a wide variety of NLP tasks.

The results obtained for TREC-5 encourage to confront TP with other proposed representations, and its application in different collections to validate the observed effectiveness. Moreover, we identify the need to establish precisely the advantages of this representation on some other task of NLP.

References

1. Baeza-Yates, R.: *Modern Information Retrieval*, Addison Wesley, 1999.
2. Booth A.: A law of occurrence of words of low frequency, *Information and Control*, 10 (4), pp. 383-396, 1967.
3. Ernesto Miñón; David Pinto & Héctor Jiménez-Salazar: Análisis de una representación de textos mediante su extracto, *Avances en la Ciencia de la Computación*, pp. 107-111, 2005.
4. Moyotl, E. & Jiménez, H.: An Analysis on Frecuency of Terms for Text Categorization, *Proc. of SEPLN-04, XX Congreso de la Sociedad Española para el Procesamiento del Lenguaje Natural*, pp. 141-146, 2004.
5. Moyotl, E. & Jiménez, H.: Experiments in Text Categorization Using Term Selection by Distance to Transition Point, *Proc. of CIC-04, XIII Congreso Internacional de Computación,* pp. 139-145, 2004.
6. Moyotl, E. & Jiménez, H.: Enhancement of DPT Feature Selection Method for Text Categorization, *Proc. of CICLing-2005, Sixth International Conference on Intelligent Text Processing and Computational Linguistics*, pp. 706-709, 2005.
7. Pérez-Carballo, José & Strzalkowski, Tomek: Natural Language Information Retrieval: progress report, *Information Processing and Management* v.36(1), Elsevier, pp. 155-178, 2000.
8. Pinto D.; Jiménez-Salazar, H. & Paolo Rosso: Clustering Abstracts of Scientific Texts using the Transition Point Technique, *Lecture Notes in Computer Science*, Vol. 3878, pp. 536-546, 2006.
9. David Pinto, Héctor Jiménez-Salazar, Paolo Rosso & Emilio Sanchis: BUAP-UPV TPIRS: A System for Document Indexing Reduction at WebCLEF. Accessing Multilingual Information Repositories, 6th Workshop of the Cross-Language Evaluation Forum, CLEF 2005, Vienna, Austria, Revised Selected Papers. Carol Peters, Fredric C. Gey, Julio Gonzalo, Gareth J.F. Jones, Michael Kluck, Bernardo Magnini, Henning Müller, Maarten de Rijke (Eds.), *Lecture Notes in Computer Science* Vol. 4022 (forthcoming).
10. Salton, G., Wong, A. & Yang, C.: A Vector Space Model for Automatic Indexing, *Communications of the ACM*, 18(11) pp. 613-620, 1975.
11. Sebastiani, F.: Machine Learning in Automated Text Categorization, *ACM Computing Surveys*, 34(1), pp. 1-47, 2002.
12. Urbizagástegui, A.R.: Las Posibilidades de la Ley de Zipf en la Indización Automática, `http://www.geocities.com/ResearchTriangle /2851/RUBEN2.htm`, 1999.
13. van Rijsbergen, C.J.: *Information Retrieval.* London, Butterworths, 1999.

A Term Frequency Range for Text Representation

David Pérez[1], José Tecuapacho[1], Héctor Jiménez-Salazar[1], Grigori Sidorov[2]

[1]Faculty of Computer Science,
Autonomous University of Puebla,
14 sur y Av. San Claudio, C.U.
Edif. 135, CP 72570, Puebla, Pue., Mexico
{dave_p_g,bicha_tecua}@yahoo.com.mx, hgimenezs@gmail.com
[2]Natural Language and Text Processing Laboratory,
Center for Research in Computer Science,
National Polytechnic Institute
Av. Juan Dios Batiz, s/n, Zacatenco, 07738, Mexico City,
Mexico
sidorov@cic.ipn.mx

Abstract. In this work, terms weighting is explored using the Vector Space Model framework. We rely on the hypothesis that terms with medium frequencies have high semantic content. This idea has been exploited before using the Transition Point approach, but manual selection of a threshold is required. We present a formula for determining these "important" frequencies automatically without any threshold setting. We made experiments for Information Retrieval and Text Extraction tasks using three subcollections of TREC-5. The results show that the suggested weighting scheme is a considerable improvement of the Transition Point approach.

1 Introduction

The Vector Space Model framework is very useful approach to text representation in Natural Language Processing tasks, like text categorization, text clustering or automatic summarization. Even when other models, such as probabilistic or fuzzy, are used, this model is mentioned as a good one [6]. In Vector Space Model, each document is represented as a vector, which indexes are all words (terms) used in a given text collection. Namely, for a text collection $\{D_1, ..., D_M\}$ with terms $V = \{t_1, ..., t_n\}$, the vector $\overrightarrow{D_i}$ of dimension n that corresponds to the document D_i is composed of entries d_{ij} with weight of each term t_j in D_i obtained according to the formula:

$$d_{ij} = tf_{ij} * idf_j, \quad (1)$$

© A. Gelbukh, S. Torres, I. López (Eds.)
Advances in Artificial Intelligence
Research in Computing Science 20, 2006, pp. 113-118

where tf_{ij} is frequency of the term j in the document i, while idf_j refers to the number of documents that use the term j, df_j. It is calculated as

$$idf_j = \log_2(\frac{2 \cdot M}{df_j}).$$

This can be explained as following: the term with high frequency in the text has large weight only if it occurs in a small number of documents. If a high frequency term occurs in many documents, then it does not convey real information because it does not allow for distinguishing between the documents.

The idea that we present in the paper is based on the fact that the terms with medium frequency should have the major weight, and while the terms are more distant from the range of medium frequencies, the less their weight is. There are descriptions of experiments that demonstrate that the usage of variations of weights (Eq. 1) using threshold and transition point (TP) is promising [7] [4]. In this paper, we propose a formula for determining medium frequencies without a threshold. It is shown that this formula allows for obtaining equal or better performance than the TP.

The work has the following structure: first we describe how to determine the range of medium frequencies of a text, then we present the suggested weighting formula, after this, the extraction of the representative sentences from a text is described, and finally, the results of the experiments are discussed.

2 Transition Point Range

One of the important tasks of text representation is the selection of a subset of terms that are good representation of a text and permit operations of categorization, clustering, searching, etc. using the selected subset instead of a whole document. There are various methods for selection of indexing terms or key words, for example, Urbizagástegui [2] used the transition point for showing the usefulness of text indexation.

Transition point is a frequency of a term that divides text vocabulary into terms of low and high frequencies. The terms that are useful for text representation are situated around the TP, because it is supposed that they have high semantic content. The formula for the calculation of TP is as follows:

$$TP = \frac{\sqrt{1 + 8 * I_1} - 1}{2}, \quad (2)$$

where I_1 represents the number of terms with frequency 1. This empiric formula seeks the identification of a frequency that is neither low, nor high. Usually, many terms correspond to low frequencies; say, more that 50% of terms in an average text have frequency 1, etc. This formula excludes them from the consideration explicitly. The calculated frequency (TP) is the lowest of the high frequencies. The alternative calculation of TP is seeking the lowest frequency that is not repeated, i.e., the first frequency that corresponds to exactly one term. It is justified by the fact that several terms usually correspond to values of low frequencies.

In this paper, we suggest to use two transition points basing on the idea of repetition of frequencies. The first TP is the lowest of the high frequencies that is repeated, TPa, i.e., we start form the highest frequency and go downwards until we find the first repetition. The second TP is the highest of the low frequencies that is not repeated, TPb, i.e., we start from the lowest frequency and go upwards until we find the first term with unique (non-repeated) frequency. Thus, we define the range of medium frequencies, namely, *transition range*, $[TPb, TPa]$.

In the following sections, we describe the application of the transition range to information retrieval and extraction of the representative sentences tasks.

3 Weighting of Terms

As we mentioned before, the documents can be represented by the weighted terms. In [7] a scheme of term weighting which takes into account the TP is presented. The method proposed there is different from Eq. 1, namely

$$d_{ij} = IDPT_{ij} \times DPTC_i, \tag{3}$$

where $IDPT_{ij} = 1/|TP_j - tf_{ji}|$ is the inverse distance of the term i to the TP of the document j, and $DPTC_i = |TP - fr_i|$, is the distance between the term i and TP, calculated for the whole collection.

For our experiment, the definition of $IDPT_{ij}$ is given by:

$$IDPT_{ij} = \begin{cases} 1 & \text{if } tf_{ji} \in [TPb_j, TPa_j], \\ 1/(tf_{ji} - TPa_j) & \text{if } TPa_j < tf_{ji}, \\ 1/(TPb_j - tf_{ji}) & \text{if } TPb_j > tf_{ji}. \end{cases} \tag{4}$$

where $[PTb_j, PTa_j]$ is the transition range of the document j. $DPTC_i$ also is adapted to the two frequencies of transition that are global now:

$$DPTC_i = \begin{cases} 1 & \text{if } fr_i \in [TPb, TPa], \\ fr_i - TPa & \text{if } TPa < fr_i, \\ TPb - fr_i & \text{if } TPb > fr_i. \end{cases} \tag{5}$$

Here $[TPa, TPb]$ constitutes the transition range of the whole collection and fr_i is the frequency of term i in the collection.

4 Representative Sentences in Texts

Let us consider the task of selection of the most "representative" sentences of a text. We base on the work [8], where the terms near TP are considered for assigning scores to sentences and generate an extract composed by three sentences with major scores. The proposed approach is as follows:

1. Preprocessing. Document splitting into sentences is performed, taking into account abbreviations, etc. The words from the stop list are eliminated from the sentences. These are words like prepositions, articles, etc.
2. Vocabulary extraction. All terms are extracted and their frequencies are calculated.
3. Transition range. The transition range is calculated according to the procedure described above. The "virtual paragraph" (VP) is generated, i.e., the paragraph, to which all terms that belong to the transition range are added.
4. Assignment of scores to sentences. Each sentence is assigned a score according to its similarity to the VP.
5. Extraction of representative sentences. Three sentences with major scores according to their similarity to the VP are taken.

The extract quality is verified by its comparison with the complete document. One of the ways of doing this is the usage of the extract instead of the full text in certain tasks like Information Retrieval. If the IR system performs in the same way, then the quality of the extract is good. For our experiments, we used Jaccard's formula to calculate the similarity between the query and each document in the collection, as in [8].

$$sim(D, q) = \frac{\#(D \cap q)}{\#(D \cup q)}.$$

5 Obtained Results

We conducted experiments with term weights assignment based on the transition range and detection of representative sentences. Several subcollections of TREC-5 were used allowing comparison of results with previous works. Further we describe subcollections and then the obtained results.

5.1 Data Description

Collection TREC-5 is a text collection of more than 50,000 documents in Spanish and 50 topics (possible queries). Each topic is assigned a set of documents that correspond to it, i.e., are relevant for this topic. The TREC-5 documents, queries, and relevance criteria were used in our experiments. We defined three subcollections from the documents according to the following algorithm: for a given topic, we add all relevant documents to the subcollection, and then add twice as many non-relevant documents. Table 1 contains the number of documents in subcollections.

The subcollections were preprocessed and words from stop lists were eliminated. The queries were preprocessed as well in the same way. Besides, all letters in queries were changed to lower case. The topics are shown in Table 2.

Table 1. TREC-5 subcollections for 6 topics.

Subcollection	Topics	#	#Relevant
1	$c_1 : c_3$	1117	211 : 164
2	$c_{10} : c_{11}$	933	206 : 105
3	$c_{14} : c_{15}$	817	281 : 6

Table 2. Topics used in evaluation.

c1	mexican oposition FTA (free trade agreement)
c3	pollution mexico city
c10	mexico important country transit war antidrug
c11	water rights rivers frontier region mexico unites states
c14	monopoly oil pemex has great influence mexico
c15	dispute fishing caused capture fishing ships unites staes

5.2 Results

In Fig. 1, the results are presented for each subcollection and for each method. The Column 1 refers to the method. For three first rows, the method based on weighting was used, while for three last rows the extract generation was applied. For each subcollection, we calculated the values of precision P, recall R, and F_1 measure, see, for example, [3].

$$P = \frac{\#\text{relevant documents obtained by the system}}{\#\text{total documents obtained by the system}}, \quad (6)$$

$$R = \frac{\#\text{relevant documents obtained by the system}}{\#\text{total relevant documents}}, \quad (7)$$

$$F_1 = (2 \cdot P \cdot R)/(P + R). \quad (8)$$

The methods that are referred to as TR use transition range, while those referred to as TP are based on transition point as it is explained in sections 3 and 4.

Fig. 1. Transition range.

Method	Subcol. 1			Subcol. 2			Subcol. 3		
	P	R	F_1	P	R	F_1	P	R	F_1
Classic	0.28	0.61	0.38	0.21	0.74	0.33	0.33	0.93	0.48
TR	0.24	0.17	**0.2**	0.17	0.2	**0.18**	0.34	0.78	**0.47**
TP	0.34	0.06	0.1	0.13	0.29	0.18	0.44	0.31	0.33
Full text	0.16	0.47	0.24	0.17	0.68	0.27	0.18	0.69	0.28
TR	0.16	0.22	**0.19**	0.19	0.39	**0.26**	0.18	0.48	**0.26**
TP	0.37	0.07	0.11	0.19	0.33	0.24	0.19	0.19	0.19

6 Conclusions

We presented an approach that allows for detection of the transition range, i.e., the range of terms with medium frequencies in a text. This range has the properties that correspond to the expected behavior of the terms, which are in the transition from terms with low frequency to terms with high frequency. It is supposed that terms in this range are the most representative terms of a text. The advantage of the approach is that it does not require choosing manually any thresholds. Certainly, the results are not as good as in the classic approach that uses $tf_{ij} \cdot ifd_j$ or as in the case of usage of the complete documents. We showed that the transition range gives better results than the transition point; however, this claim must be tested in a larger collection. It is necessary to take into account that transition range has similar behavior and inherits practically all numerous applications of the transition point. So, we can recommend the usage of the transition range in natural language processing applications instead of the transition point because of its extensive advantages.

References

1. Salton, G., Wong, A. & Yang, C.: A Vector Space Model for Automatic Indexing, *Communications of the ACM*, 18(11) pp 613-620, 1975.
2. Urbizagástegui, A.R.: Las Posibilidades de la Ley de Zipf en la Indización Automática, `http://www.geocities.com/ResearchTriangle /2851/RUBEN2.htm`, 1999.
3. van Rijsbergen, C.J.: *Information Retrieval*. London, Butterworths, 1999.
4. Moyotl, E. & Jiménez, H.: An Analysis on Frecuency of Terms for Text Categorization, *Proc. of SEPLN-04, XX Congreso de la Sociedad Española para el Procesamiento del Lenguaje Natural*, pp 141-146, 2004.
5. Moyotl, E. & Jiménez, H.: Enhancement of DPT Feature Selection Method for Text Categorization, *Proc. of CICLing-2005, Sixth International Conference on Intelligent Text Processing and Computational Linguistics*, pp 706-709, 2005.
6. Baeza-Yates, R.: *Modern Information Retrieval*, Addison Wesley, 1999.
7. Rubí Cabrera, David Pinto, Darnes Vilariño & Héctor Jiménez: Una nueva ponderación para el modelo de espacio vectorial de recuperación de información, *Research on Computing Science* 13, pp 75-81, 2005.
8. Claudia Bueno, David Pinto & Héctor Jiménez: El párrofo virtual en la generación de extractos, *Research on Computing Science* 13, pp 83-90, 2005.

Selecting Information Sources for Collaborative Filtering and Content-Based Filtering

Silvana Aciar[1], Josefina López Herrera[1] and Josep Lluis de la Rosa[1]
[1] Departament d' Electronica, Informatica i Automatica
Univertitat de Girona
17071 Campus Montilivi
Girona, Spain
{saciar, peplluis} @eia.udg.es, lopez.herrera@udg.es

Abstract. This paper addresses the problem of identifying and selecting relevant sources of information in order to enhance the accuracy of recommender systems. Recommender systems suggest to the users the items they will probably like. The large amount of information available nowadays on Internet makes the process of detecting user's preferences and selecting recommended products more and more difficult. In this paper we present a methodology to identify and select sources holding relevant information for recommender systems. This methodology is applied using two recommender methods: Content-Based Filtering (CBF) and Collaborative Filtering (CF) and showed in a real case-study, how the accuracy of the recommendations made with these methods and the selected sources increase.

1 Introduction

Information overload is one of the most important problems met by the Internet's users nowadays. The great amount of old and new information to analyze, contradictions in the available information generate noise and make difficult the identification process of relevant information. The information overload phenomena is determined by the lack of methods to compare and process the available information Recommender Systems address this problem filtering the most relevant information for the user's purpose. These systems receive as input the preferences of the users, analyze them and deliver recommendations.

Recommender systems are used in a network overloaded of information. In such a case, the search of specific information for recommender systems is a difficult task. The literature in the field of recommender systems is focused towards the methods that are used to filter the information to make the recommendations. Methods such as Content-Based Filtering (CBF) [4] [5] and Collaborative Filtering (CF) [8][9] are significant examples in this field. This paper presents a methodology for the identification of information sources, comparing the sources and to selecting most relevant information to make recommendations. This methodology allows the optimization of the search of the information obtained only from the relevant sources for the recommendations. A Multi-Agent System (MAS) called ISIIRES (Identifying, Selecting, and Integrating Information for Recommender Systems) has been designed for this

© A. Gelbukh, S. Torres, I. López (Eds.)
Advances in Artificial Intelligence
Research in Computing Science 20, 2006, pp. 119-130

purpose [2]. ISIIRES identifies information sources based on a set of intrinsic characteristics and selects the most relevant to be recommended. In this paper we present the result obtained applied the methodology using two recommender methods: CBF and CF and showed in a real case-study, how the accuracy of the recommendations increases.

The paper is organized as follows. In Section 2 are described the ISIIRES methodology and the Multi-Agent System to implement the methodology. In Section 3, the application of the recommender methods is showed. In Section 4 a Case Study with some result is described and finally, conclusions are drawn in Section 5.

2 Identifying, Selecting, Integrating Information for Recommender Systems (ISIIRES)

A methodology for the identification of information sources, comparing the sources and to selecting most relevant information to make recommendations has been proposed and has been described by Aciar et.al. in [2]. Four blocks compose the methodology which is shown in figure 1.

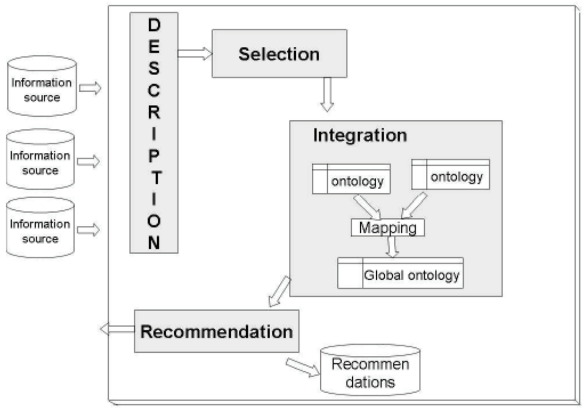


Fig. 1. Functional blocks of the methodology

2.1 Description

A set of intrinsic characteristics of the sources has been defined for the identification of relevant information to make recommendations [3]. These characteristics are:

- an abstract representation of the information contained in the sources
- criteria to compare and to select the sources.

2.2 Selection

The selection of the information sources is made based on the intrinsic characteristics and a value of trust from last recommendations. An initial value of trust = 1 is assigned to the sources that have not been used in last recommendations [1].

2.3 Integration

Mapping between the ontologies is made to integrate the information from the selected sources which are distributed and heterogeneous. The mapping has been made defining a global ontology comparing the ontologies of each source looking for similar concepts among them [2].

2.4 Recommendation

The methodology is focussed on the selection of the sources. When the relevant information sources have been selected it is possible to apply any of recommender methods. In this paper the methodology is applied using two recommender methods: CBF and CF. The Multi-Agent System (MAS) shown in figure 2 has been designed to implement the methodology [3].

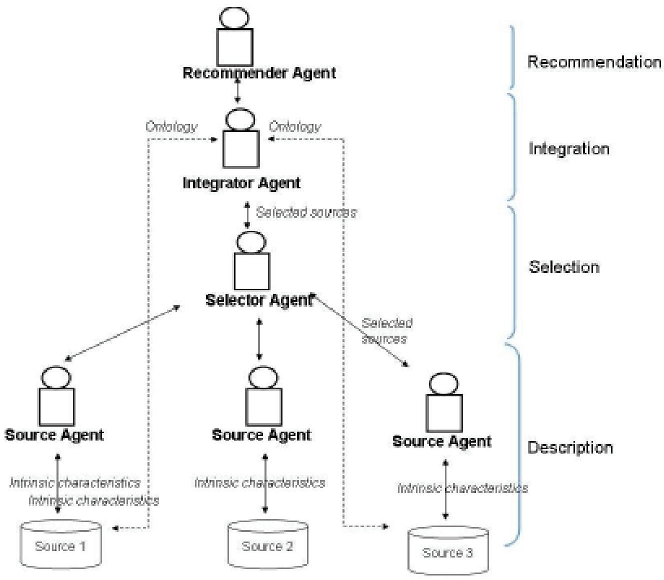


Fig. 2. MAS to implement the methodology

Four types of agents compose the MAS: Source Agent (SA), Selector Agent (SEA), Integrator Agent (IA) and Recommender Agent (RA).

– The SA managers the sources obtaining intrinsic characteristics.
– The SEA selects the sources that provide the most relevant information to make the recommendation based on the intrinsic characteristics and a trust value.
– The IA integrates the information from selected sources establishing a mapping between the ontologies of the selected sources.
– The RA is an interface agent that interacts with the users. Once the sources have been selected, it makes the recommendations and evaluates the results to keep them for future selections.

3 Applying Different Methods to Recommend

Two main methods have been used to compute recommendations: Content-Based Filtering and Collaborative Filtering, these methods are implemented in this paper with information from the selected sources.

3.1 Content-Based Filtering (CBF)

In this method the attributes of products are extracted and they are compared with a user profile (preferences and tastes). Vectors are used to represent the user profile and products in this work. The user vector is:

$$U =< u_1; u_2; \ldots\ldots\ldots\ldots; u_n >$$

The values of the user vector represent the preferences that he has for the attributes of products. Where u_i represent the weight of attribute i with respect to the user obtained from previous purchases made by the user. The product vector is:

$$P =< p_1; p_2; \ldots\ldots\ldots\ldots\ldots; p_n >$$

Where p_i represent the weight of attribute i whit respect to the product. This weight is assigned by an expert of the domain. The cosine function based on the vectorial space proposed by Salton [7] has been used to establish the relevance of products for the users using both vectors: P and U

$$Cos(P,U) = \frac{\sum_{i=1}^{n}(p_i * u_i)}{\sqrt{\sum_{i=1}^{n} p_i^2} * \sqrt{\sum_{i=1}^{n} u_i^2}} \quad \textbf{(1)}$$

The products that have a higher value of relevance are recommended the users.

3.2 Collaborative Filtering (CF)

The information provided by users with similar interests or necessities is used to determine the relevance that the products have for the user. Similarity between users is calculated for this purpose and the recommendations are made based only in this similarity, the bought products are not analyzed as it is made in the FBC. In this paper the cosine vector similarity [7] is used to compute the distance between the representation of the present user and the other users. All users are represented by vectors.

$$U =< u_1; u_2; \ldots\ldots\ldots\ldots; u_n >$$

Where u_i are the preferences of the user which is represented by the weight assigned by him to any attribute of the product, such as colour, type of product, etc. The similarity measurement is calculated by:

$$Cos(U,V) = \frac{\sum_{i=1}^{n}(u_i * v_i)}{\sqrt{\sum_{i=1}^{n} u_i^2} * \sqrt{\sum_{i=1}^{n} v_i^2}} \tag{2}$$

Where U and V are the user vectors.

