Advances in Soft Computing Techniques

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 68,** noviembre 2013. Tiraje:500 ejemplares. *Certificado de Reserva de Derechos al Uso Exclusivo del Título* No.: 04-2005- 121511550100-102, expedido por el Instituto Nacional de Derecho de Autor. *Certificado de Licitud de Título* No. 12897, *Certificado de licitud de Contenido* No. 10470, expedidos por la Comisión Calificadora de Publicaciones y Revistas Ilustradas. El contenido de los artículos es responsabilidad exclusiva de sus respectivos autores. Queda prohibida la reproducción total o parcial, por cualquier medio, sin el permiso expreso del editor, excepto para uso personal o de estudio haciendo cita explícita en la primera página de cada documento. Impreso en la Ciudad de México, en los Talleres Gráficos del IPN – Dirección de Publicaciones, Tres Guerras 27, Centro Histórico, México, D.F. Distribuida por el Centro de Investigación en Computación, Av. Juan de Dios Bátiz S/N, Esq. Av. Miguel Othón de Mendizábal, Col. Nueva Industrial Vallejo, C.P. 07738, México, D.F. Tel. 57 29 60 00, ext. 56571.

Editor responsable: *Grigori Sidorov, RFC SIGR651028L69*

Research in Computing Science is published by the Center for Computing Research of IPN. **Volume 68**, November 2013. Printing 500. The authors are responsible for the contents of their articles.

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

Volume 68
Volumen 68

Advances in Soft Computing Techniques

Alexander Gelbukh
Miguel González Mendoza
Félix Castro Espinoza (eds.)

Instituto Politécnico Nacional, Centro de Investigación en Computación
México 2013

ISSN: 1870-4069

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

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

Indexed in LATINDEX and Periodica / Indexada en LATINDEX y Periódica

Printing: 500 / Tiraje: 500

Printed in Mexico / Impreso en México

Preface
(Prefacio)

This volume of the journal "Research in Computing Science" contains selected papers on soft computing methods and techniques. The papers were carefully chosen by the editorial board based on at least two reviews by the members of the reviewing committee of the volume.

Soft computing in a broad sense is the discipline that studies approximate and near-optimal techniques in computing. This is often the case in artificial intelligence and related fields: while finding the very exact solution is not feasible, a near-optimal solution is as good for any practical application as the exact one but can be achieved using fast and low-resource algorithms.

Good examples are genetic algorithms, fuzzy logic, and neural networks. Other examples include fields where the very representation of the corresponding knowledge and the evaluation criteria are so imprecise that it is not even clear how one can reason about exact solutions. An example of this kind of imprecision is natural language processing.

In this volume we included several papers related to search and optimization, such as a multi-agent approach to search service discovery, Particle Swarm Optimization approach to optimization in the inverse robot dynamic model, optimization of course time-tabling, and a novel approach to project scheduling based on musical composition analogy, among others.

We also included a number of papers related to image processing, such as recognizing medical magnetic resonance images and characterization and classification of images in food industry.

Finally, we included a set of papers on text, semantics, ontologies, and natural language dialogue, including the issues of knowledge representation in natural language and information extraction from natural language texts.

Some of the papers in this volume are related to medical topics, especially those papers on image processing techniques and on knowledge representation with ontologies.

The volume will be useful to researchers, students, and general public interested in artificial intelligence and its practical applications.

November 2013,
Alexander Gelbukh,
Miguel González Mendoza,
Félix Castro Espinosa

Table of Contents

(Índice)

Page/Pág.

Intelligent System for Searching Nearest Services using Multi-Agent Approach

Waqar Mirza M.[1], Ana María Martinez-Enriquez[2], Aslam Muhammad[1], Afraz Z. Syed[1], Arslan Lodhi[3]

[1] Department of CS & E, UET , Lahore, Pakistan
[2] Department of Computer Science, CINVESTAV-IPN, D.F., Mexico
[3] Department of Information Technology, APPSTER Tech Ltd., Lahore, Pakistan

mirzamwaqar@hotmail.com,{maslam,afrazsyed}@uet.edu.pk, ammartin@cinvestav.mx,arslan.lo@eleget.net

Abstract. In this paper, we present the multi agent approach for developing an intelligent system for mobile users to find most of the different type of deals/services available nearby to them at unknown places. Now days, smart phone have the ability to determine the exact position of a terminal on the earth. By use of the ability the system collects the user's position and discovers the services available near to him. Many services are being offered by mobile companies and Web sites to look for locations, finding routs, and tracking devices. However, the Web content is updated after certain interval of time and is not so much interactive. For instance, the information like doctors on duty, available medicines, and beds is outdated and cause the wastage of time as well as money of visitors. We resolve this issue by means of design and development of an intelligent system whose architecture is based on multi-agents who collaborate and cooperate with each other to present the most accessible and useful services available in an interactive way. The results obtained from the experiments are very encouraging and much appreciated by the users of this system.

Keywords: Multi-agent system, GPS, location based services, Android application.

1 Introduction

Through the advent of multi agent technologies and mobile phone applications, the organizations now have many options for determining where the computing resources of an application take place [1]. Bottleneck in the overall IT infrastructure can be overcome by use of these technologies. The rapid advancements in the area of Web and Mobile applications have encouraged the provision of GPS (Global Positioning Systems) [2] based mobile application. The GPS is a system based on almost 24 satellites placed into an orbit which navigate to make up a network. GPS is available anytime, anywhere and any weather condition and also to anyone without any charges. If the user have GPS receiver in his phone then he can easily determine his position on earth. It tells the exact position of the user when he is in the range of at least three satellites of GPS network [13]. The GPS express the position in terms of latitude and longitude values.

Finding of exact location of a user is the basic of location based services (LBS). As an example, when a user wants to search any restaurant or shopping mall near to him using any mobile application, it is necessary to first determine the current location of the user and then discover all the nearest services to that position. The GPS and LBS play a vital role in mobile application to facilitate its user. It is a challenging task to find the nearest resource from a user's current location when he is travelling to an unknown location. When a user is at strange place, it is important to identify nearby services (fuel station, restaurant, vehicle service center etc.) to the needy people. This reduces the amount of stress a person feels during his journey and also save his time.

Wide range of LBS are available that can helps the user to finding a service or inform the user about his current location and ways to destination [14]. One of them is a navigation service, uses the digital map along with the labels of popular streets and buildings, use to find the direction during journey from current location to destination. Google Maps [18] is an example of this type of service where user can see different routes to reach a specific destination along with their distances and estimate times. When a digital map is extended with the facility to find the physical services then it is called yellow page service. In further extension, when a system is able to locate the user's current position and determine the nearby object and resources then these types of services is called location aware information services. However, all these services do not provide timely updated information, for instance, a tourist at any foreign place may need to visit an auto teller machine to draw money. Any location based service either offered by the bank or other vender may tell the exact site of the money cube but it is not able to inform about the case situation. If the foreigner desires to withdraw high denomination notes then the machine would not be able to answer such queries. The reason is that these systems update their values after a specific interval of time. Therefore, the problem often occurs when a user wants newest facts about any service, as he needs it urgent, and the system is unable to provide it. Likewise, cases exist where travelers or even residents have to waste a lot of time and money in terms of fuel in search the exact service as desire.

In this research, we extended the location aware information service by not only searching the nearest available service/resource but also with the current states of that service either it is presently available or not which in fact reduces the time and effort of people. We tackle this issue by the design and development of an intelligent system whose architecture is based in multi-agents. Agents can communicate, collaborate, cooperate, and act autonomously. The purpose of the system is to save the time of the user and provide comfort to him during his trip. The major objective of this research is to get a highly cost-effective application which caters the needs of the users and facilitates them in a most effective manner. The system offers many different types of businesses deals on single platform so that users can give a comparative look on multiple businesses. The purpose is to improve the existing system’s services by increasing user satisfaction, providing the quality and creating a controlled information environment. The system offers the better performance by providing desired flexibility, fast response, ability to support changes and ability to maintain the quality of services.

The rest of the paper is organized as follows: related work is discussed in Section 2. Section 3 describes in detail the intelligent architecture of LBDSS (Location Based Deals Searching System). Experiment and their obtained results are presented in Section 4. Conclusion and future perspectives are given in Section 5.

2 Related Works

Lots of work has been done in regards to help users in stranger places when they are unknown to that location and need any services immediately. All these existing system may help their user in such a situation but sometimes they are not successful in helping them correctly. The existing system update their databases after a long time that's why they do not provide accurate information to their user and instead of helping the user to remove them from problems they create more problems for them. Here we will discuss some of them in details below.

Man Lung Yiu et al. [3] studied the aggregate nearest neighbor queries to create a function that help to find the minimum aggregate distance between the special object and special network (e.g. road). Different aggregate techniques and function are considered to exploit Euclidean distance bounds, special access methods and network distance materialization structure. In their research they presented three algorithm, IER incrementally retrieves Euclidean aggregate nearest neighbors and computes their network distances by shortest path queries until the result cannot be improved. TA and CE explore the network around the query points until the aggregate nearest neighbors are discovered. These techniques are combined with special access method and shortest path materialization technique to get the minimum distance.

S. R. Balasundaram et al. [15] uses the query management to find out the most appropriate location of required services needed by a mobile user. The major objective of the paper is to provide the information to needy people who are searching for the resources during travel at their minimum distance. The location of the user is obtained using GPS and the distance of user's future location and available resource point is calculated and the resource with minimum distance available is returned to user.

Hae Don Chon et al. have developed a location based application named NAPA (Nearest Available Parking Lot Application) [16] used to assists users to find a nearest parking space on campus. NAPA uses the feature of LBS, wireless communication and directory services (LDPA) to achieve his goal. The similar kind of idea named SIAPAS [17] is also presented by G Mendez et al. The design of SIAPAS is consist of a set of six independent modules that communicate with each other using web services. Communication module keep track of parking space, GPS gets the current position, Voice provides driving assistance, GUI for user interaction, Outside Parking Manager to control global parking, Inside Parking Manager keep track of parking routes and Configuration module manages the GPS device configuration.

Also there are available some mobile application [4] [5], deals aggregator in nationwide, is not much up to the mark as they should be. The goal of these applications is to provide a unique location based mobile solution to aggregate hundreds of deal sites and make the process of saving money easier and less time consuming. When a user subscribes to these services he get alerts whenever he is close to a deal.

All the above discussed systems provide the limited information to their users. They only tell the user which deals/service is available near and how much far away from him. But they do not tell the user either the required item is currently available at that shop or either the shop is currently open or closed. Normally what happened when a user uses this ordinary application, he thought to select the most nearest location to get appropriate service and unfortunately when he reaches there, the shop founds to be closed or the required service/item is not currently available at that particular time or out of stock. So, instead of saving time they lost it and also the efforts. On the other side, the LBDSS is developed by keeping these short comes in mind. The system has the latest information whenever user needs it. So our key idea is to develop a platform for business dealers and customers for communication so they can easily get the information by just few taps of fingers.

3 Architecture of Intelligent LBDSS

The proposed system is developed by using multi-agent technology where multiple agents communicate and coordinate with each other to fulfil the assigned task. The foremost objective of this system is to facilitate the users in a most effective manner and provide him latest information about the deals or services. To use LBDSS, the user just need an android [10] [11] supported handset and internet connection in it. The user installs LBDSS client application in his mobile which co-ordinates with the server to get the deals according to the user's search criteria.

3.1 Agents in LBDSS

Software architecture describes the major components of the system and the relationship between them [6] [7]. The overall architecture of LBDSS is based on three agents who work for the users to get required information and facilitate its user by giving him up-to-date information when he requests. The detail of these three working agents is shown in Figure 1 and also discussed each of them works as individual and as a team.

User interface and Query Agent (UIQA). As the name describes the working of this agent is to provide an interface to end user by giving him the facility to use application and take queries from the user. The requested parameters are then sent to the Data Management Agent (DMA) to get filtered information. The DMA process the query and send back the fetched data to UIQA. The obtained result is then displayed on the users screen for further processing. The responsibilities of this agent is to provide an easy to use interface to make queries for searching required information and communicate with the DMA to get required result and present to user.

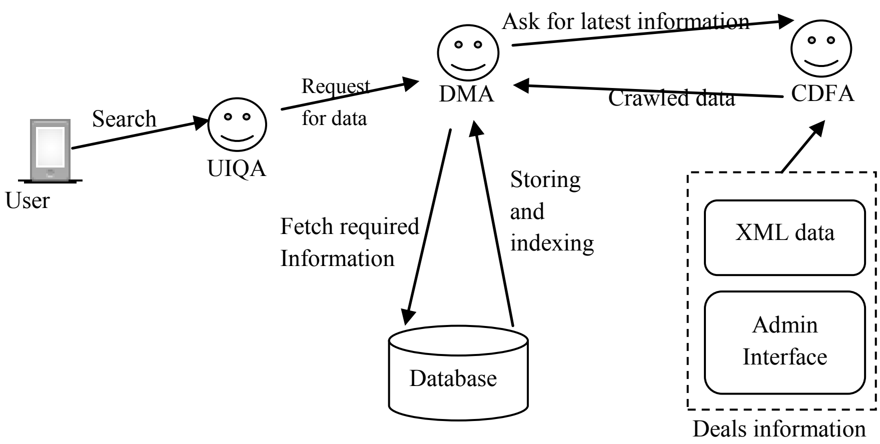


Fig. 1. General architecture of purposed system

Data Management Agent (DMA). DMA is in the middle of both agents. He got queries from the UIQA and send result back to him. The another major task done by this agent is to get crawled data from the Crawling and Data Fetching Agent (CDFA) and manage that data by making indexing of each data element and then store them in system's database. The advantage of making indexes of data element is to search them quickly and efficiently. Also if DMA did not find any up-to-date information in the system database it send request to CDFA to crawl immediately and give him latest information. The latest information means, checks services are currently available or the item user searching for is in stock or out of stock. Furthermore checks weather the shop or organization is currently open or closed. All these information is provided by the venders through an XML feed or through the HTML based interface provided to him.

Crawling and Data Fetching Agent (CDFA). The last agent CDFA is responsible of crawling the available sources and get information from them. This is a sequence based task. The CDFA visits each source one by one and send information to DMA for storing and for future using. But the most important working of CDFA is when a user requests for the information about the particular organization's deals but the LBDSS does not have up to date. Then the DMA request to CDFA to crawl that vender's source immediately and give him the latest information. Here we have two way of getting information from the venders. One way in which an administrative interface is given to a vender where he register himself and then provide all the services and products he offers. Also it is regularly updated by the staff of the organization at short interval of time about the stock and shop status and any other information which will useful for their searchers. The second source is vender's web site. We request them to give us the update information in XML format available for CDFA on a given URL. The CDFA visits that URL after a specific interval of time and get the latest information and send it to DMA.

Communication between the agents. For communication and sharing information between the agents the system uses FIPA ACL which is encoded in XML. The same technique is used by [8] [12] in their systems for agent communication and found it very efficient, fast and easy to implement in web based environments. So considering all these things, we also decided to use FIPA ACL with XML encoding for our agent's communication. A sample agent communication message is from agent-1 to agent-2 shown below. The message is encoded in XML. It can be seen that the sender and intended recipient of the message are identified by their agent-identifiers. For the sample message, the sender and receiver agent names are AG1 and AG2, respectively. The sender and receiver agent addresses are http://ag1.masc.com:5120 and http://ag2.masc.com:5120, respectively. The actual message is enclosed in message tag. The 'type' parameter defines the type of message either it is request or reply to some request and the 'id' parameter uniquely identify each message that is used by receiver to send reply of the query for reference. The 'content' tag contains the requested value required by agent from other and arguments tag contains the parameter use to filter the required information. The following sample message is a request message send by UIQA to DMA to get the list of all restaurants near to user's current location.

```
<?xml version="1.0" ?>
<sender>
  <agent_identifier>
    <name>AG1</name>
    <address>
      <url>http://ag1.masc.com:5120</url>
    </address>
  </agent_identifier>
</sender>
<receiver>
  <agent_identifier>
    <name>AG2</name>
    <address><url>http://ag2.masc.com</url></address>
  </agent_identifier>
</receiver>
<message type="request" id="MSG-1">
```

```
  <content>Nearest Restaurant</content>
  <arguments>
    <latitude>23.536723</latitude>
    <longitude>2.3223411</longitude>
    <current_time>2013-01-12 23:22</current_time>
  </arguments >
</message>
```

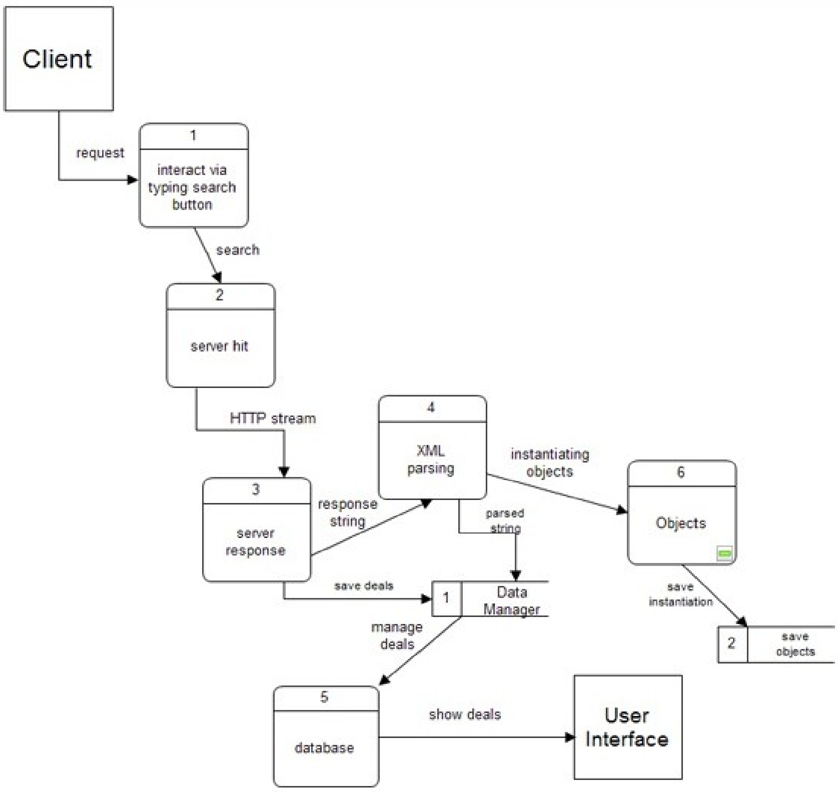


Fig. 2. Flow of information in purposed system

3.2 Methodology used for LBDSS

In this section we will discuss how actually the system works with the help of flow diagram of the system shown in figure 2. First of all user request for the required information thorough his hand held devices. For this purpose user will have to install the LBDSS client software on his mobile and mobile should have android operating system in it. The user submits his request through UIQA installed along with client application. Then UIQA send this request to server where DMA receives the request parameters and starts fetching data from available sources.

In figure 2 we can see that data from UIQA to DMA is travelled in the form of HTTP stream. Here the two agents used XML ontology [9] for communication as discussed earlier. And the request message is transferred using HTTP stream. DMA receives the request and then see in his record is the required data is in the database or not. If the data is available in system's database, then it is checked weather is it recent information or expired. If the available information is expired, the DMA request to CDFA to fetch the latest information from the vender. The vender may provide newest information through XML channel or may use the administrative interface provided by LBDSS. The latest data in XML format is sent to DMA who parse it and stored it in system's database and also sent it back to UIQA in reply to the request. UIQA takes the received information and display it to user's screen.

4 Case Studies

In order to test the performance and efficiency of the LBDSS, numbers of test experiments are performed. Initially to collect real data we visited different venders and ask them to provide data either through XML feed or use our purposed web based interface. For testing purpose we selected 30 restaurants at two major locations in the city. Then 10 end-users with android mobile, who are not much familiar with those areas, are selected and ask to visits these locations. For data inputs the android mobiles are used through which user can comfortably give the input and type the required keyword for search. User has to install the LBDSS client software on his mobile so he can get the information from the server. The LBDSS client application is easily available on our website and also on android app stores. These entire users have different brands of android mobiles.

a. search screen b. search result c. product details

Fig. 3. Screen shoots of proposed system

When user installs the software and open it in his mobile, the main screen ask him for keyword he want to search and some advance parameters if needed as shown in figure 3(a). We asked the user to search only food items, as for testing, we used only restaurants venders. After entering the desired item in keyword box the system goes to server and request for the items which are nearest to the current location. Then the search result is fetched and displayed to its users as shown in figure 3(b). The user has the option to view the details of each product as shown in figure 3(c) and also have the choice to view the direction to that vender from his current position.

At the other end, to receive information and data from the venders we have multiple options. The first one is the XML feed. This is helpful when venders have their own web application and database. Some of the venders in our list have the XML facility and agree to give us their data in XML format. So, to get the latest information about their product and its current state, we used their XML feed which is connected with their database. The XML file is access through a URL provided by those venders to us. CDFA uses this URL to parse and get data from it. After getting the data it is sent to DMA who indexed it and store in system's database. The second option for receiving latest information from the vender who did not have the facility of XML a web based interface is provided to them where they put their data regularly to keep us up to date. The venders used this Administrative application and add/edit/update/delete their product information, their stock states and any other information that is useful by us or end user. For actual system's implementation to access this section, the vender first has to get subscription from us. A signup form is available on LBDSS web based application where vender can get registered with us and available for end user to present their latest product and services.

After using this application almost 2 hour all the user are asked to give feedback about the application and tell how it help them in searching their required item. The survey results are very encouraged and the entire users give excellent rate to this application. The major feedback we received from user is they happy to receive latest information as it stops the wastage of their time to getting their required information. It helps them in searching shops or restaurants to which they are not familiar within seconds.

5 Conclusions and Future Work

Searching an item or services in an area where you are totally new or not much familiar is very difficult task. In these cases normally the people are very confused and often the service they are looking for is near to him but due to unfamiliarity about the location they goes to wrong direction and wastes their lots of time in searching. So, the purposes system is very helpful for that kind of users. The results obtained from the experiments are very successful and very impressive. All the users recommended this system as guider which guides them while they are moving in unknown location.

The obtained results encourage us to expend this idea and make it more useful for its user. So in our future plan we decided to setup mobile alerts about the deals, the user have the facility to communicate directly with the vender to get information through this application. Instead of ask vender to give us XML we plan to make a crawler who will visit the web pages of the vender and fetch information from them.

References

1. K. Virrantaus, J. Markkula, A. Garmash and Y. V. Terziyan: Developing GIS-Supported Location-Based Services. In: Proceedings of WGIS'2001 - First International Workshop on Web Geographical Information Systems, Kyoto, Japan, pp. 423−432 (2001)
2. T D'Roza and G Bilchev: An overview of location-based services. In: BT Technology Journal, vol. 21(1), pp. 20−27 (January 2003)
3. M. L. Yiu, N. Mamoulis, and D. Papadias: Aggregate nearest neighbor queries in road networks. In: IEEE Trans. Knowl. Data Eng., pp. 820−833 (2005)
4. http://www.blingdar.com/ (visited on 1st Jan 2013)
5. http://www.groupon.com (visited on 1st Jan 2013)
6. S. Mary: The Coming-of-Age of Software Architecture Research. In: Proc. of the 23rd International Conference on Software Engineering, pp. 656, (2001)
7. M. Nenad, S.R. David, F.R. David and E.R. Jason: Modeling software architectures in the Unified Modeling Language. In: ACM Transactions on Software Engineering and Methodology (TOSEM), vol. 11, pp. 2−57 (2002)
8. Khaoula ADDAKIRI, Mohamed BAHAJ, Noreddine GHERABI: XML-based agent communication and migration for distributed applications in Mobile-C. In: International Journal of Modern Engineering Research (IJMER), vol. 2(1), pp. 403−407 (Jan−Feb) (2012)
9. Extensible Markup Language (XML). The World Wide Web Consortium. http://www.w3.org/XML (visited on 1st Jan) (2013)
10. http://developer.android.com (visited on 1st Jan) (2013)
11. www.designerandroid.com (visited on 1st Jan) (2013)
12. Séverac, G., Hvilshøj, M., Bøgh, S.: XML−BASED MULTI AGENT COMMUNICATION AND SCHEDULING IN INDUSTRIAL ENVIRONMENTS; In: MITIP 2010, Aalborg, (30−31 August) (2010)
13. Davies N., Cheverst K., Efrat A.: Using and Determining Location in a Context-Sensitive Tour Guide. In: IEEE Computer, vol. 34(8), pp. 35−41 (2001)
14. Virrantaus K., Markkula J., Garmash A., Terziyan V., Veijalainen J., Katanosov A., Tirri H.: Developing GIS-supported location-based services. In: Proceedings of the Second International Conference on Web Information System Engineering, vol. 2, pp. 66−75 (2001)
15. S. R. Balasundaram, Saravanarr, A.: Resource Identification Using Mobile Queries. In: ACEEE Int. J. on Communication, vol. 1(3), pp. 25 – 27 (Dec) (2010).
16. Hae Don Chon, Divyakant Agrawal and Amr El Abbadi: NAPA :Nearest Available Parking lot Application. In: IEEE Proceedings of the 18th International Conference on Data Engineering, (2002)
17. Gonzalo Mendez, Pilar Herrero, and Ramon Valladares: SIAPAS: A Case Study on the Use of a GPS-Based Parking System. In: OTM 2006 Workshops, LNCS vol. 4277, pp. 945−954 (2006)
18. http://www.google.com/mobile/maps/ (visited on 1st Jan) (2013)

Characterizing Scars in the Cerebral Cortex by Analyzing Intensities in T2/MRI Sequences

Ivonne M. Avila-Mora[1], Sonia Mendoza[1], Kimberly García[1], Dominique Decouchant[2,3], Liliana Puente-Maury[3], Rosa Delia Delgado-Hernández[4], Oscar René Marrufo-Meléndez[4], Daniel San Juan-Orta[4]

[1] Department of Computer Science, CINVESTAV-IPN, DF, Mexico
[2] C.N.R.S. - LIG, St Martin d'Hères, France
[3] Department of Information Technologies, UAM-Cuajimalpa, DF, Mexico
[4] Instituto Nacional de Neurología y Neurocirugía, DF, Mexico
iavila@computacion.cs.cinvestav.mx, smendoza@cs.cinvestav.mx, kimberly@computacion.cs.cinvestav.mx decouchant@correo.cua.uam.mx, lpuente@correo.cua.uam.mx

Abstract. The detection of scars in the cerebral cortex usually involves a manual process performed by radiologists, who have to face multiple troubles. For example, the bad calibration of the equipment used to get images of the cerebral cortex can cause spacial and geometrical distortions in the MRI (Magnetic Resonance Imaging) sequences. Owing to the advances in algorithms capable of analyzing MRI sequences, it is possible to automatically detect scars in the cerebral cortex in a successful way. In addition, an automatic process for finding scars can decrease the subjectivity of human interpretations and serve as a tool to support diagnoses of diseases. In this paper, we propose a new methodology to detect scars in the cerebral cortex by means of the analysis of intensities in T2/MRI sequences. In particular, we implement three main algorithms: the region growing, thresholds, and artificial neural networks.

Keywords: Detection of scars, cerebral cortex, intensity, Magnetic Resonance Imaging.

1 Introduction

Nowadays, some manual processes performed by humans, such as the analysis of medical images, have been automated in order to accomplish different objectives. Several researchers have been focused on the development of projects that employing computing algorithms allow humans to get a fast and less subjective response regarding diagnoses of diseases. The kind of systems created to automate the manual processes that experts in the medical field perform is known as Computer Aided-Diagnosis (CAD) [18, 30].

There exist different kinds of medical images, e.g., Ultrasound, Computed Tomography (CT), and X rays, which are processed by CAD tools. In particular, MRI sequences are built considering the intensities of human body tissues, which are captured by the receiving antennas of a MRI machine. Later, these intensities are transformed into images, which are analyzed by radiologists and neurologists, to find anomalies in tissue structures, malfunctions, diseases, etc. Specifically talking about the brain, the analysis of intensities in MRI sequences facilitates the detection of multiple sclerosis, Alzheimer, tumors, etc.

There are different types of MRI sequences, being T1 and T2 the main ones. Each one of them has a different range of intensities, which represents a difficulty when analyzing imaging sequences through algorithms. In particular, the MRI segmentation task becomes more complex, because it is not possible to establish ranges in which organs can be located, as in the case of CT. If there were ranges of intensities established, such as the Hounsfield values in CT [28], it would be simpler to classify tissues when performing automatic imaging analysis, since their intensities are located in those ranges. Another problem is that those intensities change each time images of an organism are taken, because the MRI machine works with hydrogen spin atoms present in the patient's body at the moment the images are taken. Thus, even when the captured images correspond to the same person, their intensity values will not remain constant.

Digital analysis of medical images is an important topic of Artificial Intelligence, since a computer should have a knowledge degree in order to make decisions. That knowledge is derived from a learning phase, in which it is necessary to process labeled data. In this paper, we propose a new methodology to automate the detection of scars in the cerebral cortex. Our methodology focuses on the analysis of intensities in MRI sequences of the cerebral cortex, in particular, in T2 sequences. This analysis is made by using segmentation and pattern recognition algorithms, such as growing regions [26] and thresholding [27]. To make the final decision about the nature of a detected object, the use of artificial neural networks [13] is proposed.

This paper is organized as follows. In section 2, related work regarding the detection of cerebral injuries is presented in order to expose their strong and weak points. In section 3, the proposed methodology to automatically detect scars in the cerebral cortex is explained in detail. Following, section 4 describes the established parameters of the experimentation phase and analyzes the obtained results. Finally, in section 5, the conclusions and future work are presented.

2 Related Work

Thanks to Computer Science, the analysis of data has been automated by means of algorithms. One of the most common approaches is the statistical one. Under it, several research works have been made, such as Khayati et al.'s [15], whose objective is to find multiple sclerosis injuries in the cerebral cortex by using FLAIR sequences, specially on axial cuts. To find these injuries, they propose a Bayesian approach, which employs Markov random fields [11, 24] and adaptive mixtures [4] to compute *a priori* probabilities stored in a feature vector that is used to classify the types of analyzed tissues. Radiologists and neurologists have manually performed the selection of images in which injuries are present. They have analyzed spatial characteristics and regions of the brain to extract it from images. The feature vector is built from the intensities of FLAIR sequences. The advantage of using Markov random fields and adaptive mixtures is that it is not necessary to perform a training phase, in order to make the system learn and make decisions, since *a priori* information is already known.

Kobashi et al. [16] propose the SoS system, which consists in analyzing different types of medical images. Among them, MRI sequences are processed to get relevant information about neonates, e.g., the brain volume, the brain deformity index, the *sulcal* deformation index, the *gyral* area, and the brain representation. From these measures, the SoS model provides data that can be analyzed lately by radiologists and neurologists. The SoS model gives useful information and provides a degree of certainty, even if it relies on fuzzy methods [14].

Nakamura et al. [22] focus on measuring the brain volume when it presents multiple sclerosis injuries. Since these are known as black holes, neurologists consider that the brain loses volume when presenting such holes. To extract the brain from images, an anisotropic mask filter [3, 6] is used. This mask filter attenuates the background of images to improve the contrast of the noise or irrelevant information. Like Kobashi et al., Nakamura et al. classify different tissues (e.g., white matter, gray matter, and cerebrospinal fluid) by processing some probabilities, which are obtained from the intensities, anatomy, and morphology of the brain tissues and are used to create masks for each class.

Klöppel et al. [25] make a comparison of different methods to detect injuries in white matter, analyzing MRI sequences. They employ algorithms from the statistical approach e.g., the k-nearest neighbor algorithm [7] and support vector machines [21]. Multiple thresholds are also established based on the intensity of MRI images. On their research, Klöppel et al. combine T1 and FLAIR sequences from demented patients. It is important to notice that Klöppel et al. normalize their images, i.e., they do not work with pure intensities of tissues.

Arimura et al.'s project [2] has been in development for several years; that is why it includes the automated detection of various diseases. To start, during data processing, they use normalization and smoothing filters [8]. Depending on the injury and the extracted features, different methods are used. In case of aneurisms, Arimura et al. use a 3D Gaussian filter [29] considering the shape of injuries. Some other segmentation techniques have been employed, such as the growing regions [26], snake [9], watermarks [1], and level set [23] algorithms. For multiple sclerosis, the shape of injuries is also taken into account to establish candidates. Afterwards, the distance is measured between the places where injuries are located and the lateral ventricles, because Arimura et al. claim that sclerosis is located at a certain distance from the center of the brain. For detecting Alzheimer, support vector machines are used to measure some characteristics, such as the volume of the cerebrospinal fluid.

To detect tumors, Arimura et al. propose a neural network with the following architecture: fifteen neurons on the input layer in which each neuron corresponds to a characteristic, e.g., location of a tumor, whether it involves an edema or not, heterogeneity, etc. In the hidden layer, there are six neurons and, in the output layer, there are four neurons for: high-grade glioma, low-grade glioma, metastasis, and malignant lymphomas, respectively.

Arimura et al.'s work is extensive and really complete. They have used different algorithms, which give a wide outlook of which techniques should be used to obtain better results, when detecting a specific injury. In this case, the detected problems have involved the brain, but in a similar way, these algorithms could work with any part, type of tissues or organs of the human body.

Fu et al. [10] propose the use of artificial neural networks [13], Expectation-Maximization [5], and the C-means algorithm [20] to classify the brain tissues into white matter, gray matter, and cerebrospinal fluid.

Finally, Yamamoto et al. [30] use a false-positive reduction scheme [19], a rule-based algorithm, and support vector machines to detect multiple sclerosis. In this work, T1, T2 and FLAIR MRI sequences are analyzed.

All these research works deal with a specific illness. In this paper, we propose a new approach, which objective is to detect scars in the cerebral cortex, no matter the illness, by analyzing intensities in MRI sequences. This new approach aims at providing radiologists and neurologists with a second opinion.

3 Methodology for the Automatic Detection of Scars in the Cerebral Cortex

The following section describes the methodology proposed in this paper to automate the detection of scars in the cerebral cortex by analyzing MRI sequences. The images of these sequences have a size of 256x256 pixels in grayscale and can be represented as T1 with contrast and simple, T2 with contrast and simple, FLAIR, among others. In practice, radiologists use T2 sequences to identify scars in the cerebral cortex, since the human eye can identify lesions in the brain easily, owing to perceptible changes in the intensity of pixels. In the case of T1 sequences, intensities and shapes change with respect to T2 sequences for both the human eye and the computer. Consequently, the identification of scars in T1 sequences is not as straightforward as in T2 sequences.

3.1 Cerebral Tissue Extraction Through the Region Growing Algorithm

In all imaging sequences, tissues of all kinds can be observed, e.g., skull, ocular, fat, nerve, etc., and must be removed in order to extract the tissue of interest. This step can be accomplished through the region growing algorithm [26] with variable seed to only extract the brain tissue area, which is the target tissue of the proposed methodology. It is also possible to carry out a histogram analysis [12] because of the difference in the imaging intensities.

The main objective of the region growing algorithm is to group pixels, according to the similarity degree among the intensity values of neighboring pixels. There are two types of algorithms, semi-automatic and automatic. The former type involves an operator, while the latter only requires an operator to verify the result. Sant'anna et al. [26] propose two main tasks for the region growing algorithm: 1) to divide images into a number of homogeneous regions, where each region is labeled only once, and 2) to separate homogeneous regions from the rest with different properties. The steps of the region growing algorithm are:

1. **Selection of the seed.** In a manual way, a physician or a radiologist selects a starting point in an image that serves as seed, which will provide the coordinates of the initial pixel depending on the tissue to be extracted.
2. **Determination of the t threshold.** Necessary conditions for the algorithm to classify pixels into regions and the t threshold are determined.
3. **Establishment of the neighborhood.** Pixels are selected in order to be included in the neighborhood of the seed (see lines 7-14 from Algorithm 1).
4. **Iteration and recursion.** The neighborhood is defined for each visited pixel, so a path is determined in order to visit pixels not visited yet (see lines 1-18 from Algorithm 1).

Algorithm 1 Region growing segmentation

Require: An image, (x, y) coordinates of the seed, intensity, t threshold
Ensure: Brain tissue region extraction
1: **while** (stack $r \neq$ empty) **do**
2:     **if** (neighVisited(x, y) = 0) **then**
3:         $region \leftarrow image(x, y)$
4:         **if** (region<= (seed+ t)) & (region>= (seed-t)) **then**
5:             $neighVisited(x, y) \leftarrow 1$
6:             $segmentedImage(x, y) \leftarrow image(x, y)$
7:             pendingNeigh.push(x-1, y-1)
8:             pendingNeigh.push(x-1, y)
9:             pendingNeigh.push(x-1, y+1)
10:             pendingNeigh.push(x, y+1)
11:             pendingNeigh.push(x+1, y+1)
12:             pendingNeigh.push(x+1, y)
13:             pendingNeigh.push(x+1, y-1)
14:             pendingNeigh.push(x, y-1)
15:         **end if**
16:         $seed \leftarrow region$
17:     **end if**
18: **end while**

Within these steps, it is important to take into account three criteria. The first one corresponds to the way the seed is determined. Let us recall that the region growing algorithm can be automatic or semiautomatic. In the automatic algorithm, no intervention is required from a user to determine the coordinates of the seed. The algorithm becomes semi-automatic when the user indicates such coordinates as done in this work. This algorithm has been used because it is necessary to select a seed that corresponds to a pixel, which value is within the intensity range of the brain tissue, preventing the algorithm from selecting another tissue, such as eyes or skull. Since the segmentation process described in this paper is bi-dimensional, the seed has to be selected on each image.

The second criterion is the measure of homogeneity to determine whether a pixel belongs or not to a region of interest. At this point the t threshold is determined. We have obtained an approximate value of the brain tissue by means of the DICOM viewer called Osirix [17], a special tool for visualizing medical images. In this way, we have calculated and proposed the t threshold used for the experiments detailed in section 4.

The t threshold corresponds to the range established according to the intensity values of the region that requires segmentation. The similarity of pixels with respect to the seed is indicated by the absolute difference between the intensity of each pixel and the one of the seed. The t threshold is used to establish the following condition: if the absolute difference is less than t, then that pixel is added to the region of interest. If a large value of t is considered, there may be areas far apart, so the degree of homogeneity is lower. More, by setting a small value of t, the segmented region is more homogeneous and the data loss is less.

Finally, the third criterion refers to the conditions the algorithm must follow to stop, i.e., how and when. We have used a stack, in order to establish a path that allows the algorithm to return and to verify whether there is a neighbor that has not been visited. In this way, this stack allows the algorithm to go over the whole image looking for pixels that correspond to the brain tissue.

3.2 Detection of Suspicious Regions Using Thresholding

Thresholding is one of the simplest techniques for separating or labeling pixels in an image [27]. This technique determines a value, called threshold, that decides the class (i.e., a region) a pixel belongs to according to its intensity value. The definition of the t threshold can be expressed as follows [31]:

Let p be the analyzed pixel of an image, which has to be assigned to either a class P_0 or P_1 according to the conditions $I(p) < t$ or $I(p) > t$.

The intensity is an imaging characteristic, so the t threshold measures this property, in order to separate an object from the image background. The t threshold can be fixed or variable, i.e., its value changes according to the segmentation needs. If the intensity of the pixel being analyzed exceeds the t threshold, that pixel belongs to the class of the object of interest. On the contrary, if the intensity of that pixel is less than t, it is considered as a part of the background.

3.3 Determination of Scars Using Artificial Neural Networks

All the regions obtained during the detection phase of suspicious spots are processed, in order to be placed into two classes: scar or non-scar. An artificial neural network is used in the classification process, which input neurons correspond to the location (x, y) and the intensity of the pixels belonging to such regions.

To display the results, it is necessary to implement a scientific visualization technique, such as 3D reconstruction, and to use colors that allow radiologists to distinguish suspicious tissue regions, i.e., the system suggests a tissue susceptible to be a scar. The development of this tool is out of the scope of this paper.

4 Experiments and Results

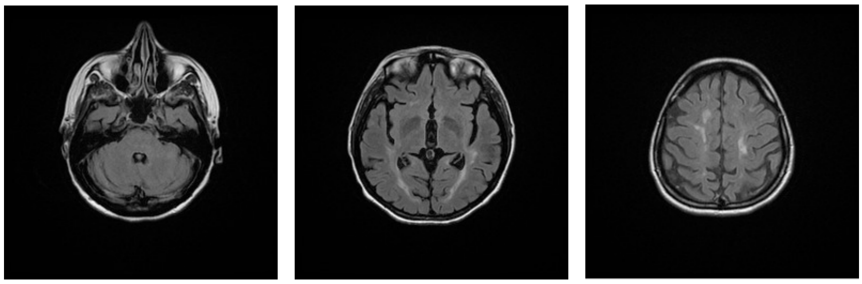
In this section, we describe two tests of the proposed methodology when applying it on the MRI sequences provided by the Mexican Neurology and Neurosurgery National Institute. On these MRI sequences, scars were previously detected through the traditional method used by experts in the medical area, i.e., observation. In particular, we describe the characteristics of T2 sequences, the parameters used in the experimental stage and the results obtained when processing medical images using the proposed methodology.

4.1 The Former Sequence

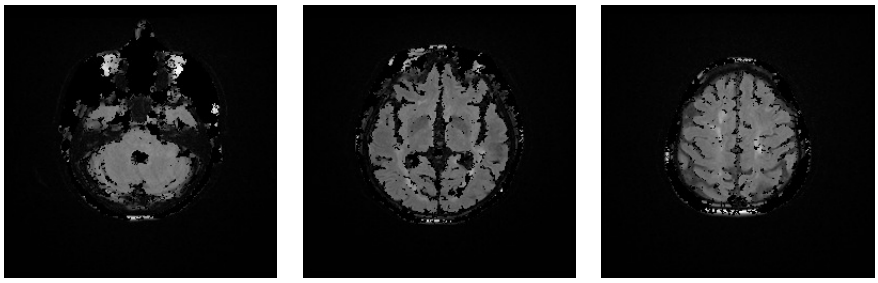
In this section, we present the results obtained from processing the former MRI sequence, which consists of 24 images of axial cuts from the T2 type. The patient is a 42 years-old female, whose disease is multiple sclerosis. Sequence values are in a range of intensities from 0 to 4052.
1) Extraction of the brain tissue region. Extracting the soft tissue is part of the preparation process of the images that are analyzed in the detection phase of scars. The main task of this first phase is the extraction of brain tissue, while avoiding losing information, e.g., removing the skull and other tissues irrelevant to this study. The parameters needed to carry out the former segmentation of images by means of the region growing algorithm are: 1) the coordinates of the seed, which are (100, 125) pixels, 2) the intensity value, corresponding to the average of the region of interest in Osirix, which according to the coordinates of the seed is 1089, and 3) the t threshold, which is 72.

Figure 1 shows some segmented images from the former sequence. Let us notice that some traces of skull and other tissues, such as eyes, are still present in the segmented images. The existence of these residues is due to the fact that the intensity values of the brain are taken as a basis to define the t threshold, so it is possible for this intensity value to be present in some other tissues, which is considered when running the region growing algorithm.
2) Detection of suspicious regions. As mentioned in Section III.B once the cerebral and skull regions have been separated, the next step is to identify, within the brain tissue, regions that can be labeled as scars (suspicious spots). It is necessary to carry out another segmentation process, in order to extract information about each one of the regions that will be considered as suspicious scars.

The viewer Osirix allows us to obtain the corresponding intensity values of each tissue. This viewer has a tool that gives the average value of the selected region. In this way, we have obtained the approximate value of a scar and established the threshold values. The proposed thresholds for extracting the brain tissue and detecting scars in T2 sequences are ± 72 and ± 385, respectively. These parameters are useful for testing both the former and latter sequences.



(a) Original image 5 (b) Original image 13 (c) Original image 23



(d) Brain extraction in image 5 (e) Brain extraction in image 13 (f) Segmented image 23

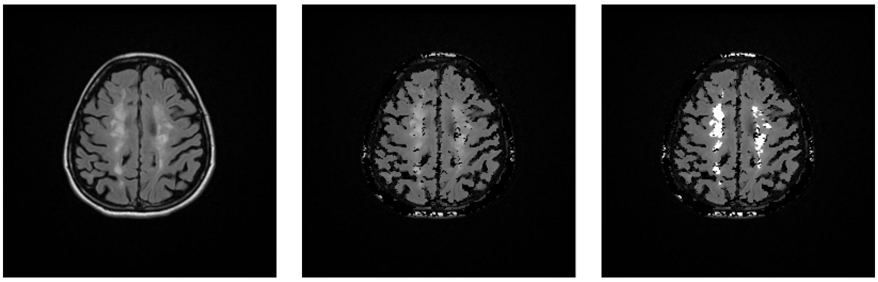
Fig. 1. Segmentation of the former sequence. We can see at the top, three images from the original sequence and, at the bottom, the corresponding segmented images, in which the brain tissue has been isolated. These two sets of images allow us to make a comparison between the original sequence and another where the skull and other tissues have been removed from each image. In figures (a), (b) and (c) there are some tissues that are not part of the brain, e.g., eyes. Once the segmentation process has been done, there are still tissues that do not correspond with the brain tissue, as shown in figures (d), (e) and (f)

Figure 2 shows the results regarding the images 21 (c) and 22 (d). The suspicious spots can be seen in (e) and (f). Once detected by the histogram analysis algorithm, the scars are highlighted by means of an intense white color, in order to be clearly identifiable. The value for the region of interest, i.e., the areas corresponding to the scars, is 1950 and the t threshold is 390.

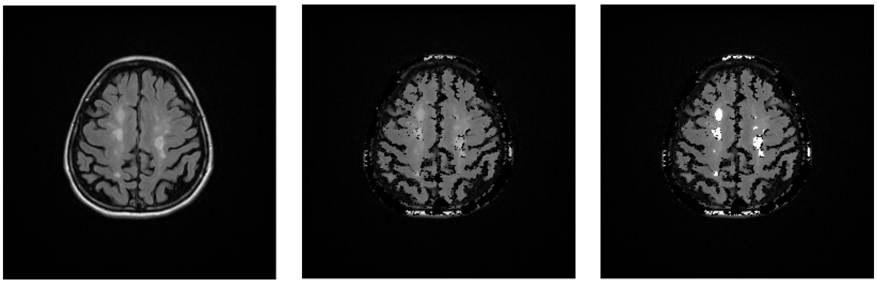
4.2 The Latter Sequence

The latter sequence consists of 24 images of axial cuts from the T2 type that correspond to a 43 years-old female patient with multiple sclerosis. The intensity levels are approximately between 0 and 3384.

Figure 3 shows the results of the segmentation performed. In particular, figures (a), (b), and (c) correspond respectively to the 18, 19 and 20 images of the original sequence. In these images, there are visually distinctive scars. Using the region growing algorithm, the cerebral tissue was obtained as observed in figures



(a) Original image 21 (b) Brain extraction in image 21 (c) Scars detected in image 21



(d) Original image 22 (e) Brain extraction in image 22 (f) Scars detected in image 22

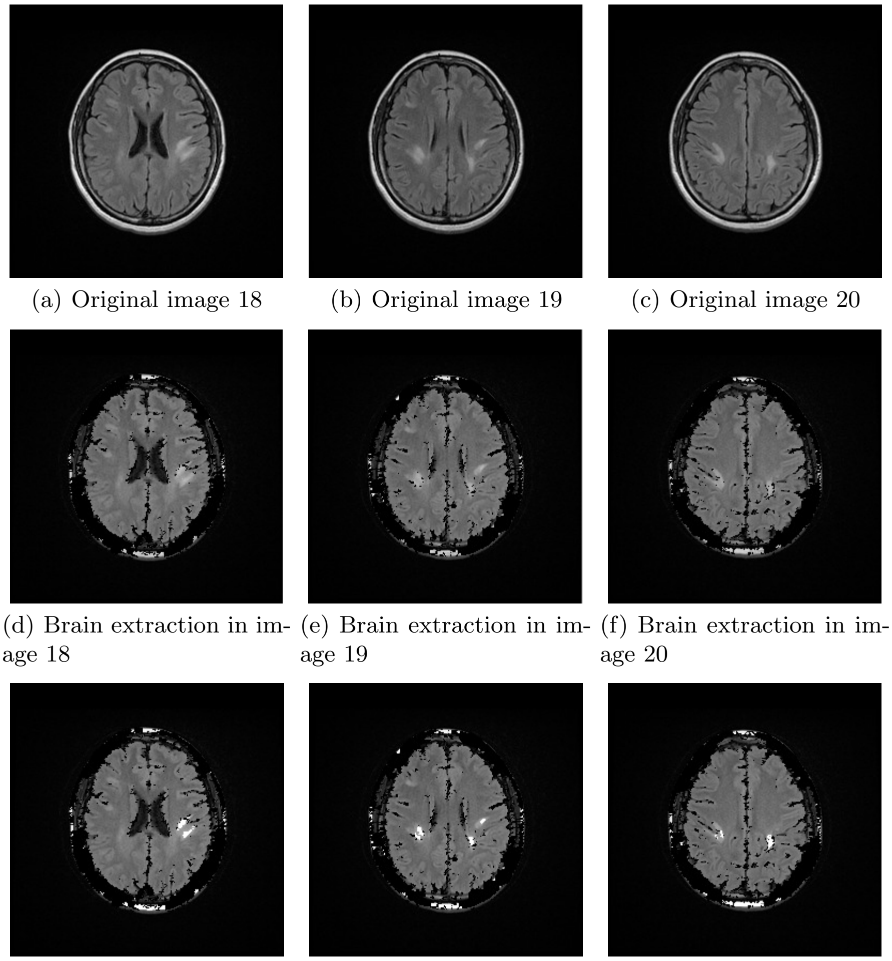
Fig. 2. Figures (a) and (b) show respectively the images 21 and 22 before being processed. Figures (c) and (d) correspond to the results of the former segmentation phase: the brain extraction. Finally, in figures (e) and (f) scars have been highlighted with a more intense color. In this case, there are other pixels that, according to the intensity values, have answered to the threshold as if they were scars, e.g., the skull tissue. This is the reason that indicates that another process has to be done: the use of artificial neural networks, so a decision of whether a tissue is or not a scar can be made.

(d), (e) and (f). Finally, in figures (g), (h) and (i), there can be distinguished some scars from the remaining tissue. In order to standardize results, the values of the parameters used to test this latter sequence were the same as those used in the former sequence.

It is important to notice that the values of parameters were tested in both sequences and that, despite the intensity variations between these sequences, it is possible to remove soft tissue and identify regions considered as scars. We have decided to analyze T2 sequences, because neurologists rely on this type of sequences when it is necessary to identify scars through manual processes.

4.3 Discussion

According to the performed tests and obtained results at applying our methodology, some observations are made.



(a) Original image 18 (b) Original image 19 (c) Original image 20

(d) Brain extraction in image 18 (e) Brain extraction in image 19 (f) Brain extraction in image 20

(g) Scars detected in image 18 (h) Scars detected in image 19 (i) Scars detected in image 20

Fig. 3. Segmentation of the latter sequence. Figures (a), (b) and (c) correspond to images 18, 19 and 20, respectively from the original sequence. Figures (d), (e) and (f) show the results obtained from the brain extraction phase done by the region growing algorithm. Figures (g), (h) and (i) show scars detected by the t threshold

MRI sequences present different ranges of intensities due to: 1) the type of sequence (e.g., T1 or T2) and their respective variants (e.g., T1 with and without contrast) and 2) the conditions of the patient's organism, which generate changes in intensities when taking images. Despite this fact, variations are not so important, i.e., a T1 sequence can be well differentiated from a T2 sequence, because the intensity values belong to an average range more or less established in both types of sequences. Nevertheless, these intensity variations make the work performed by the computer difficult, specially during the extraction of the cerebral tissue and the detection of scars.

However, most of the cerebral tissue region is segmented, it is necessary to establish the way the t threshold should be computed, in order to be highly accurate and flexible to changes in the ranges of intensities for each sequence. To identify suspicious regions, it is necessary to establish a robust t threshold, as in the case of the former segmentation, due to intensity variations.

For the methodology proposed in this paper, we have considered variations in the range of intensities for each sequence. Thus, we have established a t threshold that allows the region growing algorithm to be flexible enough to segment the greater part of the brain tissue. During the latter phase of this methodology, using another t threshold, it is possible to extract suspicious regions once the brain has been segmented.

5 Conclusions and Future Work

Through the development of this research work, a new methodology to automatically detect scars in the cerebral cortex is proposed. The work done by other researchers has been focused on creating Computer Aided-Diagnosis tools specialized in the study of diseases causing brain scars, e.g., epilepsy, multiple sclerosis, etc. Instead, the proposed methodology relies on pattern recognition algorithms to find scars in the cerebral cortex, regardless the type of disease provoking these scars. These algorithms act as a basis to create a tool that decreases the subjectivity of diagnosis and that makes the research tasks that neurologists perform at hospitals easier. Another contribution of the proposed methodology consists in making Computer Sciences stronger and wider, owing to the interdisciplinary nature of this project, which brings together different fields of study, such as neurology, physics, and bioinformatics.

Some important extensions of this work are now stated. First, we aim at reinforcing the computing of thresholds, since they must present a flexible behavior, i.e., they should adapt themselves to different intensities for the same type of sequences, T1 or T2, and to the difference existing in the range of each sequence, e.g., intensities for T2 are between 0 and 3900 in average.

As we want to extract the cerebral tissue region in three dimensions, it is necessary to adjust the region growing algorithm so that, according to the seed, this algorithm would work not only into the current image, but also into the previous and following images.

To make a final decision regarding the identified object, currently, a neural network is in development. It would decide whether such an object is a scar or not. We also aim at building a three dimensional representation of the brain to show the location of the scars in a graphical way. Finally, a multispectral analysis would be made, in order to find scars in T1 sequences.

It is important to emphasize that Computer Aided-Diagnosis tools are not accurate. Instead, they provide a closer look of what is happening inside a human organism, in order to help radiologists and physicians in the diagnosis of diseases.

The response of these tools is a second opinion, which should be verified by an expert of the medical field.

Acknowledgment

The research work presented in this paper would not have been possible without the valuable direction and advise of Dr. Adriano de Luca Pennacchia, who was a researcher of the Computer Science Department - CINVESTAV-IPN.

References

1. R. S. Alomari and A. Al-Jaber. A fragile watermarking algorithm for content authentication. *Int. J. of Computing & Information Sciences*, 2(1):27–37, 2004.
2. H. Arimura, T. Magome, Y. Yamashita, and D. Yamamoto. Computer-aided diagnosis systems for brain diseases in magnetic resonance images. *Algorithms*, 2(3):925–952, 2009.
3. M. A. Balafar, A. R. Ramli, M. I. Saripan, R. Mahmud, and S. Mashohor. Medical image segmentation using anisotropic filter, user interaction and fuzzy c-mean (fcm). *Advanced Intelligent Computing Theories and Applications. With Aspects of Contemporary Intelligent Computing Techniques. Communications in Computer and Information Science*, 15:169–176, 2008.
4. A. Bijar. Segmentation of ms lesions using active contour model, adaptive mixtures method and m r f model. *2011 7th International Symposium on Image and Signal Processing and Analysis (ISPA)*, 1:159–164, 2011.
5. S. Borman. The expectation maximization algorithm: A short tutorial. Technical report, 2009.
6. R. Demirci. Fuzzy adaptive anisotropic filter for medical images. *Expert Systems with Applications*, 27(3):219–229, 2010.
7. T. Donoeux. A k-nearest neighbor classification rule based on dempster-shafer theory. *IEEE Trans. on Systems, Man and Cybernetics*, 25(5):804–813.
8. A. Doucet and A. M. Johansen. A tutorial on particle filtering and smoothing: Fifteen years later. pages 1–37, 2008.
9. H. Fang, J. Kim, and J. Jang. A fast snake algorithm for tracking multiple objects. *Journal of Information Processing Systems*, 7(3):519–530, 2011.
10. J. C. Fu, C. C. Chen, W. J. Chai, S. T. C. Wong, and I. C. Li. Image segmentation by em-based adaptative pulse cupled neural networks in brain magnetic resonance imaging. *Computerized Medical Imaging and Graphics*, 34(4):308–320, 2010.
11. S. Geman and C. Graffigne. Markov random field image models and their applications to computer vision. *Proceedings of the International Congress of Mathematicians Berkeley*, 2:1496–1517, 1987.
12. H. Gonalves, J. A. Gonalves, and L. Corte-Real. Hairis: A method for automatic image registration through histogram-based image segmentation. *IEEE Transactions on Image Processing*, 20(3):776–789, 2011.
13. S. Haykin. Neural networks: A comprehensive foundation. *Pearson Educ*, 2005.
14. E. Hüllermeier. Fuzzy methods in machine learning and data mining: Status and prospects. *Fuzzy Sets and Systems*, 156(3):387–406, 2005.
15. R. Khayati, M. Vafadust, T. Farzad, and S. M. Nabavi. Fully automatic segmentation of multiple sclerosis lesions in brain mr flair images using adaptative mixtures method and markov random file model.

16. S. Kobashi, D. Yokomichi, K. Yamaguchi, K. Kuramoto, S. Imawaki, and Y. Hata. Computer-aided diagnosis system of systems for neonatal human. *IEEE 5th International Conference on System of Systems Engineering*, pages 1–6, 2010.
17. T. Limberg. Osirix as a resource. *MS, Biomedical Visualization, University of Illinois*, pages 1–23, 2008.
18. R. Linder, D. Mörshner, S. J. Pöppl, and A. Moser. Computer-aided diagnosis of multiple slcerosis. *Computational and Mathematical Methods in Medicine*, 10(1):39–47, 2009.
19. X. Lladó, A. Oliver, J. Martí, and J. Freixenet. Dealing with false positive reduction in mammographic mass detection. *Med Image Comput Comput Assist Interv*, 10(1):286–293, 2007.
20. A. Mekhmoukh, K. Mokrani, and M. Cheriet. A modified kernelized fuzzy c-means algorithm for noisy images segmentation: Application to mri images. *IJCSI International Journal of Computer Science Issues*, 9(1):172–176, January 2012.
21. D. Meyer. Suport vector machines. pages 1–8, 2012.
22. K. Nakamura and E. Fisher. Segmentation of brain magnetic resonance images for measurement of gray matter atrophy in multiple sclerosis patients. *NeuroImage*, 44(3):769–776, 2009.
23. S. Osher. Geometric level set methods in imaging, vision, and graphics. *Level Set Methods, Springer*, 2003.
24. J. P. Monaco and A. Madabhushi. Class-specific weighting for markov random field estimation: Application to medical image segmentation. *Medical Image Analysis*, 16(8):1477–1489, 2012.
25. Klöppel S., Abdulkadir A., Hadjidemetriou S., Issleib S., Frings L., Than T. N., Mader I., S. J. Teipel, Hüll M., and Ronneberger O. A comparisson of different automated methods for the detection of whitte matter lesions in mri data. *NeuroImage*, 57(2):416–422, 2011.
26. L. Sant'anna Bins, L. M. Garcia-Fonseca, G. J. Erthal, and F. Mitsuo. Satellite imagery segmentation a region growing approach. *Anais VIII Simpósio Brasileiro de Sensoriamento Remoto.*
27. M. Sezgin and B. Sankur. Survey over image thresholding techniques and quantitative performance evaluation. *Journal of Electronic Imaging*, 13(1):146–165, 2004.
28. Shapurian T., Damoulis P. D., Reiser G. M., T. J. Griffin, and W. M. Rand. Quantitative evaluation of bone density using the hounsfield index. *The International Journal of Oral & Maxillofacial Implants*, 21(2):290–298, 2006.
29. L. Wang, P. Liu, J. Tan, L. Wang, and X. Ding. Note: How to design gaussian filtlei, wang er with eccentricity and probe offset taken into account for roundness measurement of small radius. *Review of Scientific Instruments*, 83(5):056105 – 056105–3, 2012.
30. D. Yamamoto, H. Arimura, S. Kakeda, T. Magome, Y. Yamashita, F. Toyofuku, M. Ohki, Y. Higashida, and Y. Korogi. Computer-aided detection of multiple sclerosis in brain magnetic resonance images: False positive reduction scheme consisted of rule-based, level set method, and support vector machine. *Computerized medical imaging and graphics the official journal of the Computerized Medical Imaging Society, Elsevier Ltd*, 34(5):404–413, 2010.
31. J. Zhang and Jinglu H. Image segmentation based on 2d otsu method with histogram analysis. *2008 International Conference on Computer Science and Software Engineering*, 6.

Use of PSO for Obtaining Solution of the Inverse Robot Dynamic Model

Carlos Alberto Yahir Hervert Cano, Angel Rodríguez-Liñán,
Luis M. Torres-Treviño

Universidad Autónoma de Nuevo León, UANL, FIME, San Nicolás de los Garza, Nuevo León, Mexico
hervertkno@gmail.com, angel.rodriguezln@uanl.edu.mx,
luis.torres.ciidit@gmail.com

Abstract. The inverse dynamic model allows to compute the torques for each of the degrees of freedom of a robot knowing the positions, velocities and accelerations. In contrast, direct dynamic model is used to calculate the positions, velocities and accelerations knowing the torques for each of the degrees of freedom. Most of the times is difficult to obtain the direct dynamic model analytically because it is given by solution of the differential equations for each variable. For that reason, we propose to compute the positions, velocities and accelerations from the inverse dynamic model using a simple and powerful intelligent algorithm called Particle Swarm Optimization (PSO) without needing to obtain the direct dynamic model.

Keywords: PSO, inverse robot dynamic model.

1 Introduction

Modelling the robot dynamics allows to understand the relationship between the movement of the robot links and its forces. This relationship is obtained by dynamic models which relate mathematically:

1. The location of the robot defined by their joint variables: Position, velocity and acceleration.
2. The forces and torques applied to the joints.
3. Robot dimensional parameters, such as length, mass and inertia of their elements.

A method to obtain the robot dynamic model of n degrees of freedom with rigid links, is given by the equations of motion Euler-Lagrange [6, 1]. The Lagrangian Ł is defined as the difference of the kinetic energy K and potential energy U:

$$Ł = K - U$$

Then, the inverse dynamic model is expressed by the n equations of Euler-Lagrange:

$$\frac{d}{dt}\frac{\partial L}{\partial \dot{q}_i} - \frac{\partial L}{\partial q_i} = \tau_i \tag{1}$$

where q_i is the joint position, $\dot{q}_i$ is the joint velocity, and τ_i is the force (or torque) of the i-th link and $i = 1, 2, ..., n$. The inverse dynamic model (1) can be rewritten in their compact form and notation most widely used in robotics [2, 9]:

$$\tau = D(q)\ddot{q} + C(q,\dot{q})\dot{q} + G(q) + F_c sign(\dot{q}) + F_v \dot{q} \tag{2}$$

where $\tau \in \Re^n$, $q \in \Re^n$, $\dot{q} \in \Re^n$, $\ddot{q} \in \Re^n$ denote the vectors of applied torques, position, velocity, and acceleration in the robot joints, respectively. $D(q) \in \Re^{n\times n}$ is the inertia matrix, $C(q,\dot{q}) \in \Re^{n\times n}$ is the Coriolis and centripetal matrix, $G(q) \in \Re^n$ is the gravity vector, $F_c \in \Re^{n\times n}$ is the Coulomb friction and $F_v \in \Re^{n\times n}$ is the viscous friction.

The goal of this article is to get positions, velocities and accelerations satisfying the inverse dynamic model (2) without solving the n differential equations (1) because most of the time is very difficult solve them. We propose in this work the use of Particle Swarm Optimization (PSO). PSO and Evolutionary Algorithms are optimization tools that are inspired by natural phenomena.The PSO, in particular, was motivated by the simulation of bird flocking or fish schooling. PSO was first introduced by Kennedy and Eberhart in 1995 [7].

In recent years, the relationship between the dynamic models and Intelligent Systems as PSO has been an important merger, since the Intelligent Systems have started to get several applications in robotics as:

⇒ In parameter estimation of robot dynamics because most of the times is very difficult to know the exact parameters or even impossible to obtain all the parameters [5].
⇒ For obtaining the positions, velocities and accelerations of the inverse dynamic model [4].
⇒ In the control of swarm robots [3].

The rest of document is organized as follows: In the next section, the PSO is presented. In Section 3, the inverse dynamic model of three study cases are shown. Experimental results from simulations with PSO are illustrated in Section 4. Finally, some conclusions are given.

2 Particle Swarm Optimization

The PSO algorithm [5, 8, 11] assigns a swarm of k particles to search for the optimal solution in a m-dimensional space, where m is the amount of position components in each and every one of particles. The optimal solution to the problem consists in that any particle find a point or trajectory in the m-dimensional space that minimize some fitness function $f(x)$. The starting position of a particle is randomly set within the range of possible solutions to the problem. The range is based on an intuitive guess of the maximum and minimum possible values of each component of the particle positions p. For the i-th particle is valued the fitness function $p_{b_i} = f(p_i)$ of its current position $p_i \in \Re^m$ that determined the current best $p_{b_{i,t}}$ in the t-th iteration, with $t = \{1, 2, ..., id\}$ as the number of iterations, and has memory of its own best experience $P_{best,i} \in \Re^m$, which is compared to $p_{b_{i,t}}$ in t-th iteration, and is replaced by $p_{b_{i,t}}$ if $f(p_i(t)) < f(P_{best,i}(t-1))$. Besides its own best experience, each particle has knowledge of the best experience achieved by the entire swarm, that is the global best experience denoted by $G_{best} \in \Re^m$ such that $f(G_{best}) = \min_i(f(P_{best,i}))$. Based on the data each agent has, its velocity in the t-th iteration is determined by

$$v_i(t) = wv_i(t-1) + c_1 r_1 (P_{best,i}(t) - p_i(t)) + c_2 r_2 (G_{best}(t) - p_i(t)) \qquad (3)$$

where $v_i \in \Re^m$ is the velocity of i-th particle, $w \in \Re$ is the inertia weight, $c_1 \in \Re$ is a constant positive cognitive learning rate, $c_2 \in \Re$ is a constant positive social learning rate, $r_1 \in [0, 1]$ and $r_2 \in [0, 1]$ are random numbers re-generated at each iteration.

The position of each particle at the $(t + 1)$-th iteration is updated by:

$$p_i(t+1) = p_i(t) + v_i(t) \qquad (4)$$

After predefined conditions are satisfy, the algorithm stops and the G_{best} at the latest iteration is taken as the optimal solution to the problem.

In this work, the PSO is used to obtain an estimated value of position q, velocity $\dot{q}$ and acceleration $\ddot{q}$ of n joints of a robot for each iteration, where we know the torque τ and inverse dynamic model (2). Then, the estimated torque is given by

$$\hat{\tau} = D(\hat{q})\ddot{\hat{q}} + C(\hat{q}, \dot{\hat{q}})\dot{\hat{q}} + G(\hat{q}) + F_c sign(\dot{\hat{q}}) + F_v \dot{\hat{q}} \qquad (5)$$

where $\hat{q} \in \Re^n$, $\dot{\hat{q}} \in \Re^n$ and $\ddot{\hat{q}} \in \Re^n$ are the estimated position, velocity and acceleration by PSO, respectively. Defining the estimation error $e_i(t) \in \Re^n$ for the i-th particle in the t-th iteration

$$e_i(t) = \|\tau - \hat{\tau}_i(t)\|_1 \qquad (6)$$

where $\| \cdot \|_1$ is the norm-1. Let $f : \Re^n \rightarrow \Re$ be the fitness function to be optimized, which compute the average error [10]

$$f(e_i) = \frac{(e_{i,1} + e_{i,2} + ... + e_{i,n})}{n} \qquad (7)$$

for n freedom degrees.

The objective of the PSO algorithm in the estimation task is to find the i-th set of variables q, $\dot{q}$, $\ddot{q}$ that minimizes the function $f(e_i)$.

3 Inverse Robot Dynamic Model

In this section, 3 manipulator models are shown in order of estimate their position, velocity and acceleration by means of the PSO algorithm presented. In the 3 cases, friction effects was not taken into account in the inverse dynamic models.

Case 1: Simple pendulum.

The inverse dynamic model of the simple pendulum without friction shown in Figure 1 is:

$$\tau = ml^2\ddot{q} + mgl\sin(q) \tag{8}$$

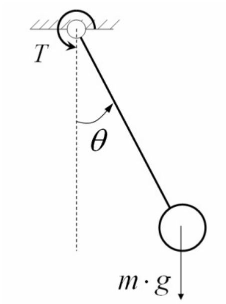


Fig. 1. Simple pendulum without friction

The simple pendulum of Figure 1 has 1 degree of freedom ($n = 1$), $q = \theta$ is the angular position, $\dot{q} = \dot{\theta}$ is the angular velocity, $\ddot{q} = \ddot{\theta}$ is the angular acceleration, the mass value was taken as $m = 1$ Kg and the length as $l = 1$ m, and the inverse model (8) is in form (2).

Case 2: Pendubot.

The inverse dynamic model of the ideal Pendubot shown in Figure 2 is:

$$\begin{aligned}
\tau_1 = {} & [m_1l_{c1}^2 + m_2l_1^2 + m_2l_{c2}^2 + 2m_2l_1l_{c2}\cos(q_2) + I_1 + I_2]\ddot{q}_1 \\
& +[m_2l_{c2}^2 + 2m_2l_1l_{c2}\cos(q_2) + I_2]\ddot{q}_2 - 2m_2l_1l_{c2}\sin(q_2)\dot{q}_1\dot{q}_2 \\
& -m_2l_1l_{c2}\sin(q_2)\dot{q}_2^2 + [m_1l_{c1} + m_2l_1]g\sin(q_1) + m_2gl_{c2}\sin(q_1 + q_2) \\
\tau_2 = {} & [m_2l_{c2}^2 + 2m_2l_1l_{c2}\cos(q_2) + I_2]\ddot{q}_2 + [m_2l_{c2}^2 + I_2]\ddot{q}_2 \\
& +m_2l_1l_{c2}\sin(q_2)\dot{q}_1^2 + m_2gl_{c2}\sin(q_1 + q_2)
\end{aligned} \tag{9}$$

The ideal pendubot of Figure 2 has 2 degrees of freedom ($n = 2$), q_i is the angular position, $\dot{q}_i$ is the angular velocity, $\ddot{q}_i$ is the angular acceleration for each

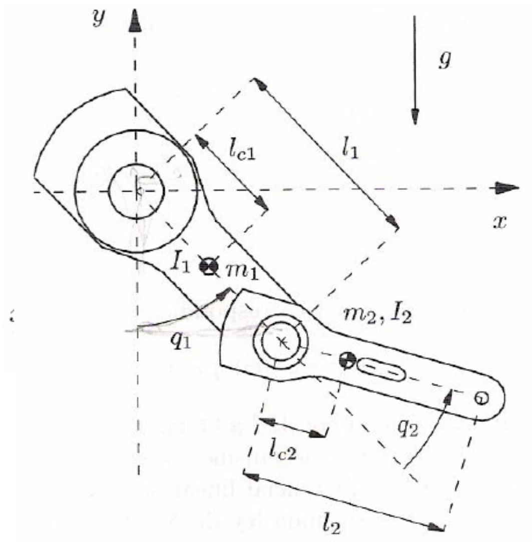


Fig. 2. Pendubot

degree of freedom respectively, the parameter values were taken as $m_1 = 5$Kg, $m_2 = 3$Kg, $l_1 = 1$m, $l_2 = 1$m, $I_1 = 0.004$Kg/m^2, $I_2 = 0.006$Kg/m^2, $l_{c1} = 0.5$m y $l_{c2} = 0.5$m, and the equations of inverse model (9) are the components of Compact Form (2).

Case 3: 3D Cartesian Manipulator.

The inverse dynamic model of a 3D cartesian manipulator shown in Figure 3 is:

$$\begin{aligned} \tau_1 &= [m_1 + m_2 + m_3]\ddot{q}_1 + [m_1 + m_2 + m_3]g \\ \tau_2 &= [m_1 + m_2]\ddot{q}_2 \\ \tau_3 &= m_1\ddot{q}_3 \end{aligned} \tag{10}$$

The 3D cartesian manipulator of Figure 3 has 3 degrees of freedom ($n = 3$), q_i is the articular position, $\dot{q}_i$ is the articular velocity, $\ddot{q}_i$ is the articular acceleration for each degree of freedom respectively, the parameter values were taken as $m_1 = 1.5$Kg, $m_2 = 1$Kg, $m_3 = 3$Kg, and the equations of inverse model (10) are the components of Compact Form (2).

4 Simulation Results

In this section, simulation results from using the PSO algorithm to estimate the torque or force of the 3 cases presented in the section above are illustrated.

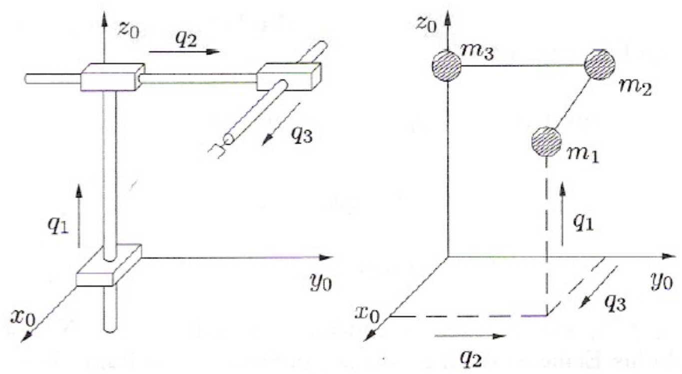


Fig. 3. 3D Cartesian manipulator

The computational experiments with the PSO algorithm for the 3 cases was done in Matlab©. The behaviour parameters of the programed PSO are the inertia weight $w = 0.5$, the cognitive learning rate $c_1 = 0.35$ and the social learning rate $c_2 = 0.35$.

Case 1: Simple pendulum.

The PSO algorithm was run with a population of 200 particles and 100 cycles. In Table 1 the torques τ corresponding to position q and acceleration $\ddot{q}$ values of inverse dynamic model (8) are shown. In Table 2 the torque $\hat{\tau}$, position $\hat{q}$ and acceleration $\ddot{\hat{q}}$ values computed from PSO algorithm minimizing the fitness (7) are shown.

Table 1. Position, acceleration and torque values from model (8)

q	$\ddot{q}$	τ
30	.15	15.165
60	.15	25.9371
90	.15	29.88
120	.15	25.9371
150	.15	15.165
180	.15	0.45
210	.15	-14.265
240	.15	-25.0371
270	.15	-28.98
300	.15	-25.0371
330	.15	-14.265
360	.15	0.45

Table 2. Position, acceleration and torque estimated values from PSO

$\hat{q}$	$\ddot{\hat{q}}$	$\hat{\tau}$
24.4735	1	15.192
128.7704	1	25.9454
66.0566	1	29.8975
64.953	-0.2441	25.9302
149	0	15.1576
185	1	0.435
209	0	-14.2679
252.2604	1	-25.0306
256	0	-28.5558
237.0218	-0.1149	-25.0377
209	0	-14.2679
185	1	0.435

In Figure 4, the torque values τ and their estimated $\hat{\tau}$ are shown for angular positions q from $0°$ to $360°$. The blue line represents the actual torque τ for the different positions of the pendulum with the known position q, velocity $\dot{q}$ and acceleration $\ddot{q}$. The red line is the torque estimated $\hat{\tau}$ with the estimated position $\hat{q}$, velocity $\dot{\hat{q}}$ and acceleration $\ddot{\hat{q}}$. It can be seen that 90 and 270 degrees are values such that more torque is required because the pendulum is in horizontal position.