3.3 Evaluating Recommendations

The purchases made by the users after the recommendations are used like feedback to evaluate the accuracy of the system. The accuracy is evaluated using the precision equation from the Information Retrieval field [6] adapted to our problem.

$$Precision = \frac{Pr}{R} \tag{3}$$

Where Pr is the number of recommended products that have been bought and R is the total number of recommended products. The precision represents the probability that a recommendation will be successful.

4 Case-Study

Eight data bases in the consumer package goods domain (retail) have been used. The data bases are related tables containing information of the retail products, 1200 customers and the purchases realized by them during the period 2001-2002. All data bases contain common customers.

The sources used in the experiments are the sources selected in the previous phase of the methodology, see Aciar et. al [1] for more detail about the selection of the sources.

4.1 Content-Based Filtering (CBF))

An expert of the supermarket has defined the relevant attributes of the product used in the CBF method. Based in these attributes shown in figure 3 has been established the user preferences represented by a vector:

codi	nom
2	Marca
3	Tipo de compra
4	Genero sujeto consumidor
5	Perfil consumidor
6	Edad consumidor
7	Implicacion
8	Transportable
9	Frecuencia uso
10	Tipo producto
11	Ciclo de venta
12	Complementariedad
13	Caducidad
14	Madurez
15	Fresco
16	Salud
17	Precio
18	Origen
19	Practico, almacenaje, conservacior
20	Sensibilidad_Precio_PF
21	Sensibilidad_Precio_PFS
22	Sensibilidad_Precio_SIO
23	Sensibilidad_Precio_C
24	Sensibilidad_Precio_VL
25	Sensibilidad_Precio_CIO
26	Sensibilidad_Precio_HD
27	Sensibilidad_Precio_PB

Fig. 3. Relevant attributes in the consumer package goods domain (retail) defined by an expert.

$$U =< u_1; u_2; \ldots\ldots\ldots\ldots; u_n >$$

The weight ui has been obtained from last purchases of the user using the tf-idf method (Term Frequency Times Inver Document Frequency) [6].

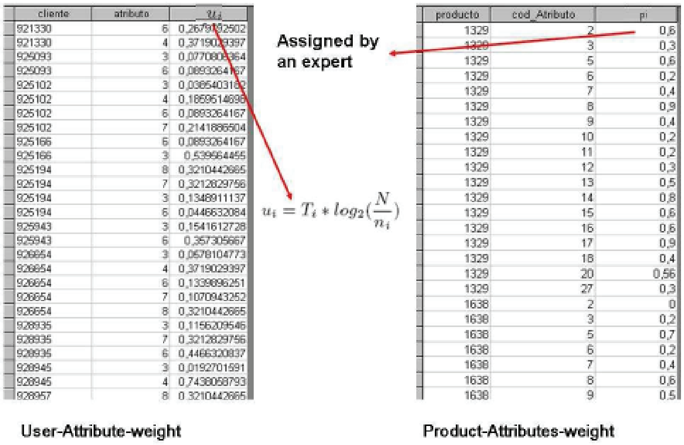
$$u_i = t_i * log_2(\frac{N}{n_i}) \quad (4)$$

Where ti is the frequency of attribute i in the purchases, ni is the number of users who have been bought a product with attribute i and N is the total number of users. The weight pi of the product vector has been assigned by the expert in the supermarket.

$$P =< p_1; p_2; \ldots\ldots\ldots\ldots\ldots; p_n >$$

The weights ui and pi of each vector are shown in figure 4. The relevance of each product for the users has been established with equation 1 using the vectors representing the users and the products (See figure 5).

The products with a value of relevance > 6 have been recommended the users. Theses recommendation are shown in figure 6.



cliente	atributo	u_i
921330	6	0,2679[illegible]2502
921330	4	0,37190[illegible]397
925093	3	0,07708[illegible]364
925093	6	0,0893264[illegible]7
925102	3	0,03854031[illegible]
925102	4	0,1859514[illegible]
925102	6	0,0893264167
925102	7	0,2141886504
925166	6	0,0893264167
925166	3	0,539564455
925194	8	0,3210442665
925194	7	0,3212829756
925194	3	0,1348911137
925194	6	0,0446632084
925943	3	0,1541612728
925943	6	0,357305667
926654	3	0,0678104773
926654	4	0,3719029397
926654	6	0,1339896251
926654	7	0,1070943252
926654	8	0,3210442665
928935	3	0,1156209546
928935	7	0,3212829756
928936	6	0,4466320837
928945	3	0,0192701591
928945	4	0,7438058793
928957	8	0,3210442665

producto	cod_Atributo	pi
1329	2	0,6
1329	3	0,3
1329	5	0,6
1329	6	0,2
1329	7	0,4
1329	8	0,9
1329	9	0,4
1329	10	0,2
1329	11	0,2
1329	12	0,3
1329	13	0,5
1329	14	0,8
1329	15	0,6
1329	16	0,6
1329	17	0,9
1329	18	0,4
1329	20	0,56
1329	27	0,3
1638	2	0
1638	3	0,2
1638	5	0,7
1638	6	0,2
1638	7	0,4
1638	8	0,6
1638	9	0,5

Fig. 4. Weights to obtain the user vector and the product vector

cliente	producto	Relevance
91850	32318	0,3633179926
91850	34971	0,3633179926
91850	34864	0,3166948446
91850	34781	0,6488153614
91850	34699	0,3969170281
91850	34533	0,3166948446
91850	32998	0,3166948446
91850	32993	0,3166948446
91850	32865	0,3969170281
91850	32492	0,3633179926
91850	32491	0,3633179926
91850	32490	0,3633179926
91850	32378	0,3633179926
91850	33535	0,3969170281
921330	44683	0,4541998345
921330	48587	0,8585791209
921330	48497	0,5599386555
921330	48472	0,8585791209
921330	48194	0,4541998345
921330	47748	0,4541998345
921330	47343	0,5599386555
921330	46127	0,5599386555
921330	46087	0,5599386555
921330	45976	0,8585791209
921330	43890	0,8585791209
921330	45479	0,4541998345
921330	44452	0,4541998345
921330	49326	0,5599386555
921330	45606	0,5599386555
921330	55178	0,8585791209
921330	65745	0,4541998345
921330	63473	0,4541998345
921330	60339	0,4541998345
921330	59930	0,8585791209

Fig. 5. Relevance of the products for the users

cliente	producto_recom	Descripcion
561339	92750	SERVILL. BLANCA CAPRABO
561339	166960	CREMA CACAO 1 C. 475 G NOCILLA.
561339	23998	CÓCTEL DE FRUTAS 1 L GRANINI
561339	61256	MAGDALENA VALENC 350 G CAPRABO
561339	113700	BIO C/F.BOSQUE X4 500G DANONE
561339	121764	ACEITE OLIVA 1 1 L CAPRABO
561339	115678	BEBIDA SOJA CALCIO 1L DIET-RADIS
561339	127622	SOLUBLE NAT. EXT 100 G CAPRABO
561339	143112	LATA SIN CAFEINA 33CL COCA COLA
561339	935000	AJOS SECOS 1/4 I
561339	63473	CERVEZA LATA 33CL CAPRABO
561339	542310	PATATAS TRADICION 170G EAGLE SNAC
561339	479628	MOZZARELLA RALL. 200 G M.D.
561339	274372	VINO TINTO BRIK 1 L DON SIMON
561339	19122	CAVA BRUT NATURE 75CL C.CARALT
561339	544330	PAN FRESCO FAMILI. 600 SILUETA
561339	133361	Q.MOZZARELLA LONCHAS 2 CAPRABO
561339	23170	12 ROLLO HIGIENI CAPRABO
561339	124971	ACEITE VIRGEN EXTRA 1 YBARRA
561339	51664	MAQ.DESECH.LADY EXTRA-II
561339	121192	LATA CERVEZA 33CL RGEMESSTER
561339	53269	PAN DE LECHE 400 G BELLA EASO
561339	440362	ZUMO MELOCOTONX3 600ML KASFRUIT
561339	9000003	Frutas y Verduras
561339	61256	MAGDALENA VALENC 350 G CAPRABO
561339	9000003	Frutas y Verduras
561339	143112	LATA SIN CAFEINA 33CL COCA COLA
561339	510004	LAUREL BOLSA 10 G DANI
561339	24677	CHAMPU NORMAL 400ML WELLA BALS
561339	143111	LATA 33CL COCA COLA
561339	483052	PARMESAN RALLADO 40 G KRAFT
561339	374385	TOMATE TRITURADO 410 G APIS
561339	121690	ACEITE OLIVA 1 L LA GLORIA

Fig. 6. Recommendations made using CBF

4.2 Collaborative Filtering (CF)

The attributes shown in figure 3 defined by the expert of the supermarket have been used to obtain the user vectors.

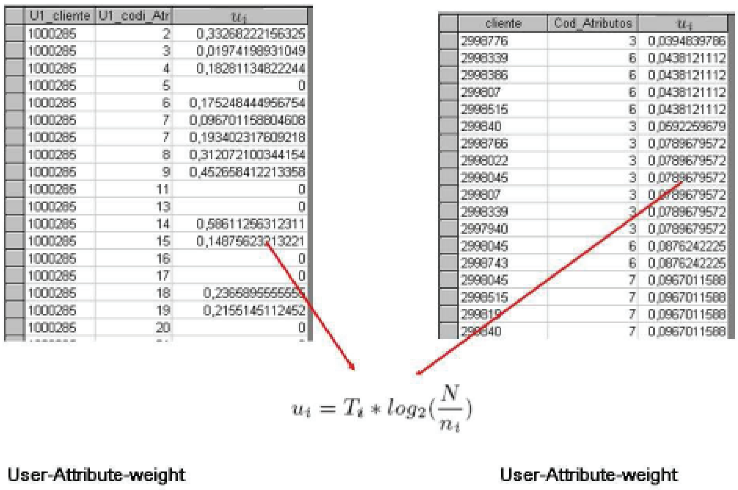
$$U =< u_1; u_2; \ldots\ldots\ldots\ldots; u_n >$$

The weight ui has been obtained from last purchases of the user using the tf-idf method (Term Frequency Times Inver Document Frequency) [6] like in the CBF

$$u_i = t_i * log_2(\frac{N}{n_i}) \quad (5)$$

Where ti is the frequency of attribute i in the purchases, ni is the number of users who have been bought a product with attribute i and N is the total number of users. The weights ui obtained for each user are shown in figure 7.

The similarity between users has been established with equation 2 using the vectors representing the users (See figure 8)



U1_cliente	U1_codi_Atr	u_i
1000285	2	0,33268222156325
1000285	3	0,01974198931049
1000285	4	0,18281134822244
1000285	5	0
1000285	6	0,175248444956754
1000285	7	0,096701158804608
1000285	7	0,193402317609218
1000285	8	0,312072100344154
1000285	9	0,452658412213358
1000285	11	0
1000285	13	0
1000285	14	0,58611256312311
1000285	15	0,14875623213221
1000285	16	0
1000285	17	0
1000285	18	0,23658955555555
1000285	19	0,2155145112452
1000285	20	0

cliente	Cod_Atributos	u_i
2998776	3	0,0394839786
2998339	6	0,0438121112
2998386	6	0,0438121112
299807	6	0,0438121112
2998515	6	0,0438121112
299840	3	0,0592259679
2998766	3	0,0789679572
2998022	3	0,0789679572
2998045	3	0,0789679572
299807	3	0,0789679572
2998339	3	0,0789679572
2997940	3	0,0789679572
2998045	6	0,0876242225
2998743	6	0,0876242225
2998045	7	0,0967011588
2998515	7	0,0967011588
299819	7	0,0967011588
299840	7	0,0967011588

$$u_i = T_i * log_2(\frac{N}{n_i})$$

Fig. 7. Weights for the user vectors – CF

	VectorClienteU1_cliente	VectorClienteU2_cliente	SIM
	1000285	299840	0,7426971197
	1000285	2997940	0,7426971197
	1000285	2998776	0,7426971197
	1000285	2998773	0,7426971197
	1000285	2998766	0,7112738823
	1000285	2998515	0,7112738823
	1000285	2998386	0,6798506450
	1000285	2998342	0,6798506450
	1000285	2998339	0,7112738823
	1000285	2997973	0,7426971197
	1000285	2998743	0,7112738823
	1000285	2997953	0,7112738823
	1000285	2998291	0,6484274077
	1000285	2998022	0,7112738823
	1000285	2998031	0,7426971197
	1000285	2998045	0,7426971197
	1000285	299807	0,6798506450
	1000285	299819	0,7112738823
	1000285	2998196	0,6798506450

Fig. 8 Similarity between users - CF

The products bought by other users with a value of similarity > 6 have been recommended to the user. Theses recommendation are shown in figure 9.

Cliente	Productos_recomendados	Descripcion
1000285	931830	PLATANO CANARIO EXTRA BANDEJA
1000285	39667	PAGES TALLAT 430GUN ROCAS. 76
1000285	1818	ESPINACA CORTADA 400 G FINDUS
1000285	68155	MAYONESA FRASCO 450ML CAPRABO
1000285	131086	TOMATE MONTSERRAT BJA.
1000285	36072	CAMINO MEGATRUCK C/PAL GOZAN 138
1000285	455393	MERMEL.NARANJA 350 G HELIOS
1000285	124928	ENSALADILLA 750 G FINDUS
1000285	16638	TORTILLA PATATA/CEBOLL G.CAMPI#A
1000285	68986	FRESH TABLETS BREF WC
1000285	130690	PEPINILLOS 345 G HELIOS
1000285	412400	PASAS CALIFORNIA 200 G CAPRABO
1000285	59905	SALSA LIGERA 225ML YBARRA
1000285	13470	ESTRO+ESP.NO RAY SCOTCH BRI
1000285	22862	MACARRONES HUEVO 250 G EL PAVO
1000285	110546	PATATAS CHURRERIA 170 CAPRABO
1000285	8493	TOMATE FRITO BRK 400 G CAPRABO
1000285	127515	PIMIENTOS ROJOS 185 G CAPRABO
1000285	121665	ACEITE OLIVA 0,4 1 L BORGES
1000285	22858	SPAGHETTI HUEVO 250 G EL PAVO
1000285	15426	GUAN.SATINADO P SCOTCH BRI
1000285	113727	Q.DESN.NATURALX2150 G DANONE
1000285	4022	DET.LIQUIDO MAQ.1000ML FINOSEDIL
1000285	179100	HARINA 1000 G CAPRABO
1000285	456908	CONFT.MANZANA 345 G HERO
1000285	340785	MEJILLON ESCABX3 240 G RIANXEIRA
1000285	46127	BIOF.DES.CO/MUX4 500 G DANONE
1000285	281557	VINO BLANCO PESC 75CL PERELADA
1000285	122939	PLATANOS FLOW-PACK
1000285	13285	ESTROP.VERDE GTE SCOTCH BRI
1000285	13955	FILETE TARRO 100 G AZKUE
1000285	466344	CALDO PESCADO08P 84 G KNORR

Fig. 9. Recommendations made using CF

4.3 Evaluating Recommendations

The experiments have been made implementing both methods: CBF and CF with the information of all the sources (8 data bases) without the methodology. The precision of recommendations has been evaluated using equation 3 obtaining the results shown in figure 10 and figure 11.

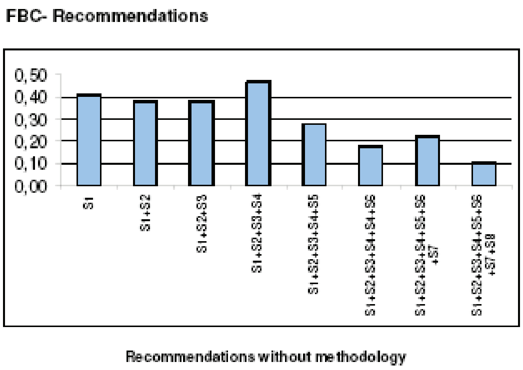


Fig. 10. Accuracy of the recommendations using CBF with all sources

In figures 12 and 13 can be observed the accuracy of the recommendations made using the CBF and the CF with information of the selected sources in the methodology. In the graphs are showed, how the accuracy of the recommendations made with these methods and the selected sources increase. The selection of the sources is established based on intrinsic characteristics of each source.

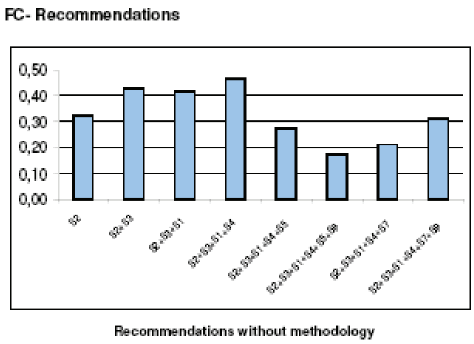


Fig. 11. Accuracy of the recommendations using CF with all sources

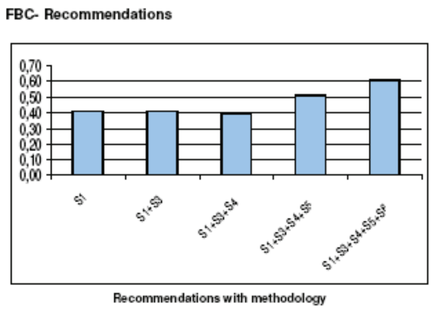


Fig. 12. Accuracy of the recommendations using CBF only with the selected sources in the methodology

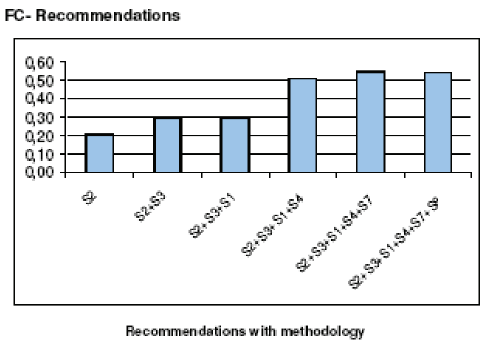


Fig. 13. Accuracy of the recommendations using CF only with the selected sources in the methodology

5 Conclusions

The large amount of information available nowadays on Internet makes the process of detecting user's preferences and selecting recommended products more and more difficult. A methodology has been developed to make this task easier and to optimize the information search to recommend resulting in better recommendations. In the methodology has been defined a set of intrinsic characteristics of information sources. The characteristics represent relevant information contained in the sources used to make recommendations. The selection of relevant sources is made based in these characteristics. The user preferences are established from the selected sources. The methodology has been used with two recommender methods: CBF and FC obtaining good results in each one of them. The results obtained in a real case-study show how the accuracy of the recommendation made with the selected sources increase.

Referentes

1. Aciar, S, Lpez Herrera, J and . de la Rosa, J.Ll 'Fc en un sma para seleccionar fuentes de información relevantes para recomendar', XI Conferencia de la Asociación Española para la Inteligencia Artificial (caepia'05). Workshop de Inteligencia Computacional: Aplicaciones en Marketing y Finanzas. Santiago de Compostela, España, (Noviembre 2005).
2. Aciar, S, Lpez Herrera, J and . de la Rosa, J.Ll 'Integrating information sources for recommender systems', Artificial Intelligence Research and Development - IOS Press. VIII Catalan Association for Artificial Intelligence Congress (CCIA 2005),Alghero, Sardinia, Italy, (October 2005).
3. Aciar, S, Lpez Herrera, J and . de la Rosa, J.Ll 'Sma para la búsqueda inteligente de información para recomendar', I Congreso Español de Informática,(Cedi 2005), Simposio de Inteligencia Computacional, SICO2005 (IEEE Computational Intelligence Society, SC)Granada, Espana, (Septiembre 2005).
4. Balabanovic,M and Shoham, Y. 'Fab: Content-based, collaborative recommendation', Communications of the ACM, 40(3), 66–72, (March 1997).
5. Lieberman, H. 'Letizia : An agent that assists web browsing', Proceedings of the International Joint Conference on Artificial Intelligence, Montreal, (1995).
6. Salton, G. and Buckley, C. 'Automatic text processing', he Transformation, Analysis and Retrieval of Information by Computer, (1989).
7. Salton, G. and McGill, M.J. 'Introduction to modern information retrieval', McGraw-Hill Publishing Company, NewYork, (1983).
8. Shardanand, U. and Maes, P. 'Social information filtering: Algorithms for automating word of mouth', SIGCHI Conference, (1995).
9. Smyth, B. and Cotter, P. 'A personalized television listings service', Communications of the ACM, (2000).

Music Information Retrieval Trough Melodic Similarity Using Hanson Intervallic Analysis

Johnny F. Serrano α,β *and José M. Iñesta* α

α Departamento. de Lenguajes y Sistemas Informáticos
Universidad de Alicante, España
β Centro de Investigación en Computación e Informática
Universidad Latina, Costa Rica
johsersa@gmail.com, inesta@dlsi.ua.es

Abstract. In music information retrieval, the melodic similarity is used as main feature for the detection of relevant information. Among the possible applications there is a detection of plagiarism of ideas exposed by an artist in the past, the payment for copyrights using detection of musical pieces in radio transmissions, the computer-aided composition, etc. There exist several techniques exposed in melodic similarity that use diverse statistical and probabilistic analysis. The objective in this work is to establish a text words equivalent to musical notation using a representation based on intervallic relationships and duration, and to evaluate three textual information retrieval techniques applied to this representation, as well as to propose changes to improve the system's performance.

1 Introduction

The music representation through descriptors such as duration, pitch, contour, intervals, etc., is common in the system analysis and musical information retrieval [1,2]. The two dimensions in which a score is codified (vertical for pitch and horizontal for duration) like symbolic registration of the music have been represented in very different ways, in order to carry out analysis as probabilistic techniques like hidden Markov chains [3], entropy analysis [4] and statistical correlations [4], among other [5].

The intervallic analysis will be the fundamental unit for the melodic similarity task, where the musical fragments are analyzed in their intervallic relationships in a tempered chromatic scale [6]. The interval is the basic unit of analysis, putting aside the pitch and the tonality as direct descriptors. Some previous works have used this concept making a cartesian notation of the intervals [7] or determining the correlation of the intervallic progressions [8]. A well-known work that makes special emphasis in the relationships among contiguous intervals may be the one of Narmour [9]. The author establishes a series of groups of contiguous intervals (implication patterns - resolution) that allow analyzes melodies through their transformation in symbols. An important disadvantage of this focus is that it ignores the relationships of durations, which is an inconvenience for the information retrieval.

© A. Gelbukh, S. Torres, I. López (Eds.)
Advances in Artificial Intelligence
Research in Computing Science 20, 2006, pp. 131-142

This work explored the conversion to text words from the extracted intervals of MIDI documents. This focus is not completely new in the sense that other authors have previously used techniques coming from the analysis of texts to solve problems related with the processing of the music information.

One possible encoding is the one proposed by Downie [10] or Doraisami and Rüger [11] that have used models based in n-grams for melody identification. On the other hand, in [12] the authors use words models to identify musical styles.

Three techniques of textual information retrieval have been used: Cosine [13], Kaszkiel [14] and Pivoted Cosine [15] to check the effectiveness in the determination of the retrieval for relevant information through melodic similarity. Also, this work proposes some modifications in the techniques mentioned above to improve the performance.

2 Methodology

2.1 Conversion to Text Words

Starting from a musical fragment (see Fig. 1), it takes the first note and the distance is calculated in semitones with regard to the following note. This distance (interval) is represented with the letters d-s-n-m-p-t of the Hanson system [6] plus the five vowels a-e-i-o-u to form the intervals of two octaves and one half[1] [16] (see Fig. 2). Thus, the intervals are coded into characters in this form.

The original work of Hanson was addressed to the modern music analysis. Because of that, it is convenient to introduce some modifications to adapt the code to this work. A third letter is added to the representation as follows: "l" to represent a descendent interval and "r" for an upward interval. The representation of an interval of zero semitones was not considered originally in the Hanson's analysis but it has been considered in this work to obtain a more complete representation of the melodies. For it the word "loo" has been used. This example shows some intervals and its corresponding textual codes.

Also, the duration of the two notes that conform it is considered as the duration of the interval representing units according to their musical figure, where the value 1 is given to a black note. This way, the duration can take fractional values. A couple of examples of Hanson intervallic representation of the melody can be appreciated in Fig. 3. This representation is a simple and comprehensible musical notation of a fragment. Through the definition of any initial note, one melody can be reproduced in different tonalities keeping its similarity [16]. Therefore, when untying the tonality of the melody the analysis is simplified and it is focused to the relationship between the intervals and their durations.

[1] These syllables have been used for many years in teaching systematic intervallic solfege.

Interval in semitones: 1 2 3 4 5 6 7 8 9 10 11 12 . .
Corresponding codification: da sa na ma pa ta pe me ne se de te . . .

Fig. 1. Interval analysis.

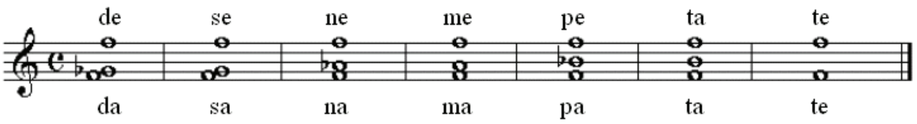


Fig. 2. Syllables' examples of the systematic-intervallic solfege. In the superior part of the staff, the syllables corresponding to the code of the interval are indicated between the intermediate note and the superior one of each chord, and the lower code corresponds to the intermediate note in relation to the lowest one.

Fig. 3. Hanson intervallic representation for two music fragments. The code showed over the scores is the key corresponding to the RISM database (see section 2.3).

2.2 The Information Retrieval System

One of the objectives of the presented work is to export techniques, commonly used in textual information retrieval, to the field of music information. In this way, we will consider that the documents we work with are monophonic melodic fragments, and the terms of these candidates are composed in textual codes for each one of the intervals present in such candidates.

Given a database of melodic fragments, the indexation will made through the following procedure. First, the musical fragments are converted to text according to the methodology described. Then, an index is extracted out of each fragment, which is addition of durations for all the intervals of the same type found in this fragment. In this way, the concept of "frequency" which is characteristic of the field from the textual information retrieval is assimilated as the duration of the notes in each melodic fragment. For it is determined as follows:

- The frequency (i.e., the duration) of the terms (intervals of one type) in the document (f_t) (see Table 1). For example, in the first candidate, the interval "sal" appears 3 times with duration of 2 units in each interval, because of that, the value of the f_t of the interval "sal" is 6.

Table 1. Indexation for frequencies of terms corresponding to Fig. 3 music fragments.

Music Fragment	***Interval***	f_t
702.001.990-1.1.1.mid	dal	3
	loo	2
	per	5
	sal	6
450.039.857-1.1.1.mid	loo	12
	mar	4
	sal	8
	sar	8

- The frequency of the candidates of the collection (f_d) that each one of the terms contains (see Table 2).

Table 2. Indexation by candidates' frequencies of Fig. 3 music fragments. There are only two candidates in the collection, therefore $f_d \leq 2$.

Interval	f_d
dal	1
loo	2
mar	1
per	1
sal	2
sar	1

2.3 Data Set

The corpus used for the experimentation has 581 melodic fragments with an average of 14 intervals for melody. The format of the melodies is MIDI. These melodies conform the answers to 11 queries generated by the team of Rainer Typke of the Utrecht University [17], being the most similar (relevant) candidates evaluated and ordered by 35 experts in the musical field starting from the collection RISM A/II out of more than 456.000 melodies.

The 11 consultations have around 50 similar melodies. These are also ordered according to a degree of similarity and in groups of same similarity. In the Fig. 4, a query example and an extract of the melodies that are more relevant to this query are visualized

2.4 Evaluation

In this section the measures proposed are presented to evaluate the precision, recall and capacity of the algorithms used in this work to retrieval the candidates in an appropriate order of relevance. Two methods will be used for this evaluation:

- Evaluation according to the TREC[2] [18] competitions that considers all the answers with the same degree of relevance, where the (not interpolated) average precision, R-precision, the precision at 5 candidates and the charts of precision versus recall interpolated will be mentioned.
- ADR (average dynamic recall) evaluation [19] also takes into consideration a degree of relevance of each candidate, where groups determine the degrees of relevance.

Considering R as the number of relevant candidates for a query:

- It takes the first *R* candidates returned by the information retrieval system.
- In each group division, the precision is calculated.
- The precision curve is integrated and is divided by *R*.

The result is a value between 1 and 0, where 1 is equal to retrieval the entirety of relevant candidates, in its correct order.

2.5 Experimentation

In the experimentation, the information retrieval techniques of the Cosine, Kaszkiel and of the Pivoted Cosine were used. Next, the formulas for the calculation of the query's weight, the candidate's weight and their similarity are shown.

[2] TREC: Text Retrieval Conference.

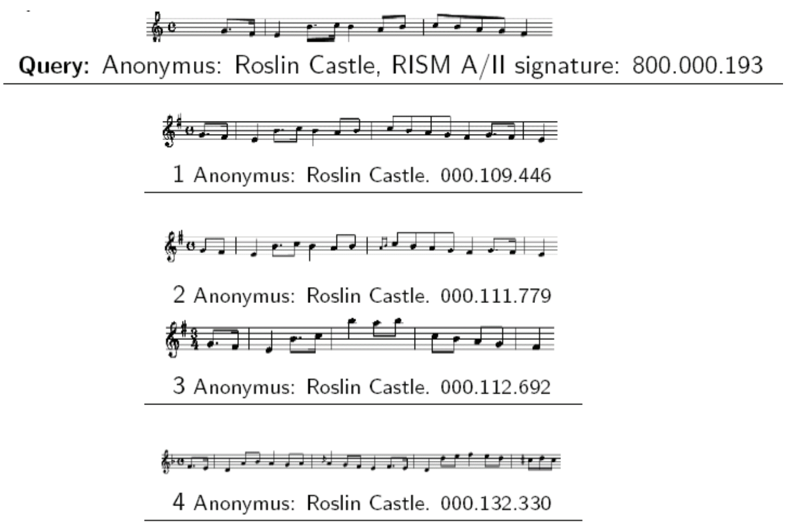


Fig. 4. Query example (above) with the 4 more relevant candidates in order of relevance.

2.5.1 Cosine

Given two fragments, one for query, Q, and other the candidate, D, selected in a database where there are N candidates, the similarity is defined between both as:

$$sims(Q,D) = \frac{\sum_{i=1}^{k} q_i \cdot d_i}{\|Q\| \cdot \|D\|} \;, \tag{1}$$

where $\|Q\| = \sqrt{\sum_{i=1}^{k} q_i^{\,2}}$, $\|D\| = \sqrt{\sum_{i=1}^{k} d_i^{\,2}}$ and k is the number of terms in Q.

The weight of each term is defined as $q_i = ft_{q,i} \cdot \log_e(\frac{N}{fd_i})$ for the terms of the query and $d_i = ft_{d,i} \cdot \log_e(\frac{N}{fd_i})$ for the each candidate. (2)

These expressions $ft_{d,i}$ represent the frequency of the term *i-th* of the candidate and $ft_{q,i}$ the frequency of the term *i-th* of the query.

2.5.2 Kaszkiel:

In an equivalent way, this measure of similarity among the consultation Q and the candidates D is defined as:

$$sims(Q,D) = \frac{\sum_{i=1}^{k} q_i \cdot d_i}{\|Q\| \cdot \|D\|} \quad , \tag{3}$$

where $\|Q\| = \sqrt{\sum_{i=1}^{k} q_i^{\,2}}$, $\|D\| = \sqrt{\sum_{i=1}^{k} d_i^{\,2}}$, and the weight of the terms:

$$d_i = \log_e(ft_{d,i} + 1) \ , \ q_i = \log_e(ft_{q,i} + 1) \cdot \log_e(\frac{N}{fd_i} + 1) \tag{4}$$

The initial evaluation was made using these techniques shown above. The results were not more favorable than that technique of Cosine. It was determined that the normalization of the natural logarithm affected negatively because it attenuated in an excessive way the frequencies, therefore it was replaced by the square root of the frequencies. In this way, the weights of the terms of the equation (4) were finally substituted for:

$$d_i = \sqrt{ft_{d,i}} \ , \ q_i = \sqrt{ft_{q,i}} \cdot \log_e(\frac{N}{fd_i} + 1) \tag{5}$$

2.5.3 Pivoted cosine

The third technique used to evaluate the similarity between the query and the candidates, is defined as:

$$sims(Q,D) = \sum_{i=1}^{k} \frac{q_i \cdot d_i}{W_d} \quad \text{, and the weights:}$$

$$d_i = 1 + \log_e(ft_{d,i} + 1) \ , \ q_i = 1 + \log_e(ft_{q,i} + 1) \cdot \log_e(\frac{N+1}{fd_i}) \tag{6}$$

In the same way as for Kaszkiel's case and since the Pivoted Cosine neither improved the technique of the Cosine, the normalization of the natural logarithm of the equation was replaced (6) for the square root of the frequencies:

$$d_i = 1 + \sqrt{ft_{d,i}} \ , \ q_i = 1 + \sqrt{ft_{q,i}} \cdot \log_e(\frac{N+1}{fd_i}) \tag{7}$$

The normalization factor that appears in the equation (6) would be:

$$W_d = (1 - slope) + slope \cdot \frac{L_B}{\overline{L_B}} \ , \tag{8}$$

being *slope* the factor of participation of the normalization, L_B the candidate longitude in Bytes, and $\overline{L_B}$ the average longitude of the all candidates in Bytes. But in the results presented in section 3.1 of this work, modification of this normalization factor has been used, because the norm has been used with regard to the average duration of all candidates' intervals. As follows:

$$W_d = (1 - slope) + slope \cdot \frac{L_I}{\overline{L_I}} \quad , \tag{9}$$

where L_I is the total duration of the candidate's intervals and $\overline{L_I}$ is the average of the total duration of intervals for the candidates.

3 Results

3.1 Preliminary Results

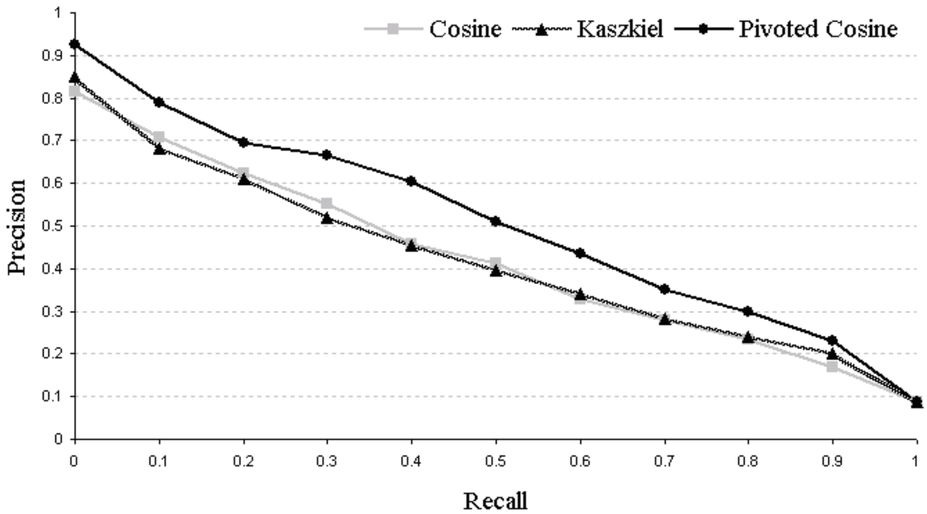
The first test was to evaluate the performance applying the three methods, using as weight of the query and weight of the candidate, the respective frequencies. In the case of the Cosine, the frequencies were normalized between 0 and 1 with regard to the maximum frequency found inside the analyzed candidate. And in the case of the Pivoted Cosine, the norm was used with regard to the average of candidates' intervals duration (23.55) with a *slope* of 0.4. The modifications proposed for the normalization of the weights in the cases of Kaszkiel and pivoted cosine improved the results regarding those obtained in the preliminary tests with the original equations. The results of the evaluation of the 11 queries can be seen in Graph 1 and in Table 3.

It can be seen that the technique of the Pivoted Cosine provides better results with 55.18% of ADR for the average of the 11 queries. Observing the results individually for each question, it can be noticed how it affects the Pivoted Cosine since the result of some questions favors small candidates, but in other ones the result is good, close or above 70%.

Table 3. Preliminary Results

Technique	*AP not int.*[3]	*Prec. at 5*[4]	*R-prec.*[5]	*ADR*[6]
Cosine	0.4062	0.6364	0.4245	0.4569
Kaszkiel	0.3968	0.5455	0.4144	0.4436
Pivoted Cosine	0.4900	0.7091	0.4805	0.5518

[3] AP not Int. : Not interpolated average precision
[4] Prec. at 5: Precision at 5 retrieval candidates
[5] R-prec.: R-precision.
[6] ADR: Average Dynamic Recall



Graph 1. Precision versus recall for the preliminary results

3.2 Normalization Change

As the objective is finding the most relevant candidates according to the query, in the case of melodic similarity it is important to make a hard analysis of the intervals duration. In order to do this, the factor Rft_i is established as the relationship between the durations of the candidate and the durations of the query. This value is calculated dividing the minor value by the major, and a numeric factor is obtained between 1 and 0 indicating the degree of similarity among the duration of the intervals of the same candidate's type with regard the query:

$$\text{if } ft_{q,i} > ft_{d,i} \text{ , } Rft_i = \frac{ft_{d,i}}{ft_{q,i}} \quad \textbf{(10)}$$

$$\text{if } ft_{q,i} < ft_{d,i} \text{ , } Rft_i = \frac{ft_{q,i}}{ft_{d,i}} \quad \textbf{(11)}$$

Next, the Cosine is evaluated using the Rft_i as the candidate's weight:

$$d_i = Rft_i \cdot \log_e(\frac{N}{fd_i}) \quad \textbf{(12)}$$

In the case of Kaszkiel, the candidate's weight would be Rft_i multiplied by the frequency of the query term:

$$d_i = \sqrt{Rft_i \cdot ft_{q,i}} \quad \textbf{(13)}$$

And in the Pivoted Cosine the candidate's weight would be $Rft_i + 1$ multiplied by the candidate's frequency. Also, the norm will be in relation to the total duration of the intervals of the query and not to the average of the duration of the candidates:

$$d_i = 1 + \sqrt{ft_{d,i} \cdot (Rft_i + 1)} \ , \quad W_d = (1 - slope) + slope \cdot \frac{L_I}{L_{qI}} \ , \qquad (14)$$

where L_{qI} is the total duration of the query intervals.

The results of the evaluation of the 11 queries are shown in the Table 5. It can be noticed that the use of Rft_i improves the performance of all methods. The cosine's technique improves the best result previously obtained, mainly in the precision, because Rft_i helps to compare the frequencies improving the results of the correlation.

Table 5. Results using the intervals duration relationships[7].

Technique	*AP not int.*	*Prec. at 5*	*R-prec.*	*ADR*
Cosine	0.4645	0.7818	0.4703	0.5495
Kaszkiel	0.4798	0.8364	0.4653	0.5684
Pivoted Cosine	0.5314	0.8000	0.5112	0.6303

The Pivoted Cosine generates the best result with 63.03% of ADR for all 11 queries. The improvement obtained through Rfti is very significant (9% in precision and 8% in ADR).

Observing the results one by one for each query, it can be noticed how the result affects the Pivoted Cosine of some queries, since it favors small candidates, but in other, the result exceed 75%.

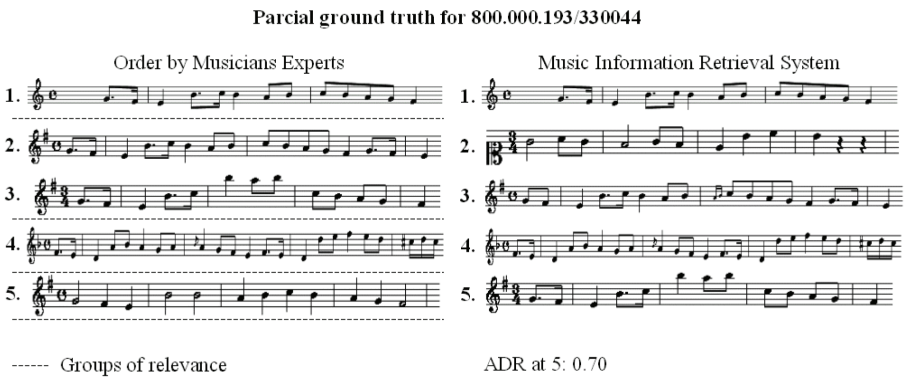


Fig. 5. Comparison results for a query between the music experts and the music information retrieval system.

---

[7] *slope* = 0.3

Comparing results can be seen in the Fig. 5 for a query between the music experts and the music information retrieval system. Although the information retrieval system did not return the first 5 candidates in the proper order, the order is acceptable because the candidates maintain a relevant similarity in relation to the query.

4 Conclusions

It has been demonstrated that the intervallic relationships are important elements to take into consideration for the melodic similarity task and that it is possible to make an equivalent to text words and use the current techniques of textual information retrieval.

In the case of Kaszkiel and Pivoted Cosine, a change was necessary in the normalization replacing the function of the natural logarithm for the square root of the frequency. Also, the improvements introduced to the three methods by the relationship of frequencies (Rft_i) demonstrated to be of great help, and they improved much results in the evaluation of the TREC as well as the ADR.

It is possible to increase the results using a balanced normalization of the musical fragments using the total duration of the intervals as the quantity of intervals that compose them.

Another solution could be the application of an initial filter for the recovery of most relevant candidates through these quick techniques, and then using a more precise algorithm, to obtain the final results.

Another pending task is studying the behavior of the system in function of the size of the musical fragments that are compared. The availability of a great database is necessary to carry out these tasks.

Acknowledgment

This work was possible thanks to the support of the Spanish Agency of International Cooperation (ref: AECI:B/2098/04). The Ministry of Science and Technology of Costa Rica and the CONICIT (ref: 84-2005), as well as the projects CICyT TIC 2003-08496-C04 and GV043-541.

References

1. Marsden, A. Pople (eds.) Computer Representations and Models in Music. Academic Press. 1992.
2. W.B. Hewlett, E. Selfridge-Field (eds.) The Virtual Score. Representation, Retrieval, Restoration. The MIT Press. 2001.
3. Ning Hu, Roger B. Dannenberg, Ann L. Lewis. A Probabilistic Model of Melodic Similarity. Proceedings of the International Computer Music Conference (ICMC), 2002.

4. Craig Stuart Sapp, Yi-Wen Liu, Eleanor Selfridge-Field. Searh-Effectiveness Measures for Symbolic Music Queries in Very Large Databases. Proceedings of 5th International Conference on Music Information Retrieval, ISMIR. 2004.
5. Rainer Typke, Frans Wiering, Remco C. Veltkamp. A survey of Music Information Retrieval Systems. Proceedings of 5th International Conference on Music Information Retrieval, ISMIR. 2004.
6. Howard Hanson. Music Theory. Harmonic Materials of Modern Music. ISBN: 0891972072 Publisher: Irvington Pub. 1960.
7. Michael Clausen, Frank Kurth, Roland Engelbrecht. Content-based Retrieval in MIDI and Audio. ECDL WS Generalized Documents. 2001.
8. Roland Engelbrecht. Statistical Comparison Measures for Searching in Melody Databases. Tech. Report. University of Bonn. 2002.
9. Eugene Narmour. The Analysis and Cognition of Basic Melodic Structures. Chicago University Press, 1990.
10. Stephen Downie. Evaluating a simple approach to music information retrieval: conceiving melodic n-grams as text. Tesis Doctoral. University of Western Notario. 1999.
11. S. Doraisami & S. Rüger. Robust polyphonic music retrievals with n-grams. Journal of Intelligent Information Systems, 21(1): 53-70.
12. C. Pérez-Sancho, José M. Iñesta, J. Calera-Rubio. Style recognition through statistical event models. Sound and Music Computing (SMC'04) Conference Proceedings, 135-139 2004.
13. W.B. Frakes, D.C. Baeza-Yates, editors. Information Retrieval: Data Structures and Algorithms. Prentice-Hall, 1992.
14. M. Kaszkiel, J. Zobel and R. Sacks-Davis. Efficient Passage Ranking for Document Databases. ACM Transactions on Information Systems, 17(4), pp 406-439. October 1999.
15. Singhal. A., Buckley, M. Mitra. Pivoted document length normalization. Proceedings of the 19th annual international ACM-SIGIR conference on research and development in information retrieval. 1996.
16. Bernal Flores. Music Theory. Systematically Orderly Sonorities. Universidad de Costa Rica, 1979.
17. Rainer Typke, Marc den Hoed, Justin de Nooijer, Frans Wiering, Remco C. Veltkamp. A Ground Truth For Half A Million Musical Incipits. Proceedings of the Fifth Dutch-Belgian Information Retrieval Workshop. Pp.63-70. 2005.
18. Edie Rasmussen. Evaluation in Information Retrieval. The MIR/MDL Evaluation Project White Paper Collection #3. Pp. 45-49. 2003.
19. Rainer Typke. Proposed measure for comparing ranked lists, for MIREX ’05 (Symbolic Melodic Similarity). 5th Dutch-Belgian Information Retrieval Workshop. 2005.
20. Ning Hu, Roger B. Dannenberg and George Tzanetakis. Polyphonic Audio Matching and Alignment for Music Retrieval. 2003 IEEE Workshop on Applications of Signal Processing to Audio and Acoustics. October 19-22, 2003, New Paltz, NY.
21. Emilios Cambouropoulos, Costas Tsougras. Influence of Musical Similarity on Melodic Segmentation and Algorithms. Proceedings of the International Conference on Sound and Music Computing (SMC04), Paris, France.

Robotics

A Game Theory Approach to the Robot Motion Tracking Problem

Carlos Rodríguez Lucatero

UIA Ciudad de México, Departamento de Ingeniería,
Prolongación Paseo de la Reforma 880, México D.F., 01210
profesor.lucatero@gmail.com

Abstract. The paper studies the problem of tracking a target robot by an observer robot. The strategy of the target robot is not known in advance. The observer robot will try to learn the target's robot strategy by keeping a model about the target robot behaviour. This will be done by modelling the tracking problem as a repeated two player game, where the robots objective is to look for strategies that maximizes their expected sum of rewards in the game. We make the assumption that the robot motion strategies can be modelled as a finite automata. First we suppose that they behave competitively and then we relax this constraint to explore the case of a more general kind of interaction between the target and the observer.