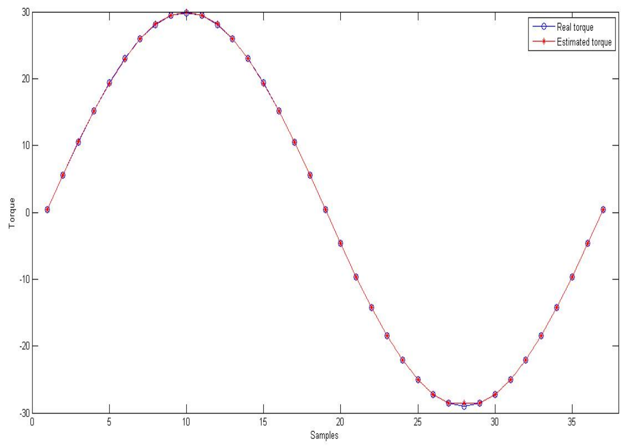


Fig. 4. Torque comparison of pendulum

Case 2: Pendubot.

The PSO algorithm was run with a population of 600 particles and 100 cycles. In Table 3 the torques τ_1, τ_2 corresponding to position q_1, q_2, velocity $\dot{q}_1, \dot{q}_2$ and acceleration $\ddot{q}_1, \ddot{q}_2$ values of inverse dynamic model (9) are shown. In Table 4 the torque $\hat{\tau}_1, \hat{\tau}_2$, position $\hat{q}_1, \hat{q}_2$ and acceleration $\ddot{\hat{q}}_1, \ddot{\hat{q}}_2$ values computed from PSO algorithm minimizing the fitness (7) are shown.

In Figure 5, the torque values τ_1, τ_2 and their estimated $\hat{\tau}_1, \hat{\tau}_2$ are shown for angular positions q_1, q_2 from 0° to 180°. The blue line represents the actual torque τ_i for the different positions of the pendubot with the known position q_1, q_2, velocity $\dot{q}_1, \dot{q}_2$ and acceleration $\ddot{q}_1, \ddot{q}_2$. The red line is the torque estimated $\hat{\tau}_1, \hat{\tau}_2$ with the estimated position $\hat{q}_1, \hat{q}_2$, velocity $\dot{\hat{q}}_1, \dot{\hat{q}}_2$ and acceleration $\ddot{\hat{q}}_1, \ddot{\hat{q}}_2$. In the Figure 5 can be seen that desired torque and the torque estimated are almost identical. The joint 1 requires more torque because it is the joint where the weight of both links rests.

Table 3. Position, velocity, acceleration and torque values from model (9)

q_1	q_2	$\dot{q}_1$	$\dot{q}_2$	$\ddot{q}_1$	$\ddot{q}_2$	τ_1	τ_2
0	0	0.2	0.15	0	0	0	0
0	180	0.2	0.15	0	0	0	0
20	0	0.2	0.15	0	0	5.0328	5.0328
40	0	0.2	0.15	0	0	9.4586	9.4586
60	20	0.2	0.15	0	0	32.9028	14.512
80	40	0.2	0.15	0	0	47.3456	12.7821
100	60	0.2	0.15	0	0	51.6521	5.0848
120	80	0.2	0.15	0	0	47.9806	-4.9737
140	100	0.2	0.15	0	0	40.2699	-12.6845
160	120	0.2	0.15	0	0	32.1278	-14.4395
160	140	0.2	0.15	0	0	21.8585	-12.705
180	120	0.2	0.15	0	0	33.8757	-12.6916
180	140	0.2	0.15	0	0	25.1434	-9.4201
180	160	0.2	0.15	0	0	13.3785	-5.0123
180	180	0.2	0.15	0	0	0	0

Case 3: 3D Cartesian Manipulator

The PSO algorithm was run with a population of 800 particles and 100 cycles. In Table 5 the forces τ_1, τ_2, τ_3 corresponding to acceleration $\ddot{q}_1, \ddot{q}_2, \ddot{q}_3$ values of inverse dynamic model (10) are shown. In Table 6 the force $\hat{\tau}_1, \hat{\tau}_2, \hat{\tau}_3$ and acceleration $\ddot{\hat{q}}_1, \ddot{\hat{q}}_2, \ddot{\hat{q}}_3$ values computed from PSO algorithm minimizing the fitness (7) are shown.

In Figure 6, the force values τ_1, τ_2, τ_3 and their estimated $\hat{\tau}_1, \hat{\tau}_2, \hat{\tau}_3$ are shown for different articular acceleration values $\ddot{q}_1, \ddot{q}_2, \ddot{q}_3$. The blue line represents the actual force τ_i for the different known accelerations $\ddot{q}_1, \ddot{q}_2, \ddot{q}_3$ of the cartesian

Table 4. Position, velocity, acceleration and torque estimated values from PSO

$\hat{q}_1$	$\hat{q}_2$	$\dot{\hat{q}}_1$	$\dot{\hat{q}}_2$	$\ddot{\hat{q}}_1$	$\ddot{\hat{q}}_2$	$\hat{\tau}_1$	$\hat{\tau}_2$
0	0	1	0	0	0	-0.03	-0.01
0	0	1	0	0	0	-0.03	-0.01
16	0	1	0	-0.1	1	5.	4.59
320	180	0	0	0	0	9.45	9.45
310	160	0	0	0	0	32.28	13.82
76.95	46.3	1	1	-0.3	1	47.39	13.5
89	74	1	1	0	0	51.84	5.74
236	91	1	1	1	1	47.12	-5.02
194	82	1	1	1	0	39.76	-12.18
147	125	0	0	1	0	32.78	-14.81
238	33	0	0	1	0	22.19	-12.69
159	95.18	1	1	-0.3	1	34.04	-12.17
182	41.72	1	0	0	0	24.74	-9.59
55	154	1	1	0	1	13.95	-5.72
360	360	1	0	0	0	-0.03	-0.01

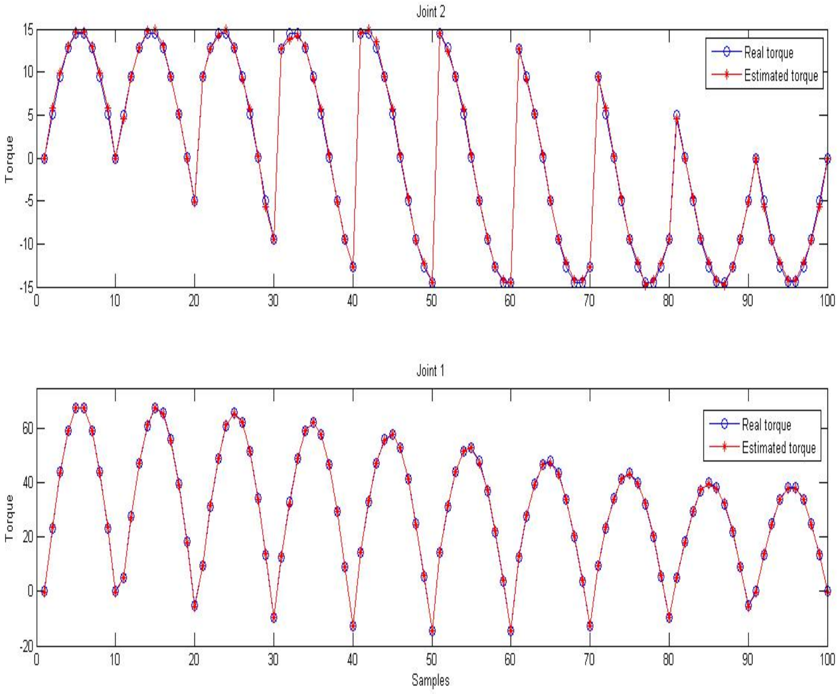


Fig. 5. Torque comparison of Pendubot

manipulator. The red line is the force estimated $\hat{\tau}_1, \hat{\tau}_2, \hat{\tau}_3$ with the estimated accelerations $\hat{\ddot{q}}_1, \hat{\ddot{q}}_2, \hat{\ddot{q}}_3$. It can be seen that desired force and the force estimated are almost identical. The joint 1 requires more force because it is the joint where supports the weight of entire robotic manipulator. The force in joint i-th only depends on the acceleration i-th to perform the motion.

Table 5. Acceleration and force values from model (10)

$\ddot{q}_1$	$\ddot{q}_2$	$\ddot{q}_3$	τ_1	τ_2	τ_3
0	10	20	0.5396	25	30
0	20	0	0.5396	50	0
0	20	10	0.5396	50	15
10	0	10	60.5395	0	15
10	0	20	60.5395	0	30
10	10	0	60.5395	25	0
20	0	20	120.5396	0	30
20	10	0	120.5396	25	0
20	10	10	120.5396	25	15
20	10	20	120.5396	25	30

Table 6. Acceleration and force estimated values from PSO

$\hat{\ddot{q}}_1$	$\hat{\ddot{q}}_2$	$\hat{\ddot{q}}_3$	$\hat{\tau}_1$	$\hat{\tau}_2$	$\hat{\tau}_3$
0	11	23	0.5396	27.5	33.5
0	20	1	0.5396	50	1.5
0	19	7	0.5396	47.5	13.5
10.3864	0	9.8226	62.8579	0	14.7338
10	0	21	60.5395	0	31.5
10	10	0	60.5395	25	0
20	0	18	120.5396	0	28
20	10	2	120.5396	25	2
20	10	11	120.5396	25	16
20	10	20	120.5396	25	30

5 Conclusions

Due to Forward Dynamic Model is difficult to obtain; a time-efficient, easily implemented, and flexible method based on Particle Swarm Optimization was applied to estimation of robot dynamics. As was shown through of simulations for a simple pendulum, pendubot and 3D cartesian manipulator, the estimation of dynamic variables (position, velocity, acceleration and torque) is very effective

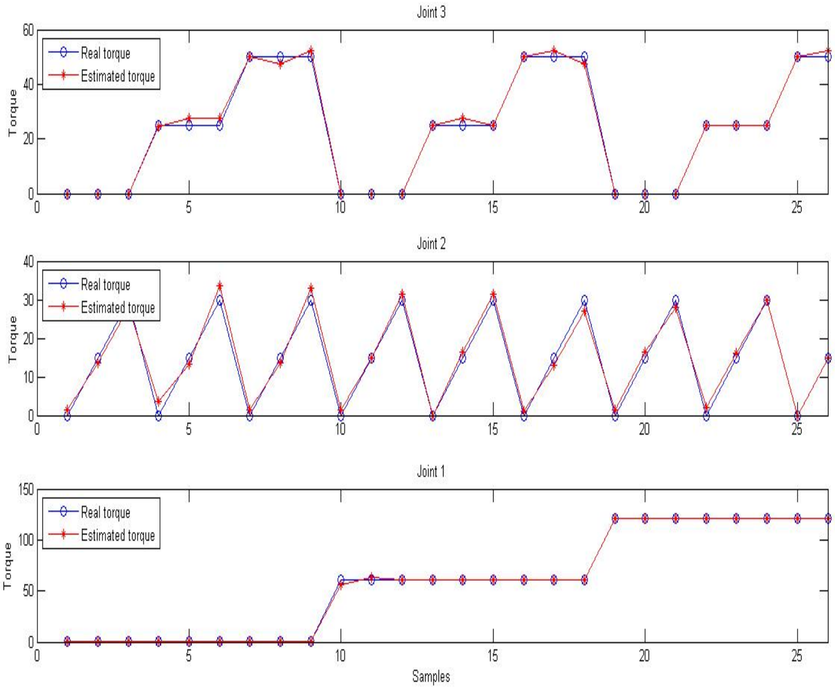


Fig. 6. Force comparison of 3D Cartesian manipulator

and precise. This method is easily executable for any user without no great robot dynamics knowledge for any manipulator with an arbitrary number of degrees of freedom. As future works are considered the parameter estimation (mass, length, Coulomb friction, viscous friction) when dynamic variables are known and position control of manipulator using PSO.

Acknowledgments Author Carlos Hervert thanks to Mexican CONACyT (Consejo Nacional de Ciencia y Tecnología) for scholarship grants. The work of Angel Rodríguez-Liñán was supported by PAICYT (Programa de Apoyo a la Investigación Científica y Tecnológica) de la UANL under grant IT562-10.

References

1. Barrientos, A., nin, F.P., Balaguer, C., Aracil, R.: Fundamentos de Robótica. Mc-Graw Hill
2. Cortéz, F.R.: Robótica Control de Robots Manipuladores. Alfaomega, México, D.F. (2012)

3. Estévez-Carreón, J., García-Ramírez, R.: Necessary conditions for the application of the shooting method to an optimal control of position in a manipulating robot (14), 72–80 (2006)
4. Grigoriadis, G., Mertzios, B., Tourassis, V.: Fast implementation of robot inverse dynamics with distributed arithmetic via a simd architecture. IEEE TRANSACTIONS ON SYSTEMS, MAN, AND CYBERNETICS 29(2), 209–216 (March 1999)
5. Jahandideh, H., Namvar, M.: Use of pso in parameter estimation of robot dynamics part one: No need for parameterization (2012)
6. Kelly, R., nes, V.S.: Control de movimiento de robots manipuladores. Automática Robótica, Prentice Hall, México, D.F.
7. Kennedy, J., Eberhart, R.: Particle swarm optimization. Proceedings of the IEEE International Conference on Neural Networks pp. 1942–1948 (1995)
8. Lu, L., Luo, Q., yong Liu, J., Long, C.: An improved particle swarm optimization algorithm. IEEE (2008)
9. Swevers, J., W. Verdonck, J.S.: Dynamic model identification for industrial robots. IEEE CONTROL SYSTEMS MAGAZINE (October 2007)
10. Weidong-Ji, Keqi-Wang: An improved particle swarm optimization algorithm. IEEE International Conference on Computer Science and Network Technology pp. 585–589 (2011)
11. Zhu, H., Pu, C., Eguchi, K., Gu, J.: Euclidean particle swarm optimization. 2nd International Conference on Intelligent Networks and Intelligent Systems pp. 669–672 (2009)

SAT Model for the Curriculum-Based Course Timetabling Problem

Nelson Rangel-Valdez[1], Jose Torres-Jimenez[2], Jorge Omar Jasso-Luna[1], Mario Humberto Rodriguez-Chavez[1]

[1] Universidad Politécnica de Victoria, Departamento de Tecnologías de la Información, Cd. Victoria, Tamaulipas, Mexico
[2] CINVESTAV Tamaulipas, Cd. Victoria, Tamaulipas, Mexico
nrangelv@upv.edu.mx, jtj@cinvestav.mx, ojassol@upv.edu.mx, mrodriguezc@upv.edu.mx

Abstract. Two widely used problems are the Satisfiability problem (SAT) and the Curriculum-Based Course Timetabling (CB-CTT) problem. The SAT problem searches for an assignment that make true a certain boolean formula. On the other side, the CB-CTT involves the task of scheduling lectures of courses to rooms, considering teacher availability, a specified curricula, and a set of constraints. Given the advances achieved in the solution of the SAT Problem, this research proposes a SAT Model of the CB-CTT problem, to aid in the construction of timetables. To demonstrate that the model can aid in the solution of real instances of the CB-CTT problem, a case of study derived from a university in Mexico was considered. This special case of CB-CTT involves the constraint where each teacher cannot teach more than one course in the same curriculum, which is included in the set of 3 hard constraints and 2 soft constraints analyzed in this research. According to the results obtained, the considered complete SAT solver required a few minutes to find a solution for the instance.

Keywords: Curriculum-based course timetabling problem, SAT model.

1 Introduction

The Curriculum-Based Course Timetabling (CB-CTT) is a problem that occurs at the beginning of each term in many universities. To solve this problem, different constraints must be taken into account, and they can vary depending of the considered particular case. Mainly, the constraints are associated with the availability of classrooms, or teachers, or with the number of classes to be assigned, etc. The general case of this problem is NP-Complete [1], which means that trying to find the optimal solution involves the consumption of great amount of computational resources. Despite of its complexity, it has been tackled with many approximated strategies over the years [2].

At the Polytechnic University of Victoria (PUV), the CB-CTT problem involves a complex combination of hard and soft constraints, that occurs at the beginning of each term. Currently, the problem is solved manually by the program directors, requiring several weeks to get an initial solution based on existing data.

This work presents a SAT model for the special case of the CB-CTT problem found at the PUV, which includes the constraint that no teacher can teach more than one course per curriculum. An instance derived from the problem was solved using a complete SAT solver, reported in the literature. It is important to point out that the approach reduces the time to construct the schedules from two weeks to a couple of minutes; it also satisfies the 3 hard constraints, and the 2 soft constraints considered for this paper.

The remaining of this article is organized in the following way: Section 2 formally defines the special case of the CB-CTT problem that, to the best of our knowledge, does not completely match its more general definition [3]. Section 3 presents the work related to the solution of similar scheduling problems. Section 4 shows the SAT model proposed for the solution of the special case of the CB-CTT problem. Section 5 shows the results of the experiments performed over an instance taken from the PUV. Finally, Section 6 presents the conclusions derived from the research.

2 Curriculum-Based Course Timetabling

The *Curriculum-Based Course Timetabling* (CB-CTT) problem found at the PUV represents a special case of the CB-CTT problem. In order to define it, the following basic elements are presented:

Time-slots. A day is split in a fixed number of non overlapping time slots, which are equal to all the days.

Courses, and Teachers. Each course consists of a fixed number of lectures (also denoted as classes) to be scheduled during the week. The course is taught by a teacher. Each teacher specifies an availability chart, i.e. a daily set of the time slots in which a lecture must be assigned to him/her.

Rooms, or Classrooms. There is a fixed number of classrooms in which the lectures must be scheduled.

Curricula, Curriculum, or Groups. A *curriculum* is a group of courses such that any pair of them cannot be schedule at the same time, because they share common students.

Idle Time-slots. One unassigned time slot between two assigned time slots in a curriculum daily schedule, or a teacher availability chart, is considered an Idle Time slot (or IT).

Figure 1 presents an simplified entity-relationship model describing the data in the PUV necessary to create an instance of the CB-CTT problem.

A feasible timetable for the PUV is one in which all lectures have been scheduled at a time slot and a classroom, so that the hard constraints $\{H_1, ..., H_3\}$ are satisfied. In addition, a feasible timetable has some desirable conditions, which are described by the two soft constraints $\{S_1, ..., S_2\}$ considered int he problem.

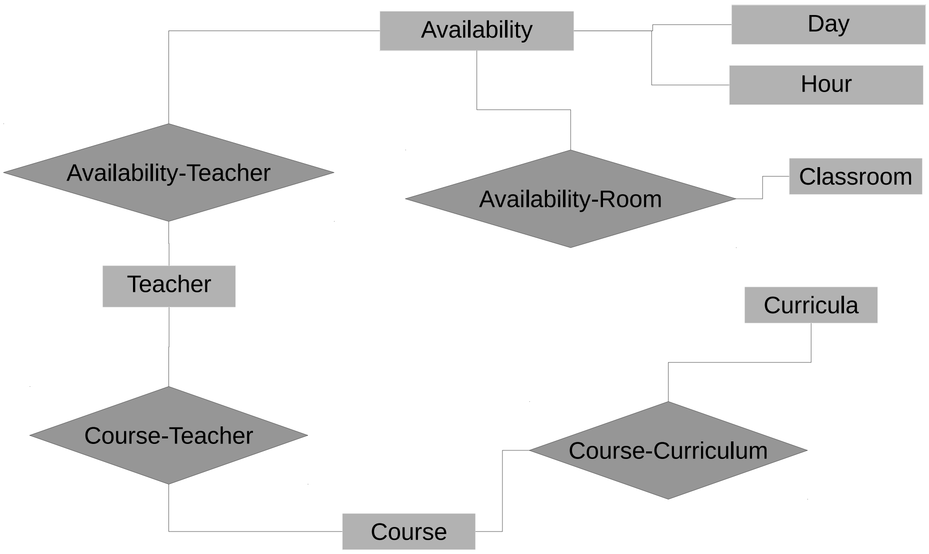


Fig. 1. Information required to derive a PUV-CBCTT instance

The three hard constraints (identified as H_i) and two soft constraints (identified as S_i) are:

- H_1. **Lectures:** All lectures of a course must be assigned to a particular time slot, in a classroom.
- H_2. **Conflicts:** Lectures of courses in the same curriculum, or taught by the same teacher, cannot be scheduled in the same time slot. Moreover, courses in the same curriculum cannot be scheduled with the same teacher. Additionally, two lectures cannot be assigned to the same classroom in a particular time slot.
- H_3. **Availability:** If the teacher of a course is not available at a given time slot, then no lectures of a course can be assigned to that time slot.
- S_1. **Minimum Curriculum IT:** Given that the curriculum is associated with a group of students, this constraint means that the agenda of students should be as compact as possible.
- S_2. **Minimum Teacher IT:** It is preferred that the agenda of a teacher should be as compact as possible.

According with the information previously presented, the specialized case of the CB-CTT problem mainly differs from others CB-CTT formulations, as the one presented in [4], in which: a) it is scheduled in a daily basis; b) it does not consider the room occupancy; and, c) it includes the additional constraint that a teacher cannot teach more than one course per curriculum. A formal definition for this problem, which will be named from now on as PUV-CBCTT problem, is presented in the next subsection.

2.1 Problem Formulation of PUV-CBCTT

Let's define $C = \{c_1, c_2, ..., c_n\}$ as the set of courses to be scheduled; $P = \{p_1, p_2, ..., p_u\}$ as the set of time slots in which a day is split; $T = \{t_1, t_2, ..., t_m\}$ as the set of teachers; $R = \{r_1, r_2, ..., r_w\}$ as the set of available classrooms; $CR = \{cr_1, cr_2, ..., cr_s\}$ as the set where each element cr_h, $1 \leq h \leq s$, is a curriculum; $TC = \{tc_1, tc_2, ..., tc_m\}$ as the set where each element tc_q, $1 \leq q \leq m$, contains the group of courses that teacher q might taught; and, $TA = \{ta_1, ta_2, ..., ta_m\}$ as the set where each element ta_q, $1 \leq q \leq m$, contains the time slots in which teacher q is available to teach a lecture. Then, the PUV-CBCTT problem can be defined as the problem of finding a timetabling array $\mathcal{M}$ of size $w \times u$, where each cell $m_{i,j}$, for $1 \leq i \leq w$ and $1 \leq j \leq u$, contains a tuple $(cl_k, t_q) \in C \times T$, subject to:

1. $\bigcup_{\forall i,j} \mathcal{C}(m_{i,j}) = C$;
2. $\mathcal{CR}(m_{i_1,j}) \bigcap \mathcal{CR}(m_{i_2,j}) = \emptyset$, for any $i_1 \neq i_2$ having fixed the time slot j;
3. $\mathcal{T}(m_{i_1,j}) \bigcap \mathcal{T}(m_{i_2,j}) = \emptyset$, for any $i_1 \neq i_2$ having fixed the time slot j;
4. $\mathcal{T}(m_{i_1,j_1}) \bigcap \mathcal{T}(m_{i_2,j_2}) = \emptyset$, for any different pairs $(i_1, j_1), (i_2, j_2)$, where $\mathcal{CR}(m_{i_1,j_1}) = \mathcal{CR}(m_{i_2,j_2})$;
5. $\mathcal{C}(m_{i,j}) \in tc_{\mathcal{T}(m_{i,j})}$, for any i, j;
6. $p_j \in ta_{\mathcal{T}(m_{i,j})}$, for any i, j.

where $cl_{k,l}$ stands for the lecture l of course c_k; $\mathcal{C} : m_{i,j} \rightarrow c_k$ is a function that obtains the course c_k assigned to time slot j in classroom i; $\mathcal{CR} : m_{i,j} \rightarrow cr_h$ is a function that obtains the curriculum cr_h associated with the course c_k derived from $\mathcal{C}(m_{i,j})$; and, $\mathcal{T} : m_{i,j} \rightarrow t_v$ is a function that finds the teacher t_q associated to the course c_k derived from $\mathcal{C}(m_{i,j})$. Finally, it is important to point out that the idle time must be minimized during the search for a solution. A summary of the previously described sets is presented in Table 1.

Table 1. Different sets required for the formal definition of the PUV-CBCTT

Symbol	Meaning
$C = \{c_1, c_2, ..., c_n\}$	A set of courses to be scheduled.
$P = \{p_1, p_2, ..., p_u\}$	A set of time slots in which a day is split.
$T = \{t_1, t_2, ..., t_m\}$	A set of teachers.
$R = \{r_1, r_2, ..., r_w\}$	A set of available classrooms.
$CR = \{cr_1, cr_2, ..., cr_s\}$	A set where each element cr_h, $1 \leq h \leq s$, is a curriculum.
$TC = \{tc_1, tc_2, ..., tc_m\}$	A set where each element tc_q, $1 \leq q \leq m$, contains the group of courses that teacher q might taught.
$TA = \{ta_1, ta_2, ..., ta_m\}$	A set where each element ta_q, $1 \leq q \leq m$, contains the time slots in which teacher q is available to teach a lecture.

3 Related Work

Several authors, among them, Schaerf [5] and Werra [6], consider that the automatization of the Timetabling Problem (TTP) cannot be done completely. The reasons they give are two: a timetable is not easily shown in an automated system, and on the other hand, as the search space is enormous, human intervention may be useful to guide the search to directions that the system alone would not easily go. Due to these reasons, most systems allow human intervention to adjust the final solution, and are called interactive timetabling.

Werra [6] formally explains several TTP problems and presents their respective formulations. He also describes the most important research in which graph theory is applied. For the purpose of this research, the remaining of this section presents some of the most representative works of the CB-CTT problem, that has been developed in the recent years.

Lu and Hao [4] shows an approach based in Tabu Search that solves the more general case of the CB-CTT. In this problem a set of lectures must be assigned into a weekly timetable. The hard constraints considered are related with those taken into account in this research, with the only exception of the covering constraint, i.e. the problem studied does not involve the assignment of courses to teachers.

Almilli [7] reports the use of a hybrid strategy for the solution of the Educational Timetabling Problem. The approach combines Simulated Annealing and Genetic algorithms to solve a problem where the entities considered were courses, classrooms, students, and time slots. This problem differs from the case of the CB-CTT in which several constraints are not considered, e.g. the inner problem does not consider the assignment of teachers to courses and classrooms, nor the availability of teachers.

Pothitos et al. [8] describes the course timetabling problem in a similar way than Lu and Hao do in [4]. The strategy followed to solve such problem consists in mapping it to the domain of a Constraint Satisfaction Problem (CSP) instance. There, a CSP solver is used to build a solution. The considered problem in that work also lacks the assignment of courses to teachers in its definition.

Abdullah et al. [9] presents the solution of a similar CB-CTT problem like the one presented here. One of the slight differences is that it does not include the assignment of courses to teachers. The algorithm used was a combination of the Great-Deluge strategy and an electromagnetism-like mechanism.

Finally, the more recent approach presented by Rues-Maw and Hsiao [10] describes the use of a Particle Swarm Optimization approach for the solution of another variation of the course timetabling problem. This case of the CB-CTT includes tighter constraints, not considered for the special case at the PUV. In addition, the constraint involving teachers and the curricula is not taken into account in the work of Rues-Maw and Hsiao.

In summary, all the revised articles related with this research do not involve the constraint where a teacher cannot teach more than one course in the same curriculum. Because of that, the present document bases its study on the solution of that special case of the CB-CTT problem, which includes the commented constraint.

The next section shows the methodology followed to solve the special case of the CB-CTT problem at the PUV.

4 SAT Model for Solving the PUV-CBCTT Problem

The SAT model proposed for the solution of the PUV-CBCTT problem is formed by seven diferent sets of restrictions with the following purposes: 1) to guarantee that each class must be assigned to just one classroom; 2) to guarantee that each lecture must be assigned to just one hour; 3) to avoid overlaps in classrooms; 4) to avoid scheduling classes in hours were the teacher is not available; and, 5) to avoid that two different classes of the same teacher or group be scheduled in the same time slot.

In order to define the restrictions of the model the boolean variables depicted in Table 2 are defined.

Table 2. Boolean variables used in the SAT model of the CB-CTT problem

$$X_{i,j} \leftarrow \begin{cases} 1 \text{ if course } c_i \text{ is assigned to classroom } r_j \\ 0 \text{ otherwise} \end{cases}$$

$$Y_{i,j} \leftarrow \begin{cases} 1 \text{ if course } c_i \text{ is assigned to time slot } p_j \\ 0 \text{ otherwise} \end{cases}$$

4.1 Restrictions to Guarantee a Classroom per Class

The first set of restrictions consists in the assignment of classrooms to courses. For this purpose the sets defined in Equation 1 are proposed. There, while the first set guarantees that at least each course must have a classroom assigned, the second set assures that only one must be assigned.

$$\begin{gathered} \bigwedge_{i=1}^{n} \left(\bigvee_{j=1}^{w} X_{i,j} \right) \\ \bigwedge_{i=1}^{n} \bigwedge_{j=1}^{w} \bigwedge_{k=j+1}^{w} \left(\overline{X_{i,j}} \vee \overline{X_{i,k}} \right) \end{gathered} \quad (1)$$

4.2 Restrictions to Guarantee a Lecture of a Classroom is assigned to just one hour

Once that a classroom is assigned, the following step to consider in the definition of the SAT Model is the assignment of time slots to each lecture of a course. The sets of constraints shown in Equation 2 are defined for this purpose. The first set assures that a lecture has at least one time slot assigned, while the second complements the restriction to ensure that it has just one.

$$\begin{gathered}\bigwedge_{i=1}^{n}\left(\bigvee_{j=1}^{u} Y_{i,j}\right)\\ \bigwedge_{i=1}^{n}\bigwedge_{j=1}^{u}\bigwedge_{k=j+1}^{u}\left(\overline{Y_{i,j}} \vee \overline{Y_{i,k}}\right)\end{gathered} \tag{2}$$

4.3 Restrictions to Avoid Overlaps in Classrooms

In order to avoid that two different classes are taught in the same classroom the set of restrictions shown in Equation 3 is considered. This set forms a single clause for each different combination of two classes sharing the same classroom and time slot, such that if both of them are true, then the clause will be false, satisfying the required assignment.

$$\bigwedge_{i=1}^{w}\bigwedge_{j=1}^{u}\bigwedge_{m_1=1}^{n}\bigwedge_{m_2=m_1+1}^{n}\left(\overline{X_{m_1,i}} \vee \overline{Y_{m_1,j}} \vee \overline{X_{m_2,i}} \vee \overline{Y_{m_2,j}}\right) \tag{3}$$

4.4 Restrictions to Assure Teacher Availability

The present restriction must assure that the courses are taught in time slots where the teacher is available. For this purpose a set of unit clauses is formed per teacher, through the set of boolean variables $Y_{i,j}$. This set will contain a clause with a single literal $\overline{Y_{i,j}}$ for each course c_i and time slot p_j in which the professor that teaches that class is not available.

4.5 Restrictions to Avoid Overlaps in Teacher and Curricula Time slots

This subsection presents the set of restrictions shown in Equation 4. The purpose of them is the assignment of classes to classrooms, considering the constraint where a teacher cannot be assigned twice to the same curricula. The function $f : c_i \rightarrow t_j$ obtains the teacher t_j that teach the course c_i, and the function $g : c_i \rightarrow cr_j$ obtains the curricula cr_j to which the course c_i belongs to. The two-literal clauses take into account each possible combination of undesired situations for these restrictions, such that it will turn false the formula if one of them occurs.

$$\begin{gathered}\bigwedge_{\forall(m_1,m_2)|f(m_1)=f(m_2)}\bigwedge_{i=1}^{u}\left(\overline{Y_{m_1,i}} \vee \overline{Y_{m_2,i}}\right)\\ \bigwedge_{\forall(m_1,m_2)|g(m_1)=g(m_2)}\bigwedge_{i=1}^{u}\left(\overline{Y_{m_1,i}} \vee \overline{Y_{m_2,i}}\right)\end{gathered} \tag{4}$$

4.6 Analysis of the Model

This subsection summarizes the complexity of the SAT model presented in this section in Table 3. There, it is shown the approximated number of clauses and literals per clause required to transform an instance of the PUV-CBCTT to a SAT formula. The values n, w, u, correspond to the number of courses, rooms, and time slots in the PUV-CBCTT instance.

Table 3. Characterization of the clauses in the SAT formula resulting from the transformation of an instance of PUV-CBCTT, following the proposed SAT model

Restriction	No. of Clauses	Literals per Clause
1	n	w
	$n \cdot w \cdot w$	2
2	n	u
	$n \cdot u \cdot u$	2
3	$w \cdot u \cdot n \cdot n$	4
4	$n \cdot u$	1
5	$n \cdot n \cdot u$	2

Note that the overall formula require two sets of boolean variables, each having $n \cdot w$, and $n \cdot u$ clauses, respectively. The number of literals per clause are mainly less than or equal to 4, and just a few of sizes n and u. In general, it is possible to comment that the formula produced by the SAT model proposed is simple, and only requires a number of clauses proportional to $O(w \cdot u \cdot n \cdot n)$, and a number of boolean variables to $O(n \cdot w)$.

5 Experimental Results

In order to demonstrate the viability of the approach to solve the PUV-CBCTT problem, the instance of it described in Table 4 was considered. This instance was modeled using the SAT model presented in this paper, and solver using The complete SAT solver used for this problem was the boolean satisfaction and optimization library in Java, SAT4J[1].

A summary of the results derived from the experimental design are shown in Table 5; in this table, the symbol $-$ means that the factor F was not necessary in the evaluation function. The time given is measured in milliseconds. The configurations shown in bold are the best for each evaluation function. Note that the SAT solver could find a solution for this instance in a few seconds, which indicates that the model can be used to solve more complex cases of this problem.

[1] http://www.sat4j.org/

Table 4. Instance of the PUV-CBCTT problem, taken from a real case derived from a Mexican University

(a) General Information

Information	Amount	Description
No. of time slots	12	A day has 12 non overlapping time slots
No. of courses	70	The courses to be scheduled during the day
No. of teachers	27	Each teacher is available in any time slot.
No. of classes per teacher	3	Each teacher can taught at most three classes.
No. of classrooms	17	Maximum number of classrooms available
No. of groups	14	The curricula that describes the instance

(b) Curricula Description

No.	Curriculum	Course	Lectures	Teacher	No.	Curriculum	Course	Lectures	Teacher
1	1	787	1	46	36	8	173	1	67
2	1	788	1	46	37	8	174	1	59,65
3	1	789	1	46	38	8	175	1	58
4	1	790	1	2,28	39	8	214	1	36,64
5	1	791	1	48,49,56	40	8	215	1	2
6	2	790	1	2,28	41	9	173	1	67
7	2	791	1	48,49,56	42	9	174	1	59,65
8	3	790	1	2,28	43	9	175	1	58
9	3	791	1	48,49,56	44	9	214	1	36,64
10	4	161	1	45,48	45	10	181	1	50
11	4	162	1	43	46	10	183	1	61
12	4	163	1	44,47	47	10	184	1	3
13	4	164	1	45	48	10	185	1	53
14	4	205	1	36	49	10	186	1	50
15	4	206	1	75	50	10	222	1	24
16	4	207	1	3,73	51	10	223	1	29
17	5	161	1	45,48	52	11	187	1	63
18	5	162	1	43	53	11	188	1	60
19	5	163	1	44,47	54	11	189	1	47,60
20	5	164	1	45	55	11	190	1	50,67
21	5	205	1	36	56	11	191	1	61
22	5	206	1	75	57	11	224	1	24
23	5	207	1	3,73	58	12	187	1	63
24	6	170	1	48,73	59	12	188	1	60
25	6	171	1	47,58	60	12	189	1	47,60
26	6	172	1	44,56	61	12	190	1	50,67
27	6	210	1	64	62	12	191	1	61
28	6	211	1	28	63	12	224	1	24
29	6	212	1	49	64	13	199	1	53,72
30	7	170	1	48,73	65	13	200	1	43,44
31	7	171	1	47,58	66	13	201	1	72
32	7	172	1	44,56	67	13	203	1	63
33	7	210	1	64	68	14	199	1	53,72
34	7	211	1	28	69	14	200	1	43,44
35	7	212	1	49	70	14	201	1	72

Table 5. Summary of the performance of SA using each of the different configurations considered

Number of Variables	3,594
Number of Clauses	841,550
Time to find a solution	3.726 seconds
Satisfiable	YES

6 Conclusions

The research presented in this document involves the solution of a special case of the Curriculum-Based Course Timetabling (CB-CTT), which can be found in the Polytechnic University of Victoria (PUV), in Mexico, and has not been completely addressed in the literature.

The special case of the CB-CTT problem involved the inclusion of the constraint of not assigning a teacher to more than one course per curriculum, and the solution of the task that assigns courses to teachers. In this document it is shown a formal definition for the problem.

A SAT model for the CB-CTT problem is presented, and a real world instance is used to test its validity. The instance was solved using a complete SAT solver, which could find a solution in a few seconds satisfying all the considered restrictions, the 3 hard constraints and the 2 soft constraints.