1 Introduction

Many applications require continuous monitoring of a moving *target* as is the case of movie filming of moving actors whose motions are not known in advance and is necessary to keep inside the scope of the camera the actions of the main character. Other application can be the virtual presence for trying to keep track remotely of some moving objects as vehicles, people, etc. Another application of robot tracking can be the automated surveillance of museums where there can be a robot that follows a guest from the beginning of his visit till he leaves the museum. In the last few years many research efforts have been done in the design and construction of efficient algorithms for reconstructing unknown robotic environments [12][13][14][15] and apply learning algorithms for this end [12][13] . When we are looking for efficient interaction strategies we have to take into account the reward obtained for the actions executed in that moment as well as the consequences of the taken actions on the future behaviour of the other entities. This task can become very hard given that the effectiveness of the interaction strategy depends on the strategies of the other agents. The problem is that the strategies of the other agents are private. For dealing with these problems it is necessary to provide the agents with the capability to adapt their strategies based on

© A. Gelbukh, S. Torres, I. López (Eds.)
Advances in Artificial Intelligence
Research in Computing Science 20, 2006, pp. 145-154

their interaction history. This implies that we have to endow the agents with learning capabilities. In the near past there have been written many excellent articles on learning models of intelligent agents as those elaborated by David Carmel & Shaul Markovitch [1] [2]. In the field of Multi-agent Systems there have been written excellent papers about Markov games as a framework for multi-agent reinforcement learning (M.L. Littman [6]). In the field of robot motion planning there has been published very interesting papers about the problem of calculating motion strategies for maintaining visibility in space cluttered by obstacles, of a moving target whose movements were partially predictable and where the movements of the *target robot* as well as the *observer robot* have uncertainties. One of the main concerns on robot tracking is to keep the visibility of the *target robot* by maintaining the target on the scope of the cone of the visual sensor of the observer robot (S. La Valle et al [4] [5]). Another very important concern that can be found in many papers on robot tracking is to keep the visibility in real time of the *target robot* whose behaviour is unpredictable by using a reactive planner (R. Murrieta et al [7]). These last to papers [4] [5] and [7] will be the starting point of the present paper where we will propose some extensions under the focus of interest of game theory. In [4] [5] and [7] they make the assumption that the strategy of the *target robot* is to evade the *observer robot* and based on that they propose geometrical and probabilistic solutions of the tracking problem which consists on trying to maximize, by the *observer,* the minimal distance of escape of the *target*. We feel that the solution lacks at least in two aspects. First the *target* don't interact with the *observer* so there is no evidence that the strategy will try to escape if it doesn't knows what are the actions taken by the *observer*. The second aspect is that even if there can take place some sort of interaction between the *target* and the *observer*, the *target* is not necessarily following an evasion strategy so this may produce a failure on the tracking task.

In Section 2 we retake the formulation given in [5] and we will make some remarks about the limitations of this approach. In section 3 we will provide a precise formulation of the problem in terms of strategies in game theory and present a modelling of the tracking problem as a repeated two game player. In Section 4 we present some concluding remarks and future works to be done by us.

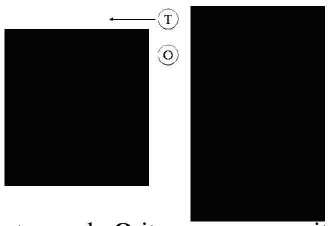
2 Formulation of the Tracking Problem as a Robot Motion Planning Problem

The problem can be posed in a worst case situation where the *target robot* try to evade the *observer robot*. Under this assumption the main goal of the strategy of the *observer robot* or *pursuer* will be to guarantee that the *target robot* or *evader* will be found for all possible motions. We can initially assume that the *pursuer* is equipped with vision or range sensing and that both robots are modelled as 2D points so each robot has a configuration space of dimension 2. This is a generalization of the *evasion-pursuit* that have been studied and formalized as a general decision problem where two agents have diametrically opposed interests. So the *pursuer* and the *evader* are modelled as points on the plane that is a bounded 2D Euclidean space cluttered by polygonal obstacles such that the task of keeping in a visibility cone the *evader* makes the problem become harder and by consequence, more appealing from the standpoint of motion planning. We will start with some definitions given in [5] [4] and [7] for the sake of clarity and the purpose of giving context to our work. In [5] are formulated two interesting research questions 1) What bounds can be established on the number of pursuers needed to solve the problem in terms of geometrical and topological complexity of the free space ? and 2) can a successful solution strategy be efficiently calculated for a given problem ? . We will try to give an answer to the second question from a game theory point of view and show that we can extend the scope of [4] using this formalism to more general *target* strategies than the *evasion* strategy. The problem of tracking have been defined in [5] as follows. Let F denote the closure of the collision free space. All *pursuer* and *evader* positions must lie in F. Let $e(t) \in F$ be the *evader* position at time $t \geq 0$. It is assumed that $e:[0,\infty) \rightarrow F$ is a continuous function, that the *evader* can execute arbitrarily fast motions and that the initial evader position $e(0)$ and path e are not known by the *pursuers*. In our case we will deal with only one *pursuer*. Let $\gamma(t)$ the position of the *pursuer* at time $t \geq 0$ and $\gamma:[0,\infty) \rightarrow F$ a continuous function representing his strategy. For any point $q \in F$ let $V(q)$ be the set of all point (i.e. linear segments joining q and any point in $V(q)$ lies in F). A strategy γ is considered as a *solution strategy* if for any $e:[0,\infty) \rightarrow F$ there exist a time $t \in [0,\infty)$ such that $e(t) \in V(\gamma(t))$. That means that the *evader* will eventually be see by the *pursuer* regardless its path. Given that the *evader's* position is unknown, one don't have access to the state at a given time, and that motivates the use the notion of *information space* that identifies all unique situations that can occur during motion strategies. In [4] are studied the motion strategies for maintaining visibility of a moving target. In this paper it is studied the problem of maintaining the visibility of the *target* with a camera mounted in an *observer* robot as motion planning problem, and assumed the following conditions 1) an *observer* nust maintain visibility of a moving *target*; 2) the workspace contains static obstacles that prohibit certain configurations of both the *observer* and *target*; 3) the workspace also contains static obstacles that occlude the target from the observer; 4) a (possibly partial) model is known for the *target*. In [4] they formulate the problem of one *pursuer* and one *evader* in the combined configuration space $X = C^{o}_{free} \times C^{t}_{free}$. They use a discrete time representation for facilitating the expressions of the uncertainty of the *target* for instance k is an index that refers to the time step that occurs at $(k-1)\Delta t$, and Δt is a fixed sampling rate. The *observer* is controlled through actions u_k from some space of actions U. The discrete time trajectory of the *observer* and the *target* were given by the transition equations $q^{o}_{k+1} = f^{o}(q^{o}_{k}, u_k)$ and $q^{t}_{k+1} = f^{t}(q^{t}_{k}, \theta_k)$ respectively where θ_k represents the unknown actions of the *target* from a space of actions Θ. In the case of a predictable *target* the transition equations is simply $q^{t}_{k+1} = f^{t}(q^{t}_{k})$. Together f^{o} and f^{t} define a state transition equation of the form

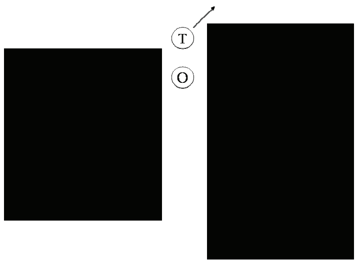
$x_{k+1} = f(x_k, u_k, \theta_k)$ where the state x_k represents the pair of configurations q_k^o and q_k^t. The visibility can be defined in many ways as for instance an omnidirectional field of view or as a fixed cone, etc. but in a more general setting it can be defined in terms of a binary relation between a visibility subspace and the space of states or more formally as $X_o \subset X$. Based on that the state trajectories can be evaluated in such a way that the *observer*'s goal is to stay in a state belonging to X_o. This control of the trajectory can be performed by applying a cost to the sequence of control inputs as follows: $L(x_1, \ldots, x_{K+1}, u_1, \ldots, u_K) = \sum_{k=1}^{K} l_k(x_k, u_k) + L_{K+1}(x_{K+1})$ where K represents the time increment for issuing a action and $l_k(x_k, u_k) = \{0 \ \ if \ \ x_k \in X_o; 1 \ \ otherwise\}$ is a loss accumulated in a single time step with which enable to measure the amount of time that the *target* is not visible and evaluate a given trajectory. If the movements of the *target* are predictable this means that q_k^t is known $\forall k \in \{1, \ldots, K+1\}$ and the transition equation is simplified to $x_{k+1} = f(x_k, u_k)$ and as consequence the state trajectory $\{x_2, \ldots, x_{K+1}\}$ can be know if once we know x_1 and the inputs $\{u_1, \ldots, u_K\}$. So for problems that don't involve the optimization of the robot trajectory, the motions of the *observer robot* can be computed by a recursive calculation of the visibility and reachability sets from stage K down to stage 1 , or telling it in words, by back-chaining. Besides that the loss functional can be minimized by dynamic programming using the relationship between the *cost-to-go* functions $L_k^* = \min_{u_k}\{l_k(x_k, u_k) + L_k^*(x_{k+1})\}$ and this can be utilized iteratively for calculating the optimal actions. The visibility polygon using omnidirectional visibility can be calculated in $O(n \lg n)$ using standard sweep algorithms. Another very interesting case appears when the *target* is partially predictable in the sense that it is know the velocity bound of the *target* in which case the dynamic programming method can be used to determine optimal strategies, but even for the very simple planar space the dimension becomes four. Due to this growth of complexity it have to be used alternative approaches that make a tradeoff between computational cost and quality of the solutions obtained. So the notions of optimal strategy become more interesting due to the fact about the uncertainty on the prediction of the *target* movements calculated as $q_{k+1}^t = f^t(q_k^t, \theta_k)$ where $\theta_k \in \Theta$ are the unknown actions. These unknown actions can be modeled in two ways. The fisrt as *nondeterministic uncertainty* and the second as *probabilistic uncertainty*. In first case one design the *observer* strategy that performs the best given the worst-case choices for θ_k. In the second case it can be assumed $p(\theta_k)$ where $p(\cdot)$ denotes a probability density function, and in that case the designed strategy of the *observer* will try to minimize the expected loss. Due to the unpredictability of the *target* strategy, it has to be designed an *observer* state-feedback strategy that respond to on-line changes. Let $\gamma_k : X \to U$ denote the strategy at stage k, $\{\gamma_1, \gamma_2, \ldots, \gamma_K\}$ the strategy, and Γ the space of possible strategies. For the case of nondeterministic uncertainty, a strategy , can be selected that yields the smallest worst-case loss: $\check{L}(x_1, \gamma^*) = \inf_{\gamma \in \Gamma} \check{L}(x_1, \gamma) = \inf_{\gamma \in \Gamma} \sup_{\gamma^\theta \in \Gamma^\theta} L(x_1, \gamma, \gamma^\theta)$ for all $x_1 \in X$, and γ^θ represents a choice of θ_k for every stage. The strategy obtained by this method guarantee the least possible loss given the worst-case action of nature. Using this formalism it can be proposed as strategy to maximize the probability of future visibility over the next m steps but the computational cost increases dramatically as a function of m. Because of that, in practice, the number of steps is limited to the case $m = 1$ and select the action u_k that maximize the probability that the target stay in the scope of the *observer* at the stage $k+1$. As a extension of the preceding approach in [7] R. Murrieta et al. use nondeterministic uncertainty and worst-case analysis for trying to solve the tracking problem for obtaining the *observer* strategy by maximizing the minimal distance to escape of the *target*, that is the shortest distance the target needs to move in order to escape the *observer's* visibility region. In this work they implement a reactive planner, that means a short term planner. They do that because of the need of a rapid response time that a normal planner cannot give. Due to the limitations of the vision sensing they introduce the notion of view frustrum or angular field of view. That is defined in terms of edges that borders either an obstacle , denoted as E_{fs}, or free space, denoted as E_{fr}. The free edge of the visibility region $V(q)$ nearest to the *target* is denoted as E^*_{fr}. To maximize the distance between the target and the boundary of the visibility region of the observer it is necessary to compute de distance between q^t_k and E^*_{fr} which is denoted $D_{q^t_k / E^*_{fr}}$. For obtaining it the distance among q^t_k and every edge of $V(q)$ must be calculated, and it can be easily done using Euclidean metric when when the E_{fr} is visible from the *observer* position, otherwise it is used a geodesic metric. So the distance between q^t_k and each free edge in $V(q)$ is the solution to the equation $D_{q^t_k / E^*_{fr}} = \min_{\{v_o, v_f\}} \{D_{q^t_k / v_o} + D_{v_o / v_f} + D_{v_f / E_{fr}}\}$ where v_o are the vertex in the *targets's* visibility region, and v_f is any vertex in $V(q)$ that sees the free edge E_{fr}.

2.1 Some Limitations of the Precedent Approach

The work done in [4] and [7] was very interesting but is limited to the case where the *target robot* is assumed to follow an evading strategy. That is due to the worst-case approach and the fact they were concerned to solve the tracking problem in real time tacking in account the uncertainty on the behaviour of a partially predictable *target*. The *evasion* strategy behaviour assumption is a very natural one but this entails the interaction between the *target* and the *observer* in such a way that the *evader* could take the best decision for evading the *observer*. This was completely ignored in [4], [5] and [7]. Lets give an schematic example. In the following image we denote with **T** and **O** the *target* and *observer* respectively. The black boxes represent obstacles in 2D and the arrow represents the movement decision taken by **T.** If **T** is following an evasion strategy it will try to minimize his distance to escape and **O** will try to maximize the minimal distance to escape of **T.**



In the case that **T** is not trying to evade **O** it can moves as it is shown in the following image.



If **O** is trying to maximize the minimal distance to escape of **T** it can lose the tracking. Another aspect that has not been considered in [4], [5] and [7] was the case where the *target* behaviour be different from the evasion one. In that case the actions or strategies calculated by the *observer* such that the *target* remains on the visibility region of the *observer* may fail if the actions of the *target* were to move to a position out of the visibility region and not necessarily to one related with the shortest distance to escape. Because of the reasons exposed above we propose as an extension of the precedent work done in [4], [5] and [7] to endow the *target* and the *observer* with a learning capacity in such a way that the *observer* can predict the behaviour of the *target* in a case different from the *evasion* strategy assumed in [4], [5] and [7].

3 Stating the Tracking Problem as a Repeated Game

One of our objections in way that the tracking problem has been formulated in [4], [5] and [7] was that it does not consider the fact that if the *target* is going to have an *evasion* behaviour it must take place an interaction between the *target* and *observer* such that the first can observe the second and take the best decision for an *evader*. It is well known that searching for an optimal interactive strategy is a hard problem because it depends on the behaviour of the others. Given that the agents involved in an interaction are autonomous their strategies are private, as is the case of the *observer* and the *target*. For dealing with this problem we propose to endow the interacting agents with a learning ability such that they can adapt their behaviours or strategies based on their interaction experience. We propose to use *model-based* approach for learning an efficient interactive strategy between the *observer* and the *target* inspired on what has been proposed in [1]. The agents keep a model of the opponent's strategy that is modified or adapted during the interaction, exploiting the current model to predict the other's behaviour and choose its own action according to the prediction. In case of failure in the prediction the agent updates the opponent's model to make it consistent with the new information. This approach give rise to two important questions 1) given a model of another agent, how an agent react optimally against it ? 2) when there is a prediction failure how an agent can adapt his opponent's model ?. To give an answer to these questions it can be used some tools of the *game theory*. The interaction between the *observer* and the *target* can be modelled as a *repeated two-player game* where the goal of each agent is to compute interaction strategies that maximizes its expected sum of rewards. This can be done efficiently under some assumptions about the kind of strategies followed by the agents as well as about the type utility functions for the *repeated games*. The first assumption is that each agent follows a regular strategy, i.e., a strategy that can be represented by a deterministic finite automata [10] and [9]. This assumption is based on the fact that in the case of having a *live complete sample* and a knowledgeable teacher that answers *membership queries* posed by the learner, it can be obtained a incremental polynomial learning DFA algorithm [9]. A second assumption is about the form of the utility functions, i.e.*discounted-sum* and *limit-of-the-means*, because it has been proved in [1] based on the work done in [8] that the best response strategy can be obtained efficiently given these common utilities functions.

Definition 1. *The tracking problem posed as a two-player-game is a tuple* $G = \left\langle f^o, f^t, u_1, u_2 \right\rangle$*, where* f^o , f^t *, are the finite set of alternative moves for the observer and the target and* $u_1, u_2 : f^o \times f^t \rightarrow \Re$ *are utility functions that define the utility joint move* (q^o, q^t) *for the players (i.e. observer and target).*

We propose that the tracking problem can be formulated as a sequence of encounters between the *observer robot* and the *target robot* and that situation can be described as a repeated game G', that is a repetition of G an indefinite number of times. At any stage k of the game, the players decide their actions $(q_k^o, q_k^t) \in f^o \times f^t$, simultaneously. A history $h(k)$ of G' is a finite sequence of joint moves chosen by the *observer* and the *target* until the current stage of the game. $h(k) = \left\langle (q_0^o, q_0^t), (q_1^o, q_1^t), \ldots, (q_{k-1}^o, q_{k-1}^t) \right\rangle$ denotes the history of movements of each robot. The empty history is denoted by ε. The set of finite histories is denoted as $H(G')$. The strategies are functions from the set of histories of games to the set of robots actions or moves $\sigma^o : H(G^{'}) \rightarrow f^o$ and $\sigma^t : H(G^{'}) \rightarrow f^t$ for the observer and the target respectively. An infinite sequence of joint moves during a repeated game $G^{'}$ between the *observer's* and the *target's* strategies is denoted by (σ^o, σ^t). The repeated game $G^{'}$ played by σ^o and σ^t defines the history $h(k)$ as follows:

$$g_{\sigma^o,\sigma^t}(0) = \varepsilon$$

$$g_{\sigma^o,\sigma^t}(k+1) = g_{\sigma^o,\sigma^t}(k) \| (\sigma^o(g_{\sigma^o,\sigma^t}(k)), \sigma^t(g_{\sigma^o,\sigma^t}(k)))$$

Definition 2*. Tracking as a two-player repeated-game over a stage game* G *is a tuple* $G^{'} = \langle \Sigma^o, \Sigma^t, U^o, U^t \rangle$ *where* Σ^o *and* Σ^t *are sets of strategies for the observer and the target respectively and* $U^o, U^t : \Sigma^o \times \Sigma^t \rightarrow \Re$ *are the utility functions.* U^o *defines the utility of the infinite sequence* g_{σ^o,σ^t} *for the observer and* U^t *for the target.*

Definition 3*.* $\sigma^o_{opt}(\sigma^t, U^o)$ *is called the optimal strategy for the observer w.r.t.* σ^t *and utility* U^o*, iff* $\forall \sigma \in \Sigma^o, [U^o(\sigma^o_{opt}(\sigma^t, U^o), \sigma^t) \geq U^o(\sigma, \sigma^t)]$.

Definition 4*.* $\sigma^t_{opt}(\sigma^o, U^t)$ *is called the optimal strategy for the target w.r.t.* σ^o *and utility* U^t*, iff* $\forall \sigma \in \Sigma^t, [U^t(\sigma^t_{opt}(\sigma^o, U^t), \sigma^o) \geq U^t(\sigma, \sigma^o)]$.

In the present work are considered two common utility functions for each robot, the first is the discount factor $U^o_{ds}(\sigma^o, \sigma^t) = (1-\gamma^o)\sum_{k=0}^{\infty} \gamma^o_k (\sigma^o(g_{\sigma^o,\sigma^t}(k)), \sigma^t(g_{\sigma^o,\sigma^t}(k)))$ for the observer and respectively $U^t_{ds}(\sigma^o, \sigma^t) = (1-\gamma^t)\sum_{k=0}^{\infty} \gamma^t_k (\sigma^t(g_{\sigma^o,\sigma^t}(k)), \sigma^o(g_{\sigma^o,\sigma^t}(k)))$ for the target for $0 \leq \gamma^o < 1$ and $0 \leq \gamma^t < 1$, the second kind of utility function is *limit-of-the-means* $U^o_{lm}(\sigma^o, \sigma^t) = \lim_{k \to \infty} \inf \frac{1}{k} \sum_{k=0}^{\infty} u^o(\sigma^o(g_{\sigma^o,\sigma^t}(k)), \sigma^t(g_{\sigma^o,\sigma^t}(k)))$ for the observer and $U^t_{lm}(\sigma^o, \sigma^t) = \lim_{k \to \infty} \inf \frac{1}{k} \sum_{k=0}^{\infty} u^t(\sigma^o(g_{\sigma^o,\sigma^t}(k)), \sigma^t(g_{\sigma^o,\sigma^t}(k)))$ for the target. So taking into account the repeated game formalism combined with robot motion planning tracking problem formulation we can make for instance

$u^t = D_{q_k^t / E_{fr}^*} = \min_{\{v_o, v_f\}} \{D_{q_k^t / v_o} + D_{v_o / v_f} + D_{v_f / E_{fr}}\}$ and

$u^o = \check{L}(x_1, \gamma^*) = \inf_{\gamma \in \Gamma} \check{L}(x_1, \gamma) = \inf_{\gamma \in \Gamma} \sup_{\gamma^\theta \in \Gamma^\theta} L(x_1, \gamma, \gamma^\theta)$ and replace them into the respective *limited-of-the-means* common utility functions then we can model the tracking problem of the kind *evader/pursuer* as a two-player repeated game. As can be seen the utility functions are more general and the *evader/pursuer* case is just an instance of the possible behaviors, so with these theoretic game model we can deal with more general behaviors depending on the utility functions u^o, u^t chosen.

3.1 Learning the Target's Automata

In this paper we assume that each robot is aware of the other robot actions, i.e. Σ^o, Σ^t are common knowledge while the preferences u^o, u^t are private. It is assumed too that each robot keeps a model of the behavior of the other robot. The strategy of each robot is adaptive in the sense that a robot modifies his model about the other robot such that the first should look for the best response strategy w.r.t. its utility function. Given that the search of optimal strategies in the strategy space is very complex when the agents have *bounded rationality* it has been proved in [10] that this task can be simplified if we assume that each agent follow a DFA strategy. In [8] has been proven that given a DFA opponent model, there exist a best response DFA that can be calculated in polynomial time. In the field of computational learning theory it has been proved by E.M. Gold [11] in that the problem of learning minimum state DFA equivalent to an unknown target is *NP*-hard. Nevertheless D. Angluin has proposed in [3] a supervised learning algorithm called *ID* which learns a target DFA given a *live-complete* sample and a knowledgeable teacher to answer *membership* queries posed by the learner. Later Rajesh Parekh, Codrin Nichitiu and Vasant Honavar proposed in [9] a polynomial time incremental algorithm for learning DFA. That algorithm seems to us well adapted to the tracking problem because the robots have to learn incrementally the strategy of the other taking as source of examples the visibility information as well as the history about the actions performed by each agent.

4 Conclusions and Future Work

As we have exposed in the present work, the *one-observer-robot/one-target-robot* tracking problem can be formulated as a two-player game and enable us to analyse it in a more general setting than the *evader/pursuer* case. The prediction of the *target* movements can be done for more general *target* behaviours than the *evasion* one, endowing the agents with learning DFA's abilities. The next step will be to elaborate the associated algorithms and implement them in a computer program. Another interesting issue is to apply the algorithms to the case of many *evaders* and many *pursuers.*

References

1. David Carmel & Shaul Markovitch, "Learning Models of Intelligent Agents," Technical Report CIS9606 Technion, 1996.
2. David Carmel & Shaul Markovitch, "How to explore your oponent's strategy (almost) optimally," Proceedings ICMAS98 Paris France, 1998.
3. Dana Angluin, "A note on the number of queries needed to identify regular languages", Information and Control, 51, 76-87, 1981.
4. S.M. La Valle, H.H. González Baños, Craig Becker, & J.C. Latombe, "Motion Strategies for Maintaining Visibility of a Moving Target", Proceedings of the IEEE International Conference on Robotics and Automation, 731-736,April 1997.
5. S.M. La Valle, David Lin, Leonidas J. Guibas, J.C. Latombe & Rajeev Motwani, "Finding an Unpredictable Target in a Workspace with Obstacles," Proceedings of the IEEE International Conference on Robotics and Automation, April 1997.
6. M.L. Littman, "Markov games as a framework for multiagent reinforcement learning," Proceedings of the eleventh International Conference on Machine Learning, 157-163 ,1994.
7. R. Murrieta-Cid, H.H. González-Baños & B. Tovar, "A Reactive Motion Planner to Maintain Visibility of Unpredictable Targets," Proceedings of the IEEE International Conference on Robotics and Automation , 2002.
8. Christos H. Papadimitriou & John N.Tsitsiklis, "The complexity of Markov decision processes", Mathematics of Operations Research, Vol. 12, No. 3, August 1987.
9. Rajesh Parekh, Codrin Nichitiu & Vasant Honavar, "A polynomial time incremental algorithm for learning DFA", Proceedings of the Fourth International Colloquium on Grammatical Inference (ICGI'98), Ames, IA. Lecture Notes in Computer Science vol. 1433 pp. 37-49. Berlin: Springer-Verlag..
10. Ariel Rubinstein, "Finite Automata Play the Repeated Prisioner's Dilemma", Journal of Economic Theory, 39, 83-96, 1986.
11. E.M. Gold, "Complexity of automaton identification from given data", Information and Control, 37, 302-320, 1978.
12. Dana Angluin, Jeffery Westbrook, & Wenhong Zhu, "Robot Navigation with range Queries", ACM STOC 96, 469-478, 1996.
13. Ronald L. Rivest, & Robert E. Schapire, "Inference of Finite Automata using Homing Sequences", Information and Computation, 103, 299-347, 1993.
14. A. Blum, P. Raghavan & B. Schieber, "Navigating in unfamiliar geometric terrain", ACM STOC 91, 494-504, 1991.
15. V. Lumelsky & A. Stepanov, "Path planning strategies for a point mobile automaton moving amidst unknown obstacles of arbitrary shape", Algoritmica, 2, 403-430, 1987.

Nonholonomic Motion Planning in Dynamically Changing Environments

Rodrigo Cuautle, Abraham Sánchez and Maria Osorio

Facultad de Ciencias de la Computación, BUAP
14 Sur esq. San Claudio, CP 72570
Puebla, Pue., México
asanchez, aosorio@cs.buap.mx

Abstract. This paper presents a motion planner for mobile robots in dynamically changing environments with both static and moving obstacles. This planner is based on lazy PRM method and the reactive control by DVZ (Deformable Virtual Zone). The planner first computes a feasible free-collision path with respect to the static obstacles, using the lazy PRM method. Then, it uses the reflex commands in order to avoid dynamic changes. Experimental results are discussed to show the effectiveness of the proposed planner.

1 Introduction

The research in robot motion planning can be traced back to the late 60's, during the early stages of the development of computer-controlled robots. Nevertheless, most of the effort is more recent and has been conducted during the 80's. Within the 80's, roboticians addressed the problem by devising a variety of heuristics and approximate methods. Motion planning can be split into two classes: holonomic motion planning and non-holonomic motion planning.

In non-holonomic motion planning, any path in the free configuration space does not necessarily correspond to a feasible one. Non-holonomic motion planning turns out to be much more difficult than holonomic motion planning. This is a fundamental issue for most types of mobile robots.

From path planning to trajectory control, the motion planning problem for mobile robots has been thoroughly investigated in the case of structured environments. Moving among unknown or badly modeled environments, practically induces the necessity of taking unscheduled and dynamic events into account and reacting as the living beings would do. Therefore, reactive behaviors play a fundamental role when the robot has to move through unstructured and dynamic environments.

Artificial reflex actions for mobile robots can be defined as the ability to react when unscheduled events occur, for instance when they move in unknown and dynamic environments. For the last fifteen years, the scientific community has been interested in the problem of reactive behaviors for collision avoidance in the domain of mobile robots [1], [2]. Another important approach that it deals

© A. Gelbukh, S. Torres, I. López (Eds.)
Advances in Artificial Intelligence
Research in Computing Science 20, 2006, pp. 155-166

with artificial reflex actions, is the potential method developed by O. Khatib, many years ago [3].

Probabilistic roadmap method (PRM) is a general planning scheme building probabilistic roadmaps by randomly selecting configurations from the free configuration space and interconnecting certain pairs by simple feasible paths. The method has been applied to a wide variety of robot motion planning problems with remarkable success [4], [5]. The adaptation of PRM planners to environments with both static and moving obstacles has been limited so far. This is mainly because the cost of reflecting dynamic changes into the roadmap during the queries is very high. On the other hand, single-query variants, which compute a new data structure for each query, deal more efficiently with highly changing environments. They however do not keep the information reflecting the constraints imposed by the static part of the environment useful to speed up subsequent queries.

In this trend, this work aims at providing a practical planner that considers reflex actions and lazy techniques to account for planning with changing obstacles. The paper is organized as follows. Section II gives an overview of the DVZ principle. Section III explains the details of the proposed planner. The performance of the planner is experimentally evaluated in Section IV. Finally, the conclusions and future work are presented in Section V.

2 The DVZ Principle

This section describes the DVZ principle. We assume that the mobile robots has no model of its surrounding space but can measure any intrusion of information (proximity-type information) at least in the direction of its own motion. The vehicle is protected by a risk zone while the deformations of the latter are directly used to trigger a good reaction.

The robot/environment interaction can be described as a deformable virtual zone (DVZ) surrounding the robot. The deformations of this *risk zone* are due to the intrusion of proximity information and controls the robot interactions. The *robot internal state* is defined to be a couple (Ξ, π), where the first component Ξ is called the *interaction component*, which characterizes the geometry of the deformable zone and the second component π characterizes the robot velocities (its translational and rotational velocities). In the absence of intrusion of information, the DVZ, denoted by Ξ_h is supposed to be a one-one function of π. The *internal control*, or reactive behavior is a relation ρ, linking these two components, $\Xi_h = \rho(\pi)$. In short, the risk zone, disturbed by the obstacle intrusion, can be reformed by acting on the robot velocities.

The geometric interaction between the moving n-dimensional robot and its moving n-dimensional environment, that is, the deformable zone surrounding the vehicle, can be viewed as an imbedding of the $(n-1)$-dimensional sphere S^{n-1} into the Euclidean n-dimensional space $\mathbb{R}^n$.

The main interest in the use of this formalism lies in the fact that each imbedding of S^{n-1} can be continuously transformed into another imbedding.

Thus, the deformations of the risk zone due to the intrusion of obstacles in the robot workspace or to the modifications of the robot velocities π (through the relation $\Xi = \rho(\pi)$) lead to the same mathematical entity (the imbedding). Fig. 1 shows different cases of the one-sphere deformations. These zones represent the various shapes of the DVZ, depending on the translational and rotational velocities of the robot. The first diagram illustrate a deformed DVZ due to the presence of an obstacle. The remaining diagrams show how the mobile robot can rebuild its DVZ, (b) by reducing the translational velocity, (c) by turning to the right, or (d) by turning to the left.

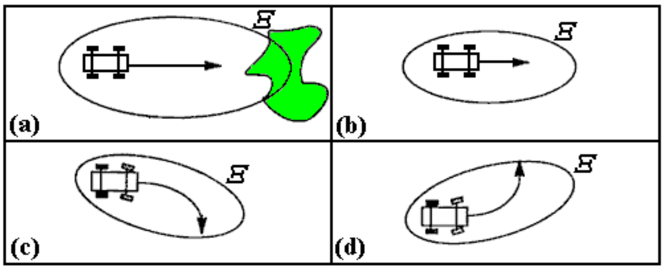


Fig. 1. Deformations of a 1-dimensional DVZ

The first cause of deformation in an interaction component is the information of intrusion due to the proximity of moving obstacles. The second cause is the internal control ($\Xi_h = \rho(\pi)$) of the robot for compensating this intrusion. The control of the internal state is done by comparing a reference interaction component Ξ_o with the deformed component Ξ. This reference depends on the accepted risk taken by the vehicle and is a matter of choice. Therefore, the reactive behavior can be modeled by a two-fold scheme. The control problem consists in generating the second deformation by internal control, with two possibilities:

- by integrally rebuilding the initial state interaction component Ξ through an action on the robot rotation (dynamic avoidance),
- by modifying the robot velocities to attain another acceptable stable state Ξ'.

2.1 Derivation of The State Equation

This subsection provides the general framework for the derivation of the state equation. This equation is formally seen as a two-fold control differential equation and imbedded in the theory of differential games.

Let $\chi = \binom{\Xi}{\sigma}$ be the vector that represents the internal state of the robot and let $\mathcal{E}$ be the state space, which is the set of all the vectors χ. The DVZ is

defined by $\Xi = \begin{pmatrix} \Xi_1 \\ \Xi_2 \\ \vdots \\ \Xi_c \end{pmatrix}$ and the robot velocities vector σ is defined by $\sigma = \binom{v}{\theta}$.
Where each component Ξ_i is the norm of the vector corresponding to the border's distance in the DVZ. These vectors belong to the straight lines that correspond to the main directions of the c proximity sensors c_i. Generally speaking, we assume that we can control the derivative ϕ of a function π for the robot velocities σ. Therefore, the control vector will be written

$$\phi = \dot{\pi} \tag{1}$$

Let $\mathcal{H}$ be the set of all internal states χ_h whose DVZ is not deformed. This set induces an equivalence relation in $\mathcal{E}$, defined by

$$\chi^1 \underset{\mathcal{H}}{\sim} \chi^2 \Leftrightarrow \chi_h^1 = \chi_h^2 \tag{2}$$

where χ_h^i is the internal state corresponding to the state χ^i but without any deformation due to the information of intrusion. In the equivalence class $[\chi]$, the vector χ_h is a one to one function for the vector π:

$$\chi_h = \rho(\pi) \tag{3}$$

which can be written (by separation of the two sets of variables)

$$\begin{cases} \Xi_h = \rho_\Xi(\pi) \\ \sigma = \rho_\sigma(\pi) \end{cases} \tag{4}$$

The derivative of eq. (4) provides the state equation when no deformation occurs (when the state vector stays on $\mathcal{H}$):

$$\dot{\chi}_h = \rho'(\pi)\dot{\pi} = \rho'(\pi)\phi \tag{5}$$

This equation is the first part of the general state equation. If we now consider deformations of the DVZ, due to the intrusion of information, we will obtain the second part of the state equation. To do it, we will denote the deformation of the state vector by $\triangle$ and study the variations of this deformation with respect to the intrusion of information. This new vector represents the deformed DVZ, which is defined by

$$\Xi = \Xi_h + \triangle \tag{6}$$

Let $I = \begin{pmatrix} I_1 \\ I_2 \\ \vdots \\ I_c \end{pmatrix}$ be the c-dimensional intrusion vector, where $I_i = d_{imax} - d_i$.
The sensor provides the measure $d_i = d_{imax}$, in the absence of obstacles.

Let $\triangle = \begin{pmatrix} \triangle_1 \\ \triangle_2 \\ \vdots \\ \triangle_c \end{pmatrix}$ be the c-dimensional deformation vector, where

$$\triangle = \alpha(\Xi_h, I) \tag{7}$$

with $\alpha(\Xi_h, I)$ being a c-dimensional vector. Each element $\triangle_i$ is defined by

$$\triangle_i = \alpha(d_{h_i}, I_i) \begin{cases} 0 & \text{if } d_i > d_{h_i} \\ d_{h_i} - d_i & \text{if } d_i \leq d_{h_i} \end{cases} \tag{8}$$

where d_{h_i} is an element of the intact DVZ (Ξ_h). By differentiating ec. (6) with respect to time, we get

$$\dot{\triangle} = \frac{\partial \alpha}{\partial \Xi_h}(\Xi_h, I)\dot{\Xi}_h + \frac{\partial \alpha}{\partial I}(\xi_h, I)\dot{I} \tag{9}$$

By letting $\psi = \dot{I}$, and using eqs. (4), (5), (6) and (9), we obtain the next control equation

$$\begin{cases} \dot{\Xi} = \left(\frac{\partial \alpha}{\partial \Xi_h}(\Xi_h, I) \times \rho'_{\Xi}(\pi) + \rho'_{\Xi}(\pi) \right) \phi + \frac{\partial \alpha}{\partial I}(\Xi_h, I)\psi \\ \dot{\sigma} = \rho'_{\sigma}(\pi)\phi \end{cases} \tag{10}$$

with

$$\begin{cases} \dot{\Xi} = \rho'_{\Xi}(\pi)\phi \\ \dot{\pi} = \phi \\ \dot{I} = \psi \end{cases} \tag{11}$$

The inputs of eq. (10) are the two control vectors ϕ and ψ. The first comes from the control module of the robot and the second from the environment itself.

3 A Reactive Lazy PRM Planner

The proposed planner integrates the lazy PRM planning method and the reactive control by DVZ in the following way: a collision-free feasible path for a mobile robot is calculated by the lazy PRM method, the robot starts moving (under the permanent protection of its DVZ), in the absence of dynamic obstacles, the control is performed by the lazy PRM method and does not require reflex commands. If there are dynamic obstacles in its path, the reactive method takes the control and generates commands to force the robot to move away from the intruder obstacles and gives back its DVZ to the original state.

In this point, the robot has lost its original path, and it is necessary to search for a reconnection path to reach its goal. The new path found is a single collision-free curve of Reeds & Shepp. If the attempt of reconnection is successful, the robot executes its new path towards the goal. The new alternative path was obtained with the lazy PRM method by using the information stored in the current robot's configuration, but if a deformation appears, the processes are interrupted by reflex actions that forces the planner to go back to the previous state. The algorithm can finish in three forms: i) the robot executes its path successfully, ii) the reflex action is not sufficient and a collision occurs, or iii) the robot does not find an alternative path to conclude its task. Figure 2 shows a high-level description of the proposed approach.

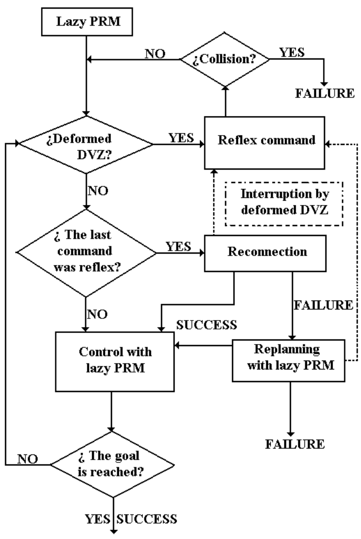


Fig. 2. High-level description of our planner

The following subsections detail the most important stages of the proposed planner. For more details, you can see [2].

3.1 Lazy PRM for Nonholonomic Robots

Lazy PRM approach for nonholonomic motion planning was presented in [6]. The algorithm is similar to the work presented by Bohlin and Kavraki [7], in the sense that the aim of our approach is to minimize the number of collision checks and calls to local method while searching the shortest feasible path in the roadmap.

Once a start-goal query is given, the planner performs A^* search on the roadmap to find a solution. If any of the solution edges are in collision, they are removed from the roadmap and then A^* search is repeated. Eventually, all edges may have to be checked for collisions, but often the solution is found before this happens. If no solution is found, more nodes may need to be added to the roadmap [8]. The most important advantage of this approach, is that the collision checking is only performed when needed. In this case, all edges don't have to be collision checked as in the original PRM case. Experiments show that only a very small fraction of the graph must be explored to find a feasible path in many cases. Single queries are handled very quickly, indeed, no preprocessing is required.

3.2 Generation of Reflex Commands

The DVZ form is used in our experimental design according to equations (1) to (12).

$$d_{h_i} = K_1 V_1^2 \cos^2(\beta_i + K_2\dot{\theta}) + d_i^{sec} \tag{12}$$

where K_1 and K_2 are constants, V_1 and $\dot{\theta}$ are the velocities of the robot, β_i is the angle of the sensor c_i with respect to the transverse axis of the robot, and d_i^{sec} is a safe distance in the direction of the sensor c_i.

For the first case in equation (8), $(d_i > d_{h_i})$, the DVZ is not deformed by the environment, the control is performed by the lazy PRM method and the reflex actions are not require. For the second case, when $(d_i < d_{h_i})$, a reflex action is necessary, the path executed by the lazy PRM method is suspended and the robot control is taken by the DVZ method. When the DVZ is in control, it has the task of taking the robot to a state free of deformations, indicating the kinematics attitudes that should continuously have the robot. These attitudes constitute the vector π, and the control is adapted in the following way.

Let $f_i[n]$ a vector in the direction of the sensor c_i to be defined as

$$f_i[n] = \begin{cases} \triangle_i[n] - \triangle_i[n-1] & \text{if } \triangle_i[n] - \triangle_i[n-1] > 0 \\ 0 & \text{if } \triangle_i[n] - \triangle_i[n-1] \leq 0 \end{cases} \tag{13}$$

Let $F[n]$ be the addition of the vectors $f_i[n]$

$$F[n] = \sum_{i=1}^{c} f_i[n] \tag{14}$$

then, the vector $\pi[n]$ is given by

$$\pi[n] = \begin{cases} V_1[n] = V_1[n-1] + Kv* \parallel F[n] \parallel *sign(\cos(\hat{F}[n])) \\ \dot{\theta} = \dot{\theta}[n-1] + Kt * \sin(\hat{F}[n]) \end{cases} \tag{15}$$

3.3 Reconnection

After a successful reflex action, the mobile robot recovers the intact state of its DVZ, but its initial planned path will be lost (Fig. 3-b). The lazy PRM method needs to have a path to push the mobile robot to the goal and it will be necessary to provide a path for such aim. Due to the high computational cost of a complete replanning, the method will avoid it by executing a process that uses a single collision-free Reeds & Shepp curve [9] (Fig. 3-c) to reconnect with the planned path.

Initially, the algorithm tries a local path that it is interrupted by a dynamic object. The algorithm will execute a reflex action in order to reconnect with the closest point that is collision-free in the original path. If it can not reconnect after a certain number of attempts, maybe because the possible reconnection paths are blocked with obstacles, the robot will remain immovable for a certain time before executing a new attempt (see Fig. 3-d). The process will be repeated several times, but if the DVZ was deformed by an intrusion, the reconnection process will be modified and will execute the reflex commands.

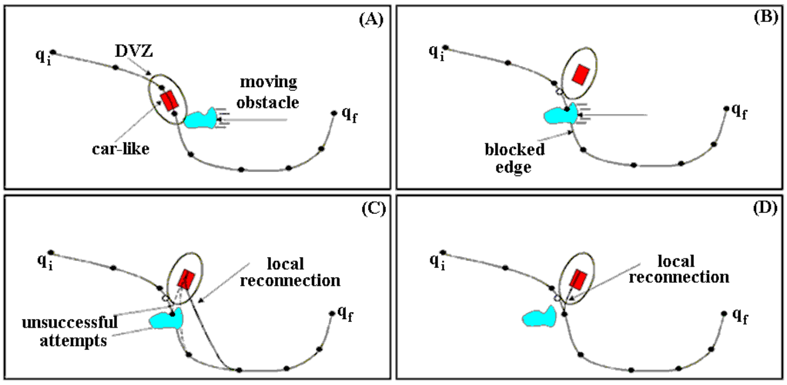


Fig. 3. Cases of the reconnection process

3.4 Replanning

If the reconnection attempts fails, it may happen that paths are blocked by many dynamic objects, or a moving object is parked obstructing the planned path. In this case, the planner executes the lazy PRM method (the initial configuration is the current configuration in the robot). The lazy PRM will be called several times until it returns a collision-free path. If after some attempts a collision-free path can not be found, the planner reports failure.

The model cannot distinguish if an intrusion is caused by a moving or a static obstacle because the DVZ method does not use any model of the environment. To solve this problem, it is necessary to use an auxiliary image that represents the environment and it is updated every time the replanning or reconnection procedures are called. When the sensors in the robot detect an obstacle that deforms the DVZ, the intruder object coordinates are revised to see if there was already an obstacle, registered in the auxiliary image; if this is the case, the system assumes the presence of a fixed obstacle and there is no need for a reflex action, otherwise, it will certainly assume that the object is in movement.

4 Experimental Results

This section presents experimental results for car-like robots obtained by using the planner described above to different scenes. The planner has been implemented in Builder C++ and the tests were performed on an Intel © Pentium IV processor-based PC running at 2.4 GHz with 512 MB RAM.

After having executed our planner in different scenes, in the majority of the cases the motion planning problem is solved satisfactorily. Our planner produces a first roadmap by sampling configurations spaces uniformly. It computes the shortest path in this roadmap between two query configurations and test it for collision. The robot starts moving under the permanent protection of its DVZ. In absence of dynamic obstacles, the robot does not require reflex commands and the control is executed with lazy PRM. If there are dynamic obstacles in its path, the reactive method takes the control and generates commands to force the robot to move away from the intruder obstacles and gives back its DVZ to the original state. The moving obstacles have a square form and move at constant velocity in straight line. Whenever they collide with another object they assume a new random direction in their movement.

Fig. 4 shows an environment that contains a circular obstacle, the scene is completely closed. This example also contains 10 dynamic obstacles moving randomly at the same velocity than the mobile robot.

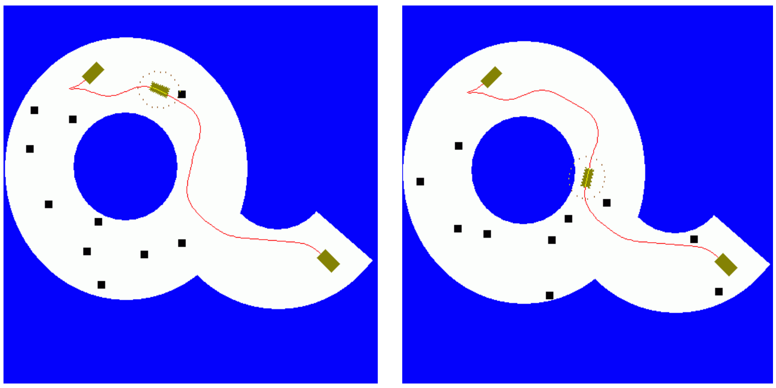


Fig. 4. An example of a query and its solution path in an environment with 10 moving obstacles. The robot starts moving under the permanent protection of its DVZ

In order to evaluate the performance of the planner, we performed tests on the environment of Fig. 5 for several roadmap sizes and different number of moving obstacles. The different settings are summarized in the tables 1, 2 and 3.