Acknowledgements. Authors want to thank the support received from projects: 58554 Cálculo de Covering Arrays, 51623 CONACyT - Gobierno del estado de Tamaulipas; and, the PROMEP projects UPV-PTC-023, titled *Análisis de la Dificultad del Problema de Minimización del Ancho de Banda en Grafos por Medio de Parejas de Invariantes*, and UPV-CA-4, entitled *Estudio del Impacto de la Experiencia de Usuario en los riesgos de integración durante el desarrollo de Sistemas Generadores de Horarios.*

References

1. Garey, M., Johnson, D.: Computers and Intractability: A Guide to the Theory of NP Completeness. Freeman: San Francisco. (1979) ISBN: 0-7167-1045-5.
2. Pillay, N.: A survey of school timetabling research. Annals of Operations Research (2013) 1–33
3. Bonutti, A., Cesco, F., Gaspero, L., Schaerf, A.: Benchmarking curriculum-based course timetabling: formulations, data formats, instances, validation, visualization, and results. Annals of Operations Research **194**(1) (2012) 59–70
4. Lü, Z., Hao, J.K.: Adaptive tabu search for course timetabling. European Journal of Operational Research **200**(1) (2010) 235 – 244
5. Schaerf, A.: A survey of automated timetabling. Technical report, Centrum voor Wiskunde en Informatica (CWI), Amsterdam, The Netherlands (1995)
6. Werra, D.D.: An introduction to timetabling. . European Journal of Operational Research **19** (1985) 151–162

7. Al-Milli, N.: Hybrid genetic algorithms with simulating annealing for university course timetabling problems. Journal of Theoretical & Applied Information Technology **29**(2) (2011) 100 – 106
8. Pothitos, N., Stamatopoulos, P., Zervoudakis, K.: Course scheduling in an adjustable constraint propagation schema. In: Tools with Artificial Intelligence (ICTAI), 2012 IEEE 24th International Conference on. Volume 1. (2012) 335–343
9. Abdullah, S., Turabieh, H., McCollum, B., McMullan, P.: A hybrid metaheuristic approach to the university course timetabling problem. Journal of Heuristics **18**(1) (2012) 1–23
10. Ruey-Maw, C., S., H.F.: Solving university course timetabling problems using constriction particle swarm optimization with local search. Algorithms **6** (2013) 227–244

Preferred Extensions as Minimal Models of Clark's Completion Semantics

Mauricio Osorio, Alejandro Santoyo

Universidad de las Américas - Puebla,
Depto. de Actuaría, Física y Matemáticas, Mexico
osoriomauri@googlemail.com, jasrvro@hotmail.com

Abstract. Dung established the connections between several logic programming semantics and various argumentation framework semantics. In this paper we present a characterization of the preferred semantics of argumentation frameworks (which is defined in terms of a maximal admissible set *w.r.t.* set inclusion) in terms of minimal logic models of Clark's completion. Additionally, we make use of integer programming for computing preferred extensions by a method defined by Bell *et al.* [3], which translates a logic program into an integer program which in turn can be solved by an *ad hoc* solver.

Keywords: Argumentation frameworks, preferred extensions, Clark's completion semantics, answer set programming, integer programming.

1 Introduction

Argumentation theory has been a research area in several disciplines such as logic, psychology, philosophy, linguistics, and legal theory. However, even though argumentation theory has a long history, it was just until the arrival of Dung's seminal paper on abstract argumentation theory [7], when this field attracted the attention of many researchers.

Particularly, argumentation theory has become an increasingly important and exciting research topic in Artificial Intelligence (AI). The main purpose of argumentation theory is to study the fundamental mechanism humans use in argumentation and to explore ways to implement this mechanism on computers.

Currently formal argumentation research has been strongly influenced by Dung's abstract argumentation theory [7]. This approach is mainly oriented to manage the interaction of arguments by introducing a single structure called *Argumentation Framework (AF)*. An argumentation framework basically is a tuple of sets: a set of arguments and a set of disagreements between arguments called attacks. Indeed an argumentation framework can be regarded as a directed graph in which the arguments are represented by nodes and the attack relations are represented by arrows.

In [7], four argumentation semantics were introduced: *stable semantics*, *preferred semantics*, *grounded semantics*, and *complete semantics*. The central notion of Dung's semantics is the *acceptability of the arguments*. Even though each of these argumentation semantics represents different patterns of selection of arguments, all of them are based on the basic concept of *admissible set*. Informally speaking, an admissible set presents a coherent and defendable point of view in a conflict between arguments.

Dung showed that argumentation can be viewed as logic programming with *negation as failure*. In this setting, he showed that the sets of arguments which can be considered as admissible can be regarded as *logic models* of a given logic program. This result is of great importance because it introduces a general method for generating metainterpreters for argumentation systems and regards argumentation semantics from another point of view in order to identify non-monotonic reasoning features of them. Following this issue, the preferred semantics was characterized by the p-stable semantics [12] in [5]. Moreover, the preferred semantics was characterized by the stable model semantics in [11].

In this work we introduce new results which complete the understanding of Dung's semantics in terms of logic programming semantics with negation as failure. By considering an argumentation framework AF and a uniform mapping of AF into a logic program Π_{AF}, we show that the Clarks's completion minimal models of Π_{AF} characterize the preferred extensions of AF, and also use integer programming fro computing such models.

It is worth mentioning that in section 4 we present some preliminary results of an ongoing research related to find out if integer programming can be used for computing any argumentation framework extension. However, even though we present encouraging results, we still have to be cautious about the conclusions it is possible to draw from them, that is way the section was called "Proof of Concept".

The rest of the paper is divided as follows: In Section 2, we present a basic background about logic programming, Clark's completion semantics, how to compute the minimal models of Clark's completion using mixed integer programming, argumentation theory, and how to map an argumentation framework into a logic program. In Section 3, we present our study about the relationship between minimal models of Clark's completion and preferred extensions. In Section 4 we present a proof of concept experiment for computing preferred extensions using integer programming as part of an ongoing research. In the last section, we outline our conclusions and future work.

2 Background

In this section, we first define the syntax of a valid program, after that the Clark's completion semantics [6] is presented, then the method define by Bell *et al.* for computing the Clark's completion minimal models, then we present some basic concepts of argumentation theory, and finally how to map an argumentation framework to a logic program.

2.1 Clark's Completion

The Clark's Completion of a given logic program is an old concept that has been intensively explored in logic programming literature in order to identify basic properties of logic programming semantics with negation as failure [1, 6]. It is defined as follows: Given a normal logic program P, its completion $Comp(P)$ is obtained in two steps:

1. Each normal clause $a_0 \leftarrow a_1, \ldots, a_j, not\ a_{j+1}, \ldots, not\ a_n \in P$ is replaced with the formula: $a_0 \leftarrow a_1 \wedge \ldots \wedge a_j \wedge \sim a_{j+1} \wedge \ldots \wedge \sim a_n$.
2. For each symbol $a \in \mathcal{L}_P$, let $Support(a)$ denotes the set of all formulae with a in the head. Suppose $Support(a)$ is the set: $\{a \leftarrow Body_1, \ldots, a \leftarrow Body_m\}$, in which each $Body_i (1 \leq i \leq m)$ is of the form $a_1 \wedge \ldots \wedge a_j \wedge \sim a_{j+1} \wedge \ldots \wedge \sim a_n$. Replace $Support(a)$ with the single formula: $a \leftrightarrow Body_1 \vee \ldots \vee Body_m$.
   If $Support(a) = \emptyset$ then replace it by $\sim a$.

2.2 Translating a Logic Program into an Integer Program

In this section we will show a method defined in [3] to translate $Comp(P)$ into a mixed integer program which then is used to compute the minimal models of $Comp(P)$. Thus it is required some definitions.

Definition 1. *[3]*

1. *A variable X is called binary variable if it can only take on a value of either 0 or 1.*
2. *For all $A \in B_{\mathcal{L}}$, let X_A be a binary variable corresponding to A. The set $\{X_A | A \in B_{\mathcal{L}}\}$ is called a binary variable representation of $B_{\mathcal{L}}$.*

If L is a ground literal, we use X_L as short-hand for the binary variable X_A if L is the positive ground atom A, and for the expression $(1 - X_A)$ if L is the negative atom *not* A.

Definition 2. *[3]*

1. *A binary variable assignment is a mapping $S : \{X_A | A \in B_{\mathcal{L}}\} \rightarrow \{0, 1\}$.*
2. *Let I be an interpretation. Define the binary variable assignment S_I corresponding to I as follows:*

$$for\ all\ A \in B_{\mathcal{L}}, S_I(X_A) = \begin{cases} 1 & if\ A \in I; \\ 0 & otherwise. \end{cases}$$

Definition 3. *[3] Suppose we consider the ground clause C below:*

$$A \leftarrow B_1, \cdots, B_n, not\ D_1, \cdots, not\ D_m.$$

We use $if(C)$ to denote the linear constraint:

$$X_A \geq 1 - (\sum_{i=1}^{n}(1 - X_{B_i})) - (\sum_{j=1}^{m} X_{D_j}).$$

Given a logic program P, we use the notation $if(P)$ to denote the set $\{if(C) | C \in grd(P)\}$.

Definition 4. *[3] Let P be a normal logic program and C be a formula in $Comp(grd(P))$.*

1. *If C is of the form: $\neg A$, then the constraint version of C, denoted by $lc(C)$, is: $X_A = 0$.*
2. *if C is of the form: $A \leftrightarrow E_1 \vee \cdots \vee E_k$ where $E_i \equiv L_{i,1} \;\&\; \cdots \;\&\; L_{i,m_i}$ for all $1 \leq i \leq k$, and none of the E_i's is "true", then the constraint vesion $lc(C)$ of C is given by the set of constraints $\{if(A \leftarrow E_i) | 1 \leq i \leq k\}$ (called "if-constraints") together with the additional set of constraints (called "only-if" constraints):*
$$X_A \leq Y_1 + \cdots + Y_k$$
where, for all $1 \leq i \leq k$, Y_i is a binary variable defined by the following constraints:
$$Y_i \leq X_{A_{i,1}}$$
$$\cdots$$
$$Y_i \leq X_{A_{i,m_i}}$$
$$Y_i \geq 1 - \sum_{j=1}^{m_i} (1 - X_{A_{i,j}})$$
where $E_i = L_{i,1} \;\&\; \cdots \;\&\; L_{i,m_i}$.
3. *For the special case when C is of the same form as above and some E_i, $1 \leq i \leq k$ is empty, $lc(C)$ is simply: $X_A = 1$.*

The solutions to the above set of constraints with binary variables, corresponds to the Herbrand models of $Comp(grd(P))$. The process we just followed leads us to Definition 5 and Theorem 1.

Definition 5. *[3] Let P be a logic program. The constraint version of P, denoted by $lc(P)$, is the set of constraints versions of all formulas in $Comp(grd(P))$.*

Theorem 1. *[3] Let P be a logic program, let I be an interpretation, and let S_I be the binary variable assignment as defined in Definition 2. Then, S_I is a solution of $lc(P)$ iff I is a model of $Comp(grd(P))$.*

Theorem 1 says that using $lc(P)$ to compute the Herbrand models of $Comp(P)$ (the models that make that all the clauses in $Comp(P)$ hold) is sound and complete, however, Comp(P) might not be consistent, therefore we need the following corollary:

Corollary 1. *[3] Let P be a logic program. Then $Comp(P)$ has an Herbrand model iff lc(P) has a solution.*

Theorem 2. *[3] Let P be a logic program, let M be a model of $Comp(grd(P))$, and let S_M be a binary variable assignment corresponding to M as defined in Definition 2. Then, S_M is an optimal solution of $lc(P)$ minimizing $\sum_{A \in B_{\mathcal{L}}} X_A$, iff M is a card-minimal model of $Comp(grd(P))$.*

Corollary 2. *[3] Let P be a logic program, M be an herbrand interpretation, and S_M be the binary variable assignment corresponding to M as defined in Definition 2. If S_M is an optimal solution of $lc(P)$ that minimizes $\sum_{A \in B_\mathcal{L}} X_A$, then M is a minimal (w.r.t. set inclusion) Herbrand model of $Comp(P)$.*

2.3 Argumentation Theory

Now, we define some basic concepts of Dung's argumentation approach. The first one is an argumentation framework.

Definition 6. *[7] An argumentation framework is a pair $AF := \langle AR, attacks \rangle$, where* AR *is a finite set of arguments, and* attacks *is a binary relation on* AR, i.e. attacks $\subseteq AR \times AR$.

Any argumentation framework can be regarded as a directed graph. For instance, if $AF := \langle \{a, b\}, \{(a, b), (b, a)\} \rangle$, then AF is represented as it is shown in Figure 1. We say that *a attacks b* (or b is attacked by a) if $attacks(a, b)$ holds.

Let us observe that an argumentation framework is a simple structure which captures the conflicts of a given set of arguments. In order to select *coherent points of views* from a set of conflicts of arguments, Dung introduced a set of *patterns of selection of arguments*. These patterns of selection of arguments were called *argumentation semantics*. Dung defined his argumentation semantics based on the basic concept of *admissible set*:

Definition 7. *[7] A set S of arguments is said to be conflict-free if there are no arguments a, b in S such that a attacks b. An argument $a \in AR$ is acceptable with respect to a set S of arguments if and only if for each argument $b \in AR$: If b attacks a then b is attacked by S. A conflict-free set of arguments S is admissible if and only if each argument in S is acceptable w.r.t. S.*

By considering the concept of admissible set, in [7], Dung introduced four basic argumentation semantics.

Definition 8. *[2] Let $AF := \langle AR, attacks \rangle$ be an argumentation framework and $S \in AR$. We introduce a function $F : 2^{AR} \rightarrow 2^{AR}$ such that $F(S) = \{A \mid A\, is\, defended\, by\, S\}$.*

Definition 9. *[2] Let $AF := \langle AR, attacks \rangle$ be an argumentation framework and S be a conflictfree set of argument. S is said to be a complete extension iff S = F(S).*

Definition 10. *[2]* [1] *Let $AF := \langle AR, attacks \rangle$ be an argumentation framework. An admissible set of argument $S \subseteq AR$ is* preferred *if and only if S is a maximal (w.r.t. inclusion) complete extension of AF.*

[1] This definition differs from the Dung's original, and in fact it is a characterization that it was proved to be equivalent

Since the first argumentation semantics were introduced in [7], Dung's argumentation semantics have given place to different formal studies about the properties of them. One of these formal studies has been to regard them as formal non-monotonic reasoning. In this setting, one can find that the argumentation semantics are closely related to logic programming semantics with *negation as failure*. In the following sections, we will study the preferred extension and a particular relationship with logic programming semantics with negation as failure.

2.4 Mapping from Argumentation Frameworks to Logic Programs

The first step for studying the structure of an argumentation framework as a logic program is to get the logic program that represents such a structure. To this end, we will present the method defined in [13] to map from an argumentation framework into a logic program. Let us observe that this mapping basically is *a declarative representation* of an argumentation framework by having in mind the ideas of *conflict-freeness* and *reinstatement* which are the basic concepts behind the definition of admissible sets.

In this mapping, the predicate $def(x)$ is used, the intended meaning of $def(x)$ is "x is a defeated argument" which means that x cannot be part of an admissible set. A transformation function *w.r.t.* an argument is defined as follows.

Definition 11. *Let $AF := \langle AR, Attacks\rangle$ be an argumentation framework and $a \in AR$. We define the transformation function $\Pi(a)$ as follows:*

$$\Pi(a) = \bigcup_{b:(b,a)\in Attacks} \{def(a) \leftarrow \ not\ def(b)\} \cup$$

$$\bigcup_{b:(b,a)\in Attacks} \{def(a) \leftarrow \bigwedge_{c:(c,b)\in Attacks} def(c)\}$$

The transformation function Π with respect to an argumentation framework AF is defined as follows:

Definition 12. *Let $AF := \langle AR, attacks\rangle$ be an argumentation framework. We define its associated normal program as follows:*

$$\Pi_{AF} := \bigcup_{a\in AR} \{\Pi(a)\}.$$

As one can see in Π_{AF}, the language of Π_{AF} only identifies the arguments which can be considered as defeated. By considering *total interpretations*, as the ones suggested by logic programming semantics as stable model semantics [10], we can assume that any argument which is not defeated in a model of Π_{AF} will be acceptable. This means that given an argumentation framework $AF = \langle AR, Attacks\rangle$ if M is a model of Π_{AF}, then any atom $def(x)$ which is false in M will identify an argument x which is acceptable. This assumption suggests a normal clause of the following form:

$$acc(x) \leftarrow not\ def(x).$$

where $acc(x)$ denotes that the argument x can be considered as accepted. This clause essentially fixes as acceptable any argument which is not fixed as defeated in Π_{AF}.

3 Preferred Extensions as Logic Programming Semantics

This section presents the characterization of the preferred extension in terms of the minimal models of Clark's completions, which allows us to propose that integer programming could be used for computing the preferred extension of an argumentation framework, and even other argumentation framework semantics such as the complete extensions.

Please note that while argumentation frameworks extensions are defined in terms of accepted arguments, this work was developed in terms of the defeated ones.

3.1 Preferred Extensions as Minimal Models of Clark's Completion

The first step is to map a given argumentation framework AF into a logic program Π_{AF}, and then characterize it as complete extensions in terms of *logic models* introduced in [4], to this end consider the following proposition:

Proposition 1. *[4] Let $\langle AR, attacks\rangle$ be an argument system. A set $S \subseteq AR$ is a complete extension iff S is a model of the formula*

$$\bigwedge_{a \in AR} ((a \rightarrow \bigwedge_{b:(b,a) \in attacks} \neg b) \wedge (a \leftrightarrow \bigwedge_{b:(b,a) \in attacks} (\bigvee_{c:(c,b) \in attacks} c))).$$

Using this proposition and the following definition:

Definition 13. *[13] Let $AF := \langle AR, attacks\rangle$ be an argumentation framework. Let $A \subseteq AR$, then $m(A) = \{def(x) \mid x \in AR \setminus A\}$.*

It was proved the correspondence between complete extensions and models of $Comp(\Pi_{AF})$. This relationship was formalized by the following Theorem:

Theorem 3. *[13] Let $AF := \langle AR, attacks\rangle$ be an argumentation framework. Let $A \subseteq AR, A$ is a complete extension of AF, iff $m(A)$ is a model of $comp(\Pi_{AF})$.*

Once more note that $m(A)$ is defined in terms of defeated arguments instead of the accepted ones, *i.e.* if $m(A)$ is a model of $comp(\Pi_{AF})$, then it is not directly a complete extension but its complement. Thus, we have the following corollary:

Corollary 3. *Let $AF := \langle AR, attacks\rangle$ be an argumentation framework. Let $A \subseteq AR, A$ is a maximal complete extension of AF, iff $m(A)$ is a minimal model of $comp(\Pi_{AF})$.*

Proof. Since $m(A)$ is defined in terms of defeated arguments instead of the accepted ones, then to a maximal complete extension corresponds a minimal model of $comp(\Pi_{AF})$.

Now, considering Corollary 3 we have the following theorem:

Theorem 4. *Let $AF := \langle AR, attacks\rangle$ be an argumentation framework. Let $A \subseteq AR$, A is a preferred extension of AF, iff $m(A)$ is a minimal (w.r.t. set inclusion) model of $comp(\Pi_{AF})$.*

Proof. .

- *From Theorem 3 we know that A is a complete extension of AF, iff $m(A)$ is a model of $comp(\Pi_{AF})$.*
- *From Definition 10 we know that a model A is a preferred extension, iff A is also a maximal (w.r.t. inclusion) complete extension of AF*
- *From Corollary 3 we know that A is a maximal complete extension of AF, iff $m(A)$ is a minimal model of $comp(\Pi_{AF})$.*
- *Therefore, A is a preferred extension of AF, iff $m(A)$ is a minimal (w.r.t. set inclusion) model of $comp(\Pi_{AF})$.*

Example 1. In order to illustrate the proof, we will determine the preferred extensions of the argumentation framework depicted in Figure 1.
1.- Mapping this argumentation framework AF into a logic program, we get Π_{AF} as follows:
$defeated(b) \leftarrow not\ defeated(a).\quad defeated(b) \leftarrow \top.$
$defeated(a) \leftarrow not\ defeated(b).\quad defeated(a) \leftarrow \top.$
2.- Then we get $comp(\Pi_{AF})$:
$defeated(a) \leftrightarrow not\ defeated(b) \vee defeated(a)$
$defeated(b) \leftrightarrow not\ defeated(a) \vee defeated(b)$
3.- The models of $comp(\Pi_{AF})$ are: $\{a\}, \{b\}, \{\}$, therefore they also are *complete extensions* of AF (Theorem 3).
4.- A maximal (*w.r.t.* inclusion) *complete extensions* is also a preferred extension (Definition 10), therefore the models {a}, {b} are also preferred extensions.
5.- However, considering that we worked with defeated arguments, we should take into account the minimal models of $comp(\Pi_{AF})$ which are the maximal models' complement (Corollary 3). In our case the complement of {a} is {b}, and the complement of {b} is {a}.
6.- Therefore {b} and {a} are the preferred extension of AF (Theorem 3).

The characterization of preferred extensions of an argumentation framework in terms of minimal models of Clark's completions, and the method defined by Bell *et al.* for computing such minimal models can be put together in a method for computing preferred extensions by integer programming, as stated in the following corollary:

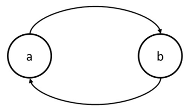


Fig. 1. Graph representation of $AF := \langle\{a, b\}, \{(a, b), (b, a)\}\rangle$.

Corollary 4. *Let $AF := \langle AR, attacks\rangle$ be an argumentation framework, let P be the logic program mapped from AF, and let M be an Herbrand interpretation, and S_M be a binary variable assignment corresponding to M as defined in Definition 2. If S_M is an optimal solution of $lc(P)$ that minimizes $\sum_{A\in B_{\mathcal{L}}} X_A$, then M is a minimal (w.r.t. set inclusion) Herbrand model of $comp(P)$, and a preferred extension of AF.*

Proof. Direct from Definition 2 and Theorem 4.

4 Computing Argumentation Framework Extensions with Integer Programming: An Ongoing Research

In this section we present a technical result in a proof of concept of Corollary 4, however, the purpose of this section is not to make a detailed description of the experiment we made, but to make a little reflection about a new possible method for computing argumentation framework extensions. However, even though the results are not conclusive and the experiment is not finished, it is worth commenting this experiment.

In order to have a measure of the performance of the method based on integer programming for computing the preferred extension of an argumentation framework, we considered that it would be important to compare it against other method, and we chose the method based on Answer Set Programming (ASP) encodings.

The integer programming method used the *ad-hoc* Xpress[2] solver to solve the integer program associated to each argumentation frameworks instance, and for the method based on ASP it was used DLV[3].

4.1 Computing Preferred Extension using Mixed Integer Programming

To carry out the experiment it was necessary to develop some pieces of software, which were used according to the following procedure:

1. We developed a java program to map AF_i instances into logic programs Π_{AF_i}.

[2] http://www.fico.com/en/Products/DMTools/Pages/FICO-Xpress-Optimization-Suite.aspx

[3] http://www.dlvsystem.com/

2. We also developed a C++ program to compute the completion of each Π_{AF_i} generated in previous step, to get $comp(\Pi_{AF_i})$.
3. The same C++ program was used to generate $lc(\Pi_{AF_i})$.
4. The same C++ program also generated, for each $lc(\Pi_{AF_i})$, a *.mos* program which is the language used by Xpress for mathematical programming.
5. Within each *.mos* program it was included the algorithm described after this procedure for computing all the minimal models of each $lc(\Pi_{AF_i})$.
6. Each *.mos* program was executed by Xpress, and the execution times were recorded within each program.

In order to compute all the minimal models of each integer program it was used the following algorithm, and the execution times were recorded immediately after computing all of them.

Algorithm: Minimal Models of $Comp(P)$
Let P be a logic program and $lc(P)$ be constructed as described in Section **??**. *In the following, S is intented to contain all minimal models of $Comp(P)$ and AC is a set of additional constraints.*

1. *Set S and AC to $\emptyset$.*
2. *Solve the integer program: Minimize $\sum_{A \in B_{\mathcal{L}}} X_A$ subject to $lc(P) \cup AC$.*
3. *If no optimal solution can be found, halt and return S as the set of minimal models.*
4. *Otherwise, let M be the model corresponding to the optimal solution found in step 2. Add M to S.*
5. *Add the constraint $\sum_{A \in M} X_A \leq (k-1)$ to AC, where k is the cardinality of M. Then go to step 2.*

The sixty instances used for the experiment ranged from 20 to 50 nodes, and they were taken from the Database and Artificial Intelligence Group Web Page[4].

4.2 Computing Preferred Extension using Answer Set Programming Encodings

For the method based on ASP, which is a declarative programming paradigm under the stable models semantics [9], we used the encodings in [8], where Egly *et al.* presented the system ASPARTIX for reasoning problems in different types of argumentation frameworks by means of computing the answer sets of a datalog program.

ASPARTIX was developed in DLV and is capable to compute several extensions, among them the preferred extensions. For space reasons we can not show the answer set program used to compute the preferred extensions, however the program is available at ASPARTIX's web page[5].

With regard to the experiment, the DLV solver was used for processing each instance, and the execution times were recorded after computing all the minimal models associated to each instance by a C++ program.

[4] http://www.dbai.tuwien.ac.at/proj/argumentation/cegartix/#download
[5] http://www.dbai.tuwien.ac.at/research/project/argumentation/systempage/

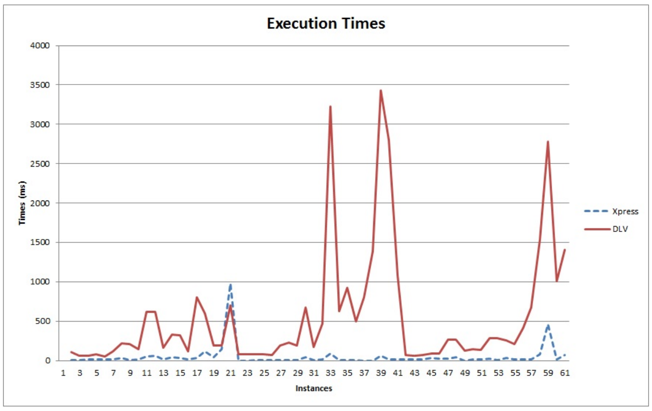


Fig. 2. Graph with Xpress and DLV execition times.

4.3 Proof of Concept Results

Figure 4.3 depicts the results we had in this proof of concept stage, and even though we could not include the table with detailed information, the figure is pretty clear, Xpress outperformed DLV. Remember that the main point at this stage was to find out if an *ad-hoc* integer programming solver can be used for computing preferred extensions of given argumentation framework, therefore the results are not conclusive since the research is not ended.

Thus, at this stage, these results allows us to think that we can go further, in order to be able to draw definitive conclusions.

Maybe the reader can ask why a new method and why integer programming. Well, we should consider that the integer programming method provide a mathematical representation of a given problem, and that integer programming solvers have a long history of development and achievements. Therefore we can infer that this method could become a good alternative for computing argumentation framework extensions.

On the other hand, even if we got just encouraging results, we think that this work represents a good starting point, since it also constitutes a new possible method for computing preferred extensions of argumentation frameworks.

Please, note the importance of this work lies not on the number of theorems or the difficult to reach them, but to make note that the known connection between logic programming and mathematical programming can be used for argumentation frameworks.

5 Conclusions and Future Work

Since Dung introduced his abstract argumentation approach, he proved that his approach can be regarded as a special form of logic programming with negation as failure. In this paper we have showed that preferred extensions can be characterized in terms of minimal models of Clark's completion semantics, by considering a unique mapping of an argumentation framework AF into a logic program.

It is worth mentioning again, that these kind of results also help to understand the close relationship between two successful approaches of nonmonotonic reasoning: argumentation theory and logic programming with negation as failure.

On the other hand, these results helped to build a bridge between argumentation framework semantics and integer programming, but it is required to make an exhaustive experiment to determine the real potential of this method.

References

1. Baral, C.: Knowledge Representation, Reasoning and Declarative Problem Solving. Cambridge University Press, Cambridge (2003)
2. Baroni, P., Caminada, M., Giacomin, M.: An introduction to argumentation semantics. Knowledge Eng. Review 26(4), 365–410 (2011)
3. Bell, C., Nerode, A., Ng, R.T., Subrahmanian, V.S.: Mixed integer programming methods for computing nonmonotonic deductive databases. Journal of the ACM 41(6), 1178–1215 (1994)
4. Besnard, P., Doutre, S.: Checking the acceptability of a set of arguments. In: Tenth International Workshop on Non-Monotonic Reasoning (NMR 2004). pp. 59–64 (June 2004)
5. Carballido, J.L., Nieves, J.C., Osorio, M.: Inferring Preferred Extensions by Pstable Semantics. Iberoamerican Journal of Artificial Intelligence (Inteligencia Artificial) ISSN: 1137-3601, (doi: 10.4114/ia.v13i41.1029) 13(41), 38–53 (2009)
6. Clark, K.L.: Logic and Databases, chap. Negation as Failure, pp. 293–322. Plenum Press (1978)
7. Dung, P.M.: On the acceptability of arguments and its fundamental role in nonmonotonic reasoning, logic programming and n-person games. Artificial Intelligence 77(2), 321–358 (1995)
8. Egly, U., Alice Gaggl, S., Woltran, S.: Answer-set programming encodings for argumentation frameworks. Argument & Computation 1(2), 147–177 (2010), http://www.tandfonline.com/doi/abs/10.1080/19462166.2010.486479
9. Gelfond, M., Lifschitz, V.: The stable model semantics for logic programming. pp. 1070–1080. MIT Press (1988)
10. Gelfond, M., Lifschitz, V.: Classical Negation in Logic Programs and Disjunctive Databases. New Generation Computing 9, 365–385 (1991)
11. Nieves, J.C., Osorio, M., Cortés, U.: Preferred Extensions as Stable Models. Theory and Practice of Logic Programming 8(4), 527–543 (July 2008)
12. Osorio, M., Navarro, J.A., Arrazola, J.R., Borja, V.: Logics with Common Weak Completions. Journal of Logic and Computation 16(6), 867–890 (2006)
13. Osorio, M., Nieves, J.C., Santoyo, A.: Complete extensions as clark's completion semantics. Accepted paper in ENC 2013 (2013)

Method of Musical Composition and Static Topologies for Resource Constrained Project Scheduling: A Case Study

Rafaela Blanca Silva López, Rosa Elena Cruz Miguel, Eric Alfredo Rincón García, Roman Anselmo Mora Gutiérrez, Ponsich Antonin

Dpto. de Sistemas, Universidad Autónoma Metropolitana, Unidad Azcapotzalco
México D.F., Mexico
{rbsl,recm,rigaeral,mgra,aspo}@correo.azc.uam.mx

Abstract. Resource constrained project scheduling problems have a computational complexity that makes it difficult to obtain optimal solutions using exact methods. Thus, heuristic techniques have been used to generate feasible solutions in acceptable computational times. In this paper, we analyse static population topologies for an algorithm based on the Method of Musical Composition (MMC) to solve the problem of course scheduling in a university in a minimum number of quarters. We show that the social network topology used within the MMC operating mode has a significant influence on the performance of the algorithm.

Keywords: Resource constrained project scheduling problem, static population topologies, Method of Musical Composition.

1 Introduction

The resource constrained project scheduling problem (RCPSP) consists in scheduling a set of activities with deterministic processing times, resource requirements and precedence relations between activities. The aim is to find a schedule with minimum makespan (total project duration) respecting both precedence relationships and resource limits. The RCPSP has attracted increasing interest in both academic and engineering fields, including medical research [1], software development [2], audit scheduling [3] and market research interviewers scheduling [4]. The methods for solving the RCPSP have ranged from exact methods [5, 6] to heuristics techniques [7]. However only small sized instances of the RCPSP can be solved optimally using exact algorithms within reasonable time, due to its complexity which has been proved to be NP-hard [8]. Thus, in recent years, the need for solving large, realistic project instances has motivated a shift in research trends towards heuristic approaches.

In this paper, we consider a case study of the Autonomous Metropolitan University (UAM, Mexico) where a set of courses must be scheduled, in such a way that a student can achieve his/her degree in the minimum number of quarters. Due to the complexity of this problem we decide to apply a heuristic method to find solutions within acceptable computing times. However, in recent decades a variety of heuristics have been developed [9], which seek for an improvement of the solution quality or a reduction of the computing time required to generate them. Thus we decided to use a recently proposed population-based heuristic technique known as Method of Musical Composition (MMC). Its performance has been shown to be satisfactory in the context of continuous optimization problems and its application to discrete problems seems a promising alternative. For more details about MMC, its advantages and drawbacks, and differences with other population-based techniques, we refer the reader to [10]. In order to improve the performance of the algorithm we varied the way an individual in the MMC interacts with its neighbors, as proposed in [11, 12]. We tested five commonly used neighborhood configurations and discovered that some of them, for instance, the "star" neighborhood, that performed better than the classical ones on a suite of standard test problems.

In the next section we describe the resource constrained project scheduling problem, while the university project is introduced in section 3. In section 4 we explain the features and operation mode of the MMC. The proposed implementation of the algorithm and result analysis are presented in section 5. Finally conclusions are presented in section 6.

2 The Resource Constrained Project Scheduling Problem

A project consists of a set $\mathcal{J}$ of N activities, $\mathcal{J} = \{1, ..., N\}$, and a set $\mathcal{R}$ of K renewable resources, $\mathcal{R} = \{1, ..., K\}$. Each resource $k \in \mathcal{R}$, is available in limited amounts, R_k, and is renewable from period to period. The duration of activity j is denoted by d_j. Each activity has to be processed without interruption. The precedence relationships are given by sets of immediate predecessors P_j indicating that an activity $j \in \mathcal{J}$ may not be started before each of its predecessors $i \in P_j$ is completed. Activity j requires r_{jk} units of resource k in each period. The activities $j = 1$ and $j = N$ are dummy activities, which represent the start and end of the project, respectively. The dummy activities are added to the project to act as source and sink of the project, and it is assumed that they do not request any resource and their durations are equal to zero.

A schedule can be presented as $\mathcal{S} = \{s_1, ..., s_N\}$, where s_i denotes the starting time of activity i, with $s_1 = 0$. The objective is to determine the starting time of each activity, so that the total project duration, s_N, is minimized, and both the precedence and the resource constraints are respected.

In consideration of the above assumptions the resource constrained project scheduling problem can be formulated as follows:

$$\min\{\max\{f_i : i = 1, 2, ..., N\}\} \tag{1}$$

subject to:

$$s_j - s_i \geq d_i \ \forall i \in P_j, \ j = 1, 2, ..., N \tag{2}$$

$$\sum_{A_t} r_{jk} \leq R_k, \ k = 1, 2, ..., K, \ t = s_1, s_2, ..., s_N \tag{3}$$

Where f_i is the completion time of activity i $(i = 1, ..., N)$ and A_t is the set of ongoing activities at time t.

3 University Project

The UAM's study program consists of courses that must be completed in a predefined number of quarters. All courses have credits and, in some cases, prerequisites and other requirements. These requirements are divided into two groups:

R1. Enrolment to some courses is subject to the successful completion of other predefined classes (precedence constraints).
R2. Enrolment to some courses is subject to have a minimum credit numbers completed.

Besides, the maximum and minimum amount of credits necessary for a student to be able to enroll without special overload approval from his/her college office is restricted.

R3. First quarter, maximum 46 credits.
R4. The remaining quarters, maximum 60 credits.
R5. For all quarters, minimum 21 credits.

Recently UAM's study program was modified, resulting in the elimination and creation of courses, affecting the order in which courses should be completed. These changes required a revision of the new plan, both to create a new schedule for the courses in different quarters and to determine if the new plan could be completed within the time set by the university, i.e. 13 quarters. This work was carried out manually and required an effort of several days to design a solution able to meet the conditions labeled as **R1-R5**.

We decided to modify this problem in order to find the minimum number of quarters required for a student to complete his/her degree. In this case, the problem can be modelled as a RCPSP, where the objective is to determine the starting quarter of each course, so that the total career duration is minimized, while respecting precedence constraints (**R1-R2**) and resource constraints (**R3-R5**).

As mentioned previously, due to the computational complexity of these problems, most authors have focused on the development of heuristic solution methods to find good quality schedules. Thus, we decide to implement the MMC technique whose application in RCPSP has not been reported in the specialized literature.

4 Method of Musical Composition

The Method of Musical Composition (MMC) is a novel heuristic (see [13, 10]), designed according to the following ideas:

- Musical composition can be viewed as an algorithm, since this process uses rules, principles and a finite number of steps to create original music of some particular style.
- Musical composition is a creativity system, involving interactions among agents.
- Usually, agents learn from their experience and use what they learn for future decisions.

In the MMC algorithm, each solution is called a tune and is represented by an D-dimensional vector:

$$tune = [x_1\ x_2\ \ldots\ x_D] \tag{4}$$

Initially, a composer i in the society creates a set of tunes, namely an artwork, which is registered in a score matrix ($P_{\star,\star,i}$). Subsequently, and while the termination criterion is not met, the following steps are carried out: *(a)* the social network linking the different composers is modified; *(b)* composers exchange information (tunes) among each other; *(c)* each composer i decides whether or not using the received information (by including or not within its artwork a tune from other composer) and therefore builds his acquired knowledge (denoted by matrix $ISC_{\star,\star,i}$); *(d)* the i^{th} composer then builds his knowledge background ($KM_{\star,\star,i}$) as the union of $P_{\star,\star,i}$ and $ISC_{\star,\star,i}$; *(e)* the resulting knowledge matrix $KM_{\star,\star,i}$ is used as a basis for creating a new tune $x_{i,new}$; *(f)* finally, the composer makes a decision on whether updating or not his score matrix $P_{\ star,\star,i}$: based on the quality of $x_{i,new}$, this latter will replace the worst tune within $P_{\star,\star,i}$. The termination criterion typically consists of reaching a maximum number of arrangements ($max\_arrangement$), or iterations. The basic structure of the *MMC* metaheuristic is presented in Algorithm 1.

We refer the reader to [13] for a more complete description of the MMC algorithm.

5 Implementation and Numerical Results

5.1 Implementation

In this section, the specific implementation of the proposed algorithm, within the RCPSP framework, is described. Each solution is a schedule that can be represented as a vector $\mathcal{S} = (s_1, ..., s_N)$, where s_i denotes the quarter when course i begins.

Algorithm 1: MMC algorithm

```
1  Create an artificial society with rules of interaction among agents.
2  for each composer i in society do
3  |   Randomly initialize an artwork.
4  end
5  repeat
6  |   Update the artificial society of composers.
7  |   Exchange information between agents.
8  |   for each composer i in the society do
9  |   |   Update the knowledge matrix.
10 |   |   Generate and evaluate a new tune (x_{i,new}).
11 |   |   if x_{i,new} is better than the worst tune in the artwork of composer i
   |   |   (x_{i,worst}) then
12 |   |   |   Replace x_{i-worst} by x_{i,new} in the artwork.
13 |   |   end
14 |   end
15 |   Update the solution set.
16 until termination criterion is met;
```

1. First the algorithm generates for each composer a social network, i.e. a set of arcs representing the relationships between composers. In this case, we tested five commonly used topology networks: random, ring, linear, tree, and star, see next section for a detailed definition of these topologies. The social network remains fixed throughout the algorithm, so the process of society links updating (row 6 of the above-mentioned algorithm) is omitted.
2. Each composer creates a set of solutions using a uniform random number generator that provides integer numbers in the range $[0,Max_{Quarter}]$.
3. Each composer i compares his worst tune, $\mathcal{S}_{i-worst}$, with composers' solutions in his/her social network. Then, he/she updates his/her acquired knowledge, $ISC_{\star,\star,i}$, with those solutions that are better than $\mathcal{S}_{i-worst}$.
4. Each composer i creates a new tune, $\mathcal{S}_{i,new}$, using three solutions. The first solution is the best tune in his/her knowledge background , $\mathcal{S}_{best} \in KM_{\star,\star,i}$. The second solution, $\mathcal{S}_{KM,rnd}$, is randomly selected in $KM_{\star,\star,i} - \{\mathcal{S}_{best}\}$. The third solution is randomly selected from his/her artwork, $\mathcal{S}_{i,rnd} \in P_{\star,\star,i}$. To create $\mathcal{S}_{i,new}$ the starting quarter of the $j-th$ course is randomly selected from the periods assigned in $\mathcal{S}_{best}$, $\mathcal{S}_{KM,rnd}$ and $\mathcal{S}_{i,rnd}$. The cost of $\mathcal{S}_{i,new}$ is evaluated in terms of the objective function and the number of unsatisfied restrictions.
5. Each composer i makes a greedy decision on whether updating or not his/her score matrix. If the new solution $\mathcal{S}_{i,new}$ has a lower cost than $\mathcal{S}_{i,rnd}$, $\mathcal{S}_{i,new}$ replaces $\mathcal{S}_{i,rnd}$.

5.2 Numerical Experiments

In this case we used an instance of 66 courses that must be scheduled in the minimum number of quarters, in such a way that restrictions **R1-R5** are satisfied.

Note that the size of this instance cannot be reduced since it would not result in a realistic challenging problem. Moreover, an exact solver, GUSEK[1], was applied to the instance proposed in this section. Neverthelees, the method could not converge after a 48 hours run, therefore confirming the NP-hard charateristics of the problem.

In Table 1 we present the 66 courses considered in this case. The first and fourth columns provide an identifying label for each course. The second and fifth columns provide the number of credits corresponding to each course. The third and sixth columns provide the precedence list for each course. In Table 2 we present seven courses where students are required to have completed a minimum number of credits to be enrolled.

Table 1. Courses, credits and precedence

Course	Credits	Precedence	Course	Credits	Precedence
1	4		34	9	19
2	9		35	9	27
3	3		36	3	29
4	4		37	9	34
5	3		38	9	8; 20
6	7		39	9	22; 32; 33
7	9	1	40	6	
8	6	2	41	6	
9	3	3	42	12	31; 36
10	9	6	43	8	31; 36
11	6	6	44	9	32
12	9	7	45	7	36
13	3	7	46	9	27; 37
14	6		47	9	35; 46
15	3	12; 13	48	12	46
16	9	11	49	9	22; 30
17	7	10; 11	50	6	
18	6	14	51	6	
19	9	16	52	9	27; 44
20	9	12	53	9	42; 47
21	7	17	54	6	50
22	8	17	55	7	42; 43; 46
23	12	17	56	8	43
24	6		57	6	42; 43; 46
25	6	16; 12	58	8	42; 43
26	9	19	59	3	54
27	9	23; 22	60	8	42; 53
28	3	24	61	6	
29	8	17	62	6	
30	9	17	63	7	55; 56; 57
31	8	22; 26	64	18	59
32	9	17	65	6	54
33	9	21	66	9	24; 43

[1] http://gusek.sourceforge.net/gusek.html

Table 2. Courses and minimum number of credits required

Course	Minimum credits
14	50
40	150
41	150
50	280
59	360
61	150
62	150

As mentioned previously, five social networks, based on different topologies, which have been commonly used to improve the performance of social algorithms [12], were considered for the implementation of the MMC-based algorithm:

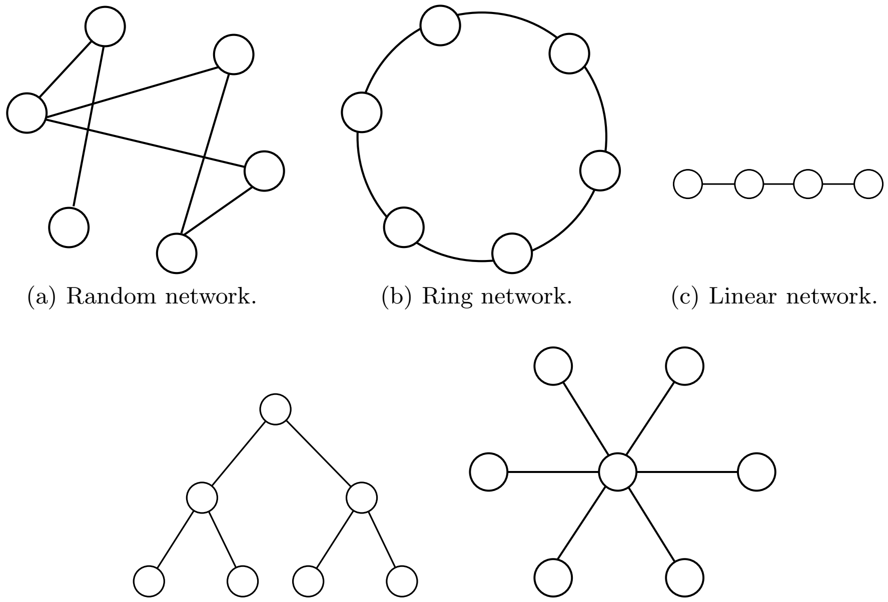
1. **Random**: Is composed by randomly generated arcs.
2. **Ring**: A circular list of composers (i.e. a permutation) is created, such that each composer sends (receives) information to his/her successor (from his predecessor).
3. **Linear**: Composers are arranged over a line, where most composers are connected to two other composers. However, the first and last composers are not connected as they would be in the ring structure.
4. **Tree**: One “root” composer connects to other composers, which in turn connect to other composers, forming a tree structure.
5. **Star**: One central composer is connected to all the remaining composers.

We applied a brute force calibration procedure by several executions of the algorithm using a random social network. This way, the number of composers and tunes for each composer were set to 25 and 3, respectively. We decided to use these values for all social networks in order to provide a fair comparison basis between different topologies. Finally, the number of generations was set to 90,000.

Besides, in order to deal with the stochastic effect inherent to heuristic techniques, 100 independent executions were performed for each topology. Each run produced a single solution and the resulting 100 solutions were subsequently divided into two groups according to the number of quarters required to schedule the 66 courses: solutions that required 11 quarters and solutions that required 12 or more quarters. We must remark that all generated solutions satisfied **R1-R5**. The results for all topologies are presented in Table 3.

Table 3. Results for different topologies

Topology	11 quarters	12 or more quarters
Random	9	91
Ring	12	88
Linear	5	95
Tree	6	94
Star	61	30



(a) Random network. (b) Ring network. (c) Linear network.

(d) Three network. (e) Star network.

Fig. 1. Example of the networks used in this article

It is clear, from the observation of Table 3, that the star and ring topologies outperform the other ones, since they produce 11-quarters solutions for more than 60% and 10% of the runs, respectively. Using this information we deduced that the social network is an important parameter for the performance of the algorithm. We decided to modify the algorithm to apply a variant of the best topologies. Thus two rings and two stars topologies were used while other parameters remain unchanged. In this case, two circles (stars) with 13 and 12 composers were considered, and a link between two composers, one in each circle (star), was used to exchange knowledge between both “sub-societies”. As can be seen in table 4, these variants improved the performance of the proposed algorithm, where the two star topologies clearly provided the best results.

Table 4. Results for two circles and two stars topologies

Topology	11 quarters	12 or more quarters
Two rings	43	57
Two stars	78	22

6 Conclusion

In this paper we presented a resource constrained project scheduling problem, which consists of placing a set of courses in the minimum number of quarters, subject to precedence relationships and limited resources. The example presented in this work includes 66 courses and was created using real-world information from the Mexican Autonomous Metropolitan University's study program. 55 courses has at least one precedence restriction, and 7 courses requires a minimum number of completed credits to be available for students. Quarters with limited resources, 46 and 60 credits, are used to schedule the courses, and an additional requirement, at least 21 credits for each quarter, is also involved.

A discrete implementation of the a heuristic technique, namely the Method of Musical Composition, was proposed for the solution of this problem. To analyze the performance of the algorithm different static social networks, based on five topologies, were used. After several executions we observed that star or ring networks were acceptable options. In order to improve the performance of the algorithm we decided to apply a variant of these topologies. Thus two circles and two stars topologies were implemented. The computational experiments proved that a two stars topology produces better solutions than its counterparts. These results indicate that the MMC based algorithm proposed is able to find feasible solutions for the RCPSP described in this work, but its performance highly depends on the social network topology. Our future work includes the study of dynamic topologies within our algorithm.

References

1. Hartmann, S.: Scheduling medical research experiments – an application of project scheduling methods (1997)
2. Alba, E., Francisco Chicano, J.: Software project management with gas. Inf. Sci. **177**(11) (June 2007) 2380–2401
3. Dodin, B., Elimam, A., Rolland, E.: Tabu search in audit scheduling. The a. gary anderson graduate school of management, The A. Gary Anderson Graduate School of Management. University of California Riverside (1996)
4. Hartmann, S.: PROJECT SCHEDULING:. Lecture notes in economics and mathematical systems. Springer (1999)
5. Sprecher, A.: Resource-constrained project scheduling: exact methods for the multi-mode case. Lecture notes in economics and mathematical systems. Springer-Verlag (1994)
6. Zamani, M.R.: A high-performance exact method for the resource-constrained project scheduling problem. Computers & OR **28**(14) (2001) 1387–1401
7. Fang, C., Wang, L.: An effective shuffled frog-leaping algorithm for resource-constrained project scheduling problem. Computers & OR **39**(5) (2012) 890–901
8. Blazewicz, J., Lenstra, J., Kan, A.: Scheduling subject to resource constraints: classification and complexity. Discrete Applied Mathematics **5**(1) (1983) 11 – 24

9. Brownlee, J.: Clever Algorithms: Nature-Inspired Programming Recipes. Lulu Enterprises Incorporated (2011)
10. Mora-Gutiérrez, R.A., Ramírez-Rodríguez, J., Rincón-García, E.A., Ponsich, A., Herrera, O.: An optimization algorithm inspired by social creativity systems. Computing **94**(11) (2012) 887–914
11. Kennedy, J., Mendes, R.: Population structure and particle swarm performance. In: Proceedings of the Congress on Evolutionary Computation (CEC 2002). Volume 2., IEEE Press (2002) 1671–1676
12. Kennedy, J., Mendes, R.: R.: Neighborhood topologies in fully-informed and bestofneighborhood particle swarms. In: In: Proceedings of the 2003 IEEE SMC Workshop on Soft Computing in Industrial Applications (SMCia03, IEEE Computer Society (2003) 45–50
13. Mora-Gutiérrez, R., Ramírez-Rodrguez, J., Rincón-García, E.: An optimization algorithm inspired by musical composition. Artificial Intelligence Review (2012) 1–15

Digital Image-based Inductive Characterization and Classification to Improve the Quality Inspection of Diverse Food Products

Hiram Calvo, Salvador Godoy-Calderón, Marco A. Moreno-Armendáriz

Centro de Investigación en Computación-IPN,
México, D. F., Mexico

hcalvo@cic.ipn.mx, sgodoyc@cic.ipm.mx,
mam_armendariz@cic.ipn.mx

Abstract. With the increasingly demanding international regulations for import and export of food products, as well as with the increased awareness and sophistication of consumers, the food industry needs accurate, fast and efficient quality inspection means. Each producer seeks to ensure that their products satisfy all consumer's expectations and that the appropriate quality level of each product is offered and sold to each different socio-economic consumer group. This paper presents three study cases where digital image analysis and inductive characterization techniques have been successfully applied to improve the quality inspection process. Three very different and unrelated basic food products are studied: Hass Avocado, Manila Mango and Corn Tortillas. Each one of these products has some special and particular features that complicate the quality inspection process, but each of these products is also very important in economical terms for the sheer volume of their production and marketing. Experimental results of each case shows that the general technique has great accuracy and significantly lower costs.

Keywords: Inductive characterization, digital image analysis, corn tortilla, hass avocado, manila mango, boundstar.

1 Introduction

Nowadays, assessing the quality of food products involves mostly using physical methods. These methods can be non-destructive (produce compression, tactile testing) and destructives (penetrometers); however, even traditional non-destructive methods may represent a quality dropping, because the produce is usually delicate and can be easily damaged (Ochoa et al., 2009). For this reason, newer noninvasive techniques have been proposed. Noninvasive methods such as acoustic, spectroscopic, and digital image analysis, including multi- and hyperspectral imaging techniques, have been widely applied to evaluate the quality foods. However, acoustic and spectroscopic methods (near-infrared, nuclear magnetic resonance, and magnetic resonance imaging) could be relatively expensive and laborious, and a large number of samples, as well as a large laboratory area, could be required; therefore, an interesting alternative to study the superficial features in produces is the computerized image processing (also known as digital image analysis) that has been widely developed in the food sector. Physical characteristics, such as size, shape, morphology, color and texture properties can be measured by means of image processing (Du and Sun, 2004; Brosnan T and Sun D-W, 2004; Mendoza et al., 2007; Quevedo et al., 2008;).

Digital image analysis allows food quality evaluation by maintaining accuracy and consistency while eliminating the subjectivity of manual inspections. Image analysis have more advantages than other noninvasive techniques such as being cheap, easily adaptable to obtain measurements online, high accuracy, good correlation with visual human inspection, and is very versatile because it allows to obtain a widespread number of features from a simple digital image (Du and Sun 2004).

Careful feature selection of image analysis plays an important role for a successful classification. Some works as (Mery et al., 2010) extract as much as 2,300 features to finally use only a few, 64. This represents a system overload and hinders automatic classification systems from real applications, where very fast classification is needed due to the sheer volume of production.

In this work, we present the feature extraction of three different food produces, namely tortilla, avocado, and mango, described in the following subsections. Then, their relevant features are selected (Section 2) and then they are classified using inductive algorithms (See Section 3). Then we present the results of our experiments (Section 4) and finally we draw conclusions (Section 5).

2 Feature Selection

In this section, we describe the feature selection of three produces: tortilla, mango, and avocado for their latter classification using inductive algorithms. Selecting features carefully is nearly as important—if not more—as the classification algorithm itself. A good feature set must be compact, non-redundant, and one that allows a classifier to correctly separate the classes of interest.

2.1 Color Features

Each image is separated into Red, Green and Blue channels (RGB), and then three interest areas are determined for each image by color-reducing each channel. Those areas bring evidence of the cooking degree on the tortilla surface, or the maturity level on fruits. A tortilla can be considered raw (undercooked), well cooked or burnt (excessively cooked) by counting the surface area occupied by bright spots, average color spots or dark spots respectively. For the case of fruits, the color uniformity suggests little or no damage, while sudden changes of color represent damaged areas.

To reduce the number of intensity levels in an image channel, we establish a threshold. Background pixels are also mapped to a zero-intensity value; so that they appear to be part of the dark spots area but they are filtered out during the color-features extraction process.

Once the three interest areas within each image are selected, the following processes take place: contour identification, surface area filtering and edge characterization; then, features are extracted. We propose the following color features for produce classification

Contour identification. Two morphological *hit-or-miss* operators are proposed. Each structure element is applied four times, rotating 90° each time, and then we use the union of the results. The first structure element, composed by the values $B_{N1} = \{0, \neg\mathbf{0}, \neg 0\}$ looks for image regions matching one black pixel (the background) followed by two consecutive non-zero values (the produce surface). The second structure element $B_{C1} = \{\neg 128, \neg\mathbf{128}, 128\}$ looks for regions matching two consecutive non high-intensity pixels followed by a high-intensity pixel, thus, finding raw areas inside the tortilla, or damaged areas inside a fruit.

Color homogeneity (***cHom***) describes a uniform color distribution. For tortillas, this would be the cooking process, and, of course, the average color depends on the corn type the tortilla is made of, the damage area or the mango, or the ripening stage of the mango. In general a uniform specific color represents a better produce (a correctly baked tortilla, a non-damaged mango or avocado). Color will be less homogeneous when there are defects on the produce.

Average Brightness (*Lavg*). This feature identifies, as the name implies, the amount of light from the surface, which ideally should be equivalent throughout the surface, this value varies depending on the type of dough used for making the product in the case of the tortilla, while in the case of fruits, it represents their degree of maturity. This parameter is obtained by an arithmetical mean of all pixels contained in the surface of the produce.

Contrast *(Cnst).* Is the number of local variations in grayscale tones of the image. The more is the variation of the gray tones, the greater is the contrast; a contrast of 0 means that gray levels are constant along the image.

Variance of light intensity (*Lvar*). The non-homogeneous coloration on the surface, is caused by burnt or raw areas and poorly processed corn leftovers (for the tortilla). In the case of the mango and avocado, it represents the amount of a damaged area, or some sickness of the fruit. These defects cause the brightness of the surface not to be homogeneous, which may vary dramatically from one pixel to another. In the ideal case, the variance should be zero, so that for a uniform brightness would expect a value close to zero. For the calculation of this feature, as with the average brightness, all pixels of the surface are used and then the variance of these values is used.

Raw areas (*rawA*). The raw areas represent sections of dough that did not reach the proper cooking and have a lighter color than the rest of the surface. As with the burnt areas, the extraction of the obtained raw areas is obtained from the surface segmentation and calculation is carried out indirectly by counting the pixels of the edges.

Burned areas (*burntA*). Burned areas represent sections of varying sizes of burnt dough produced by an excess of cooking. This feature represents also the damaged areas in fruits. Coloring of these areas is common to any hue of the tortillas or fruits, making it easy to identify. For its identification we use the border extracted as described previously with the *hit-miss* transformation, by filtering the edges of the burned areas from the produce. Obtaining this feature is done in an indirect way, because by counting the edge borders we infer that as there are more edge pixels, the size of the area would be greater.

2.2 Texture Features

The texture parameters described here provide detailed information about the structural changes that occur in the peel during the ripening process of fruits (*i.e.* mango and avocado) while the changes of color described above can be used as auxiliary parameters to evaluate and determine the fruit ripeness.

The **angular second moment** (**ASM**) is a feature that measures the homogeneity of the image (Haralick et al., 1973), **Contrast** is a measure of the local variations present in the image, relating the highest contrast to the largest local variations (Haralick et al., 1973; Mendoza et al., 2007). **The fractal dimension (FD)** or fractal texture is a measure of the degree of roughness of the images; higher values of FD mean more complex or rougher grey level images, while lower values of FD can be associated to simpler or smoother images (Quevedo et al., 2008). Finally, **Entropy** measures the disorder or randomness of the image and it is an indicator of its complexity, thus, the more complex the images the higher the entropy values (Haralick et al., 1973; Mendoza et al., 2007).

2.3 Geometric Features

Since shape is important for the tortilla quality assessment, special geometric features were obtained for this purpose. In order to get a discrete representation for the tortilla's border, we divide it into 64 arc segments. Each arc is projected over the horizontal axis and the length of the projection is measured (Watanabe and Matsumoto, 1991; Cai et al. 2004). A characteristic descriptor for each tortilla side is made up with the lengths of the 64 projected arcs (Hastie et al., 2003).

Geometry-related features aim to capture those attributes related to the shape of the tortilla including curvature; symmetry and continuity. A high quality tortilla must have a nearly perfect circular shape with no holes and no bends or breaks along its border. Since a discrete characteristic chain is used to describe the border of the tortilla, this chain is compared with the corresponding chain of a perfect circle as described in (Gupta and Srinath, 1987). If both chains are identical then the tortilla border has the best quality available. The following features were extracted:

Circularity (circ). Its value is calculated by adding up the differences of each one of the diameters with regard to an average.

Defects (dfct). A defect in a tortilla occurs when a small irregularity appears on the edge, perceptibly altering the circularity of the edge section; this change is often abrupt and has a short distance of no more than one centimeter.

Deformations (dfrm). The deformation in a tortilla is along one segment edge with a non-circular tendency. In this segment the edge direction usually has only one direction (straight).

3 Inductive Classification

Michalski and Chilausky (Michalski and Chilausky, 1980) in the early 1980s, developed an expert system based on learning from examples, called pLANTS / DS. It was able to perform diagnostics of diseases of the soybean plant.

One of the most common techniques used to obtain the rules of inductive learning is known as "divide and conquer" (Domingos, 1996). This technique, which appeared in the early 80's, is named after the method applied to construct a rule induction, dividing the initial set of knowledge rules and selecting the rules that provide better coverage rates. Important works using this technique are Michalski, 1983, Michalski et al. 1986; Clark and Tim, 1989, and Rivest, 1987. Several innovations and/or adaptations of the techniques proposed by them arose for nearly a decade after the publication of these mentioned works.

One of these works to improve learning technique was performed by the same Michalski in the mid-1980s. The STAR method is based on the principle of "divide and conquer" and it further allowed the resolution of everyday problems or applications that have large numbers of possible solutions. Finally, it emerged and positioned itself as the BOUNDSTAR and REDUSTAR methodologies.

3.1 The BOUNDSTAR Algorithm Implementation

The implementation of the BOUNDSTAR algorithm yields as result a series of learning rules that will characterize each class. Since the BOUNDSTAR algorithm strives to find a $G(e|E0,m)$ set, then: e is the event from $E0$ such as ($e \in E0$), m is the set of events that best characterizes a specific class. In our case-study we have m1, m2 and m3 for selecting class1, class2, and class3 respectively.

LEF is the decision criterion that determines the preference order for the events selected by m, considering the class coverage percentage. As the coverage grows, so will the preference. The complete algorithm flows as follows:

Pseudocode for the BOUNDSTAR algorithm

1. By using the LEF criterion, m1 events that best characterized class1 are obtained from the e events. These m1 events are ordered and then are moved to the PS1 set.
2. Each one of the PS events is extended by conjunction with the new events obtained in step (1).
3. Again, the LEF criterion is used to find a new set of events that will be aggregated to a new PS set. Those events that won't be part of the new PS set will be set aside in an "always-available", called ED set.
4. The complex features that strongly characterize the class selected by m1

are found and selected within the PS set and copied into the SOL set.

5. PS1 is again extended by conjoining its elements with those in the ED and PS sets. These sets are evaluated by LEF and only the events with the best covering percentages continue to the next stages.
6. Return to step (4) until the stop criterion is met.

For the purpose of this research, the stop criterion is finding a strong characterizer among the preserved sets, or having an empty ED set. When the algorithm stops because of this last condition, the selected solution is the best qualified by LEF. The above algorithm is repeated for m2 and m3, and that is the way in which the characterization rules are obtained.

The last step in the algorithm is the learning rule performance's analysis. That rule later serves as a classifier over a set of non-ordered samples. During the classification process, new samples are assigned to one of the three classes, or to a set of non-classified samples.

At the end of the classification stage, the learning rule's efficiency is evaluated using the precision and recall concepts.

4 Experimental Results

In this section we will present results for classifying three different produces with the aforementioned feature characterization and inductive analysis.

4.1 Tortillas

We obtained a sample of 600 tortillas, from three kinds of commercial establishments, 200 from each kind. With help from some experts in food engineering, a model of the perfect tortilla made from each possible type of corn was defined and then we proceeded to apply the induction classification. We learned from the first half of 300, and then we evaluated with the remaining 300. Details on the feature extraction of the 300 images of tortilla during the learning phase can be seen in (Moreno-Armendáriz *et al.*, 2013). In **Table 1** the set of solutions that characterize classes better, after applying the REDUSTAR algorithm are shown in bold.

Table 1. Induced rules for classifying tortillas. The rules to be used in the generalization are shown in bold

Class	Solution set	C 1	C 2	C 3
	burntA < 220 Λ *circ* < 60 Λ *Lavg* < 50	38	4	0
1	***rawA* > 1200 Λ *Lavg* < 50**	**96**	**0**	**10**
	0.3262 < *dfrm* < 0.4452 Λ *Lavg* < 50	52	0	0
2	***Lavg* > 50**	**0**	**100**	**0**
	Lvar < 85.5 Λ 0.3262 > *dfrm* < 0.4452	0	0	63

3	220 < *burntA* <450 Λ *Lvar* < 91	15	0	72
	***Lavg* < 50 Λ *rawA* < 1200 Λ *circ* > 60**	**0**	**0**	**90**

Table 2. Classified samples. The rows are read the actual class and columns are the class to which they were assigned according to their features and the learned rules of knowledge

		Classification			
		Class 1	Class 2	Class 3	No class
	Class 1	82	0	16	2
Real	Class 2	0	100	0	0
Class	Class 3	2	0	97	1

4.2 Avocado

A computer vision system similar to that described by Pedreschi et al., (2004) was employed to capture the images (1600 x 1200 pixels RGB color and JPEG format). Samples were illuminated using four fluorescent lamps, with a color temperature of 6500 K. Lamps (60 cm long) were arranged in the form of a square, 35 cm above the sample and at an angle of 45° in relation with the sample. Twenty images were obtained for each day of the maturation kinetic (two for each avocado). Each sample was placed in front of the camera in the same position and orientation. Images were obtained using a color digital camera that positioned vertically over the sample. Images of two faces of the avocados were taken on a matte gray background using the following camera settings: manual mode with the lens aperture at f = 2.8 and speed 1/15 s, no zoom and no flash.

The experimental data set was restructured to account for each avocado-side image as a distinct pattern. By doing so, we could set a matrix with 240 rows, one for each pattern (2 images of each avocado during each of the 12 days of the experiment) and 3 columns, one for each digital color parameter (L^*, a^* and b^*). This matrix (*SM*) was used as a supervision sample for the rest of the process.

Following the BOUNDSTAR algorithm, we ranked each feature according to its discrimination capability (Jollife, 2002). In such ranking, the most relevant feature was a^*, followed by b^* in second place and L^* at the end. A weight value in the interval [0,1] is assigned to each feature based on this ranking.

Next we classified the samples using the previously obtained rules, taking as an input a new pattern (the digital image of one side of an avocado) and as an output the maturity stage of the respective avocado.

Statistical analysis. The results were expressed as averages with their standard errors. We applied a one-way analysis of variance (ANOVA) with Tukey's multiple comparison tests for statistical comparisons of data. Ripening classification was developed based on two experiments, and for both cases, 240 images were used (2 images of each avocado during 12 days of the experiment, and 10 avocados per day, for 240 patterns). A first experiment was carried out exclusively with the three digital color parameters (L^*, a^* and b^*) showing that the a^* parameter presented the highest weight value (0.6), followed by b^* and L^* (0.3 and 0.1, respectively). The second experiment, which did not include b^* and L^* parameters, was done employing the six more relevant features (a^* 0.5, Correlation 0.2, ASM 0.2, IDM 0.05, FD 0.025, and Entropy 0.025). The BOUNDSTAR algorithm ranked both sets of features. In the first experiment, a successful classification was found, where 58 of 72 images were properly classified, which represents a percentage of **80.5%.** In the second experiment, 59 images of 72 for the validation phase could be correctly classified, which corresponds to **81.9%.** In both experiments, the chromaticity coordinate a^* was the parameter that described better the ripening process of avocados Hass.

4.3 Mango

After taking mangoes' photographs as described for the avocado in the previous section, we used mathematical morphology for image processing. The morphological operators used were Bot-Hay and Top-Hat used for defect segmentation. We used the feature *burntA* to quantify the damaged areas, as described in Section 0.

Once the mango's damaged surface has been quantified, we classified it according to the Mexican norm NMX-FF-058-SCFI-2006, since it establishes damage tolerances for each mango class. Herrera-Corredor et al. (2007) sets out the different types of damage a mango can show, and also, the expected damage tolerance for each class. That norm establishes 3 mango quality classes: Extra class, First class and Second class. In order to classify mangos, no distinction is made about the various defects, such as illness or mechanical damage that a mango can show. Therefore, the image segmentation processes (*burntA*) just detect any damaged surface unit regardless of the type of defect it represents. Once both mango sides' images are analyzed and the total damaged surface is calculated in square centimeters, the following formula from the Mexican norm, is used to immediately deduce the quality class of each mango sample: Minimum damage: area $\leq$ 23mm^2, High damage: area > 30mm^2 and area $\leq$ 49mm^2, Critical damage: area > 49 mm^2.

The proposed methodology for the analysis of the Manila Mango quality was tested over a 334 images set corresponding to 167 mangoes. This set was previously classified as follows: 27 mangoes for Class 1, 68 for Class 2, and 72 for Class 3. Classification results are shown in **Table 3**, from which an accuracy of 80% is calculated.

Table 3. Confusion matrix resulting from the classification

	Class 1	**Class 2**	**Class 3**
Class 1	27	0	0
Class 2	20	48	0
Class 3	3	10	59

5 Conclusions

This paper presents three study cases where digital image analysis and inductive characterization techniques have been successfully applied to improve the quality inspection process. Three different basic food products were studied: Avocado, Manila Mango and Tortillas. Each one of these products has some special and particular features that complicate the quality inspection process, but the general technique of inductive characterization with a reduced set of features presented in this work, has great accuracy and significantly lower costs.

An example of the general process of characterization can be seen in **Table 1**, where early complex features achieve the percentages of coverage. From this learning stage it can be appreciated, as anticipated in the extraction of features, that not all features that are useful for the characterization, however, there are complex features capable of characterizing a class above the other with excellent results. Examples are traits for the tortilla *Lavg* > 50 and *Lvar* > 91, which are close to 100% coverage in the class of interest, class 2. After the implementation of the BOUNDSTAR algorithm, we obtained the rule set shown in bold in in **Table 1**. This rule set represents the best solutions for each class and it can be used independently for different purposes.

We have found that the inductive characterization of three classes of producers is achieved with high percentages of coverage and precision; particularly in the case of Group 2 of tortillas presents a coverage of 100%. Importantly, despite the diversity of variables involved it was possible to find distinctive patterns of each. The proposed color and geometric features were useful for achieving this classification. The most important features in the classification, given the high percentage of coverage for the desired classes were color features. Significantly, color features become extremely important in characterizing when considering the preference of the consumer.

In general, we can say that image features obtained by image processing provide a good description of the quality of produces. In addition, the color and texture features evaluated from the images, in particular FD values, showed an adequate correlation with the traditional quality parameters. However, in this particular study, the image color features resulted in better information than the image texture parameters when a reduced number of samples and images features were used. Particularly, when applying only three image color features an adequate classification of the avocados Hass was obtained. Nevertheless, the chromaticity coordinate a^* was the best parameter to describe the ripening process. These results confirm that computer vision systems, based on image processing, can be used to determine the quality of produces. The existence and commercial availability of these systems greatly impacts on the logistic operation and budget of independent farmers and small food industries. Moreover, critical systems which analyze and determine the quality of agricultural products, such as fruits or vegetables, or even other products such as tortilla, could be very well designed and implemented following the presented methodology.

As a future work, we plan to test our implementation on hardware to make it fully automatized.

References

1. Brosnan T and Sun D-W.: Improving quality inspection of food products by computer vision. Journal of Food Engineering, 51(3), 3−16 (2004)
2. Cai, W., Yu, Q., Wang, H.: A fast contour-based approach to circle and ellipse detection. In: Fifth World Congress on Intelligent Control and Automation. 5, pp. 4686−4690 (2004)
3. Clark, P. and Tim N.: The CN2 induction algorithm, Tahoe City,CA (1989)
4. Domingos, P. Unifying Instance-Based and Rule-Based Induction. Machine Learning, vol. 24, nº 2 (1996)
5. Du C and Sun D-W.: Recent developments in the applications of image processing techniques for food quality evaluation. Trends in Food Science and Technology, 15(1), 230−249 (2004)
6. Haralick R, Shanmugam K. and Dinstein I.: Textural features for image classification. IEEE Transaction on Systems, Man, and Cybernetics SMC, 3(6), 610−621 (1973)
7. Hastie, T., Tibshirani, R., Friedman, J.: The Elements of Statistical Learning: Data Mining, Inference, and Prediction. Springer (2003)
8. Herrera-Corredor, J., Saidu, J., Khachatryan, A., Prinyawiwatkul, W., Carballo-Carballo, A., Zepeda-Bautista, R.: Identifying Drivers for Consumers Acceptance and Purchase Intent of Corn Tortilla. J. of Food Sc. 72 (9), S727−S730 (2007)
9. Jollife, I.T. Principal Component Analysis, Series: Springer Series in Statistics, 2nd ed., Springer, NY, (2002)
10. Mendoza F, Dejmek P and Aguilera J.: Colour and image texture analysis in classification of commercial potato chips. Food Research International, 40(1), 1146−1154 (2007)
11. Mery D., Chanona-Pérez, J.J., Soto, A., Aguilera, J.M., Vélez-Rivera, N., Arzate-Vázquez, I., Gutiérrez-López, G.F.: Quality Classification of Corn Tortillas using Computer Vision. J. of Food Eng. 101, pp. 357−364 (2010)
12. Michalski, R.: A theory and methodology of inductive learning. v. 20 (1983)
13. Michalski, R.S. and R. L. Chilausky, Learning by being told and learning form examples: an experimental comparison of the two methods of knowledge adquisition in the context of developing an extra system for sorbean desease diagnosis. Policy analysis and information systems, vol. 4, nº 2, pp. 125−160 (1980)
14. Michalski, R.S., I. Mozetic, J. Hong, N. Lavrac: The multi-purpose incremental learning system AQ15 and its testing application to three medical domains. In: Proceedings of the Fifth National Conference on Artificial Intelligence, Philadelphia (1986)
15. Moreno-Armendáriz, S. Godoy-Calderón, H. Calvo and O. M. Rojas-Padilla: Assessing the Quality Level of Corn Tortillas with Inductive Characterization and Digital Image Analysis. In: J.A. Carrasco-Ochoa et al. (Eds.), LNCS 7914, Springer-Verlag, pp. 40−53 (2013)
16. Norma-Oficial-Mexicana, NOM-187-SSA1/SCFI-2002: Productos y servicios. Masa, tortillas, tostadas y harinas preparadas para su elaboración y establecimientos donde se procesan. Especificaciones sanitarias. Secretaría de Economía, Estados Unidos Mexicanos (2003)
17. Ochoa S, Hertog, M. and Nicolaï, B.: Modelling the transient effect of 1-MCP on Hass avocado softening: A Mexican comparative study. Postharvest Biology and Technology, 51(1), 62−72 (2009)
18. Pedreschi F, Mery D, Mendoza F. and Aguilera J.: Classification of potato chips using pattern recognition. J. of Food Science, 69(6), E1−E5 (2004)

19. Quevedo R, Mendoza F, Aguilera J, Chanona J., and Gutierrez G.: Determination of senescent spotting in banana (*Musa cavendish*) using fractal texture Fourier image. Journal of Food Engineering, 84(1), 509−515 (2008)
20. Rivest, R.L.: Learning decision lists. Machine Learning, pp. 229−246 (1987)
21. Gupta, L., Srinath, M.D.: Contour Sequence Moments for the Classification of Closed Planar Shapes. Pattern Recognition 20 (3), pp. 267−272 (1987)
22. Watanabe, T., Matsumoto M.: Recognition of Circle Form Using Fuzzy Sequential System. In: Twenty-First International Symposium on Multiple-Valued Logic, pp. 85−92 (1991)

A Novel Text Encryption Algorithm

Elena Acevedo, Ángel Martínez, Marco Acevedo, Fabiola Martínez

Escuela Superior de Ingeniería Mecánica y Eléctrica,
Instituto Politécnico Nacional, Mexico City, Mexico

{eacevedo, macevedo, fmartinezzu}@ipn.mx

Abstract. The encryption/decryption processes is applied to text. Both algorithms involved the use of the associative models, in particular we propose the Alpha-Beta associative memories. The original text is divide in sets of 9 elements, and together with a secret key they build an associative memory which represents the encrypted text. The advantage of our proposal is the cyphertext does not have the same dimension that the plaintext. Additionally, the cyphertext is represented as an image, therefore, since the beginning the encrypted text has a different meaning.

Keywords: Artificial Intelligence, associative models, alpha-beta associative memory, text encryption.

1 Introduction

Cryptrography [1] is the science of protecting data and communications. A cryptosystem has two parts: encryption, which is done at the sender's end of the message and means to put the actual plaintext (original messages) into cyphertext (secret code), and decryption, which is done at the recipient's end and means to translate the cyphertext back into the original plaintext message. Generally an encryption or decryption algorithm will relay on a secret key [2], which may be a number with particular properties, or a sequence of bits; the algorithm itself may be well known, but to apply the decryption to a given cyphertext requires knowledge of the particular key used.

Traditional encryption algorithms are private key encryption standards (DES and AES), public key standards such as Rivest Shamir Adleman (RSA), and the family of elliptic-curve-based encryption (ECC), as well as the international data encryption algorithm (IDEA).

There are other algorithms for encrypting text. Some of them use the corresponding decimal ASCII code, convert it to binary numbers and apply a process [3], [4], [5] for changing the order of the bits. Another proposal is a technique on matrix scrambling which is based on random function [6], shifting and reversing techniques of circular queue. A symmetrical encryption algorithm [7] is proposed in this paper to prevent the outside attacks to obtain any information from any data-exchange in Wireless Local Area Network. Other algorithms [8] encipher message into nonlinear equations using public key and decipher by the intended party using private key. Some works applied modifications to traditional encryption algorithms:

AES [9], blowfish [10], DES [11] and RSA [12]. In this work we propose a text-image encryption as [13] but with a different approach.

2 Basic Concepts

Sometimes, we recall a person, a place or a feeling when we smell certain perfume, we see a blue sky or we watch a movie, in other words, we can associate a person with a perfume, a place with a blue sky or a feeling with a movie. We also can resolve a quadratic equation because we at once associate the form of this equation with its solution as follows,

$$ax^2 + bx + c = 0 \rightarrow x = \frac{-b \pm \sqrt{b^2 - 4ac}}{2a}$$

Therefore, our brain learns by association. Neural Networks try to simulate the structure of the brain by interconnecting a set of neurons, on the other side, Associative Memories try to simulate the behavior of the brain, i.e., they associate concepts.