In fact, the method's performance can be considered satisfactory if it presents a fast planning phase, reflex actions based on sensors that do not require expensive algorithms, an effective process of reconnection performed in milliseconds,

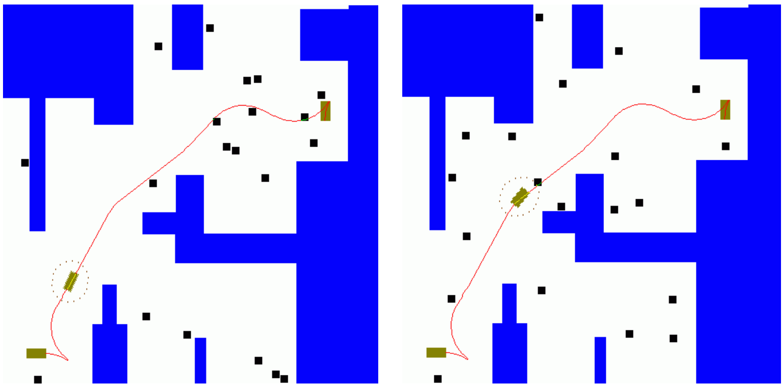


Fig. 5. An environment composed of narrow passages with 20 dynamic obstacles

Table 1. Performance data for Lazy PRM

Settings	50 nodes	50 nodes	50 nodes	100 nodes	100 nodes	200 nodes
Steering angle	25	35	45	35	25	45
Graph building	0.007	0.006	0.01	0.01	0.01	0.02
Graph searching	0.003	0.004	0.03	0.01	0.005	0.03
Coll. checking	380	425	1300	650	365	1481
Total Time (s)	0.01	0.01	0.04	0.02	0.015	0.05

Table 2. Performance data with 20 moving obstacles

Reconnections	Time for reconnection	Replanning	Time for replanning	Collision	Success
29	0.010	0	0.000	no	ok
39	0.015	2	0.000	no	ok
57	0.023	1	0.000	no	ok
5	0.010	0	0.000	no	ok
37	0.012	3	0.000	ok	no

Table 3. Performance data with 15 moving obstacles

Reconnections	Time for reconnection	Replanning	Time for replanning	Collision	Success
3	0.010	0	0.000	no	ok
10	0.020	0	0.000	no	ok
36	0.030	1	0.000	no	ok
40	0.040	2	0.000	no	ok
12	0.010	0	0.000	ok	no

and a process of replanning that is executed if the Lazy PRM and DVZ' s parameters are appropriate.

The planning time is reduced due to the incomplete collision detector whose work is complemented with the robot's sensors during the path execution. On the other hand, the assignation of direction angles to the nodes that conform the shortest paths obtained by the algorithm A^*, produces curves that allow the algorithm to omit the optimization process (i.e., the smoothing process). With respect to the reconnection process, the paths obtained with the planner are conformed by a single Reeds & Shepp curve and based on the incomplete collision detector, making short the time and close to optimal the curves obtained with the algorithm. Since the reflex actions are provided by the DVZ method, it is possible to interrupt the reconnection and replanning processes if necessary, without incurring in bigger problems. If the execution's parameters for the Lazy PRM and DVZ methods are adapted, the replanning process will not be called very often and will be successful in the absence of narrow passages.

Figure 6 presents a case where the reflex actions were not sufficient. The presence of narrow passages is an important problem to being considered.

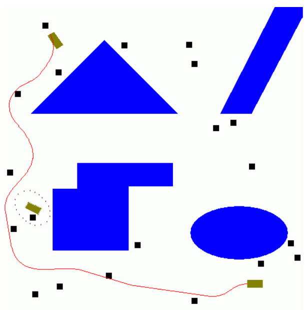


Fig. 6. The reflex actions were not sufficient, the mobile robot collides with a moving obstacle

5 Conclusions and Future Work

Even in the absence of obstacles, planning motions for nonholonomic systems is not an easy task. So far, no general algorithm exists for planning the motions of any nonholonomic system, that guarantees to reach a given goal. The only existing results deal with approximation methods, that is, methods that guarantees to reach a neighborhood of the goal, and exact methods for special classes of nonholonomic systems. Obstacle avoidance adds a second level of difficulty: not only does one have to take into account the constraints imposed by the kinematic nature of the system, but also the constraints due to the obstacles. It appears necessary to combine geometric techniques addressing the obstacle avoidance with control techniques addressing nonholonomic motions.

The results obtained in the evaluation of the reactive lazy PRM planner proposed in this work, show the importance of finding a solution for the complex problem of motion planning for nonholonomic robots in dynamic environments.

A reactive lazy PRM planner for dynamically changing environments is presented in this paper. Although some promising results are shown in its present form, the planner could be improved in a number of important ways. This approach can be extended to use real robots and to solve the problem posed by small static obstacles. Besides, some cases where the reflex action was not sufficient to avoid collisions were observed during the evaluation tests. Theses cases are difficult because they require a more intelligent behavior in order to avoid the robot to be trapped. In those cases, it can be necessary to add a process that computes the trajectories of moving objects and corrects the path in real time.

Finally, a very interesting topic in robotics, is the study of non-structured environments. This methodology can be extended to solve those cases.

References

1. Zapata R. "Quelques aspects topologiques de la planification de mouvements et des actions réflexes en robotique mobile", *Thèse d'Etat*, University of Montpellier II (in French), (1991)
2. Cuautle P. R. "Planificación de movimientos en ambientes dinámicos utilizando un enfoque lazy PRM reactivo", *Tesis de Maestria, FCC-BUAP*, (in spanish), (2005)
3. O. Khatib. "Real-time obstacle avoidance for manipulators and mobile robots", *International Journal of Robotic Research*, Vol. 5, No. 1, (1986) 90-98
4. Kavraki L. E., Švetska P., Latombe J. C., and Overmars M. H. "Probabilistic roadmaps for path planning in high-dimensional configuration spaces", *IEEE Transactions on Robotics and Automation*, Vol. 12, No. 4, (1996) 566-580
5. Švestka P. and Overmars M. H. "Motion planning for car-like using a probabilistic learning approach", *The International Journal of Robotics Research*, Vol. 16, No. 2, (1997) 119-143
6. Sánchez A., Zapata R., Arenas A. "Motion planning for car-like robots using a lazy probabilistic roadmap method", *Advances in Artificial Intelligence, Springer Verlag, Vol. 2313*, (2002) 1-10
7. Bohlin R. and Kavraki L. E. "Path planning using lazy PRM", *Proc. of the IEEE Robotics and Automation Conference*, (2000) 521-528
8. Lanzoni C., Snchez L. A., and Zapata, R. "A single-query motion planner for car-like nonholonomic mobile robots", *3rd International Symposium on Robotics and Automation (ISRA)*, (2002) 267-274
9. Reeds J. A. and Shepp R. A. "Optimal paths for a car that goes both forward and backwards", *Pacific Journal of Mathematics*, Vol. 145, No. 2, (1990) 367-393
10. Sánchez A., Zapata R., and Lanzoni C. "On the use of low-discrepancy sequences in nonholonomic motion planning", *Proc. of the IEEE Robotics and Automation Conference*, (2003) 3764-3769

RMP3D: A multipurpose Platform for Motion Planning

Josué Sánchez, Abraham Sánchez, Miguel Rodriguez, Xochitl Hernández and Gerardo García

Facultad de Ciencias de la Computación, BUAP
14 Sur esq. San Claudio, CP 72570
Puebla, Pue., México
asanchez@cs.buap.mx

Abstract. This paper reports on RMP3D, a multipurpose software platform dedicated to motion planning. RMP3D makes it possible to set up motion planning experiments very quickly. The generality comes from a dedicated software architecture allowing a rapid design of motion planners. Comprehension of concepts and algorithms involved in the robotics field can be improved through the use of an interactive visualization tool. We present an interactive tool for visualizing and editing motion planning environments, problem instances, and their solutions. RMP3D is specialized for model-based randomized planners such as Probabilistic Roadmap (PRM) methods. The paper focuses on recent results obtained in robot motion planning, graphics animation, and behavioral animation.

1 Introduction

Robot motion planning refers to the ability of a robot to automatically plan its own motions to avoid collision with the physical objects in its environment. Such a capability is crucial, since a robot accomplishes tasks by physical motion in the real world. This capability would be a major step toward the goal of creating autonomous robots. This observation has motivated much research in robot motion planning. The approaches to robot motion planning can be roughly divided into two categories: the classical motion planning or model-based motion planning, and sensor-based planning. The first approach, assumes that the robot system has an explicit representation of the robot's environment. On the other hand, in the second approach, the environment is unknown and the robot is guided directly from the sensory input without constructing internal representation for the environment. This work considers solely the model-based motion planning problem.

Motion planning problem is typically solved in the configuration space ($\mathcal{C}$), in which each placement of the robot is mapped as a point. The free configuration space, $\mathcal{F}$, is the subset of $\mathcal{C}$ at which the robot does not intersect any obstacle. The dimension of $\mathcal{C}$ depends on the degrees of freedom (dof) of the robot, which can be high. An exact computing of a high-dimensional configuration space is impractical. For this reason, a large family of model-based planners has been

© A. Gelbukh, S. Torres, I. López (Eds.)
Advances in Artificial Intelligence
Research in Computing Science 20, 2006, pp. 167-178

developed [1], [2], [3], [4]. These algorithms have been successfully used to solve challenging problems.

The most popular paradigm for model-based motion planning is the Probabilistic Roadmap Method (PRM) [1]. The key idea of the PRM is to randomly distribute a set of nodes in $\mathcal{C}$ and then connect these nodes using a simple local planner, to form a graph (or a tree) known as a *roadmap*. An important property of a roadmap is that it provides a good approximation of the connectivity of the $\mathcal{F}$. If the roadmap is successfully capturing this connectivity, motion planning may be reduced to a graph search.

Motion planning has application in many other areas, such as assembly planning, design for manufacturing, virtual prototyping, computer animation, medical surgery simulation, computational biology, etc. As stressed by Latombe [5], non-robotics applications (e.g., graphics animation, surgical planning and computational biology) are growing in importance and are likely to shape future motion planning research at least as much as robotics.

In Section II, we present the software architecture of RMP3D. The following sections give practical results obtained in problems arising in fields as diverse as robot motion planning, planning motions for animated characters and behavioral animation. Finally, Section VI presents the conclusions and future work.

2 RMP3D Architecture

RMP3D is composed of diverse modules associated with functionalities such as the modeling of the mechanical systems (geometric modeling, steering methods), geometrical tools (collision detection, distance calculation), motion planning, and a graphic interface that allows to define the problems, call the algorithms, and to display the produced results. Fig. 1 shows the structure of the motion planning software RMP3D.

- the modeling module enables the user to describe mechanical systems and environments.
- the geometric tools for the collision detection algorithms.
- the steering methods allows to compute local paths satisfying the kinematic constraints of the mechanical systems.
- the planning algorithms module contains many procedures based on randomized techniques such as PRM, Lazy PRM, Gaussian PRM.

The following steering methods are actually integrated within RMP3D:

- **Linear** computes a straight line segment between two configurations, this method works for any holonomic system like a manipulator arm.
- **Nonholonomic** computes smooth paths for both car models, Reeds & Shepp [6] and Dubins [7] or articulated mobile robots.

Other methods could be easily integrated into this library. They can also be combined to design more complex steering methods for mechanical systems subjected to different motion constraints.

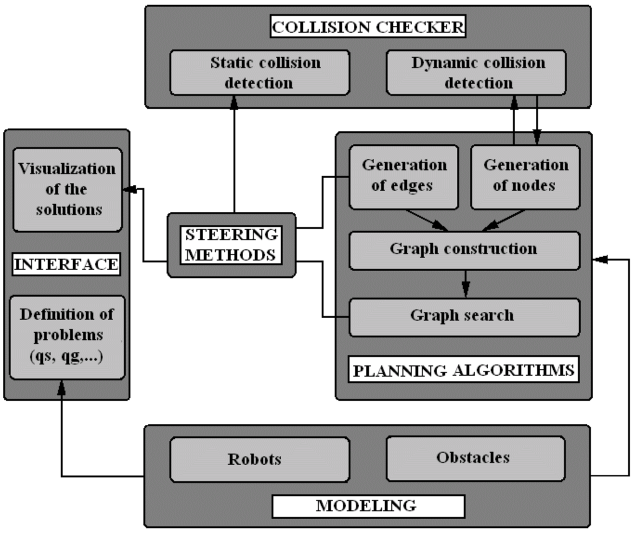


Fig. 1. Architecture of RMP3D

The planning module integrates three of the randomized techniques proposed recently. These techniques are based on the probabilistic roadmap methods that first construct a roadmap connecting collision-free configurations picked at random, and then use this roadmap to answer multiple or single queries. Additionally to the planning methods, we added some generators of determinist samples like Halton, Hammersley, Faure, Sobol, and Sukharev grid.

- Basic-PRM is based on the basic PRM scheme [1]. The key idea of this scheme is to randomly distribute a set of nodes in $\mathcal{C}$ and then connect these nodes using a simple local planner (or a steering method), to form a graph (or a tree) known as a roadmap.
- Gaussian-PRM is meant to add more samples near obstacles [8]. The idea is to take two random samples, where the distance between the sampless is chosen according to a Gaussian distribution. Only if one of the samples lies in the $\mathcal{F}$ and the other lies in $\mathcal{C}_{obst}$ do we add the free sample. It has been shown that this leads to a favorable sample distribution.
- Lazy-PRM [9], the idea is not to test whether the paths are collision free unless they are really needed. The goal of this variant is to minimize the number of collision checks. The rational behind this is that for most paths we only need to consider a small part of the graph before a solution is found.

The collision checker integrated is PQP[1] for determining whether a given path is collision-free or not (this is performed by multiple calls to the interference detection algorithm). PQP is a library for performing three types of proximity queries on a pair of geometric models composed of triangles.

[1] A collision detection package from University of North Carolina at Chapel Hill

All techniques were integrated in a single motion planning system called RMP3D, implemented in Builder C++ under Windows XP. RMP3D automates conducting experiments, i.e. statistics are automatically generated and processed, decreasing the chance for errors. All experiments were run on a 2.4 GHz Pentium 4 processor with 512 MB internal memory.

3 Robot Motion Planning

In this section, we discuss the field of model-based motion planning. In contrast to methods that construct boundary representations of configuration space obstacles, model-based methods[2] use only information from a collision detector as they search the configuration space. The simplicity of this approach, along with increases in computation power and development of efficient collision detection algorithms, has resulted in the introduction of a number of powerful motion planning algorithms, capable of solving challenging problems with many degrees of freedom (dofs).

All of the recent methods rely on some method for generating samples over the configuration space. Typically, the samples are taken at random from a statistically uniform distribution; however, this method of sampling is not as uniform as some deterministic methods [10]. Several weakness of random sampling were shown in the context of the PRM in [11], [12], [13].

Despite the success of PRM planners, narrow passages in a robot's configuration space create significant difficulty for PRM planners. A narrow passage is a small region whose removal changes the connectivity of the free space. There are several sophisticated sampling strategies that can alleviate this difficulty, but a satisfactory answer remains elusive. Indeed, many of them only solved partially the motion planning problem, e.g., the case of free-flying robots. We claim that deterministic sampling is suitable to capture the connectivity of configuration spaces with narrow passages [10]. Figures 2 and 3 show the stages of the PRM approach.

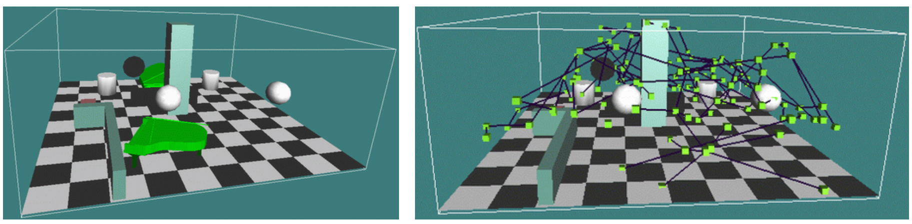


Fig. 2. The start and goal configurations for the piano mover problem and the roadmap graph

[2] These methods are called sampling-based methods

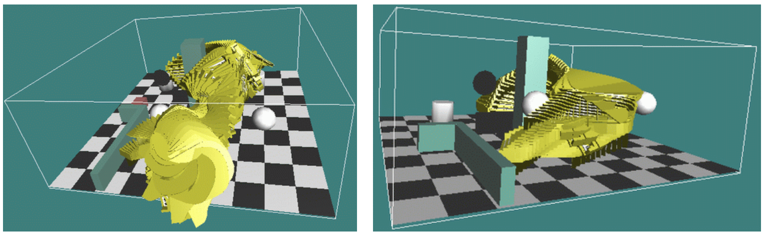


Fig. 3. An un-smoothed path for the piano problem and the path obtained by smoothing

Figures 4 and 5 show some resolute examples, the examples correspond to free-flying objets in a three-dimensional workspace, the objects have six dofs, three translational degrees and three rotational degrees.

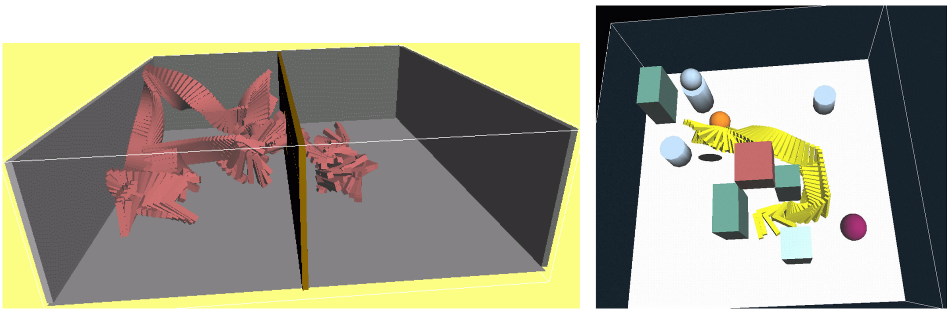


Fig. 4. Left, a famous test scene, in which an elbow-shaped robot passes through a small opening. Right, the same elbow-shaped robot in other environment

The PRM approach has been applied to many different types of motion planning problems involving different kinds of robots. Unfortunately, the different improvements suggested are difficult to compare. Each author used his or her own implementation of PRM. Also the effectiveness of one technique sometimes depends on choices made for other parts of the method.

With this tool, we provide a comparative study of a number of PRM techniques, all implemented in a single system and run on the same test scenes and on the same computer. In particular, we can compare basic sampling techniques, steering methods and node adding techniques.

4 Planning Motions for Animated Characters

The design of autonomous characters capable of planning their own motions continues to be a challenge for computer animation. We present a novel method

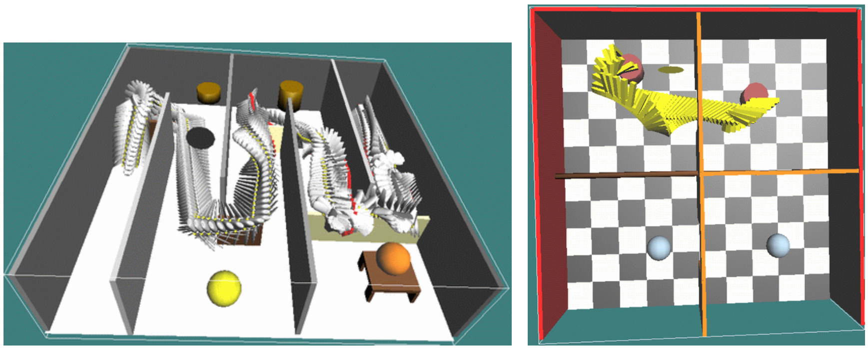


Fig. 5. Two scenes with narrow passages. Left, a spaceship in a labyrinth. Right, an object with complicated geometry in four roms with lots of space and with narrow doors between them

for animating human characters. Our method is integrated in a probabilistic roadmap method scheme. The navigation of the human character in this environment is modeled by a composition of Bezier curves. The controllers are based on motion data capture techniques.

Given start and goal positions in a virtual environment, our objective is to find a sequence of motions of a human character to move from the start and to the goal. Conventional motion planning techniques in robotics typically generate very efficient mechanical movements rather that lifelike natural-looking motions desired in computer animation applications. On the other hand, motion editing techniques in computer graphics are not equipped with a high-level planning capability to yield a desired motion. To rapidly plan convincing motions of the human-like character with high-level directives.

Although the motion planning and following concept generally applies to many types of characters and motions, we will concentrate on generating walking or running motions for a human-like character. We would like the character's motion to be smooth and continuous, natural-looking, and follow the computed path as closely as possible. For more details on this technique, you can review the work in [14]. Our procedure consists of the following three steps: roadmap construction, roadmap search, and motion generation. Figure 6 shows a high-level description of the proposed approach.

A result of complete trajectory (composition of several local paths) is presented on figure 7. One can notice several specificities in this result: the model strictly respects the initial and final configurations required. The structure of the character is modeled in two levels. Pelvis and legs are used for the locomotion, all the 18 dofs are said to be active dofs. The 34 other ones are said to be reactive dofs, they deal with the control of the arms and the spine. The pelvis is the root of five kinematics chains modelling respectively the arms, the legs and the spine.

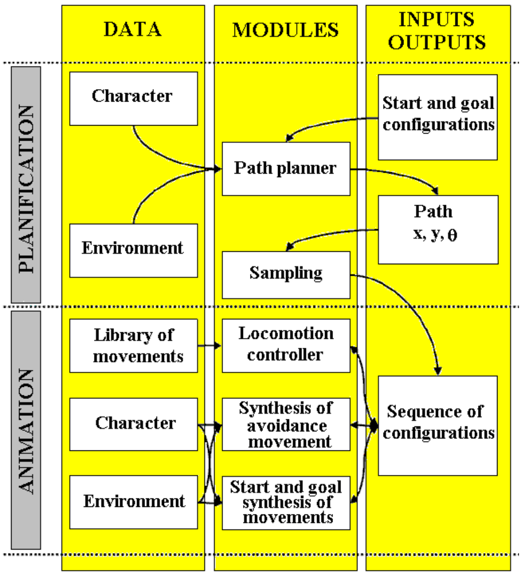


Fig. 6. High-level description of our approach

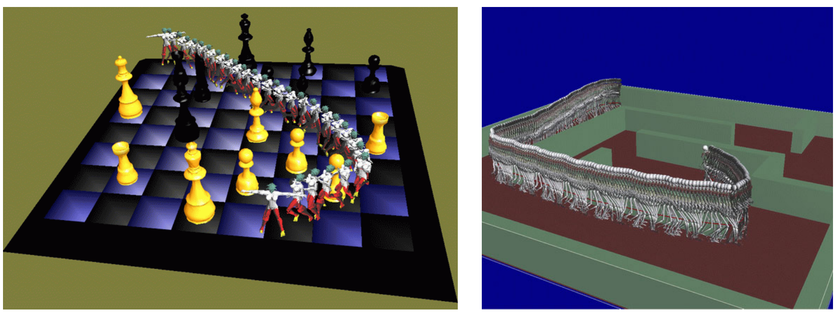


Fig. 7. Walking through the chessboard and the labyrinth

Figure 8 illustrates the result of the warping module, the goal of the warping module is to locally modify the animation of the upper bodies of the character (arms and spine) when collision occur in the animation produced by the module locomotion-controller.

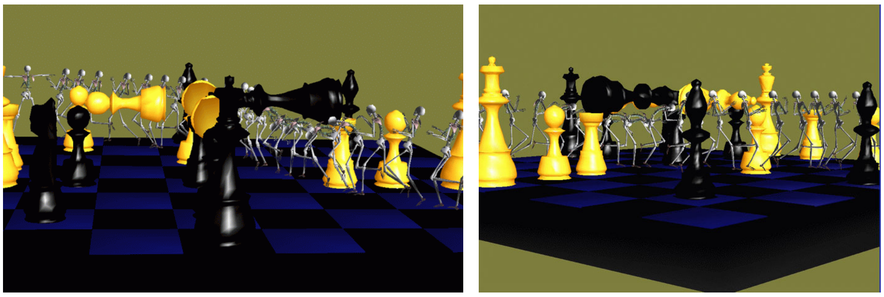


Fig. 8. Two snapshots to show the effectiveness of the warping module

The approach satisfies some computer graphics criteria such as realistic results, collision-free motion in cluttered environments and low response time. The approach has been implemented in our RMP3D architecture and successfully demonstrated on several examples. The combination of randomized motion planning techniques and motion capture editing techniques offer promising results.

We have presented a solution for digital actors locomotion problems. The solution insists on the modularity of the approach. Each component can be modified or replaced with ease. Our locomotion planner is still to be enhanced: we want to introduce the ability for the digital actor to change its locomotion behavior by crouching, crawling, etc.

Motion capture system have been widely used during the last few years for creating new animations of virtual humans. These approaches assume that realistic motions can be obtained by editing and tuning library of motion data. The computational cost is low ($O(n)$) because each articulation is computed separately. Although several convincing animations have been designed using these methods, the parameter control is not simple: a number of trial and error iterations is often needed before obtaining the desired result.

5 Behavioral Animation

Behavioral animation is a type of procedural animation, which is a type of computer animation. In behavioral animation an autonomous character determines its own actions, at least to a certain extent. This gives the character some ability to improvise, and frees the animator from the need to specify each detail of every character's motion. These improvisational characters are, in the words of Ann

Marion: "puppets that pull their own strings". An early example of behavioral animation was the 1987 boids model of bird flocking [15]. While in some limited sense autonomous characters have a mind, their simplistic behavioral controllers are more closely related to the field of artificial life than to artificial intelligence.

Like particle systems, behavioral animation is a vague term which refers to a number of techniques. Also like particle systems, behavioral animation is used to control the motion of many objects automatically. The primary difference is in the objects being animated. Instead of simply procedurally controlling the position of tiny primitives, motion is generated for actors with orientations, current state, and interactions. Behavioral animation has been used to animate flocks, schools, herds, and crowds. All of these require interaction between a large number of characters with relatively simple, rule-based motion. Fabric can also be simulated using behavioral techniques.

While many methods to simulate flocking behaviors have been proposed, these techniques only provide simplistic navigation and planning capabilities, because each flock member's behavior depends only its local environment [16], [17].

In this section, we show that complex group behavior can be generated using a roadmap providing global environment information. The roadmap contains topological information and adaptive edge weights that enables the flock to achieve behaviors that cannot modeled with local information alone. We propose new techniques for different group behaviors: homing, goal searching, and traversing narrow areas. We extended ideas from cognitive modeling to create behavior rules in individual flock members and in the roadmap. These embedded behaviors enable the creatures to modify their actions based on their current location and state.

These behaviors exploit global knowledge of the environment and utilize information gathered by all flock members which is communicated by allowing individual flock members to dynamically update the shared roadmap to reflect (un)desirable routes or regions.

Homing behavior consists of two sub-models, one representing the individual behavior of flock members and the other influencing the global behavior of the flock. Once a path is found (with our RMP3D tool), individual flock members follow the path. The path is discretized to subgoals based on and individual flock member's sensor range. Each member keeps track of subgoals and as soon as a subgoal comes within the sensory range the next subgoal becomes the steering direction for the global goal. With other interacting forces from neighboring flock members and obstacles, steering toward the subgoal has the lowest priority, so individual members still move together while moving toward the goal. This results in a flocking toward the goal and avoids getting trapped in local minima.

Goal searching is a type of exploring behavior. We achieve this behavior using a roadmap graph with adaptive edge weights. Each individual member behaves independently from its flock mates and uses the roadmap to wander around. They follow roadmap edges and there are no predefined paths. If they reach a roadmap node with several roadmap edges, they probabilistically choose a roadmap edge to follow based on weight of the edge. The edge weights represent any preferences for the current task, i.e., searching for and reaching the goal.

A naive solution to achieve narrow passage traversal by the flock is to use the homing behavior and to select two nodes as goals, first a node in front of the entrance to the passage and then a node outside the exit from the passage. One drawback is that flock members may bunch up and conflict with each other as they try to move through the passage. A strategy that may avoid the congestion problems of the naive idea is the follow-the-leader strategy. We first assemble the flock in front of the narrow passage, and then select the closest flock member to the entrance to the narrow passage as the leader. The remaining flock members are arranged into a queue that follows the leader.

Figure 10 show some results obtained with our approach. The behavior rules embedded in our roadmaps enable the flocks to modify their actions based on their current location and state. Our simulation results for the three types of behaviors studied show that the performance of the rule-based roadmap behaviors is very close to ideal behaviors that have complete knowledge.

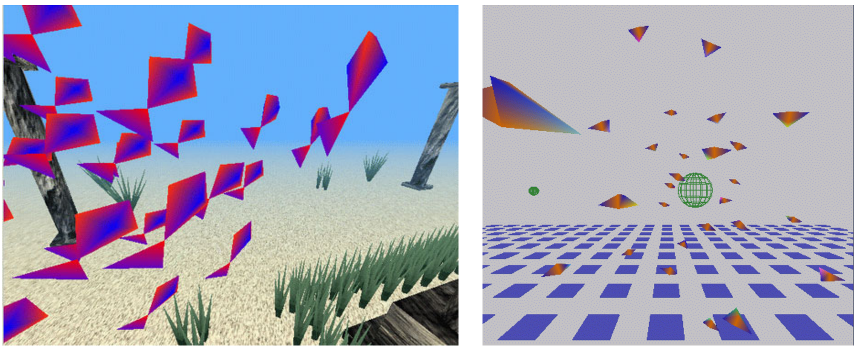


Fig. 9. Complex behaviors, such goal searching can be improved using global information

6 Conclusions and Future Work

Motion planning has been an active research field in robotics for more than 20 years. Within the 80's, roboticians addressed the motion planning problem by devising a variety of heuristics and approximate methods. While complete and deterministic algorithms for motion planning are very time-consuming as the dimension of the configuration space increases, it is now possible to address complicated problems in high dimension thanks to alternative methods (for instance PRM methods) that relax the completeness constraint for the benefit of practical efficiency and probabilistic completeness.

This paper presented the software platform RMP3D developed at the University of Puebla (Facultad de Ciencias de la Computación) for generic multipurpose

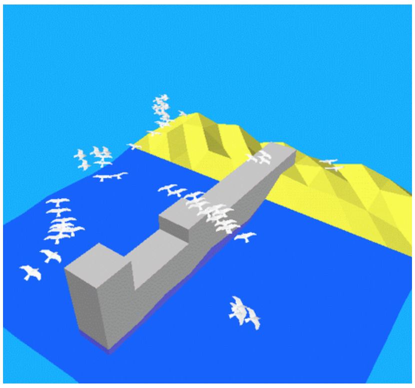


Fig. 10. Complex behaviors, such Homing can be improved using global information

applications. A large number of variant PRM planners can be constructed by combining the algorithmic components presented in our software platform.

RMP3D is a multipurpose tool for visualizing and editing motion planning environments, problem instances, and their solutions. RMP3D offers a self-explanatory graphical user interface that provides functionality that can help us understand robotics and motion planning concepts. We are convinced that researchers and students can take advantage of visualization tools such as RMP3D which in addition to visualization, allows interaction with the robots workspace.

The aim of the projet RMP3D is to develop a general planning software for providing systems with motion planning facilities. The examples shown in the paper illustrate the kind of problems that can be solved today by the algorithms integrated within RMP3D. It remains that additional work still needs to be done for improving the efficacy of the planning techniques. Another challenging issue is to develop more sophisticated planning algorithms (i.e., multiple robots, manipulation planning).

Motion planning applications are emerging today in various domains such as computer graphics, drugs design, medicine...

Finally, our paper does not introduce new algorithms, nor new analysis. It should be viewed as an experience feedback in developing motion planning technology within the context of a well focused application field. The advantages of our platform are many with respect to other proposals, but the most important thing is the level of the modelling (environments and robots), since this is possible by using Inivis AC3D.

References

1. Kavraki L. E., Švetska P., Latombe J. C., and Overmars M. H. "Probabilistic roadmaps for path planning in high-dimensional configuration spaces", *IEEE Transactions on Robotics and Automation*, Vol. 12, No. 4, (1996) 566-580

2. Švestka P. and Overmars M. H. "Motion planning for car-like using a probabilistic learning approach", *The International Journal of Robotics Research*, Vol. 16, No. 2, (1997) 119-143
3. Siméon T., Laumond J. P., and Nissoux C. "Visibility-based probabilistic roadmaps for motion planning", *Journal of Advanced Robotics*, Vol. 14, No. 6, (2000) 477-494
4. Sánchez G., and Latombe J. C. "On delaying colllision checking in PRM planning: Application to multi-robot coordination", *The International Journal of Robotics Research*, Vol. 21, No. 1, (2002) 5-26
5. Latombe J. C. "Motion planning: A journey of robots, molecules, digital actors, and other artifacts", *The International Journal of Robotics Research*, Vol. 18, No. 11, (1999) 1119-1128
6. Reeds J. A., and Shepp R. A. "Optimal paths for a car that goes both forward and backwards", *Pacific Journal of Mathematics*, Vol. 145, No. 2, (1990) 367-393
7. Dubins L. E. "On Curves of minimal length with a constraint on average curvature and with prescribed initial and terminal positions and tangents", American Journal of Mathematics, Vol. 79, (1957) 497-516
8. Boor V., Overmars M. H., and Van der Stappen A. F. "The Gaussian sampling strategy for probabilistic roadmap planners", *Proc. of the IEEE Robotics and Automation Conference*, (1999) 10181023
9. Bohlin R. and Kavraki L. E. "Path planning using lazy PRM", *Proc. of the IEEE Robotics and Automation Conference*, (2000) 521-528
10. Sánchez L. A. "Contribution à la planification de mouvement en robotique: Approches probabilistes et approches déterministes", *PhD thesis, Université Montpellier II* (in French) (2003)
11. Branicky M. S., LaValle S. M., Olson K., and Yang L. "Quasi-randomized path planning", *Proc. of the IEEE Robotics and Automation Conference*, (2001) 1481-1487
12. Geraerts R. and Overmars M. H: "A comparative study of probabilistic roadmap planners", *Proc. of Workshop on the Algorithmic Foundations of Robotics* (2002)
13. Sánchez A., Zapata R., and Lanzoni C. "On the use of low-discrepancy sequences in nonholonomic motion planning", *Proc. of the IEEE Robotics and Automation Conference*, (2003) 3764-3769
14. Sánchez T. J. "Animación de personajes virtuales utilizando técnicas probabilisticas", *BS thesis, FCC - BUAP*, (in Spanish) (2005)
15. Reynolds C. W. "Flocks, herds and schools: A distributed behavioral model", *Computer Graphics*, Vol. 21, No. 4, (1987) 25-34
16. Ward C. R., Gobet F., and Kendall G. "Evolving collective behavior in an artificial ecology", *Artificial Life*, (2001) 191209
17. Funge J., Tu X., and Terzopoulos D. "Cognitive modeling: Knowledge, reasoning and planning for intelligent characters", *Computer Graphics*, (1999) 29-38

Intelligent Tutoring Systems

Towards a Methodology for the Design of Intelligent Tutoring Systems

Enrique Sierra, Ramón García-Martínez, Zulma Cataldi, Paola Britos and A. Hossian

Software & Knowledge Engineering Center. Graduate School. Buenos Aires Institute of Technology
Electrotecnic Department. School of Engineering. University of Comahue
Intelligent Systems Laboratory. School of Engineering. University of Buenos Aires
Educative Informatics Laboratory. School of Engineering. University of Buenos Aires

rgm@itba.edu.ar

Abstract. The present article proposes a methodology for the construction of intelligent tutoring systems that can be applied to any case that implies the design of a system intended for training advanced engineering students in the operation and maintenance of mechanisms. The article offers premises for the design of the modules of knowledge domain, student and tutor, and describes control strategies implemented as meta-rules.

1 Introduction

Intelligent tutoring systems (ITS) started to be developed in the 80's, designed with the idea to provide knowledge with base in some form of intelligence in order to guide the student in the process of learning [Urretavizcaya, 2001; Sancho, 2002]. An intelligent tutor is a software system that uses techniques of artificial intelligence (AI) to represent the knowledge and interacts with the students in order to teach it to them [VanLehn, 1988]. To this definition, [Giraffa *et al*., 1997] adds the consideration of different cognitive styles of the students who use the system according to Cern [2002]. In the 90's, with the advances of cognitive psychology [Norman, 1987; Gardner, 1988], neurosciences and the new paradigms of programming [Pressman, 2002; Pfleeger, 2002], ITS have evolved from a mere instructional proposal [Cruz Feliú, 1997] towards the design of environments of new knowledge discovery and experimentation [Bruner, 1991; Perkins, 1995; 2002] according to a constructivist vision of learning processes. In spite of these achievements, ITS have still not received a generalized adoption due to the complexity implied in its design, which has limited ITS practical application. The development of STI was stuck by the lack of maturity in the development of the human cognition area and therefore it was not possible to model it computationally since the complexity of the models involved required a high cost of calculation.

© A. Gelbukh, S. Torres, I. López (Eds.)
Advances in Artificial Intelligence
Research in Computing Science 20, 2006, pp. 181-190

2 Foundations for a Intelligent Tutoring Systems Methodology

2.1 Structure of the Intelligent Tutoring System

An Intelligent Tutorial System for training in the operation and maintenance of mechanisms consists basically of three models that communicate with one another [Kearsley, 1987]. With the purpose of obtaining a suitable operation of the ITS in reparation of mechanisms, the models of the mechanism, the student and the tutor must accomplish a suitable interaction. Therefore, it will exist a set of rules [García-Martínez and Britos, 2004] that define when and how these models are used. These rules constitute a meta-model in the sense that they control the basic models and their associate rules. The meta-model has the required functionality to activate and deactivate the basic models. As an example, the rules that can be expected to be contained in the meta-model may have the following form:

```
IF      The student has finished an specific item
AND     The tutor has little confidence in its own
        assessment on the student knowledge about that
        specific item
THEN    The tutor will interrogate the student exhaustively
        on the specific item
```

2.2 The Knowledge Domain Model

A model can be understood as an entity that copies the characteristics of an object, process or concept of the real world. In fact, a model is an abstract representation of some type of mechanism. It is abstract in the sense that it really does not exist, it is something that is created in the border of a computational program. In order to be able to construct a model of a mechanism, it must be possible to decompose the mechanism in its constituent parts. That is to say, the mechanism to be modeled must have identifiable parts in which it can be decomposed. This way, the behavior of the mechanism can be described through the behavior of its parts. This description includes from the intrinsic form of operation of each component up to the way in which a given component interacts with the others. In this work, qualitative models will be used more than quantitative, that is to say, that the relations among parts are described more in terms of the qualities of the constituent entities than of mathematical expressions that are representative of the operation way of these entities. This conception is more related with the way in which the human beings seem to approach the problems in their daily interaction with the every day world. In this way, a person can know when he or she is safe to cross a street without the need to construct mentally a mathematical model in order to calculate the trajectory of the vehicles that approach to him or her. The proposed methodology to model the knowledge domain when this one is referred to a mechanism consists of the following steps:

- Step 1. Identify the components which make up the mechanism

- Step 2. Identify the relations among the components of the model
- Step 3. Specify the rules of operation of the model
- Step 4. Evaluate the model

2.3 The Student Model

The design of the student model should be centered around the questions: What is desired that the student knows about the mechanism?. What types of knowledge must have a student to be able to solve a problem of operation or repair of the mechanism? [Barr & Feigenbaum, 1983]. It is evident that, in some way, the student must know how the mechanism works. The parts and components of the mechanism are the things that make it work.. Therefore, the student must have knowledge about:

- The components of the mechanism
- The operation of the components of the mechanism
- The interrelation between the components of the mechanism
- The operation of the mechanism

If a student chooses to examine a particular component, then it is assumed that the student knows something about that component. Given the context of the problem, the selection of a component is a kind of confirmation or disconfirmation that the student understands what the component does and how it relates to the operation of the mechanism. Every time the student checks, manipulates or examines a component, this tells what he or she knows or does not know about the operation of the mechanism. In order to make inferences about what the student knows, it is necessary to make assumptions about the meaning of student actions. These interpretations constitute the central part in the development of the student model in the design of an intelligent tutoring system:

Step 1. Identify the knowledge that the student has acquired concerning the components that integrate the mechanism.

Step 2. Identify the understanding level that the student has acquired in relation to the functionality of the mechanism and how its components contribute to achieve it.

Step 3. Identify the strategies used by the student to solve the problem and to approach suitably the processes necessary to carry out the reparation of the mechanism.

2.4 The Tutor Model

The instructional model or model of the tutor [Sierra, 1999; Sierra *et al*., 2001; 2003] is a representation of the methods that will be used in the intelligent tutor to provide information to the student. This model is complex because it is thought to direct the student in his or her process of learning and to carry out adjustments in this direction automatically as the student makes progress. In a practical sense, the following problem must be solved when the tutorial module of a system of intelligent instruction is constructed: the student is manipulating the domain model or mechanism and the student model is making inferences on the basis of these manipulations. The tutor must then make use of this information in order to provide useful information to the student. In a more general form, with the object of being able to correctly define the operation of the tutorial module, it must be possible to answer the following questions: When is it necessary to instruct? What type of instruction must occur? Therefore, the proposed methodological steps for the design of the tutor model are the following ones:

Step 1. Analyze the student model in order to clearly define which are the actions that he or she can perform.

Step 2. Interpret the actions defined in Step 1 in terms of the type of knowledge that the student must have in order to carry out these actions in a correct way.

Step 3. On the basis of the different types of knowledge identified in Step 2, determine the appropriate strategies of instruction so that the student incorporates significantly this knowledge into his or her cognitive structure.

3 An Example of Intelligent Tutoring System in Mechanism Reparation Domain

3.1 The Student Model

On the basis of the considerations carried out in previous theoretical analysis with respect to what the student model should be and the proposed steps in order to achieve it, the following rules are defined in order to describe how student actions can be modeled:

3.1.1 Assessing the Student's Understanding of the Mechanism Through a Single Student Action

An assessment rule can be expressed in terms of a subsumption analysis, which may be formulated as follows:

```
IF     in the model of the mechanism, the inferred path from
       source x (the point of examination or manipulation) to
       the target  y (the point where the fault occurs)
       subsumes a sequence of path elements from source x to
       target y

THEN   there is evidence supporting student knowledge of the
       inferred path of the mechanism.
```

3.1.2 Assessing the Student's Understanding of the Mechanism Through a Series of Actions

Concerning this aspect of analysis, the following rules can be formulated:

```
IF     a student's action examines a component in the model
       of the mechanism that is closer to the target in
       terms of physical distance and evidence
THEN   there is positive evidence that the student has
       knowledge of the mechanism
IF     a student's action examines a component in the model
       of the mechanism that is physically more distant and
       is adding negative evidence to the assessment of the
       student knowledge of the mechanism
THEN   there is negative evidence that the student has
       knowledge of the mechanism
```

3.1.3 Assessing the Student's Problem-Solving Process: Divide and Conquer

The rule can be formulated as follows:

```
IF     there is evidence (by showing that the student is
       manipulating components that belong to a different
       path of inference in the mechanism model) that a
       student is using a divide and conquer approach to
       problem solving
THEN   there is increased evidence that the student has
       understanding of some problem-solving methodology
```

3.1.4 Assessing the Student's Problem-Solving Strategy: Sequential Analysis

The following rule can be defined in terms of the possible paths of inference shown in the mechanism model:

```
IF     a student's sequence of actions follows a
       breadthwise, depthwise or spiralwise path through the
       mechanism,
THEN   there is evidence that the student is using a
       sequential strategy to diagnose the problem
```

3.1.5 Assessing the Student's Knowledge of Components

The rule covering this aspect of analysis can be stated as follows:

```
IF     a student examines component x
AND    then in sequence examines the sources of component x
       (the components that feed component x)
AND    then in sequence examines the sinks of component x
       (the components that are fed by component x)
THEN   there is evidence supporting that the student has some
       understanding of component x and its relation to other
       components
```

3.1.6 Assessing Student Use of the Troubleshooting Guide

The rule in this case can be enunciated as follows:

```
IF     the number of times that the student uses the
       troubleshooting guide follows a downward trend over
       time,
THEN   there is evidence that the student's knowledge of
       problem solving and the mechanism is increasing over
       time
```

3.1.7 Assessing Student Repetitive Actions

In this case, the following rule can be formulated:

```
IF       the count associated with the performance of any action
         is over a specified threshold
OR       the sequence of actions results in an identifiable
         pattern of actions
THEN     there is evidence that the student is repeating actions
```

3.2 The Tutor Model

The preceding section described a student model containing seven rules. These rules can be roughly classified as shown in Table 1. The partitioning of the rules into three categories allows the instructional model to address three distinct kinds of knowledge and assist students while they are interacting with the tutor [Pozo, 1998; Pozo Municio; 1999]. As Table 2 shows, tutoring or instructional strategy can be organized around these classifications.

Rule	Description	Classification
R1	Infer knowledge of the mechanism from a single student action	Mechanism Knowledge
R2	Infer knowledge of the mechanism from a series of student actions	Mechanism Knowledge
R3	Is the student using a divide and conquer problem-solving strategy?	Problem-solving Knowledge
R4	Is the student using a sequential problem-solving strategy?	Problem-solving Knowledge
R5	Does the student understand components?	Component Knowledge
R6	Is the student using the troubleshooting guide?	Problem-solving Knowledge
R7	Is the student performing repetitive actions?	Mechanism Knowledge / Problem-solving Knowledge

Table 1. Student Model Rule Classifications

Rule(s)	Classification	Instruction
R5	Component Knowledge	Provide the student with instruction about the function of a specific component
R1, R2	Mechanism Knowledge	Provide the student with instruction about how the mechanism works and the relationship between components
R3, R4, R6, R7	Problem solving knowledge	Provide the student with instruction about problem solving methods that would be useful

Table 2. Summary of Student Model Rule-Based Instructional Strategy

This organization of student model rules and their relation to instruction assumes that these three kinds of knowledge are important in the process of diagnosing and repairing mechanisms. Of course, other kinds of knowledge might be appropriate for other kinds of domains and problems. Based on the idea that the data from the student model is an indication of a particular problem, a series of instructional model tutoring rules may be formulated:

Rules referred to *Component Knowledge*

```
IF      the student model indicates there is a possibility
        the student has a deficit of component knowledge
AND     the assessment is above a specified threshold
THEN    provide first-level instruction to the student about
        relevant components

IF      the assessment is above a second specified threshold
THEN    provide second-level instruction to the student about
        relevant components
```

Rules referred to *Mechanism Knowledge*

```
IF      the student model indicates there is a possibility
        the student has a deficit of mechanism knowledge
AND     the assessment is above a specified threshold
THEN    provide first-level instruction to the student about
        relevant portions of the mechanism

IF      the assessment is above a second specified threshold
THEN    provide second-level instruction to the student about
        relevant portions of the mechanism
```

Rules referred to *Problem Solving Knowledge*

```
IF      the student model indicates there is a possibility
        the student is problem solving using a sequential
        approach
AND     the assessment is above a specified threshold
THEN    provide instruction to the student about alternative
        methods of problem solving

IF      the student model indicates there is a possibility
        the student is problem solving by continuously
        referring to the technical reference
AND     the assessment is above a specified threshold
THEN    provide instruction to the student about using the
        technical reference manual less

IF      the student model indicates the possibility that the
        student is performing repetitive actions
AND     the assessment is above a specified threshold
THEN    provide instruction to the student about trying
        different actions to avoid repeating the same actions
```

4 Conclusions

The main contribution of the present communication can be seen in the guidelines given for the construction of intelligent systems to instruct and train students in the operation and repair of mechanisms. The scope of the article goes beyond the methodologies suggested in the bibliography for the construction of intelligent tutors, entering in the details of the effective implementation of this kind of systems.
The motivating effect of technology in education is verified when it is correctly applied to the generation of relevant experiences of learning. In addition, the use of simulations -and mainly with respect to the operation and maintenance of mechanisms- will allow that students trained with these technologies develop suitable mental models with high possibilities of transference to real contexts and situations. Nevertheless, it is highly recommendable that the proper techniques of educational research are applied to evaluate the effectiveness of the proposed tool, which totally justifies the formalization of later studies in this direction.