Associative Memories (AM) associate patterns x with y, which can represent any concept: faces, fingerprints, DNA sequences, animals, books, preferences, diseases, etc. We can extract particular features of these concepts to form patterns x and y.
There are two phases for designing an associative memory: Training and Recalling.
In the Training Phase (see Figure 1), the process of associate patterns x with patterns y is performed. Now, we say that the memory is built.

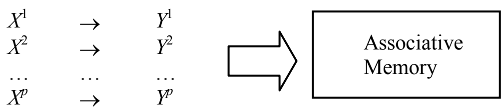


Fig. 1. Training Phase of an Associative Memory

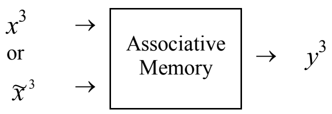


Fig. 2. Recalling Phase of an Associative Memory

Figure 2 shows the Recalling phase where the pattern x is presented to the AM and the corresponding pattern y is recalled. Also, a noisy version of a pattern x could be presented and the memory should recalls is corresponding y pattern.

The input and output patterns are represented by vectors. The task of association of these vectors is called Training Phase and the Recognizing Phase allows recovering patterns. The stimuli are the input patterns represented by the set $\mathbf{x} = \{x^1, x^2, x^3, \ldots, x^p\}$ where p is the number of associated patterns. The responses are the output patterns and are represented by $\mathbf{y} = \{y^1, y^2, y^3, \ldots, y^p\}$. Representation of vectors x^μ is

$x^{\mu} = \{x_1^{\mu}, x_2^{\mu}, \ldots, x_n^{\mu}\}$ where n is the cardinality of x^{μ}. The cardinality of vectors y^{μ} is m, then $y^{\mu} = \{y_1^{\mu}, y_2^{\mu}, \ldots, y_m^{\mu}\}$. The set of associations of input and output patterns is called the fundamental set or training set and is represented as follows: $\{(\mathbf{x}^{\mu}, \mathbf{y}^{\mu}) \mid \mu = 1, 2, \ldots, p\}$

There are two types of associative memories concerning to the nature of the input and output patterns.

A memory is ***Autoassociative*** if it holds that $x^{\mu} = y^{\mu}$ $\forall \mu \in \{1, 2, \ldots, p\}$, then one of the requisites is that $n = m$.

A memory is ***Heteroassociative*** when $\exists \mu \in \{1, 2, \ldots, p\}$ for which $x^{\mu} \neq y^{\mu}$. Notice that there can be heteroassociative memories with $n = m$.

Now, we will describe the Morphological associative model.

The fundamental difference between classic associative memories (Lernmatrix [14], Correlograph [15], Linear Associator [16] and Hopfield [17]) and Morphological associative memories [18] lies in the operational bases of the latter, which are the morphological operations: dilation and erosion. This model broke out of the traditional mold of classic memories which use conventional operations for vectors and matrices in learning phase and sum of multiplications for recovering patterns. Morphological associative memories change products to sums and sums to maximum or minimum in both phases.

The basic computations occurring in the proposed morphological network are based on the algebraic lattice structure (R, ∨, ∧, +), where the symbols ∨ and ∧ denote the binary operations of maximum and minimum, respectively. Using the lattice structure (R, ∨, ∧, +), for an m x n matrix A and a p x n matrix B with entries from R, the matrix product C = A ∇ B, also called the max product of A and B, is defined by equation (1).

$$c_{ij} = \bigvee_{k=1}^{p} a_{ik} + b_{kj} = \left(a_{i1} + b_{1j}\right) \vee \cdots \vee \left(a_{ip} + b_{pj}\right) \qquad \textbf{(1)}$$

The *min product* of A and B induced by the lattice structure is defined in a similar fashion. Specifically, the *i,j*th entry of $C = A \,\Delta\, B$ is given by equation (2).

$$c_{ij} = \bigwedge_{k=1}^{p} a_{ik} + b_{kj} = \left(a_{i1} + b_{1j}\right) \wedge \cdots \wedge \left(a_{ip} + b_{pj}\right) \qquad \textbf{(2)}$$

Suppose we are given a vector pair $\mathbf{x} = (x_1, x_2, \ldots, x_n)^t$ and $\mathbf{y} = (y_1, y_2, \ldots, y_n)^t \in \mathbf{R}^m$. An associative morphological memory that will recall the vector when presented the vector is showed in equation (3)

$$W = y \nabla (-x)^t = \begin{bmatrix} y_1 - x_1 & \cdots & y_1 - x_n \\ \vdots & \ddots & \vdots \\ y_m - x_1 & \cdots & y_m - x_n \end{bmatrix} \qquad \textbf{(3)}$$

Since W satisfies the equation $W \,\Delta\, \mathbf{x} = \mathbf{y}$ as can be verified by the simple computation in equation (4)

$$W \nabla x = \begin{bmatrix} \bigvee_{i=1}^{n} (y_1 - x_i + x_i) \\ \vdots \\ \bigvee_{i=1}^{n} (y_m - x_i + x_i) \end{bmatrix} = y \tag{4}$$

Henceforth, let ($\mathbf{x}^1$, $\mathbf{y}^1$), ($\mathbf{x}^2$, $\mathbf{y}^2$), …, ($\mathbf{x}^p$, $\mathbf{y}^p$) be p vector pairs with $x^k = (x_1^k, x_2^k, \ldots, x_n^k)^t \in \mathbf{R}^n$ and $y^k = (y_1^k, y_2^k, \ldots, y_m^k)^t \in \mathbf{R}^m$ for k = 1, 2, …, p. For a given set of pattern associations {($\mathbf{x}^k$, $\mathbf{y}^k$) | k= 1, 2, …, p} we define a pair of associated pattern matrices (X, Y), where X=($\mathbf{x}^1$, $\mathbf{x}^2$, …, $\mathbf{x}^p$) and Y=($\mathbf{y}^1$, $\mathbf{y}^2$, …, $\mathbf{y}^p$). Thus, X is of dimension n x p with i,jth entry x_i^j and Y is of dimension m x p with i,jth entry y_i^j. Since $\mathbf{y}^k \nabla (-\mathbf{x}^k)^t = \mathbf{y}^k \Delta (-\mathbf{x}^k)^t$, the notational burden is reduced by denoting these identical morphological outer vector products by $\mathbf{y}^k \times (-\mathbf{x}^k)^t$. With each pair of matrices (X, Y) we associate two natural morphological m x n memories M and W defined by

$$M = \bigvee_{k=1}^{p} \left(y^k \otimes (-x^k)^t \right) \tag{5}$$

$$W = \bigwedge_{k=1}^{p} \left(y^k \otimes (-x^k)^t \right) \tag{6}$$

With these definitions, we present the algorithms for the training and recalling phase.

Training Phase

1. For each p association ($\mathbf{x}^\mu$, $\mathbf{y}^\mu$), the minimum product is used to build the matrix $\mathbf{y}^\mu \Delta (-\mathbf{x}^\mu)^t$ of dimensions m x n, where the input transposed negative pattern $\mathbf{x}^\mu$ is defined as $(-\mathbf{x}^\mu)^t = (-x_1^\mu, -x_2^\mu, \ldots, -x_n^\mu)$.
2. The maximum and minimum operators ($\vee$ and $\wedge$) are applied to the p matrices to obtain M and W memories as equations (5) and (6) show.

Recognizing phase

In this phase, the minimum and maximum product, Δ and ∇, are applied between memories M or W and input pattern $\mathbf{x}^\omega$, where $\omega \in \{1, 2, ..., p\}$, to obtain the column vector $\mathbf{y}$ of dimension m as equations (7) and (8) shows:

$$\mathbf{y} = M \,\Delta\, \mathbf{x}^\omega \tag{7}$$

$$\mathbf{y} = W \,\nabla\, \mathbf{x}^\omega \tag{8}$$

Now, we present an illustrative example for learning and recognizing phases of a Morphological associative memory.

Suppose we want to associate a set of three pairs of patterns, then $p = 3$. The cardinality of **x** and **y** will be $n = 3$ and $m = 4$, respectively. The three pairs of patterns are:

$$x^1 = \begin{pmatrix} -255 \\ 0 \\ 0 \end{pmatrix} \rightarrow y^1 = \begin{pmatrix} 255 \\ 255 \\ 255 \\ 255 \end{pmatrix}, x^2 = \begin{pmatrix} 0 \\ -255 \\ 0 \end{pmatrix} \rightarrow y^2 = \begin{pmatrix} 155 \\ 128 \\ 0 \\ 0 \end{pmatrix}$$

$$x^3 = \begin{pmatrix} 0 \\ 0 \\ -255 \end{pmatrix} \rightarrow y^3 = \begin{pmatrix} 255 \\ 255 \\ 255 \\ 0 \end{pmatrix}$$

Now, we apply the first step of training phase for associating the pair number one

$$y^1 \times (-x^1)^t = \begin{bmatrix} 255 \\ 255 \\ 255 \\ 255 \end{bmatrix} \times -[-255\ 0\ 0] = \begin{bmatrix} 255+255 & 255-0 & 255-0 \\ 255+255 & 255-0 & 255-0 \\ 255+255 & 255-0 & 255-0 \\ 255+255 & 255-0 & 255-0 \end{bmatrix} =$$

$$= \begin{bmatrix} 510 & 255 & 255 \\ 510 & 255 & 255 \\ 510 & 255 & 255 \\ 510 & 255 & 255 \end{bmatrix}$$

The same process is performed at remain pairs of patterns, and then the maximum of each element of every matrix is obtained as follows:

$$M = \begin{bmatrix} 510 & 255 & 255 \\ 510 & 255 & 255 \\ 510 & 255 & 255 \\ 510 & 255 & 255 \end{bmatrix} \bigvee \begin{bmatrix} 155 & 410 & 155 \\ 128 & 383 & 128 \\ 0 & 255 & 0 \\ 0 & 255 & 0 \end{bmatrix} \bigvee$$

$$\bigvee \begin{bmatrix} 255 & 255 & 510 \\ 255 & 255 & 510 \\ 255 & 255 & 510 \\ 0 & 0 & 255 \end{bmatrix} = \begin{bmatrix} 510 & 410 & 510 \\ 510 & 383 & 510 \\ 510 & 255 & 510 \\ 510 & 255 & 255 \end{bmatrix}$$

Now, we present the first input vector to the *max*-type morphological associative memory

$$M \Delta x^1 = \begin{bmatrix} 510 & 410 & 510 \\ 510 & 383 & 510 \\ 510 & 255 & 510 \\ 510 & 255 & 255 \end{bmatrix} \Delta \begin{bmatrix} -255 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 510+(-255) \wedge 410+0 \wedge 510+0 \\ 510+(-255) \wedge 383+0 \wedge 510+0 \\ 510+(-255) \wedge 255+0 \wedge 510+0 \\ 510+(-255) \wedge 255+0 \wedge 255+0 \end{bmatrix}$$

$$M \Delta x^1 = \begin{bmatrix} 255 \\ 255 \\ 255 \\ 255 \end{bmatrix} = y^1$$

When we present the other two input patterns (x^2 and x^3) to the memory, we recall their corresponding output patterns (y^2 and y^3).

The process of this illustrative example will be used in the following section for explaining our proposal.

3 Proposed Model

3.1 Encryption Algorithm

In order to describe both algorithms: encryption and decryption, we will present an illustrative example.

Suppose the message:

Help me! I am in danger, please

In the first step, groups of 9 characters are formed as follows,

Help me!
I am in d
anger, pl
*ease*____

If the number of characters is not multiple of 9, then we have to add spaces.

Then, each character of the message is converted to the ASCII code, and we have:

$$y^1 = [72\ 101\ 108\ 112\ 32\ 109\ 101\ 33\ 32]$$

$$y^2 = [73\ 32\ 97\ 109\ 32\ 105\ 110\ 32\ 100]$$

$$y^3 = [97\ 110\ 103\ 101\ 114\ 44\ 32\ 112\ 108]$$

$$y^4 = [101\ 97\ 115\ 101\ 32\ 32\ 32\ 32\ 32]$$

These vectors of dimension 9 represent the output patterns. Now, we have to build the input patters, which are the private key. As we have four vectors, the four input vectors have a dimension of 4, and they are built as follows,

$$x^1 = [-300\ 0\ 0\ 0]$$

$$x^2 = [0\ -300\ 0\ 0]$$

$$x^3 = [0\ 0\ -300\ 0]$$

$$x^4 = [0\ 0\ 0\ -300]$$

With these pairs of patterns we apply the training phase for building *max* morphological associative memory. The first pair is associated,

$$y^1 \times (-x^1)^t = \begin{bmatrix} 72 \\ 101 \\ 108 \\ 112 \\ 32 \\ 109 \\ 101 \\ 33 \\ 32 \end{bmatrix} \times -[-300\ 0\ 0\ 0] = \begin{bmatrix} 72+300 & 72-0 & 72-0 & 72-0 \\ 101+300 & 101-0 & 101-0 & 101-0 \\ 108+300 & 108-0 & 108-0 & 108-0 \\ 112+300 & 112-0 & 112-0 & 112-0 \\ 32+300 & 32-0 & 32-0 & 32-0 \\ 109+300 & 109-0 & 109-0 & 109-0 \\ 101+300 & 101-0 & 101-0 & 101-0 \\ 33+300 & 33-0 & 33-0 & 33-0 \\ 32+300 & 32-0 & 32-0 & 32-0 \end{bmatrix} =$$

$$= \begin{bmatrix} 372 & 72 & 72 & 72 \\ 401 & 101 & 101 & 101 \\ 408 & 108 & 108 & 108 \\ 412 & 112 & 112 & 112 \\ 332 & 32 & 32 & 32 \\ 409 & 109 & 109 & 109 \\ 401 & 101 & 101 & 101 \\ 333 & 33 & 33 & 33 \\ 332 & 32 & 32 & 32 \end{bmatrix}$$

Remain pairs are associated as in the previous process. The built *max* associative memory is showed in Figure 3.

$$M = \begin{bmatrix} 372 & 373 & 397 & 401 \\ 401 & 332 & 410 & 397 \\ 408 & 397 & 403 & 415 \\ 412 & 409 & 401 & 401 \\ 332 & 332 & 414 & 332 \\ 409 & 405 & 344 & 332 \\ 401 & 410 & 332 & 332 \\ 333 & 332 & 412 & 332 \\ 332 & 400 & 408 & 332 \end{bmatrix}$$

Fig. 3. Associative Memory representing the encrypted text

This memory represents the encryption of the message. We obtain the greatest number from the matrix, in this case 414, and we calculate 415 – 255 = 160. We subtract 160 from all the elements in the matrix then we have the results showed by the Figure 4.

$$M = \begin{bmatrix} 212 & 213 & 237 & 241 \\ 241 & 172 & 250 & 237 \\ 248 & 237 & 243 & 255 \\ 252 & 249 & 241 & 241 \\ 172 & 172 & 254 & 172 \\ 249 & 245 & 184 & 172 \\ 249 & 245 & 184 & 172 \\ 173 & 172 & 252 & 172 \\ 172 & 240 & 248 & 172 \end{bmatrix}$$

Fig. 4. Resulting matrix when we subtract 160 from all the elements of the matrix in Figure 3.

The idea of the subtraction is to show the memory as an image, as Figure 3 shows.

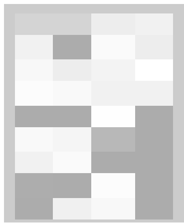


Fig. 5. Image representing the memory

At the first sight, we can imagine that the image is an encryption from another image.

If we perform a cryptanalysis, we will do it to obtain the original image without thinking that is a text.

This is the main advantage of our proposal.

Now, we will describe the decryption algorithm.

3.2 Decryption Algorithm

We recover the original text from the last matrix.

First, we add 160 to all the elements to the matrix in Figure 4. The result will be the matrix in Figure 3.

Now, we generate the input vectors or the private key.

These patterns are presented to the morphological associative memory, as follows,

$$M\Delta x^1 = \begin{bmatrix} 372 & 373 & 397 & 401 \\ 401 & 332 & 410 & 397 \\ 408 & 397 & 403 & 415 \\ 412 & 409 & 401 & 401 \\ 332 & 332 & 414 & 332 \\ 409 & 405 & 344 & 332 \\ 401 & 410 & 332 & 332 \\ 333 & 332 & 412 & 332 \\ 332 & 400 & 408 & 332 \end{bmatrix} \Delta \begin{bmatrix} -300 \\ 0 \\ 0 \\ 0 \end{bmatrix} = \begin{bmatrix} 372+(-300) \wedge 373+0 \wedge 397+0 & 401+0 \\ 401+(-300) \wedge 332+0 \wedge 410+0 & 397+0 \\ 408+(-300) \wedge 397+0 \wedge 403+0 & 415+0 \\ 412+(-300) \wedge 409+0 \wedge 401+0 & 401+0 \\ 332+(-300) \wedge 332+0 \wedge 414+0 & 332+0 \\ 409+(-300) \wedge 405+0 \wedge 344+0 & 332+0 \\ 401+(-300) \wedge 410+0 \wedge 332+0 & 332+0 \\ 333+(-300) \wedge 332+0 \wedge 412+0 & 332+0 \\ 332+(-300) \wedge 400+0 \wedge 408+0 & 332+0 \end{bmatrix}$$

$$M\Delta x^1 = \begin{bmatrix} 72 \\ 101 \\ 108 \\ 112 \\ 32 \\ 109 \\ 101 \\ 33 \\ 32 \end{bmatrix} = y^1$$

We perform the same process with the rest of the patterns, and we obtain the original output vectors.

$$y^1 = [72\ 101\ 108\ 112\ 32\ 109\ 101\ 33\ 32]$$
$$y^2 = [73\ 32\ 97\ 109\ 32\ 105\ 110\ 32\ 100]$$
$$y^3 = [97\ 110\ 103\ 101\ 114\ 44\ 32\ 112\ 108]$$
$$y^4 = [101\ 97\ 115\ 101\ 32\ 32\ 32\ 32\ 32]$$

Each element of the patterns are converted to its corresponding ASCII code, then all the patterns are concatenated and, finally we have the original text.

4 Experiments and Results

We perform 1000 tests with different messages, length of the messages and values in the private key.

The Table 1 shows three examples of messages with different values of private key and lengths.

Table 1. Examples of messages with different lengths

Original message	Value of private key	Recovered message
Help me! I'm in danger, please (30 characters)	-10	felp[mf[cI_m?in?ddngerb[pleaseWi[Wi[
	-20	\elpQmeQYIUm5in5dangerXQpleaseM_QM_Q
	-50	Help3me3;I7m in danger:3please/A3/A3
	-80	Help me! I'm in danger, please # #
	-82	Help me! I'm in danger, please ! !
	-83	Help me! I'm in danger, please
	-84	Help me! I'm in danger, please
	-90	Help me! I'm in danger, please
	-100	Help me! I'm in danger, please
	-300	Help me! I'm in danger, please
	-500	Help me! I'm in danger, please
	-1000	Help me! I'm in danger, please
Then there was the bad weather. It would come in one day when the fall was over. We would have to shut the windows in the night against the rain and the cold wind would strip the leaves from the trees in the Place Contrescarpe. (227 characters)	-10	*Not recovered*
	-20	*Not recovered*
	-50	*Not recovered*
	-80	*Not recovered*
	-87	*Not recovered*
	-88	*Recovered*
	-89	*Recovered*
	-90	*Recovered*
	-100	*Recovered*
	-300	*Recovered*
	-500	*Recovered*
	-1000	*Recovered*
4090 characters, a sheet approximately	-255	*Recovered*

From Table 1, we can observe that the value of the private key must be higher when the number of characters is increased to assure the original message be recovered. We can see that for some values of private key the recovering fails.

5 Conclusions

Associative models have been applied in classification, prediction, pattern recognition and feature selection. In this paper we demonstrated that this approach can also be applied in text encryption.

The algorithm is symmetric and it has a private key.

The original massage is recovered always: the number of characters is not a limitation; however, the value of the private key must be increased if the number of characters in the message is increased too.

Acknowledgments. The authors would like to thank the Instituto Politécnico Nacional (COFAA, SIP and EDI), and SNI for their economic support to develop this work. Ángel Martínez is grantee PIFI.

References

1. Stanoyevitch, A.: Introduction to Cryptography with mathematical foundations and computer implementations. CRC Press, pp. 1−2 (2011)
2. McAndrew, A.: Introduction to Cryptography with open-source software. CRC Press, pp. 4−5 (2011)
3. Ayushi: A Symmetric Key Cryptographic Algorithm. International Journal of Computer Applications (0975−8887) vol. 1(15) (2010)
4. Solanki,K. H., Patel, C. R.: New Symmetric Key Cryptographic algorithm for Enhancing Security of Data. International Journal Of Research In Computer Engineering And Electronics. vol. 1(3) (Dec 2012)
5. Mathur, A.: A Research paper: An ASCII value based data encryption algorithm and its comparison with other symmetric data encryption algorithms. International Journal on Computer Science and Engineering (IJCSE), vol. 4(9), Sep 2012, pp.1650−1657 (2012)
6. Kiran Kumar M., Mukthyar Azam, Shaik Rasool, S.: Efficient Digital Encryption Algorithm Based On Matrix Crambling Technique. International Journal of Network Security & Its Applications (IJNSA), vol. 2(4), October 2010, pp. 30−41 (2010)
7. Ramesh G., Umarani, R.: A Novel Symmetrical Encryption Algorithm with High Security Based On Key Updating. International Journal of Communication Engineering Applications-IJCEA, vol. 2(5), November-December 2011, pp. 329−341 (2011)
8. Buba, Z. P., Wajiga, G. M.: Cryptographic Algorithms for Secure Data Communication. International Journal of Computer Science and Security (IJCSS), vol. 5(2), pp. 227−243 (2011)
9. Hameed, S., Riaz, F., Moghal, R., Akhtar, G., Ahmed, A., Dar, A. G.: Modified Advanced Encryption Standard For Text And Images. Computer Science Journal vol. 1(3), December 2011, pp. 120−129 (2011)

10. Agrawal, M. and Mishra, P.: A Modified Approach for Symmetric Key Cryptography Based on Blowfish Algorithm. International Journal of Engineering and Advanced Technology (IJEAT), vol. 1(6), August 2012, pp. 79−83 (2012)
11. Kruti, S., Gambhava, B.: New Approach of Data Encryption Standard Algorithm. International Journal of Soft Computing and Engineering (IJSCE), vol. 2(1), March 2012, 322−325 (2012)
12. Jamgekar, R. S., Joshi, G. S.: File Encryption and Decryption Using Secure RSA, International Journal of Emerging Science and Engineering (IJESE), 1(4), February 2013, pp. 11-14 (2013)
13. Abusukhon, A., Talib and Ottoum, M. I.: Secure Network Communication Based on Text-to-Image Encryption. International Journal of Cyber-Security and Digital Forensics (IJCSDF), vol. 1(4), pp. 263−271 (2012)
14. Steinbuch, K.: Die Lernmatrix, Kybernetik. vol. 1(1), pp. 36-45 (1961)
15. Willshaw, D., Buneman, O. and Longuet-Higgins, H.: Non-holographic associative memory. Nature, vol. 222, pp. 960−962 (1969)
16. Anderson, J. A.: A simple neural network generating an interactive memory. Mathematical Biosciences, vol. 14, pp. 197−220 (1972)
17. Hopfield, J.J.: Neural networks and physical systems with emergent collective computational abilities. Proceedings of the National Academy of Sciences, vol. 79, pp. 2554−2558 (1982)
18. Ritter, G., Sussner, X. P. and Diaz de León, J. L.: Morphological Associative Memories. IEEE Transactions on Neural Networks vol. 9, pp. 281−293 (1998)

Business-Media Analysis for Information Extraction

Ekaterina Pronoza, Elena Yagunova

Saint-Petersburg State University, Saint-Petersburg, Russian Federation

{katpronoza,iagounova.elena}@gmail.com

Abstract. In this paper an approach to primary business-media analysis for further information extraction is proposed. We consider business events representation by looking into part of speech (POS) distribution across tagged n-grams. Two Russian business-media corpora, Russian Business Consulting (RBC) and Commersant, are analyzed, and it is shown that they differ not only in style or themes coverage but also in the range of contexts for the words which mark business-events. Purchase, merger and ownership events are given a closer look at, and it is shown that they are mostly represented by noun phrases in both corpora rather than verbal phrases.

Keywords: Information extraction, business media.

1 Introduction

This paper considers the primary stage of news corpus analysis as part of business events extraction. This stage is closely connected to collocation analysis and aims at extracting key named entities and concepts hierarchy as well as investigating business events' representation. The former is a source of information for the local semantic dictionary whereas the latter may suggest specific ways of further business events extraction including part-of-speech patterns construction.

In this paper, two Russian business-media corpora are analyzed and it is shown that they both reveal noun phrases domination in business-events representation.

These Russian business-media sources demonstrate difficulties in Information Extraction (IE) imposed by the language (e.g., free word order characterizes Russian in general and especially in the media and in language specific domains). Our corpora analysis is very important for event extraction rules development taking into account the domain and stylistic features of the sources in question.

2 Related Work

In [5] an approach based on using collocation statistics for information extraction is proposed. Recognition rules are automatically extracted from the collocation database, and collocational context of words (co-occurrences) is treated as features for unknown proper names classification.

According to the approach proposed in [7], filtering techniques are applied to collocation sets as part of concepts extraction from text corpora. An approach for semantic network construction from the knowledge found in text corpora is presented.

In [10] IE algorithm based on local contextual information is presented. They propose a method of recognizing major constituents of a text as the most relevant collocational expressions and an algorithm which models relevant facts extraction.

A system of crime-related IE for the Arabic language is presented in [1]. Collocation analysis is used to obtain the concordance of the key words and also as part of further local syntactic analysis to define the type of crime, nationality, location, etc (indicator nodes) and supply data for the local grammar.

In [8] it is emphasized that for fully unsupervised event extraction, extensive linguistic analysis which increases the importance of text genre style and topic specification for IE (see also [11]) is essential. The idea of statistical comparison between text types and genres goes back at least as far as [2]. In [12] the linguistic cues indicating uncertainty of events in three genres (not only news) are studied; significant difference in lexical usage across genres is demonstrated.

The approach closest to ours is proposed in [11], where event representation across genre is considered. “Subject-verb-object” pattern statistics is analyzed and it is demonstrated that such statistics can differ across different genre/text types because event structure is strongly related to the genre of the corpus.

In this paper we also consider business events representation across the news corpora but in our approach n-grams, collocations (n-grams extracted by an associate measure) and n-gram part of speech (POS) statistics is used. Unlike [11], we do not propose any patterns (like S-V-O triplets) beforehand – on the contrary, the statistics obtained is used for patterns development. Moreover, our approach demands neither syntactic parser nor any special pattern-mining tool.

3 Data

The data used in the experiments consists of two business-media corpora: “РосБизнесКонсалтинг” (Russian Business Consulting, RBC) of 9 905 342 tokens (including 8 316 573 words) and “КоммерсантЪ” (Commersant) of 30 187 316 tokens (24 718 590 words). Both corpora include all the news articles released by these periodicals in 2011.

They were chosen for being main business-media representatives in Russia. We assume that RBC offers short-spoken businesslike news coverage while Commersant has more freedom of style. They also differ thematically: RBC specializes in financial and economic news and in Commersant there is usually a much broader range of events reported (including general political issues, etc.). Therefore it can also be an important task to compare the forms of business events representation in these media.

4 Methods and Instruments

4.1 Methods, Instruments and Main Assumptions

As part of primary analysis the corpora in question were tokenized and then lemmatized using Aot.ru (AOT)[1] morphology tools. Morphological disambiguation problem was solved by adopting the most frequent POS suggested by Aot.ru. Then n-gram frequencies were extracted.

Logarithmized Dice coefficient was chosen to be the measure for collocation extraction [4, 6, 9]:

$$\mathrm{Dice}(\mathrm{x},\mathrm{y}) = \log_2 \frac{2f(xy)}{f(x)+f(y)}, \tag{1}$$

where $f(\mathrm{x}), f(y)$ refer to the frequencies of the words x and y and $f(xy)$ refers to the frequency of xy word combination. Dice coefficient is considered to be an adequate measure for our problem as it evaluates association degree between collocates based on compatibility and co-occurrence constraints. Moreover, this measure provides robustness of results for both large and small corpora, as shown in [4].

We only analyzed n-grams which consisted of two words not separated by a comma. A lexeme (and not word form) was chosen to constitute unit of analysis because it was important to consider all possible syntactic and semantic roles n-grams play in the sentence. It was also necessary to solve the disambiguation problem which would be unavoidable if we did not take parts of speech (POS feature) into account. For example, the normal form “*продать*” /sell/ is offered by AOT for the participle “*проданный*” /sold/ or “*продающий*” /selling/, the verb “*продать*” /sell/, the infinitive “*продать*” /to sell/ and the gerund “*продав*” /having sold/, and it is important to divide these four cases as they may indicate the form of business events representation. POS features adopted in this paper are noun, adjective, noun pronoun, verb, participle, gerund, infinitive, predicative pronoun, adjective pronoun, cardinal numeral, ordinal numeral, adverb, predicative, preposition, conjunction, interjection, particle, parenthesis and adjective and participle short forms.

We consider Dice coefficient extension for trigrams:

$$\mathrm{Dice}(\mathrm{x},\mathrm{y},\mathrm{z}) = \log_2 \frac{3f(xyz)}{f(x)+f(y)+f(z)}, \tag{2}$$

where $f(\mathrm{x}), f(y)$ and $f(z)$ refer to the frequencies of the words x, y and z and $f(xyz)$ refers to the frequency of xyz word combination.

[1] An open source project providing tools for morphological, syntactic and semantic analysis for the Russian language: `http://aot.ru`.

The absolute frequency threshold of both bigrams and trigrams is chosen to be equal to 4 as we consider it to be a reasonable value for our task when corpora of several millions words are concerned. Now and later both “bigram” and “collocation” terms are used to refer to bigrams with the only difference that when calculating POS distribution for bigrams we sum frequency values (i.e., the number of occurrences in the corpus) when grouping them by a POS bigram (e.g., <Noun Noun>) while for collocations POS distribution we do not take the number of occurrences of a particular bigram into account which is equivalent to their frequency values being equal to 1.

4.2 Goal and Main Phases of Analysis

Our goal is primary business-media analysis, and the main phases of the media corpora processing may be described as follows:

- n-gram frequencies calculation,
- Dice coefficient calculation for bigrams and trigrams and collocations (extracted by Dice coefficient) analysis,
- tag words list construction for the business events (see Table 1),
- automatic business events tagging (for layoff, appointment and bankruptcy events),
- manual business events tagging (for the rest of the events),
- n-gram statistics calculation for the analyzed business events (including part of speech distribution).

The business events to be extracted from media texts are

- the purchase of assets, shares, etc. made by some company (PURCHASE tag),
- companies merger (MERGER tag),
- ownership of companies (OWNERSHIP tag),
- appointments, retirements or taking up a post (POST tag),
- employee layoff (LAYOFF tag),
- contract signing (CONTRACT tag),
- business investments (INVESTMENT tag),
- bankruptcy (BANKRUPTCY tag).

The tags given in the brackets will be adopted to refer to the event in this paper. In fact, these tags will be used in further IE as database table headers while fulfilling scenario templates.

Let us assume a tag word for a business event to be a lexeme (actually, a collocate) which describes this business event in one way or another. For example, a noun “*продавец*” /*seller*/ and a verb “*продать*” /*sell*/ would be tag words for the PURCHASE business event, while a noun “*банкрот*” /*bankrupt*/ and a verb “*обанкротиться*” /*go bankrupt*/ would be tag words for the BANKRUPTCY business event. It is easy to see that tag words for PURCHASE do not necessarily indicate a purchase made by a company while tag words for BANKRUPTCY describe exactly the event we are analyzing. Therefore our business events can be divided into two classes according to the feature of their tag words described below (for automatically or manually business events detection).

For the events like LAYOFF, POST and BANKRUPTCY tag words always describe the events we are interested in and it is possible to mark all the bigrams which include one of these words with business events tags automatically.

On the other hand, we have business events like PURCHASE, MERGER, OWNERSHIP, CONTRACT and INVESTMENT, and for these events their respectful tag words do not always specify the event we need. Therefore we should look though the list of bigrams containing these tag words and mark only those which indeed refer to the event we are interested in, and this cannot be done automatically. Some of these tag words are outlined in Table 1.

Table 1. Tag words for the business events processed manually

EVENT	TAG WORDS
PURCHASE	Купить /buy, purchase/; покупка /a purchase/; продавать /sell/; продажа /sale/
INVESTMENT	Инвестирование /investment/; инвестировать /invest/; инвестор /investor/
CONTRACT	Договор /contract/; контракт /contract/; подписать /sign/; сделка /bargain/
OWNERSHIP	Главный /chief/; дочерний /subsidiary/; филиал /branch/
MERGER	Объединение /union/; объединяться /unite/; поглощение /takeover/; слиться /merge/

4.3 Approach Restrictions

Trigrams are not considered with respect to the business events because a trigram collocation is usually a bigram collocation extension, and adding the third collocate to a bigram does not change the way it represents a particular business event. For example, a noun phrase "*покупка компания*" /*company purchase*/ cannot become a verbal phrase when a verb is added to the bigram as its action is represented by a verbal noun. It is an example of predicative bigram without any verbal component.

Furthermore, we do not take non-contiguous bigrams into account and therefore are unable to choose MWEs for tag words.

As for business events representation, we consider three following events: PURCHASE, MERGER and OWNERSHIP. INVESTMENT and CONTRACT which are manually marked with tag words are not included into the analysis part because they demonstrate too much ambiguity (e.g., a contract is not necessarily the contract between two companies, and the word "*investment*" can be used in metaphoric sense).

Business events for which bigrams are marked automatically are not considered in this paper because their representation simply depends only on the list of tag words we supply the classification program with. For example, for BANKRUPTCY, such tag words are "*банкрот*" /*bankrupt*/, "*обанкротиться*"/*go bankrupt*/, "*разориться*"

/*go bankrupt*/, and for POST event they are "*должность*" /*post*/, "*пост*" /*post*/, "*отставка*" /*retirement*/.

5 N-grams and Collocations

5.1 Bigrams and Trigrams. General Characteristics

Top bigram collocations of the RBC corpus sorted in the descending order by Dice coefficient and by their frequency (with frequency threshold value equal to 41) were analyzed (all in all, there are 16698 bigrams).

Top bigram collocations (a few hundred) of this list refer to the three classes of multiword expressions (MWE):

− a Proper Name person (or a part of it), like *Усама Бен* /*Osama Bin*/, *Пан Ги* /*Ban Ki*/, *Broco Алексей* /*Broco Alexey*/, *Фог Расмуссен* /*Fogh Rasmussen*/, etc.;
− a Proper Name company (organization), e.g., *Middle East*, *IPE Brent*, *Церих Кэпитал* /*Zerich Capital*/;
− a media cliché and compound function words, like "*сослаться на*" /*refer to*/, "*применительно к*" /*with reference to*/, "*несмотря на*" /*despite*/, etc.

Top trigram collocations for the RBC show the same tendency. Some of the compound phrases partially represented in the bigram collocations list are given here as full phrases, e.g., *Middle East Crude* and *Церих Кэпитал Менеджмент* /*Zerich Capital Management*/. There is also a political context indication, e.g., in MWE like "*расколоть страна на*" /*split country into*/, "*экс-кандидат в президент*" /*ex-presidential candidate*/ and "*злободневный вопрос российский*" /*burning question russian*/. These MWE also prove that financial and economic news dominate in the RBC corpus (see "*фиксинг по золото*" /*gold fixing*/, "*наличный рынок драгоценный*" /*cash market precious*/ and "*нерыночный актив или*" /*non-market asset or*/).

As in the RBC case, the Commersant's bigram and trigram collocations also constitute the three main types of MWE: a proper Name person, a Proper Name company and a media cliché and compound function words. The only difference from RBC is poorer financial terminology representation together with social and political terms domination (e.g., "*возбудить уголовный дело*" /*launch criminal case*/, "*полномочный представитель президент*" /*presidential plenipotentiary*/).
We would also like to emphasize a specialty of person MWE in both RBC's and Commersant's corpora[2].

---

[2] *Broco Алексей* /*Broco Alexey*/ collocation takes 11^{th} place in top RBC bigrams list (frequency = 154, Dice = 0.99 (out of 1). This is apparently Alexey Matrosov, who is the chief of analytical department of ГК Broco company. At the same time *Алексей Матрос* /Alexey Matrosov/ collocation is less frequent in the corpus (frequency = 122, Dice = -3.04). This tendency can also be illustrated by an example from the Commersant's corpus: we have *Совлинк Ольга* /*Sovlink Olga*/ (Olga Belenkaya, the chief of analytical department of Sovlink company) with frequency = 57 and Dice = 57 and *Ольга Беленький* /*Olga Belenkiy*/ with

Famous politicians' names mostly appear in the top bigram collocations list in the <first name, last name> form, rather than in <last name, post> or <last name, organization/country> combinations. A couple of examples are given in Table 3 (with F1 and D1 for RBC and F2 and D2 for Commersant as frequency and Dice).

Table 2. Some person collocation characteristics

Bigram	F1	D1	F2	D2	Bigram	F1	D1	F2	D2
Барак Обама /Barack Obama/	599	0.44	1381	0.36	*Ангела Меркель /Angela Merkel/*	242	0.39	250	0.26
США Барак /USA Barack/	567	-3.5	487	-3.5	*Германия Ангела /Germany Angela/*	200	-2.4	130	-3.4
президент Барак /president Barack/	80	-6.4	152	-6.8	*ФРГ Ангела /FRG Angela/*	88	-0.7	48	-1.43
президент Обама /president Obama/	< 4	-	320	-5.7	*канцлер Ангела /chancellor Angela/*	27	-2.9	62	-1.4
господин Обама /Mr. Obama/	< 4	-	133	-7.8	*госпожа Меркель /Mrs. Merkel/*	< 4	-	26	-7.13

We can conclude that the famous people are mainly given in the <first name, last name> form, whereas others are represented with reference to a company. This assumption is important for further IE as it shows how POST events should be handled.

5.2 POS Characteristics of Collocations

Since one of the goals of our study is finding out the information necessary for building the patterns for business events extraction system, we consider POS-based statistics. Bigrams POS distribution is estimated (without frequency threshold this time) for both corpora. A list of top 10 POS bigrams is shown in Table 3.

Table 3. POS distribution for RBC and Commersant bigrams

POS1 POS2	RBC		Commersant	
	freq	%	freq	%
Noun Noun	608879	14	1604411	12
Preposition Noun	555211	13	1305335	9.9
Noun Preposition	460552	11	1085355	8.2
Adjective Noun	366324	8.3	826811	6.3
Noun Verb	179283	4.1	444857	3.4
Preposition Adjective	158440	3.6	398119	3
Noun Adjective	148556	3.4	347163	2.6
Verb Preposition	141581	3.2	351963	2.7
Noun Conjunction	117303	2.7	386254	2.9
Verb Noun	101084	2.3	273452	2.1

frequency = 4 and Dice = -8.66 (almost minimal). *Analysis Михаил /Analysis Mikhail/* (Mikhail Korchemkin, the chief director of East European Gas Analysis) and *АвиаПорт Олег /AviaPort Oleg/* (Oleg Panteleev, the chief editor of AviaPort agency) collocations reveal the same tendency.

Thus, <Noun Noun>, <Noun Preposition> and <Preposition Noun> bigrams dominate in both corpora. We shall use non-parametric Mann-Whitney U-test for statistical hypotheses testing as our data is nominal and aggregated (grouped by POS).

RBC and Commersant show similar POS distribution and according to Mann-Whitney U-test the difference is statistically insignificant at the 0.05 level.

In Table 4 top 10 POS collocations with largest Dice value are shown.

Table 4. POS Distribution for RBC and Commersant Collocations

POS1 POS2	RBC		Commersant	
	amount	%	amount	%
Noun Noun	32604	16	116750	17
Adjective Noun	16886	8.3	51624	7.3
Noun Verb	14742	7.2	51718	7.3
Noun Preposition	13322	6.5	35297	5
Preposition Noun	12223	6	31386	4.4
Noun Adjective	11135	5.5	36246	5.1
Verb Noun	6929	3.4	25444	3.6
Preposition Adjective	5550	2.7	14445	2
Noun Conjunction	5144	2.5	16964	2.4
Conjunction Noun	4448	2.2	16107	2.3

According to Mann-Whitney U-test, the difference between RBC and Commersant POS distribution for the collocations is significant at the 0.05 level. In particular, the RBC corpus is more heterogeneous (with respect to collocations POS distribution) than the Commersant’s (with standard deviation value equal to 1.66 against 1.34).

6 Business Events Tagging

As we have previously shown, some tag words can unambiguously define an event. And here we introduce the idea of a lexeme’s context variety. If a lexeme (i.e., a tag word) has small context variety, it is a perfect event classification tool (this is what we have for POST, LAYOFF and BANKRUPTCY), but if its context variety is large, an arbitrary bigram containing this lexeme cannot be automatically attributed to an event. We also introduce context determinacy, which is opposite to context variety: the larger context variety is, the smaller context determinacy we have, and vice versa.

We have calculated the latter as a portion of bigrams containing a given tag word and attributed to a particular event out of all the bigrams containing this tag word. Context determinacy values for the tag words (for the business events we are interested in) are shown in Table 5, with L1, R1 for RBC’s left and right context determinacy, and L2 and R2 for Commersant's context determinacy respectively.

Table 5. Tag words context determinacy

TAG WORD	L1, %	R1, %	L2, %	R2, %	TAG WORD	L1, %	R1, %	L2, %	R2, %
MERGER									
объединить /to unite/	0	9	0	1	соединяться /to unite/	0	31	-	-
сливаться /to merge/	0	9	-	-	присоединение /adjunct/	0	10	0	2
слияние /merger/	33	51	2	19	объединение /union/	-	-	0	3
объединяться /to unite/	100	0	17	0	объединиться /to unite/	-	-	7	0
поглощение /takeover/	0	31	0	20					
OWNERSHIP									
материнский	66	100	4	61	филиал /branch/	62	84	67	74
дочерний /subsidiary/	62	98	15	96	представительство /agency/	3	48	0	21
PURCHASE									
покупка /purchase/	0	15	0	13	купить /to buy/	3	0	5	5
покупать /to buy/	0	39	8	4	продать /sell/	0	3	2	3
продажа /sale/	0	1	0	5	приобрести /to purchase /	2	6	8	5
продавать /to sell/	4	0	2	2	покупатель /purchaser/	-	-	0	1
приобретение /purchase/	0	27	0	17	приобретать /to purchase/	-	-	11	5

Apparently Commersant has a bit smaller context determinacy than RBC (and larger context variety). According to Chi-Square test, the difference between left determinacy value for RBC and Commersant is significant at the 0.001 level.

RBC also reveals larger standard deviation than Commersant for both left and right context (0.096 and 0.109 for RBC against 0.022 and 0.071 for Commersant). It suggests that the Commersant corpus is more homogeneous than the RBC's when business events representation is concerned but, as we only consider three business events, there is not enough data to prove whether this is statistically significant.

7 Business Events Representation

7.1 PURCHASE

As far as the PURCHASE event is concerned, the RBC corpus shows small representation forms variety. Noun phrases (NPs) like <Noun Noun> dominate throughout all frequency levels, e.g., *покупка компания* /*company purchase*/, *приобретение актив* /*asset purchase*/, *приобретение сеть* /*chain purchase*/, etc. The Commersant corpus has wider constructions variety, and <Noun Noun> phrases dominate among the most frequent ones, e.g., *приобретение доля* /*share purchase*/, *приобретение актив* /*asset purchase*/, *покупка ООО* /*OOO purchase*/ (see Table 6).

According to Mann-Whitney U-test, there is no significant difference between the two distributions at the 0.05 level. But if we group all part of speech bigrams into NPs[3] and verbal phrases (VPs)[4] we shall have the total portion of noun phrases equal to 56% against the portion of VPs equal to 44% for RBC and 52% against 48% for Commersant respectively. In other words, noun phrases are slightly more frequent in

[3] <Noun Noun>, <Noun Adjective>, <Participle Noun>, <Noun Participle> and <Noun Participle (short)>

[4] <Infinitive Noun>, <Infinitive Adjective>, <Noun Infinitive>, <Noun Verb>, <Verb Noun> and <Gerund Noun>

both corpora than verbal phrases, and this difference is smaller in Commersant's case which leads to the suggestion that the latter is more "balanced".

Table 6. POS distribution for PURCHASE (RBC, Commersant)

POS1 POS2	Commersant		RBC	
	freq	%	freq	%
Noun Noun	226	35	662	48
Infinitive Adjective	176	27	-	-
Noun Adjective	123	19	9	0.6
Infinitive Noun	78	12	101	7
Noun Verb	26	4.0	321	25
Participle Noun	15	2.3	33	2
Verb Noun	5	0.7	158	11
Noun Infinitive	-	-	42	3
Noun Participle (short)	-	-	13	0.9
Noun Participle	-	-	10	0.7
Gerund Noun	-	-	10	0.7

7.2 MERGER

If we consider MERGER, it is easy to see that in RBC NPs dominate across all frequency levels: *слияние Skype* /*Skype merger*/, *поглощение концерн* /*concern takeover*/, *присоединение дочерний* /*adjunct subsidiary*/, etc. In Commersant NPs are even more frequent: *слияние актив* /*asset merger*/, *объединение Газпром* /Gazprom union/, *слияние ОАО* /*OAO merger*/, etc. All these NPs are predicative structures (with verbal nouns with intact verbal valences). POS distribution of the bigrams for MERGER is given in Table 7.

Table 7. POS distribution for MERGER (RBC, Commersant)

POS1 POS2	RBC		Commersant	
	freq	%	freq	%
Noun Noun	177	42	170	45.5
Adjective Noun	140	33	150	40
Participle Noun	40	10	6	1.6
Noun Adjective	33	8	14	3.7
Noun Numeral	15	4	-	-
Verb Noun	12	3	-	-
Noun Verb	4	1	23	6.2
Infinitive Noun	-	-	11	3

Mann-Whitney U-test shows no significant difference between the two distributions at the 0.05 level. Let us again sum portions of NPs and VPs for both corpora. For RBC corpus we have 96% and 4% of NPs and VPs respectively, and for Commersant the results are 91% and 9%.

This shows that MERGER in about 9 cases out of 10 is represented by NP rather than by VP in both corpora and that Commersant has a little more balance when the difference between the portions of NPs and VPs is concerned for the MERGER event.

7.3 OWNERSHIP

OWNERSHIP representation has a certain specialty: in Russian a relatively small number of verbs are appropriate for its description (e.g., the verb "*удочерить*" /*adopt as a daughter/* is unlikely to be used with respect to a company, and "*представлять компанию*" /*represent a company*/ is referred to the employees and not the subsidiary company). As a result, NPs dominate in both corpora for this event. Some examples presented in Table 8, and in Table 9 OWNERSHIP bigrams POS distribution is given.

Table 8. OWNERSHIP bigrams (examples)

RBC	Commersant
дочерняя компания /subsidiary company/	дочерняя авиакомпания /subsidiary aviacompany/
материнская компания /parent company/	материнская структура /parent structure/
сеть дочерний /subsidiary chain/	дочерний ООО /subsidiary OOO/

Table 9. POS distribution for OWNERSHIP (RBC, Commersant)

POS1 POS2	RBC		Commersant	
	freq	%	freq	%
Adjective Noun	365	52	1022	62
Noun Adjective	126	18	197	12
Noun Noun	124	18	358	22
Adjective Adjective	29	4	43	3
Adjective pron. Adjective	23	3	-	-
Verb Adjective	16	2	4	0.2
Verb Noun	7	1	5	0.3
Preposition Adjective	6	0.9	-	-
Participle Adjective	4	0.6	-	-
Participle Noun	-	-	18	1

As far as the difference between these two distributions is concerned, it is not statistically significant at the 0.05 level according to Mann-Whitney U-test. Summed up portions of NPs and VPs are equal to 97% and 3% for RBC and 99% and 1% for Commersant. The latter shows a wide variety of <Adjective Noun> phrases with the same noun and various adjectives, and this seems to be the reason of the difference. And yet OWNERSHIP is represented almost by NPs only in both corpora.

8 Conclusion and Future Work

Collocation analysis of the RBC and the Commersant corpora shows that these two business media differ in both style and themes. RBC is a finance specialized periodical while Commersant covers a broad range of issues.

There is no statistically significant difference between the two corpora with respect to business events representation. NPs dominate for ***merger*** and ***ownership*** and are slightly more frequent than verbal phrases for ***purchase***. A bigram consisting of two nouns is the most frequent type of ***purchase*** and ***merger*** representation form, while the <Adjective, Noun> pair mostly represents ***ownership*** (for both corpora).

Analysis of tag words context determinacy for ***purchase***, ***merger*** and ***ownership*** reveals that Commersant has significantly wider variety of tag word left context than RBC. In fact, the Commersant corpus is somewhat more balanced with respect to business events representation form.

Our future work plans include development of the Russian business domain ontology (which involves identifying the most significant noun groups and organizing them into lexicon attached to the ontology), identification of the most significant predicate structures (both NPs and VPs) and search patterns implementation (with patterns appropriate for Russian media language) based on them, with NPs prior to VPs (at least for the three business events mentioned above) since, according to the statistics described in the paper, they cover most of business events references in the media corpora.

Acknowledgement. We would like to thank our friends and colleagues, especially Lidia Pivovarova, for their invaluable guidance, support and encouragement.

References

1. Alruily, M.F.: Using Text Mining to Identify Crime. Patterns from Arabic News Report Corpus (2012)
2. Biber, D.: Variation across Speech and Writing. Cambridge University Press (1991)
3. Cowie, J., Wilks, Y.: Information Extraction. In: Dale, R., Moisl, H., Sommers, H. (eds) Handbook of Natural Language Processing, ch. 10, pp. 241−269, Dekker (2000)
4. Daudaravicius, V.: Automatic Identification of Lexical Units. Informatica, 34 (1), pp. 85−91 (2010)
5. Dekang, L.: Using Collocation Statistics in Information Extraction. In: Proceedings of the Seventh Message Understanding Conference (MUC-7) (1998)
6. Dice, L.R.: Measures of the Amount of Ecologic Association between Species, vol. 26 (3), pp. 297−302 (1945)
7. Heyer, G., Quasthoff, U., Wolff, Ch.: Information Extraction from Text Corpora: Using Filters on Collocation Sets. In: Proceedings of the Third Int'l. Conference on Language Resources and Evaluation (LREC 2002), vol. 3, pp. 1103−1107. Las Palmas (2002)
8. Grishman, R.: Structural Linguistics and Unsupervised Information Extraction. In: Automatic Knowledge Base Construction and Web-scale Knowledge Extraction (AKBC-WEKEX 2012), pp. 57−61 (2012)
9. Manning, Ch.D., Schutze, H.: Foundations of Statistical Natural Language Processing. The MIT Press, London (1999)
10. Perrin, P., Petry, F.E.: Extraction and Representation of Contextual Information for Knowledge Discovery in Texts. Information Sciences – Informatics and Computer Science: An International Journal, vol. 15, pp. 125−152 (2003)
11. Pivovarova, L., Huttunen, S., Yangarber, R.: Event Representation across Genre. In: Proceedings of NAACL Workshop on Events: Definition, Coreference and Representation (2013)
12. Szarvas, G., Vincze, V., Farkas, R., Mófra, G., Gurevych, I.: Crossgenre and Cross-Domain Detection of Semantic Uncertainty. Computational Linguistics, 38 (2), pp. 335−367 (2012)

Development of Upper Domain Ontologies for Knowledge Preservation of Unani Medicines

Talha Waheed[1], Ana M. Martinez-Enriquez[2], Sobia Amjad[3], Aslam Muhammad[4]

[1,3,4] Department of CS & E, UET, Lahore, Pakistan
[2] Department of CS, CINVESTAV, D.F., Mexico

{twaheed, maslam}@uet.edu.pk, sobia.amjad@live.com, ammartin@cinvestav.mx

Abstract. Observing the role of traditional medicines in global healthcare, World Health Organization signifies the need to preserve knowledge of this valuable intellectual property which is being lost or inaccessible, as it is either undocumented or local in context. Unani medicines, a 2500 years old system, has been practiced in Asia, is facing the same situation. Little computerization effort has been done so far earlier in this domain. To preserve the knowledge of Unani medicines, a formal semantic structure is required that is machine readable and reusable. This paper defines a conceptual structure of Unani medicines using upper domain ontology that includes core philosophy, diseases, diagnosis, symptoms, drugs, and treatment of patients. Developed in Protégé, designers have no past experience in ontology development; information collected from books and expert interviews. The proposed ontology serves as a backbone in upcoming knowledge management framework of Unani medicines.

Keywords: Tibb Unani, Unani medicines, traditional medicines, domain ontology, knowledge management framework.

1 Introduction

Traditional medicines have been practiced across the globe since ancient times, the practices continue even today. It is commonly believed that they are cheap and have fewer side effects. Despite of their cultural place, recent research indicates that some herbs have considerable medicinal benefits, but hundreds of traditional medicines still need investigation. World Health Organization (WHO) estimated that 70% of the population in Africa and Asia still depends on traditional medicines to fulfill their health care needs [1] and 30% population of the world have to rely solely on traditional medicines [2]. Global market for traditional medicines was estimated at US $ 114 billion in 2015 in a recent report [3].

WHO recognized its importance in its strategy. To promote the use of traditional medicines for health care in member countries, the organization suggested preserving the indigenous knowledge of traditional medicines [1]. Unfortunately most of the knowledge is undocumented, and with the deaths of knowledgeable practitioners this great asset is being lost. Whereas, documented knowledge is in local languages, diverse, non-standard, and mostly inaccessible. If the knowledge of their principles, concepts, formulations and practices is preserved and standardized, it can be validated according to modern scientific principles [4] and ultimately it can be included in mainstream healthcare systems.

Unani medicines (locally known as Tibb-e-Unani in sub-continent) is a modified Greco-Arabic system of traditional medicines. In 2500 years of its practices, it absorbed medical traditions of Greek, Arab, Persian, and Indian regions [5]. It is mainly practiced in Pakistan, Afghanistan, India, Bangladesh, Nepal, Srilanka, Iran, Malaysia, Indonesia, Arab region and countries of central Asia. The field of Unani medicines is rich in its documented heritage. There exist thousands of its books on principles and philosophy (Kulliaat), herbal formulations (Nuskha Jaat), and therapies (Mualijaat). The material is written mostly in Arabic, Persian, Urdu, and Indian languages. Unfortunately, the system of Unani Medicines is still not scientifically well studied at larger scale. There are different kind of complexities involved in this work, the material is multi-lingual, geographically and time wise widespread, terminology, and principals are nonstandard, and lack of governmental support are a few.

Before conducting any scientific study, there is a need to preserve and translate knowledge of Unani medicines into international languages so that global access for this information becomes possible. By now there exists a single effort from Indian government to preserve its knowledge [23], but it has its own limitations. Although efforts to preserve knowledge of other traditional medicinal systems of Korea [6], Japan [7], Thailand [8-11], Africa [12-14] and China [15-22] have already been started. Despite the fact that the philosophy and practices of Unani medicines overlaps with other domains of traditional medicines, it is inherently different from these domains with its unique principles, concepts, and formulations that cannot be represented and processed with already developed systems of similar domains. However, their comparison and integration is possible in future.

For compilation, preservation and sharing of Unani medicines knowledge, a web-services based knowledge management framework is required that stores semantically annotated, multilingual, standard, and reusable knowledge into a central web repository. The framework provides collaborative environment between domain experts, researchers, practitioners, and academicians, and acts as an infrastructure for further computerization of Unani Medicines. The knowledge management framework is based on formal semantic structures of ontologies. For the upcoming framework, an ontology for Unani medicines is being proposed in this paper, up to our best knowledge, this is the first ontology in this domain. The ontology mainly contains upper level concepts of Unani medicines, their properties and relationships with other concepts. Our main objective is to provide a platform for preservation, standardization, and validation of Unani system of medicines according to modern scientific principles and ultimately making it able to be included in mainstream healthcare systems. The proposed ontology will be used in the design and development of different semantic-based AI and expert system applications e.g. for diagnostics of diseases, prescription of treatment, etc.

The document organization is as follows: Section 2 presents the work already done using ontologies in similar domains of traditional medicines. Section 3 elaborates seven phases of ontology modeling technique that is employed in the research along with the description of ontology and a usage scenario.

2 Related Work in Similar Domains

A non-exhaustive survey of some ongoing efforts of ontology modeling in other systems of traditional medicines have been presented below, certain practices such as chiropractic, massage, reiki, reflexology, meditation or yoga are not included.

Jang et al. developed ontology of traditional Korean medicines for symptoms, diseases, and treatments [6]. An effort for Kampo ontology, traditional Japanese medicines, is presented by Arita et al. [7]. Multiple efforts to develop ontology of Thai herbal knowledge can also be seen in [8-11].

For the domain of African traditional medicines, Atemezing and Pavon [12] proposed an ontology, which is further enhanced by Ayimdji et al. [13]. Oladosu et al. designed an ontology about African Traditional Herbs [14]. For the domain of Traditional Chinese medicines (TCM), Chen [15] and Mao [16] proposed large scale domain ontology to overcome the problem of semantic heterogeneity; however, this ontology is not available to the public. TCM is rich in semantic modeling, where multiple efforts on semantic web and ontologies of TCM are available [17-22].

Along with above mentioned efforts Traditional Knowledge Digital Library (TKDL) [23] requires special attention. TKDL is an online knowledge repository developed under the supervision of Indian Government, which contains definitions, principles, drug formulations and concepts of Indian traditional medicines. It includes Unani Medicines along with Ayurvedic, Yoga and Siddha. In TKDL the knowledge of traditional medicines available in local language (Urdu, Persian, Arabic, Hindi, etc.) is translated into 5 international languages (English, French, German, Spanish and Japanese). By 2011 around 2.5 Lac drug formulations have been transcribed in TKDL. A classification system, Traditional Knowledge Resource Classification (TKRC) is based on the International Patent Classification (IPC) structure, is specially evolved for this project.

Traditional Knowledge is transcribed in TKRC symbols by using Unicode, XML and Metadata methodology. Samaddar mentioned [24] that TKDL stores knowledge in a text-based database without formal ontologies. So absence of ontologies or similar formal structure reduces its scope. Second limitation of TKDL is that it is unable to offer web-services, which are required for integration with other software applications. Third major issue is that in TKDL keyword based manual queries are allowed, but semantic queries are not possible. So in the present form TKDL does not support semantic web, which requires ontologies to structure data for extendibility, reusability and automatic machine processing. To enable TKDL for semantic web, formal ontologies, web-services and sematic query support is required that serves as a base for Unani medicines computerization.

3 Ontology Modeling for Unani Medicines

Ontologies became popular for organizing and sharing knowledge in the last decade. Whereas "Ontology is a formal, explicit specification of a shared conceptualization" [25]. Ontology should provide descriptions for classes (or things) in the domain of interest, relationship among class members and the attributes that these classes should possess. Huang mentioned that construction of a domain model or ontology is an important part in the development of a knowledge-based system [26]. Depending upon the application of the ontology, different kinds of ontologies exist. Domain ontology is one of the kinds that describe the concepts pertaining to a specific domain.

Developing ontology is a complex task that requires a high degree of analytical and abstract thinking. Therefore, there exist several ontology development methods, each of them with their own objectives depending on the specific needs of application. The method employed in our development is suggested by Noy and McGuiness, it helps beginners in creating their first ontology [28]. In this method, there are following seven phases of research task for the construction of ontology:

3.1 Determine the Domain and Scope of the Ontology

In the first phase of modeling, domain and scope of the ontology have been clearly defined. Noy and McGuinness raises some basic questions in their methodology [28] that clarifies the objectives of the ontology and limits its scope. So domain of our ontology is Unani medicines, and it is proposed for preservation and standardization of knowledge. At present, it only covers upper level domain knowledge of Unani medicines that includes core principles, symptoms, diagnoses, patient, treatment and, disease. It helps researchers, academicians and Unani medicines practitioners (locally known as hakeem) to share their common understanding of the domain. For software developers, it also provides semantic structure and knowledge base thru web-services interface for writing different computing applications.

3.2 Consider Reusing Existing Ontologies (if any)

Up to our best knowledge, there exists no relevant ontology that can be reused in our case so proposed Unani medicines ontology is initially being developed from scratch. However, in future it can be compared and linked with existing ontologies of other traditional medicines domain, so integration and reuse of their software applications and data become possible.

3.3 Enumerate Important Terms in the Ontology

This is the phase where important terms, concepts and properties of the domain ontology are defined that required explanations. Expert interviews, literature reading, observations, and questionnaires helps in listing these terms. For enumerating the terms, explicit knowledge of domain is captured from Unani medicines source books and written material and tacit knowledge is gathered by interviewing domain experts and local practitioners. Outcome of this stage for our proposed ontology resulted core terms as Family History, Drugs, Diseases, Examination, Signs and Symptoms, Element, Organs, Temperaments, Humours, Pneumas, Limitations, etc.

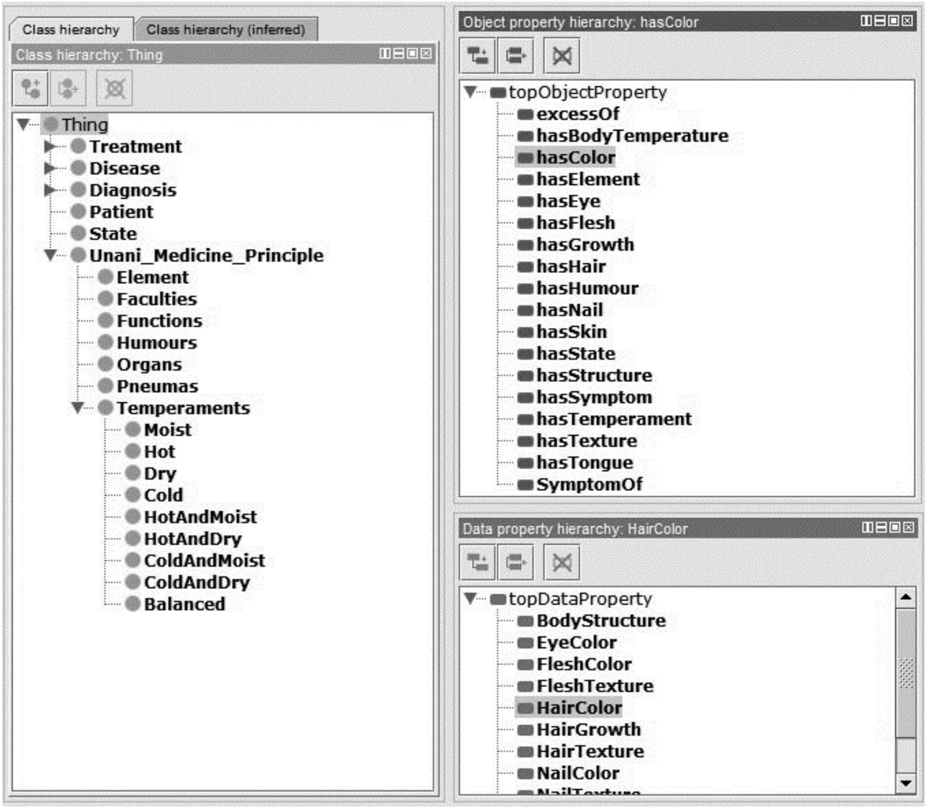


Fig. 1. Class Hierarchy, Object and Data properties of proposed ontology

3.4 Define Class and Class Hierarchy

In the fourth phase, classes (a concept in the domain) and their super and sub-class hierarchy has been defined. Gruber mentioned that there are several possible approaches in developing a class hierarchy [25]. Top-Down approach is employed in our research in which hierarchal structure and top level concepts of the ontology are identified as super classes. The design of class hierarchy and level of details depends on the possible uses of the ontology that is required for the application.

For our proposed ontology classes of Unani medicines and their hierarchy (see Fig 1) are defined using Protégé [27]. All classes in the hierarchy are subclasses of Thing, as per convention of Protégé. So, six main classes of ontology Treatment, Disease, Diagnosis, Patient, State and Unani_Medicine_Principle are derived from Thing, each of them with their own hierarchy of subclasses. It has also been verified from domain experts that top level classes cover all major terms of Unani medicines that have been found in phase 3.

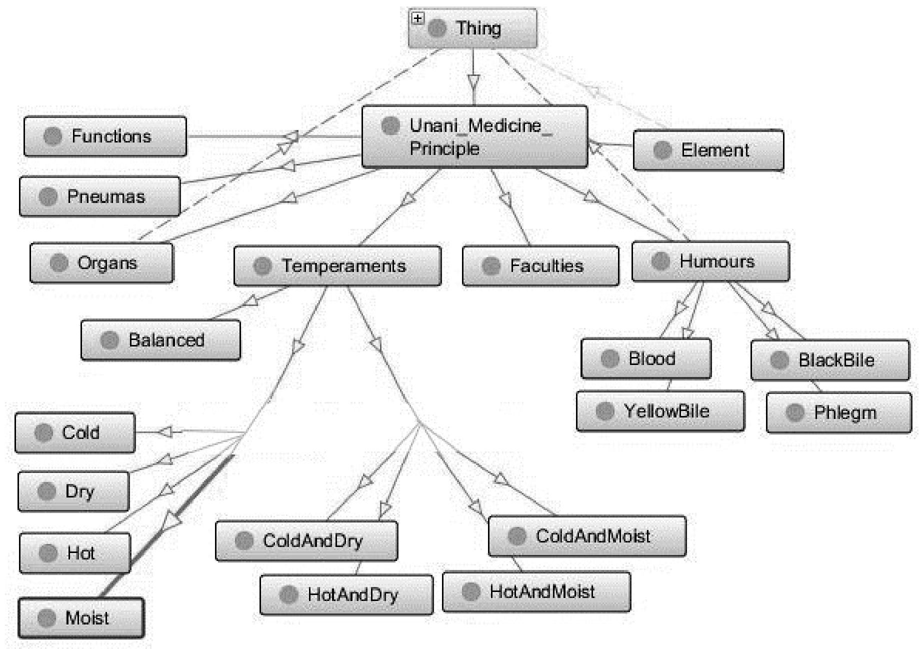


Fig. 2. Ontology Graph of Unani_Medicines_Principle in Protégé

The **Unani_Medicine_Principle** class describes all the core concepts and philosophy of Unani Medicines. It distinguishes Unani medicines from other domains of traditional medicines like Ayurveda and Chinese medicines, each of them with their own core concepts and principles.

After these 6 main classes, sub-classes of each of them have been defined. Only subclasses of Unani_Medicine_Principles need further elaboration. Unani medicines is a wholistic approach towards the whole human body and divide it into seven core components, known as Umoor-e-Tabiya [29] in Urdu language. These are Element (Arkaan), Temperaments (Mizaj), Humours (Akhlaat), Organs (Aa'za), Faculties (Quwa), Functions (Afa'al) and Pneumas (Arwaah). (see Fig. 2). Some of them are further elaborated in their subclasses.

3.5 Define Class Properties (or Slots)

Classes (or concepts) and class hierarchy alone does not preserve all the required semantics of the domain. Every instance of a class has some associated properties (or attributes). Some of the properties are data properties and some are object properties.

The **Data properties** may be simple or complex. Simple contains primitive data of concepts that comprises of strings, numbers, etc. Whereas complex properties contains other objects (See Fig. 1, lower right window for data properties.)

The **Object properties** are defined to relate two classes (or concepts) and hence their objects (two classes may be same or different). In ontologies, object properties may exist independently or may be related to one or more class. Following is a short description of object properties which are defined for our ontology (See Fig. 1, upper right window for object properties.)

excessOf relates Patient with Humour, it shows the excess of any of four Humour values: Blood, Phlegm, YellowBile or BlackBile, i.e. excess may cause specific Disease related to that Humour.

hasBodyTemperature Patient with its Symptom for having body temperature, increased, decreased, or normal.

hasColor relates Patient with Symptom of Eye, Hair, Skin, or Tongue color, that helps in identification of Temperament or Disease, and ultimately in Diagnosis.

hasElement relates any of Temperament value with related value of Element according to Unani medicines principle.

hasGrowth relates Patient with Symptom of abnormal or normal Hair growth, that leads to Diagnosis process.

hasHumour provides any of four Humour value of Patient that is in excess.

hasState relates any of four Element with relevant value from four State.

hasStructure relates Patient with broad or short Body Structure as Symptom.

hasSymptoms relates Patient Disease with any associated Symptom.

hasTemperament links concept of Temperament with concepts of Patients, Humours or Organs.

hasTexture relates Patient with Symptom of Abnormal Hair or Nail texture, providing clue in Diagnosis of Diseases.

SymptomOf relates Symptoms with Disease; it is inverse of hasSymptom property.

Similarly object properties of **hasTongue**, **hasSkin**, **hasNail**, **hasHair**, **hasEye** and **hasFlesh** relates Patient to his physical attributes.

3.6 Define the Facets (or Constraints) of the Properties

In this phase facets (constraints and restrictions) of the properties, have been applied. Facets limit the set of possible values for any property. Commonly applied facets are cardinality and Type of Value. Currently only one restriction in the ontology is enforced that patient is a person that has at least one disease associated with it (otherwise the person is healthy, not a patient).

3.7 Create Instances

Once all the classes, subclasses, object and data properties, facets have been defined, individuals of classes are created. Individuals (like objects in object oriented paradigm) are specific instances of classes. In proposed ontology a lot of individuals have been created (See left side window in Fig. 3).

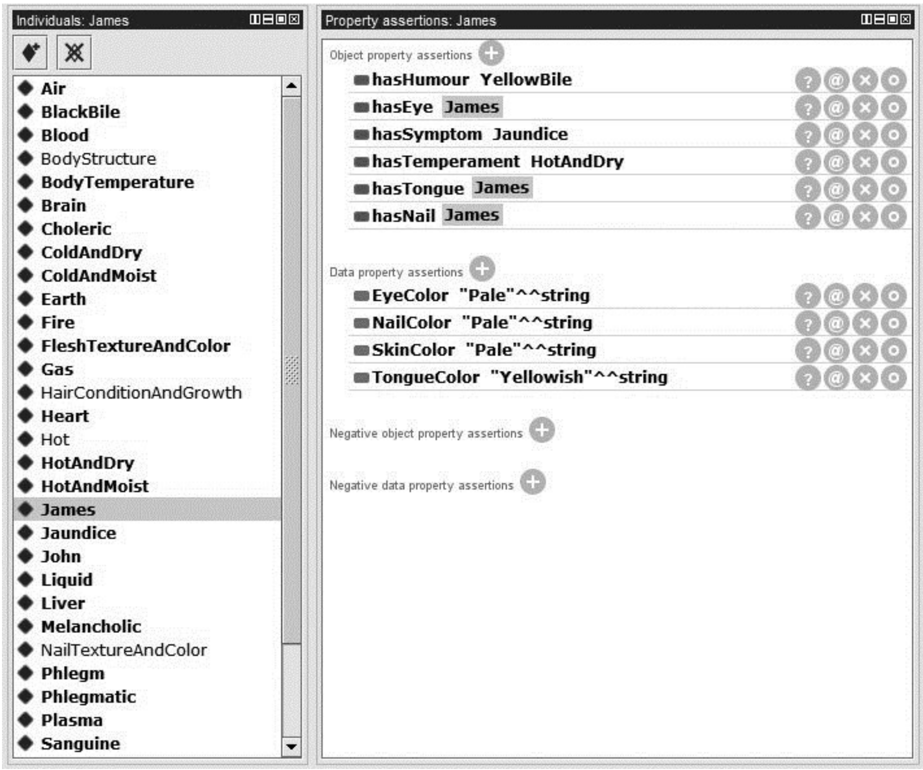


Fig. 3. An individual of a class at left its data and object properties in lower right

A real world scenario is presented in Fig. 3 that elaborates how our proposed ontology can be used in a clinical decision support system that assists Unani practitioners.

For instance, whenever a patient arrived, naming James, an instance of class Patient is created for him and practitioner using the system input values of data properties of James; TongueColor is Yellowish, NailColor is Pale, EyeColor is Pale and SkinColor is Pale too. There exists a rule in the system, which infers his Humour using the given data as YellowBile.

```
RULESTART "YellowbileHumour"
    IF (TongueColor is Yellowish AND
        NailColor is Pale AND
        EyeColor is Pale AND
        SkinColor is Pale)
    THEN Humour is YellowBile
ENDRULE
```

Concepts of Humor of Patient and YellowBile are linked (by hasHumor object property), whereas Temperament of YellowBile is already semantically linked in our ontology to HotAndDry (by object property hasTemperament) and by applying reasoning system suggests that James has chances of Jaundice (object property of hasSymptom is semantically linked in ontology with Jaundice). So the ontology links semantically related concepts and helped in semantic searching.

In the same way, by using the proposed ontology, decision support system can infer new facts from family history and signs and symptoms of James, and helps in prescription of drugs for the treatment of Jaundice.

4 Conclusion and Future Work

We proposed upper domain ontology for knowledge preservation of Unani medicines, an Asian system of traditional medicines. It is the first ever ontology for this field. The ontology currently covers the core principles of Unani medicines, patient, diagnosis, symptoms, disease, and treatment. The ontology organizes common understanding for information of the domain both for human and for software agents; it serve as the base semantic structure for linking concepts of Unani medicines with each other, by restricting the types and values of each concept and helps in semantic searching and navigation. All of these above mentioned features are previously not possible in text-based databases, like traditional knowledge digital library, that have no built-in conceptual structure and offer only keyword based search. In upcoming work, ontology will be further expanded to broaden its depth and breadth and validated by employing it in expert systems, clinical decision support systems and different AI and machine learning applications.