References

1. Urretavizcaya, M. 2001. *Sistemas inteligentes en el ámbito de la educación*. Revista Iberoamericana de Inteligencia Artificial. Nº 12, pp. 5-12. ISSN 1137-3601
2. Sancho, L. 2002. *Sistemas Tutores Inteligentes: Una alternativa para el uso de computadoras en educación*. Education Net. Red Global de educación a distancia.

(DistEdNet) universidad Estatal a Distancia. Consultado el 10/07/04. www.uned.ac.cr/servicios/global/ensenanza /instruccion/articulos/sistemas.html
3. VanLehn, K 1988. *Student Modelling. M. Polson. Foundations of Intelligent Tutoring systems*. Hillsdale. N.J. Lawrence Erlbaum Associates, 55-78.
4. Giraffa, L.; Nunes, M. and Viccari, R. 1997. *Multi-Ecological: an Learning Environment using Multi-Agent architecture*. MASTA'97: Multi-Agent System: Theory and Applications. Proceedings.. Coimbra: DE-Universidade de Coimbra.
5. Cern, S. (Ed.) 2002 *Intelligent tutoring systems*. Springer Verlag Pub.
6. Norman, D. 1987. *Perspectiva de la ciencia cognitiva*. Paidos. Barcelona.
7. Gardner, H. 1988. *La nueva ciencia de la mente: Historia de la psicología cognitiva*. Paidós. Barcelona.
8. Pressman, R. 2002. *Ingeniería del software. Un enfoque práctico*. 5th Ed., México: McGraw Hill.
9. Pfleeger, S. 2002. *Ingeniería de software. Teoría y práctica*. Prentice Hall.
10. Sommerville, I. 2002. *Ingeniería de software*. Addison Wesley.
11. Cruz Feliú, J. 1997. *Teorías del aprendizaje y tecnología de la enseñanza*. Trillas. México.
12. Bruner, J. 1991. *Actos de significado. Más allá de la revolución cognitiva*. Madrid: Alianza.
13. Perkins, D. 1995. *La escuela inteligente*. Gedisa. Barcelona.
14. Perkins, D. 2002. *King´s Arthur round table. How collaborative conversations create smart organizations*. John Wiley & Sons.
15. Kearsley, G. 1987. *Artificial Intelligence and Instruction*, Reading, MA: Addison Wesley.
16. García–Martinez, R. and Britos, P. 2004 *Ingeniería de Sistemas Expertos*. Nueva Librería. Bs. As.
17. Barr, A. and Feigenbaum, E. 1983 *Handbook of Artificial Intelligence*, Morgan Kaufmann.
18. Sierra, E. A. 1999. *A Cognitivist Instructional Approach applied to the design of intelligent tutoring systems*, Proceedings of the Argentine Symposium on Artificial Intelligence, 221-232.
19. Sierra, E., Hossian, A. and García Martínez, R. 2001. *Selección de Estrategias Instruccionales. Abordaje desde la Teoría del Conocimiento*. Revista del Instituto Tecnológico de Buenos Aires. 25(1): 24-36.
20. Sierra, E.; Hossian, A. and García-Martínez, R. 2003. *Sistemas Expertos que recomiendan estrategias de instrucción. Un Modelo para su Desarrollo*. Revista Latinoamericana de Tecnología Educativa. 1(1): 19-30. Facultad de Educación. Universidad de Extremadura. ISSN: 1695-288X.
21. Pozo, J. I. 1998. Teorías cognitivas del aprendizaje. Morata. Madrid.
22. Pozo Municio, I. 1999. Aprendices y Maestros. Alianza.

Learning and Applying Artificial Intelligence with Mobile Robots

Julio César Sandria Reynoso[1,2], Mario Morales García[2], Arturo Mendoza Rendón[2]

[1] Instituto de Ecología, A.C., Departamento de Informática,
Xalapa, Veracruz, 91070, México
sandriaj@ecologia.edu.mx
[2] Universidad de Xalapa, Ingeniería en Sistemas de Cómputo Administrativo,
Xalapa, Veracruz, 91190, México

Abstract. Learning artificial intelligence sometimes is a very frustrating task when you try to understand it with a lot of theory and a little practice. This paper intends to illustrate that learning artificial intelligence could be an attractive and entertainment task when your try to build robots that show some intelligent behavior like see, hear, speech, move, and even learn. We use Lego mobile robots and Java for learning and applying some techniques from artificial intelligence like neural networks, genetic algorithms, computer vision, speech recognition, and speech synthesis.

1 Introduction

The use of physical robots as a teaching tool has been extended around the world. With the development of the Lego Mindstorms Robotics Invention System, some universities use them for educational purposes. The Robotics Academy in the Carnegie Mellon University (www.rec.ri.cmu.edu/education) offers a complete teaching package for children, to introduce them to technology, hardware, electronic control, computer programming and mechanics. It has been used to instill engineering skills, scientific interest, computer acquisition, general ideas and creativity among students [6]; also, applying Piaget's theories of cognitive development, to help students to understand concepts about complex dynamic systems, like how global behavior can emerge from local dynamics [8].

In the artificial intelligence arena, Lego based robots have been used for teaching neural networks [4] and for building low cost robotics laboratories for teaching artificial intelligence [5].

In this paper we use Lego mobile robots for teaching/learning some artificial intelligence themes in an Artificial Intelligence Course. Based on Nilsson's book [9], which covers these themes with software agents, we begin to work a neural network, a genetic algorithm and a computer vision algorithm with physical agents – our mobile robots.

With the idea of using only one programming language, available to Lego robots, we use Java. This way, we can build intelligent Lego robots with Java, as it provides APIs for programming systems that can see, hear, speak [7], and even learn [12, 13].

© A. Gelbukh, S. Torres, I. López (Eds.)
Advances in Artificial Intelligence
Research in Computing Science 20, 2006, pp. 191-198

We use the Lego Mindstorms Robotics Invention System 2.0 for building the Lego mobile robots; leJOS 2.1.0, a little Java operating system for downloading and running Java programs inside the robots; Java 2 for compiling the Java programs under LeJOS, and some APIs for computer vision and speech recognition.

1.1 Lego Mobile Robots

The Lego Mindstorms Robotics Invention System (RIS) is a kit for building and programming Lego robots. It has 718 Lego bricks including two motors, two touch sensors, one light sensor, an infrared tower, and a robot brain called the RCX.

The RCX is a large brick that contains a microcontroller and an infrared port. You can attach the kit's two motors (as well as a third motor) and three sensors by snapping wire bricks on the RCX. The infrared port allows the RCX to communicate with your desktop computer through the infrared tower.

In this work, we use a Roverbot as it is constructed in the Lego Mindstorms Constructopedia, the guide for constructing Lego robots.

1.2 Java Technology

We use Java 2 Technology to program all algorithms we show, using own code and predefined packages and Java APIs as shown in Table 1.

Table 1. Java Technology used with Lego robots

Java Technology	Mean and use
J2SDK	**Java 2 Software Development Kit** to compile and run Java programs
LeJOS	**Lego Java Operating System** to run Java programs inside the RCX
LMbpn	A Lego Mindstorms Backpropagation **Neural Network** (own)
LMsga	A Lego Mindstorms Simple **Genetic Algorithm** (own)
JFM	Java Media Framework for **Computer Vision**
JSAPI & Sphinx	Java Speech API and Sphinx for **Speech Recognition**
JSAPI & Sphinx	Java Speech API and Sphinx for **Speech Synthesis**

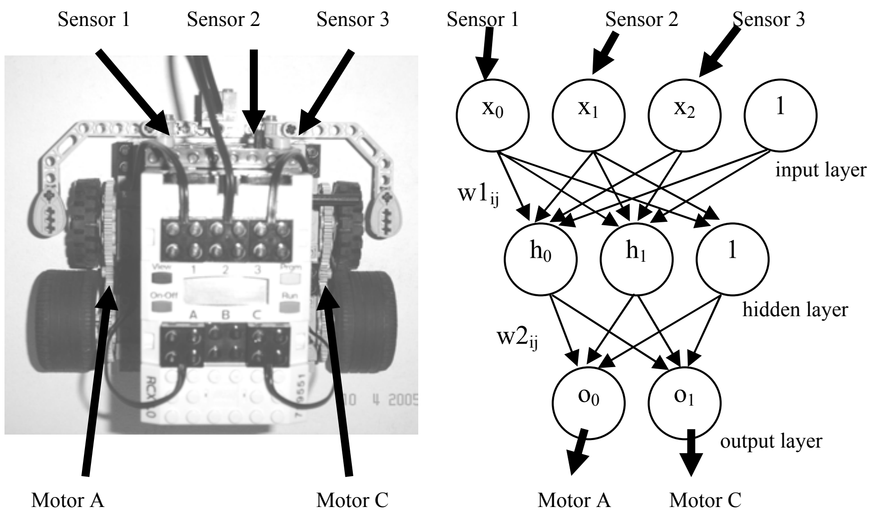
1.3 LeJOS

LeJOS is a small Java-based operating system for the Lego Mindstorms RCX. Because the RCX contains just 32 KB of RAM, only a small subset of the Java Virtual Machine and APIs can be implemented on the RCX. LeJOS can be downloaded from lejos.sourcefoge.net. For setting up a LeJOS installation refer to the LeJOS documentation or Ferrari *et. al.* [2].

2 Artificial Intelligence applications

2.1 Neural Network: The Backpropagation algorithm

For the neural network example, the task was to learn and apply the backpropagation algorithm as it is described by Rich and Knight [11]. We had to model a backpropagation network for our mobile robot. The robot has three inputs (two touch sensors and one light sensor) and two outputs (two motors) as shown in Fig. 1.a. So, we can use a three layer backpropagation network as shown in Fig. 1.b.



a) The roverbot to model b) The backpropagation network

Fig. 1. The robot input/outputs and its backpropagation network

The use of a three layered network and three units in the hidden layer was just an arbitrary decision influenced by teaching purposes.

To define input-output vector pairs for using in the backpropagation network, from the robot input-output (sensor-motor), we had to identify what was going to learn the robot. We defined four basic behavior rules:

1. **Forward**: if sensor 1 is off, and sensor 2 is over white floor, and sensor 3 is off, then Motor A and Motor C go forward (the robot goes forward).
2. **Turn right**: if sensor 1 is on, then Motor A goes forward, and Motor C goes backward (the robot turns right).
3. **Turn left**: if sensor 3 is on, then Motor A goes backward, and Motor C goes forward (the robot turns left).
4. **Backward**: if sensor 2 is over black floor, then Motor A and Motor C go backward (the robot goes backward).

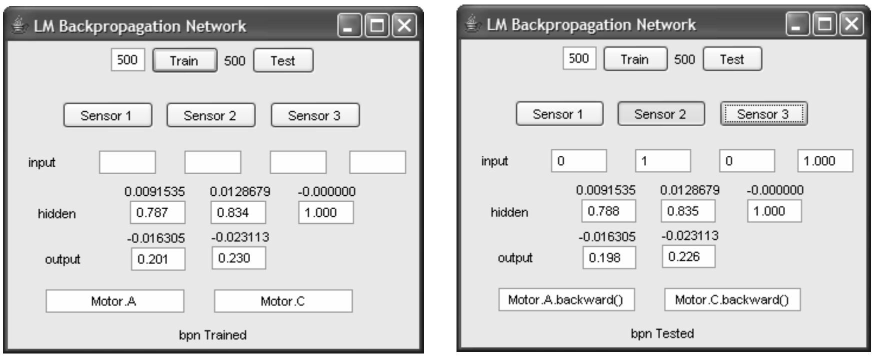
We translated these rules to training examples for the backpropagation network as shown in Table 2.

Table 2. The rules to learn as training examples

Rule	Training examples				
	Sensor 1	Sensor 2	Sensor 3	Motor A	Motor C
Forward	0	0	0	1	1
Turn right	1	0	0	1	0
Turn left	0	0	1	0	1
Backward	0	1	0	0	0
	Input vectors			Output vectors	

The input-output vector pairs are the examples we used to train the backpropagation network. This way, our mobile robot learnt to move forward, turn right, turn left and backward, based in its sensor states. Additionally, we did not define a rule when both, sensor 1 and sensor 2 were both on, but the backpropagation network gave the robot an emergent behavior for such case.

For the development process we used a Java-based graphic interface (Fig. 2), where it is possible to train and test the neural network. This way, at the final development cycle, we trained the backpropagation network, and found that it learnt the rules in 500 epochs, that required less than 1 second in a Windows XP system with 256 MB RAM, and a 1.8 GHz processor.



a) BPN training phase b) BPN testing phase

Fig. 2. A Java-based graphic interface for the development process.

The program code was compiled with the LeJOS compiler (`lejosc`) and downloaded to the RCX with the LeJOS loader (`lejos`), as shown in [12]. The program ran well in the RCX, but requires about 5 minutes to train 500 epochs. It is better to train the network in a desktop computer and then download the program to the RCX.

2.2 Genetic Algorithm

In order to give the mobile robot an artificial intelligence program that could by itself choose the correct way to response to the signals being received by its sensors an Artificial Neural Network application was created, but of course the program needs to be trained first, which poses a slight computational problem, the RCX by itself takes a considerable amount of time to process the information needed to be correctly trained, in this case when using the backpropagation algorithm took about 500 epochs to get the responses right, which translates to 5 minutes of processing time at the RCX. And, to learn an obvious emergent behavior – not defined by given rules in Table 2 – like both sensor 1 on and sensor 3 on, it took to the backpropagation network more than 5000 epochs to learn move backward.

The way chosen to tackle this problem was to optimize the backpropagation algorithm with the use of a simple genetic algorithm as described by Goldberg [3] immersed in the error minimization function, and get all the processing done on a desktop computer. This way, what was tried to accomplish was to get the correct weight values for each of the layers on the neural network, and then transfer those values to the neural network on the RCX, and so completing the training of the network without having to wait for the brick to get the entire math done.

The first thing done to get the genetic algorithm working was to create some useful functions that would be needed along the whole process more than one time. These were the functions to convert from real numbers to binary ones; the function to obtain the current values if the population when applied to the mathematical evaluation function, the roulette count for each of the population values and of course the function to convert back the binary numbers to real ones.

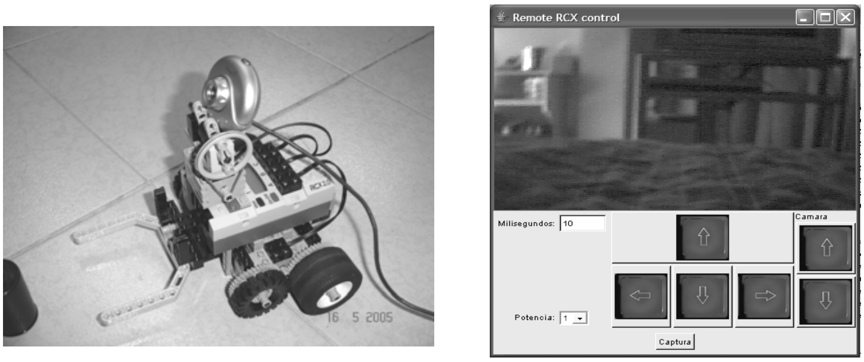
The next functions to get done where the most important ones, the coupling function and the mutation function. For the former one a few randomization operations were needed, especially when choosing the couples to reproduce and the point of insertion of each portion of the binary value that would finally create the new values. The latter one uses also a randomization function, but it's used to create a probability operation to choose if any of the binary values should mutate when passing to the next generation which would represent a change of one of its binary numbers.

After getting all these functions done the only thing needed is to create an encapsulation for them, that is a program that uses the functions in the correct order and that using recursive programming can evolve the evaluation of the mathematical function in order to get the correct values of the weights by passing the values obtained after the mutation and coupling functions back to the genetic algorithm so it can start processing them all over again.

2.3 The Computer Vision Algorithm

One of the biggest issues in artificial intelligence is to simulate the way the human vision works, the computational power needed to correctly interpret the images captured in real time represents a huge obstacle to the development of this area. DeSouza and Kak survey the developments of robot vision navigation since 1980s [1]. Here, we applied just only one algorithm, mainly trying to discern different objects viewed across a camera. The whole point of it was to get the mobile robot to recognize and differentiate between two different objects and advance towards them in order to grab the one the user has ordered to be captured. This was planned to be done by using edge detection and remote controlling the robot.

Before even attempting to program anything the correct tools must be present, and one of these tools was the Java Media Framework [7] which presents libraries and methods to properly use the images captured by a camera, in this case a small web-cam attached to the robot's body (Fig. 3.a). Other tools are the LeJOS Vision classes in `vision.jar` and remote control classes `rcxcomm.jar` and `pcrcxcomm.jar` to remotely control the Lego robot through the infrared tower, manually using buttons in an own graphic interface (Fig. 3.b) or automatically using commands in the vision program.



a) A small webcam attached to the robot's body

b) The Java-based graphic interface to control the robot

Fig. 3. A small webcam for the robot vision

For this program one of the most basic edge recognition algorithms was used –the Sobel algorithm [10]– which uses convolution masks to determine the difference between pixels that are next to each other and find the edges on an image.

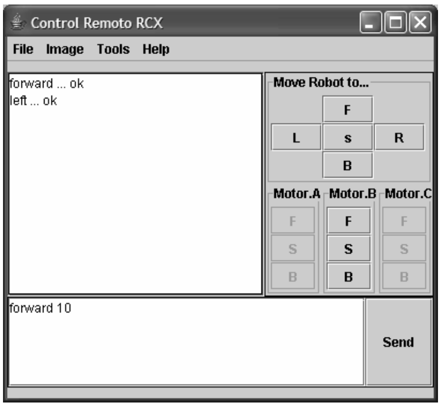
The biggest issue present when programming this application was to manipulate correctly the images by it's pixels. Therefore a function that can convert an image to an array of pixels on integer values it's needed. Fortunately Java provides a very easy to use method on its image libraries that is called PixelGrabber and it does precisely the work that is needed to be done. With the pixels already on an integer array it is only needed then to translate the Sobel algorithm into a function which at the end returns an array of pixels where the edges found get a different color. Another useful Java method can render an image out of the pixel array finally showing on screen the original image with the edges marked.

In this case we used two different objects to work with, a cylinder and a sphere. After getting the Sobel function working the next thing needed is to create a function that searches for the edge pixels and checks if they all make a geometrical figure a rectangle or a circle. The search is done by searching for pixels next to each other. To find the rectangle if two parallel lines are found connected to two perpendicular ones the image is recognized. For the sphere the pixels need to be connected in one of four ways - to the left and below, right and below, left and above and right and above – so the circle is drawn.

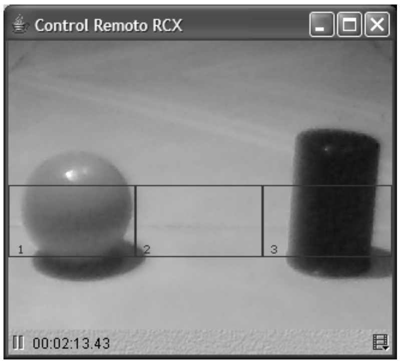
After that the work becomes simpler, a small program that lets the user choose between the object to look for needs to be done and using the Java Media Framework the image being captured it's shown to the user so he can check how the mobile robot looks for the geometrical figure on it.

2.4 Speech Recognition and Synthesis

In order to "talk" with our mobile robot a chat interface was designed to give orders to the robot and to receive response from it (Fig. 4.a). This way, we can see what the robot see (Fig. 4.b), and tell what to do, e.g., move forward, backward, turn right, turn left, move camera up or down, and so on.



a) A remote control chat interface



b) The robot vision

Fig. 4. The remote control interface

Trying to give simple voice orders to the robot and receive responses from it in the chat interface, we used Sphinx 4, a Java Framework for speech recognition [14]. For the speech recognition we used a little grammar like is required by Sphinx:

```
grammar move;

public <greet> = ( go | turn | camera ) ( up | down |
left | right | stop | backward | forward );
```

3 Conclusion

Learning artificial intelligence in a one semester course using mobile robots is a more different and entertainment task than the traditional way – with a lot of theory and some programs. At the beginning students become disconcerted when the teacher tells them they have to learn and apply some artificial intelligence algorithms in a little Lego toy. But, it becomes very interesting to understand first how to program the Lego robot, and second, trying to program some algorithms, using only the Java language.

References

1. DeSouza, G.N., Kak, A.C.: Vision for Mobile Robot Navigation: A Survey. IEEE Transactions on Pattern Analysis and Machine Intelligence (2002) 24:2, 237-267.
2. Ferrari, G., Gombos, A., Hilmer, S., Stuber, J., Porter, M., Waldinger, J. Laverde, D.: Programming Lego Mindstorms with Java. Syngress Publishing, Rockland MA (2002).
3. Goldberg, D.E.: Genetic Algorithms in Search, Optimization, and Machine Learning. Addison-Wesley (1989).
4. Imberman, S.P.: Theaching Neural Networks Using Lego-Handyboard Robots in an Artificial Intelligence Course. SIGCSE Technical Symposium on Computer Education (2003).
5. Kumar, D., Meeden, L.: A Robot Laboratory for Teaching Artificial Intelligence Resource Kit. Proceedings of the Twenty-ninth SIGCSE Technical Symposium on Computer Science Education, Daniel Joyce (ed.), ACM Press (1998).
6. Law, K.W., Tan, H.K., Erwin, B.T., Petrovic, P.: Creative Learning in School with Lego Programmable Robotics Products. 29th ASEE/IEEE Frontiers in Education Conference (1999).
7. Meloan, S.: Futurama: Using Java Technology to Build Robots that can See, Hear, Speak, and Move. Sun Microsystems Inc, July (2003).
8. Miglino, O., Lund, H.H., Cardaci, M.: Robotics as an Educational Tool. Journal of Interactive Learning Research (1999) 10:1, 25-48.
9. Nilsson, N.J.: Inteligencia Artificial. Una nueva síntesis. McGraw-Hill, España (2001).
10. Parajes, G., De la Cruz, J.M.: Visión por Computador, imágenes digitales y aplicaciones. Alfaomega-RaMa, México (2002).
11. Rich, E., Knight, K.: Inteligencia Artificial. 2da ed. McGraw-Hill, España (1994).
12. Sandria Reynoso, J.C.: A Neural Network for Java Lego Robots. JavaWorld. May 16 (2005).
13. van Dam, B.: Simple Neural Network as robot brain. generation5, www.generation5.org (1999).
14. Walker, W., Lamere, P., Kwok, P., Raj, B., Singh, R., Gouvea, E., Wolf, P., Woelfer, J.: Sphinx-4: A Flexible Open Source Framework for Speech Recognition. Sun Microsystems Inc. (2004).

Author Index

Índice de autores

Impreso en los Talleres Gráficos
de la Dirección de Publicaciones
del Instituto Politécnico Nacional
Tresguerras 27, Centro Histórico, México, D.F.
Mayo de 2006.
Printing 500 / Edición 500 ejemplares.