References

1. WHO: WHO traditional medicine strategy 2002−2005. WHO, Geneva (2002)
2. Patwardhan, B.: Traditional Medicine: Modern Approach for Affordable Global Health. WHO-Commission on Intellectual Property Rights Innovation and Public Health (CIPIH), Study Nine on TM, WHO, Geneva. (2005)
3. Global Industry Analyst: Alternative Medicine: A Global Outlook. Report published by Global Industry Analyst, Inc. San Hose (2012)
4. Tinnaluck, Y.: Modern science and native knowledge: collaborative process that opens new perspective for PCST. J. Quark No. 32: Apr−Jun 2004, Spain (2004).
5. Magnor, L. N.: A History of Medicine. 2nd Ed., Taylor and Francis, USA (2005).
6. Jang, H., Kim, J., Kim, S.-K., Kim, C., Bae, S.-H., Kim, A., Eum, D.-M., Song, M.-Y.: Ontology for Medicinal Materials Based on Traditional Korean Medicine. Bioinformatics 26(18), 2359−2360 (2010)
7. Arita, M., Suwa, K., Yoshimoto, M., Hirai, A., Kanaya, S.: Ontology Checking and Integration of Crude-Drug and Kampo Information. 2nd Int. Conf. on BMEI, Tianjin, pp 1−4 (2009)
8. Nantiruj, T., Maneerat, N., Varakulsiripunth, R., Izumi, S., Shiratori, N., Kato, T., Kato, Y. and Takahashi, K.: An e-Health Advice System with Thai Herb and an Ontology, In

Proceeding of the 3rd International Symposium on Biomedical Engineering (ISBME 2008), pp.315−319 Thailand (2008)

9. Kato T., Maneerat N., Varakulsiripunth R., Kato Y., Takahashi K.: Ontology-based E-health System with Thai Herb Recommendation. In Proceeding of the 6th International Joint Conference on Computer Science and Software Engineering (JCSSE2009), pp.172−177, Thailand (2009)
10. Kato T., Maneerat N., Varakulsiripunth, R., Izumi, S., Takahashi, H., Suganuma, T., Takahashi, K., Kato, Y., Shiratori, N.: Provision of Thai Herbal Recommendation Based on an Ontology. In 3rd Int. Conf. on Human System Interaction (HSI2010), pp.217-222, Poland (2010)
11. Izumi, S., Kuriyama, D., Itabashi, G., Togashi, A., Kato, Y., Takahashi, K.: An Ontology-based Advice System for Health and Exercise. In Proceedings of the 10th International Conference on Internet and Multimedia Systems and Applications (IMSA2006) pp.95−100 (2006)
12. Atemezing, G., Pavon, J.: An Ontology for African Traditional Medicine. International Symposium on Distributed Computing and Artificial Intelligence 2008 (DCAI'08): In Series: Advances in Intelligent and Soft Computing, Salamanca, Vol. 50, pp. 329−337 (2008)
13. Ayimdji, A., Koussoubé, S., Fotso, L.P., Konfé, B.O.: Towards a "Deep" Ontology for African Traditional Medicine. In J. of the Intelligent Information Management, pp.244−251 (2011)
14. Oladosu, J.B., Adigun M.O., Mbarika V.: Towards a Pharmaceutical Ontology for African Traditional Herbs. In Proceedings of the World Congress on Engineering and Computer Science (WCECS 2012), Vol. I, San Francisco (2012)
15. Chen, H., Mao, Y., Zheng, X., Cui, M., Feng, Y., Deng, S., Yin, A., Zhou, C., Tang, J., Jiang, X., Wu, Z.: Towards Semantic e-Science for Traditional Chinese Medicine, BMC Bioinformatics, Vol.8(3) (2007)
16. Mao, Y., Wu, Z., Tian, W., Jiang, X., Cheung, W.K.: Dynamic sub-ontology evolution for traditional Chinese medicine web ontology. J. Biomed. Inform., 41(5), pp.790−805 (2008)
17. Cheung K.H., Chen H.: Semantic Web for data harmonization in Chinese medicine. In J. of Chinese medicine, 5(2) (2010)Jokiniemi, J.: Ontologies and Computational Methods for Traditional Chinese Medicine, Master's Thesis, School of Science & Technology, Aalto University, Finland. (2010)
19. Zhao J.: Publishing Chinese medicine knowledge as Linked Data on the Web. In J. of Chinese Medicine, 5(27) (2010)
20. Qi, X., Meng, C., Cungen, C., Baoyan, L., Changlin, L., Hongguan, J.: Establishment of the TCM Meta Conceptual Model Based on Domain Ontology. In J. of World Science and Technology Modernization of Traditional Chinese Medicine, Vol.11(4) pp.621-625 (2009)
21. Yu, T.: Ontology Engineering for Chinese Medicine - Connecting eastern and western medicine, Technical Report of CCNT DartGrid Group, Zhejiang University, Hangzhou, Zhejiang, China (2008)
22. Traditional Knowledge Digital Library, http://www.tkdl.res.in/
23. Samaddar, S.G., Chaudhary, B.D: Development of a Framework for Traditional Knowledge Digital Library (TKDL-S) Semantic Web Portal for Santal Medicinal System. J. of Science and Culture, Vol. 76 (11–12) pp. 544−551 Kolkata India, Nov-Dec (2010)
24. Gruber, T.R.: Toward principles for the design of ontologies used for knowledge sharing. In International J. of Human-Computer Studies - Special issue: the role of formal ontology in the information technology Vol. 43 (5−6) pp. 907−928 (1995)

25. Huang, M.J., Chen, M.Y.: Integrated design of the intelligent web-based Chinese Medical Diagnostic System (CMDS) – Systematic development for digestive health. Expert Systems with Applications, Vol. 32(2) pp.658−673 (2007)
26. Protégé Ontology Editor and Knowledge-base framework, http://protege.stanford.edu/
27. Noy, N.F., McGuiness, D.L.: Ontology Development 101: A Guide to Creating Your First Ontolotgy. Technical Report KSL−01−05, Stanford Knowledge System Laboratory (2001)
28. Ahmad S.I.: Introduction to Al Umur Al Tabiya - Principles of Human Physiology in Tibb. Saini Printer, Pahari Dhiraj, New Delhi (1980)

Ontology-based Knowledge Representation for Supporting Medical Decisions

José A. Reyes-Ortiz, Ana L. Jiménez, Jaime Cater, Cesar A. Meléndez, Patricia B. Márquez, Marlon García

Health Digital Systems,
Distrito Federal, Mexico

{alejandro.reyes, ajimenez, jaimecater, cmelendez, pmarquez, marlon.garcia}@saludhds.com.mx

Abstract. The realization of medical diagnoses is the result of decisions taken by doctors, which consists in identifying the signs, symptoms, risk factors and medical background to determine the clinical diagnosis for patients. This process can be performed improperly due to factors such as inaccurate interpretation of information. This paper presents a computational model of representation of medical knowledge to support decision-making task during a medical consultation in order to reduce the chance of misdiagnosis in general medicine. This representation is based on ontologies that provide a mechanism for structuring knowledge to become computer-understandable information, shared by information systems, formalized and using a common vocabulary. Our ontological model is able to infer a list of clinical diagnoses from the data of signs, symptoms, risk factors and medical background.

Keywords: Knowledge representation, clinical diagnosis, ontologies, medical decisions.

1 Introduction

Medical diagnosis is a procedure, which identifies a disease, syndrome or condition of a patient by using several variables among which are relevant the symptoms, risk factors and clinical studies.

The automation of this process is a challenge for computational systems. In this area doctors generate the final diagnoses manually in systems based in the Electronic Health Record (EHR), which selects the final diagnosis from the international catalog of diseases called ICD10 [1]. This process is tedious, very time consuming and can be generated incorrectly due to misuse of the doctor-patient relationship, interrogation and misdirected clinical misinterpretation of medical semiotics or due to doctor's physical wear. It has been estimated that the percentage of medical errors during hospital care is among 3.5% to 16.6% [2]. These problems reflect the importance of generating information systems that enable medical support during the decision making process to provide a final diagnosis. A computational model that represents medical knowledge should support such systems.

This paper puts forward a computational model based on ontology representation, which is used to support decision-making in clinical diagnostic generation from input data as semiotics, medical history and risk factors. This model is able to infer a list of likely clinical diagnoses relying on a mechanism of inference (rules) on the stored information. The model is based on ontologies due to its ability to represent and share knowledge in an automated manner, the use of a shared vocabulary in the field of medical diagnostics and the ability to formalize knowledge in ontological language accepted as standard. In addition, ontologies are able to infer new knowledge by relying on inference machines, so they are a novel technology to knowledge representation in order to improve diagnosis process from certain medical conditions, such as semiotics, medical history and risk factors.

The rest of this paper is organized as follows. Section 2 presents the elements involved in the process of obtaining a medical diagnosis and the theoretical foundations of knowledge representation based on ontologies. Section 3 presents the work related to the task of obtaining automatic or semi-automatic clinical diagnoses, as well as state of the art work of carrying out knowledge representation through ontologies. The knowledge representation model based on ontologies proposed in this paper is presented in section 4. Section 5 presents the implementation of the proposed ontology, which sets in all classes, relationships and instances of our ontology. Finally, Section 6 presents the conclusions and Section 7 provides the corresponding acknowledgments.

2 Theoretical Framework

The work presented in this article is based on a knowledge representation model to support the work of medical decision-making in clinical diagnostics. Therefore, in this section we present the elements needed to make a medical diagnosis and the theoretical foundations of knowledge representation based on ontologies.

2.1 Medical Diagnosis

Medical diagnosis is a procedure, which identifies a disease, syndrome, or any health condition. This process is carried out using various criteria, such as: symptoms, risk factors, sex and age of patients, laboratory, desk studies and vital signs, such as temperature, blood pleasure, pulse rate and respiration rate.

The most relevant criteria for diagnosis are symptoms with anatomy and evolution, risk factors such as addictions, family-inherited background including the relative (family member) involved. Symptoms are identified by diagnostic printing process, which is clinical reasoning in order to explain the condition. To consolidate a diagnosis it is necessary to identify internal and external risk factors. Once the final diagnosis is made, the doctor proceeds with the treatment, control and monitoring of the disease.

In the area of clinical care, a doctor assigned to the consultation performs the process described above. This process can be done incorrectly, resulting from various factors, the most common documented mistakes are: improper use of the doctor-patient relationship, misdirected interrogation and clinical misinterpretation of semiotics [3].

Computer systems can support this process. However, knowledge representation models are necessary to organize the information needed to make the diagnosis in order to make this information manageable by a machine. This model can be understood by a machine to perform inference on the information and get the diagnoses that meet the symptoms and risk factors.

Our work proposes a model of representation of the ontology-based information to make it manageable for an expert system that supports decision-making in medical diagnosis.

2.2 Knowledge Representation

Knowledge representation is an Artificial Intelligence area research which aims to facilitate the inference of new knowledge from the elements represented symbolically. The knowledge representation also involves an analysis of the proper and efficient cognitive reasoning to perform a certain task [4].

The proper use of given information leads to a set of facts or assumptions in a knowledge domain. This information makes logical sense that helps to formalize the semantics of how reasoning should work.

Knowledge representation can be carried out with the support of the ontologies, which function as the means to accomplish this task. Ontologies become the most viable means to accomplish this representation because of its benefits and features [5] and [6]:

1. They offer a specification using a conceptual model.
2. The model is processed by a computer.
3. Allows sharing knowledge using a common vocabulary.
4. The vocabulary used is accepted by a group of domain experts

In our approach, ontologies are proposed to represent medical information necessary to apply inference rules and get a set of likely clinical diagnoses. These ontologies are our model of knowledge representation.

3 Related Work

In the reviewed literature, there are studies that have supported the process of carrying out a medical diagnosis; in this respect computational approaches have been proposed for the generation of probable diagnoses, which have been addressed semiautomatically or automatically. In this regard there are two types of approaches: a) the learning-based statistical approaches such as [7] presented an update of the algorithm C4.5 to "learn" from examples and determine the diagnosis, a statistical approach using Support Vector Machines (SVM) algorithm is discussed in [8], b) finally, approaches based on rules generated manually by experts [9] and [10]. In this respect, there are no studies that support ontology to represent medical knowledge and perform clinical diagnostic process.

The knowledge representation based on ontology is a technique that has been used in domains such as bioinformatics and molecular biology in order to capture existing processes in these domains [11], a protein ontology is proposed in [12] to support computer systems in biology in general providing a proper interpretation of the data, and to model the relationships in proteins, [13] proposes a model of an ontology-based annotation for disease phenotypes, which facilitates the discovery of new phenotype-genotype relationships among species.

The indications of generating computer systems that support medical diagnoses are based on statistical approaches, leaving aside the semantic interpretation of the input data. Meanwhile, the representation of knowledge is an area that has been applied extensively in medicine, specifically in bioinformatics. However, research on ontology-supported medical diagnosis is a challenging task that has no investigation involved.

Knowledge representation of diagnosis has been addressed from other perspectives as probabilistic and logic based approaches. Hence, [14] proposes an approach that uses Bayesian networks in order to argue and deal with the uncertainty problem of fault diagnosis well, the Bayesian network structure is established according to the cause and effect sequence of faults and symptoms, which are the only medical conditions considered in the model of authors; [15] applies the concept of a fuzzy set as knowledge representation to improve the decision making process.

Therefore an attempt is made by authors to design and develop a diagnosis system. The system developed is evaluated using a simple set of symptoms that is added to clinical data in determining diabetes and its severity. [16] offers an approach to create a new medical knowledge representation model, bases on the use of probabilistic theory. Their work start from the description, realized by an expert of the medical knowledge describing the relation between symptoms and diagnoses the proposed approach consists on building a probabilistic model including the Medical Knowledge Base. Moreover, the proposed approach integrates several probabilistic reasoning mechanisms based on the considered knowledge.

Most of the diagnosis systems are designed to possess the clinical data and symptoms associated with a specific disease as knowledge base or they use few or simple medial conditions. Our work presents a novel approach for knowledge representation based on ontologies that unifies the terminology used in diagnosis, facilitates diagnostic prediction and understanding of diagnoses. We consider several medical conditions that are used to determine a clinical diagnosis.

4 Proposed Knowledge Representation Model

This paper puts forward an ontological model to represent medical information, which facilitates obtaining a diagnosis. The information is modeled based on the concepts (disease, symptoms, anatomy, risk factors and background) and the relationships between them (symptom-anatomy, disease-symptom, etc.).

The proposed ontological model is presented in a modular fashion, which has been divided into the following sections: diseases (age group and gender); symptoms with anatomy, intensity and evolution, and risk factors.

A diagnosis must be an instance of the disease concept and it is governed by the International Catalogue of Diseases ICD-10 [1]. Therefore, three sub-ontologies are created that shape the complete model: *Diseases, Symptoms and RiskFactors*. Figure 1 shows the complete model of the Diagnosis Ontology and their relationships between sub-ontologies.

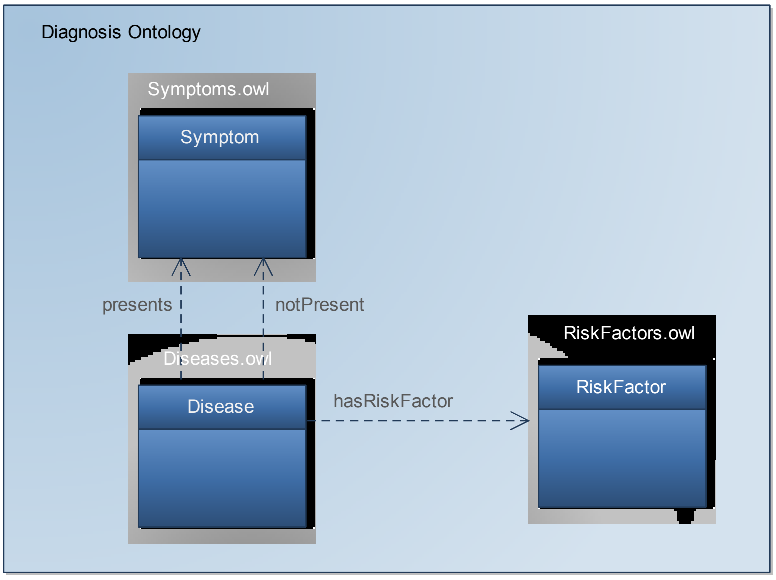


Fig. 1. Diagnosis ontology model

4.1 Diseases

Diseases Ontology has the *Disease* class and has 21 subclasses: *Infectious, Neoplasia, BloodDisease, Endocrine, MentalDisorder, NervousSystemDisease, EyeDisease, EarDisease, CirculatorySystemDisease, RespiratorySystemDisease, DigestiveTractDisease, DermatologicDisease, OsteomusclarSystemDisease, GenitourinarySystemDisease, Pregnancy-Birth, PerinatalInfection,* *CongenitalMalformation, AbnormalClinicalFinding, Trauma-Poisoning and MorbidExtremeCauses.*

Disease class is related to class *Gender, AgeGroup, Symptom* and *RiskFactor*. Figure 2 shows *Diseases* Ontology with some examples of subclasses and their corresponding relationships.

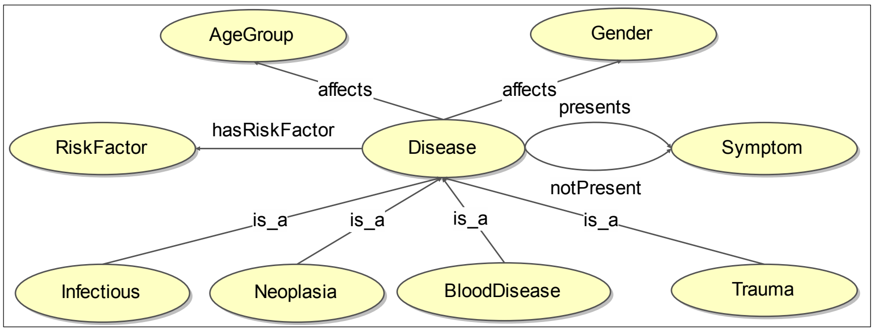


Fig. 2. Diseases ontology

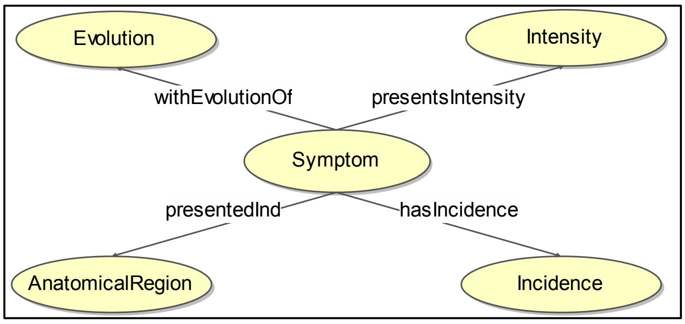


Fig. 3. Symptoms ontology

The *presents* and *notPresent* relations express the absence or presence of certain disease symptoms. On the other hand, the *affects* relationship represents both gender and age group likely to be diagnosed with a certain disease. Finally, the *hasRiskFactor* relationship is used to represent circumstances that allow the existence of a diagnosis.

The *Disease* class instances are the list of possible diagnoses, they belong to the ICD-10 and some examples of *Infectious* subclass instances are paratyphoid *fever*, *pulmonary tuberculosis* and *neonatal tetanus*.

4.2 Symptoms

Symptoms ontology shapes knowledge presented in the disease symptoms. The main class of this ontology is *Symptom* which has four relationships *presentedIn*, *withEvolutionOf*, *presentsIntensity* and *hasIncidence* towards *AnatomicalRegion, Evolution*, *Intensity*, and *Incidence* classes respectively. Figure 3 shows the ontology created to represent the relationship between the symptoms and the diseases.

Examples of Symptom class instances are: *headache*, *pruritus*, and *asthenia*. Meanwhile anatomical regions express human body parts where certain symptom happens, e.g. *stomach*, *legs*, *temporal region of the head*. The intensity, evolution and incidence are relations of the symptoms that are characteristic for certain diseases, which help to differentiate and make a wise decision.

4.3 Risk Factors

A risk factor is a condition or component that can cause a disease to be diagnosed. The main class of this ontology is *RiskFactor*, which has three subclasses *PersonalBackground*, *Addiction* and *FamilyInheritedBackground*. Figure 4 shows the ontology to represent the risk factors that indicate the presence of a certain disease diagnosed.

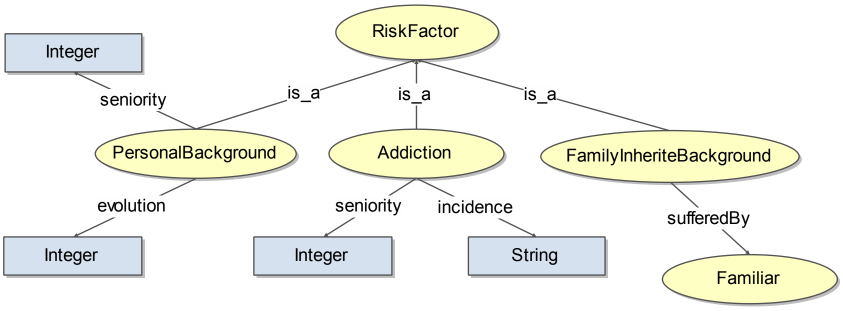


Fig. 4. Risk factor onrology

The *PersonalBackground* class has two properties of object relationships seniority and evolution which are Integer type. Meanwhile, the Addiction class has two properties of object relationships: seniority, which is Integer type; and incidence, which is String type. Finally, the *FamilyInheritedBackground* class has a relationship to Family class called *sufferedBy*, which is used to express the relative (*father*, *mother*, *uncle*, *grandfather/grandmother*, *brother/sister*) who suffer/s a disease.

5 Implementation

The implementation of ontology was made relying on the framework based on knowledge and ontology editor called Protégé [17]. The ontology was encoded using the Web Ontology Language in its version 2.0, which provides classes, properties, individuals and data type values [18]. This language is stored as documents under the Semantic Web standards.

The global ontology called Diagnostic Ontology contains 78 classes or concepts, 23 relations of data type, 29 object type relations. Obtaining a diagnosis is based on the mapping of the input data (patient data, history, symptoms, etc.) with the information stored in the ontology for diseases. We are currently working on this phase.

The information stored in the ontology by classes, relations, individuals and types of data is validated and accepted by a group of medical experts that ensure a common vocabulary in the area of diagnostics.

The fact of using an ontological model to represent information enables the exchange of knowledge, the use of a common vocabulary in applications that are supported by this ontology and reuse of knowledge represented.

6 Conclusions

In this paper we have presented a model that integrates ontological concepts (classes), real entities (entities or individuals) and interactions between classes (relations). These elements add the meaning and the information necessary to perform a diagnosis. This is supporting the process of decision-making by the doctor in order to obtain a final diagnosis.

The proposed model turns medical knowledge in a semantic structure used by a machine for various tasks, such as automatic retrieval of information and inferences. The ontological model presented focuses on three sub-ontologies *Disease*, *Symptoms* and *RiskFactors*. In the *Diseases* Ontology, semantic structured information of diagnoses is represented and is the core of the global model. *Symptoms* ontology is used to represent the information that leads to disease. *RiskFactors* ontology is a model that expresses the necessary conditions for a disease to be diagnosed.

The ontology created is used to support the task of obtaining a list of probable diagnoses. However, in order to obtain this list is not enough with the ontological model, it is essential a set of inference rules to extract new knowledge from existing data. This task leads a research and further work.

We are currently working on these rules to achieve the automatic generation of a list of likely diagnoses from input data. This generation will support medical decision-making, specifically in the generation of the final diagnosis by a doctor. Our approach does not propose the full replacement of a physician in the clinical diagnosis process for patients. However, it proposes a support for physician in decision-making in order to improve the clinical diagnosis process.

Our ontological model can be extended to consider new medical conditions such as family history, gynecological history, laboratory studies, vital signs (temperature, blood pressure and pulse rate), among others. It will be closer to all medical considerations in making a diagnosis manual process.

Our proposed ontology cannot only be used to support decision-making in diagnoses; it can also be used for the task of extracting information on diseases. In addition, it can be used to support a question answer system about the symptoms of diseases, such as: *What are the symptoms of X?* or *What risk factors are conducive to*

X?. This task also involves a great challenge and Natural Language Processing techniques are needed in order to understand the question.

Acknowledgements. This project was funded through the National Council of Science and Technology of Mexico (CONACYT) under project No. 164 487.

References

1. Organización Mundial de la Salud: Clasificación Internacional de Enfermedades, décima versión CIE-10. Informe de la Conferencia Internacional para la Décima Revisión de la Clasificación Internacional de Enfermedades (1998)
2. Martínez, C. M.: Errores médicos en la práctica clínica, del paradigma biologicista al paradigma médico social. Revista Cubana Salud Pública 32(1). Ciudad de La Habana (2006)
3. Díaz, J., Gallego, B., León, A.: El diagnóstico médico: bases y procedimientos. Revista Cubana de Medicina General Integral. Revista Cubana Medicina General Integral 22(1), Ciudad de La Habana (2006)
4. Davis, R., Shrobe, H., Szolovits, P.: What Is a Knowledge Representation? Association for the Advancement of Artificial Intelligence 14(1), pp. 17−33 (1993)
5. Gruber T.: Toward Principles for the Design of Ontologies Used for Knowledge Sharing. International Journal of Human-Computer Studies - Special issue: the role of formal ontology in the information technology 43(5−6), pp. 907−928. USA (1993)
6. Borst, W.: Construction of Engineering Ontologies for Knowledge Sharing and Reuse. Universiteit Twente, PhD thesis, Enschede (1997)
7. Alexopoulos, E., Dounias, G. D., Vemmos, K.: Medical Diagnosis of Stroke using Inductive Machine Learning. In: Proceedings of Workshop on Machine Learning in Medical Applications, Advance Course on Artificial Intelligence ACAI99, pp. 91−101, Chania, Greece (1999)
8. Shashikant, U. G., Ghatol A. A.: Heart Disease Diagnosis Using Machine Learning Algorithm. In: Proceedings of the International Conference on Information Systems Design and Intelligent Applications. Advances in Intelligent and Soft Computing 132, pp. 217−225, India (2012)
9. Drastal, G. A., Kulikowski, C. A.: Knowledge-Based Acquisition of Rules for Medical Diagnosis. Journal of Medical Systems 6(5), pp. 433−445, Springer, Netherlands (1982)
10. Jianwei, X., Ke, X.: A novel extracting medical diagnosis rules based on rough sets. In: The 2009 IEEE International Conference on Granular Computing, pp. 608−611, IEEE Press, Nanchang, China (2009)
11. Steven, R., Goble, C.A., Bechhofer, S.: Ontology-based knowledge representation for bioinformatics. Brief Bioinform 1(4), pp. 398−414, United Kingdom (2000)
12. Chang, E., Sidhu, A., Dillon, T., Sidhu, B.: Ontology-based Knowledge Representation for Protein Data. In: Dillon, T. and Yu, X and Chang, Elizabeth (ed.), INDIN 2005: 3rd International Conference on Industrial Informatics, Frontier Technologies for the Future of Industry and Business. IEEE Press, pp. 535−539. Australia (2005)
13. Washington, N. L., Haendel, M. A., Mungall, C. J., Ashburner, M., Westerfield, M., Lewis, S. E.: Linking Human Diseases to Animal Models Using Ontology-Based Phenotype Annotation. PLoS Biol. 2009 Nov; 7(11) USA (2009)

14. Baoping, C., Yonghong, L., Qian, F., Yunwei, Z., Zengkai, L., Shilin, Y., Renjie, J.: Multi-source information fusion based fault diagnosis of ground-source heat pump using Bayesian network. Applied Energy, vol. 114, pp. 1−9. (2014)
15. Anouncia, M. S., Maddona, C. L., Jeevitha, P., Nandhini, R. T.: Design of a Diabetic Diagnosis System Using Rough Sets. Cybernetics and Information Technologies, vol. 13 (3), pp. 124−169. (2013)
16. Alsun, M., Lecornu, L., Solaiman, B.: Possibilistic Medical Knowledge Representation Model. J. Adv. Inf. Fusion, vol. 7(2), pp. 101−113. (2012)
17. Tudorache, T., Noy, N., Tu, S., Musen, M.: Supporting Collaborative Ontology Development in Protégé. The Semantic Web – International Semantic Web Conference, vol. 5318, pp. 17−32. (2010).

Smith, M., Horrocks, I., Krötzsch, B. G.: OWL 2 Web Ontology Language: Conformance (Second Edition). Eds. W3C Recommendation. Latest version available at http://www.w3.org/TR/owl2-conformance/. (2012)

Clasificación de actos de habla en diálogos basada en los verbos de habla

Grigori Sidorov

Laboratorio de lenguaje natural y procesamiento de texto,
Centro de Investigación en Computación (CIC),
Instituto Politécnico Nacional (IPN), México DF,
México

www.cic.ipn.mx/~sidorov

Resumen. En este artículo presentamos la clasificación de los actos de habla para el manejo de diálogos que usa como base la clasificación de los verbos de habla presente en los lenguajes naturales. Consideramos dos clasificaciones de los verbos de habla en español y realizamos su comparación. Observamos que a pesar de las diferencias en los niveles altos de las clasificaciones, los niveles bajos que corresponden a los actos de habla tienen mucha similitud, por lo que podemos utilizarlos para la el manejo de diálogo en los sistemas computacionales. Adicionalmente el artículo contiene las listas de lexemas que corresponden a los actos de habla.

Palabras clave: Clasificación de actos de habla, robótica móvil, diálogo en lenguaje natural, verbos de habla.

1. Introducción

Los seres humanos vivimos en la sociedad, somos seres sociales, por lo que es indispensable nuestra interacción con otras personas. Para esta interacción utilizamos nuestro lenguaje, de hecho, es una de las funciones más importantes del lenguaje. Normalmente, para la expresión de nuestras ideas y su transmisión a otros estamos necesitamos establecer un diálogo en el lenguaje natural. El diálogo es un proceso de intercambio de réplicas entre dos o más personas que cumple con ciertas reglas.

Las reglas básicas que deben cumplir las réplicas en un diálogo son conocidas como las máximas de Grice [1]. Esas reglas establecen los principios de una comunicación exitosa. Son unos principios muy generales e intuitivamente bastante claros, de hecho, siempre los utilizamos en nuestra propia comunicación con otras personas. Son cuatro máximas en total:

- Máxima de calidad que requiere que las réplicas no contengan la información deliberadamente falsa o infundamentada.
- Máxima de cantidad que dice que no hay que decir ni demasiado ni poco, sino solamente lo necesario.
- Máxima de relevancia que establece que las réplicas deben tener alguna relación con el tema de diálogo (digamos, algo puede ser muy cierto y muy bien expresado, pero estar completamente fuera del contexto).
- Máxima de claridad que exige que la forma de expresión sería la más entendible posible, es decir, sin ambigüedades y sin enredos.

Cabe mencionar que con el desarrollo de la ciencia de la computación y específicamente de la robótica, cuando, primero, las computadoras —que son cerebros de los robots—, ya pueden reconocer el habla, tienen cierto conocimiento del mundo exterior y pueden contestar de manera coherente, y, segundo, los robots ya tienen una forma aceptable —en algunas casos ya hasta parecen a los humanos— y pueden moverse en su ambiente de manera razonable, algunos participantes de los diálogos ya pueden ser los robots. Claro que en la etapa actual su participación es bastante limitada, sin embargo es una de las aplicaciones muy interesantes de la lingüística computacional.

El esquema general de procesamiento de información en el diálogo con participación de un robot es el siguiente: el robot reconoce la voz de la frase de entrada (realmente lo hace la computadora, es decir, el software de reconocimiento de voz), después se realiza el procesamiento de la frase utilizando las técnicas tradicionales de la lingüística computacional (análisis morfológico, sintáctico [2], hasta llegar al análisis semántico [3]), se analiza la frase en el contexto del diálogo, se determina el acta de habla de cada frase (su interpretación pragmática), se prepara una respuesta que corresponde a este acta de habla, se genera la frase de la respuesta, se pronuncia la respuesta en caso necesario, se realiza la acción en caso que la frase correspondía al acta de habla que requería alguna acción.

Un concepto importante en este esquema de procesamiento es precisamente el concepto del acta de habla, que se discute más abajo.

La idea principal en este artículo es proponer una clasificación de actos de habla basándose en los verbos de habla que existen en los idiomas reales, como el inglés o el español, es decir, la clasificación de actos de habla ya existe reflejada en algún lenguaje. Realmente, esos verbos deben reflejar la necesidad de las personas en los actos de habla, claro, entre otras cosas.

La estructura del artículo es la siguiente. Primero estamos discutiendo el concepto de los actos de habla y presentamos algunas clasificaciones de ellos propuestas para el manejo de diálogo (sección 2). Después analizamos la estructura de los verbos de habla tomados de dos fuentes —del tesauro de Roget para el idioma inglés [4] y del libro de texto [5]— considerando sus traducciones al español; asimismo presentamos las unidades léxicas que corresponden a cada elemento de las clasificaciones (sección 3). Realizamos la comparación entre esas clasificaciones (sección 4). Finalmente presentamos las conclusiones.

2. Los actos de habla

Los actos de habla es un concepto que establece la relación entre las frases o sus componentes (los lexemas) y las acciones que ya existen o que deben ser realizadas en el mundo exterior. En este sentido, este concepto pertenece al dominio de la pragmática [2], [6].

La primera clasificación de los actos de habla fue propuesta por su creador J. Searle [7]:

- Afirmaciones (*assertives*): se afirma que lo que se dice es verdad, por ejemplo, *profesar*.
- Directivas (*directives*): son comandos que requieren alguna acción de la otra parte, por ejemplo, *pedir*, *ordenar*, *aconsejar*.
- Promesas (*commissives*): el hablante contrae ciertos compromisos, como *prometer*.
- Expresiones (*expressives*): se comunica el estado del hablante o su evaluación de las acciones de la otra parte, por ejemplo, *felicitar*, *excusarse*, *agradecer*.
- Declaraciones (*declarations*): esencialmente son los verbos performativos, por ejemplo, *declaro*, *juro*, etc.

Como esa fue una primera clasificación, se puede observar que es demasiado general y para su uso en el mundo real se necesitan unas clasificaciones mucho más finas.

Como hemos mencionado en nuestro trabajo anterior [8], es comúnmente aceptado, por ejemplo, [9], [10], [11], [12], [13] que el esquema general de un diálogo sencillo es la siguiente:

Saludo—Apertura—Negociación/Acción—Finalización—Despedida.

Muchas clasificaciones de actos de habla están directamente orientadas a este esquema del diálogo sencillo. Por ejemplo, Levinson [6] menciona las siguientes pares de actos de habla y también incluye las reacciones típicas:

- petición → aceptar / rechazar,
- oferta → aceptar / rechazar,
- invitación → aceptar / rechazar,
- evaluación → acuerdo / desacuerdo,

- pregunta → respuesta esperada / respuesta inesperada,
- acusación → negación / admisión,
- disculpa → minimización,
- agradecimiento → minimización (ej., "de nada"),
- saludo → saludo.

Otro modelo también orientado al diálogo de este tipo es el modelo DAMSL de Allen y Gore [14], que contiene cuatro dimensiones, cada una incluye algunas subdimensiones y los actos de habla que les corresponden.

1. Estado comunicativo (no operable, abandonado, o monólogo vs. diálogo).
2. Nivel de información.
   Actos de habla:
   - tarea,
   - mantenimiento de tarea,
   - mantenimiento de comunicación.
3. Análisis hacia adelante (*Forward Looking Function*), es decir, las acciones subsecuentes se verán afectadas.
   Actos de habla:
   - Declaración *(statement),*
   - Compromiso de la acción subsecuente *(committing-speaker-future-action),*
   - Convencional *(conventional),*
   - Performativo *(explicitperformative),*
   - Exclamación *(exclamation),*
   - Otro (lo que dice el hablante): suposición, confirmación,
   - Influencia de la acción subsecuente *(influencing-addressee-future-action)* (como en la teoría de Searle):
     - Una sugerencia o una lista de opciones,
     - Un comando real,
     - Petición de información.
4. Análisis hacia atrás (*Backward Looking Function*), es decir, se establece la relación con el discurso anterior.
   Actos de habla:
   - Concordancia *(agreement):* aceptar, rechazar, mantener comunicación,
   - Entendimiento *(understanding),*
   - Respuesta *(answer)* que corresponde a la petición de información.

Una clasificación más detallada de actos de habla se presenta, por ejemplo, por Jurafsky y Martin [15]:

- Agradecer (*Thank*),

- Saludar (*Greet*),
- Introducir (*Introduce*),
- Despedirse (*Bye*),
- Solicitar_comentario (*Request_comment*),
- Sugerir (*Suggest*),
- Rechazar (*Reject*),
- Aceptar (*Accept*),
- Solcitiar_sugerencia (*Request_suggest*),
- Iniciar (*Init*),
- Argumentar (*Give_reason*),
- Retroalimentar (*Feed_back,*)
- Deliberar (*Deliberate*),
- Confirmar (*Confirm*),
- Clarificar (*Clarify*),
- Desviarse (*Digress*),
- Motivar (*Motivate*),
- Desechar (*Garbage.*)

Se puede observar que en su mayoría esas clasificaciones parten del esquema mental de la estructura del diálogo impuesto por el sentido común. En este artículo trataremos de construir una clasificación de actos de habla partiendo de la parte estrictamente lingüística. Para eso vamos a considerar los verbos que existen en el idioma español y están relacionados con el habla.

3. Verbos de habla y actos de habla

La idea principal de este artículo es proponer una clasificación de los actos de habla partiendo de los datos que están reflejados en el lenguaje, y no de algún esquema puramente lógico. Es claro que las clasificaciones —basada en la lógica y basada en el lenguaje— pueden parecerse en algunos aspectos, pero esperamos que la clasificación lingüística presente más detalles y más aspectos. En un paso siguiente hay que determinar qué aspectos sean relevantes para el manejo de diálogo.

Para realizar esa tarea hemos considerado dos fuentes de los datos lingüísticos: tesauro de Roget [4] y una más reciente clasificación de los verbos de habla tomada de [5].

Adicionalmente utilizamos tanto las fuentes mencionadas como otros diccionarios para determinar los lexemas que corresponden a los verbos de habla.

Cabe mencionar que hemos excluido de esa lista algunos verbos que aunque pertenezcan al campo semántico de los verbos de habla no pueden expresar los actos de habla como tal, por ejemplo, *estar en silencio*, *sisear*, etc.

A continuación presentamos ambas clasificaciones. En la tabla 1 dos primeras columnas corresponden a la clasificación, la tercera a actos de habla, y la última contiene los lexemas que permiten la detección del acto de habla correspondiente, es decir, su presencia en la frase nos hace suponer que tal acto de habla podría estar presente. El número ante la lista de lexemas corresponde al acto de habla en la misma fila (que también tiene este número). Primero presentamos los verbos, después va el separador "//", y al final se encuentran palabras de otras clases gramaticales. En caso de los verbos hemos hecho el mejor esfuerzo de tener la lista completa utilizando un diccionario de sinónimos, en caso de palabras de otras clases gramaticales, son más bien para la ayuda de los desarrolladores de sistemas prácticos que los pueden ocupar, pero las listas no son completas.

Hay que mencionar que usualmente tomamos el sentido dominante de la palabra (normalmente es el primer sentido de diccionario), pero sería útil marcar los sentidos en la tabla y realizar el procedimiento de desambiguación de sentidos de las palabras en la frase de entrada.

Tabla 1. Primera clasificación (basada en Roget) junto con los marcadores léxicos

	Situación general	**Situación particular**	**Acto de habla**	**Lexemas que permiten la detección del acta de habla**
1	Formar una idea (intelecto)	Preliminar	1 Preguntar 2 Responder	1. pedir, hacer, preguntar, invitar, interrogar, publicar, indagar, averiguar, informarse, inquirir, cuestionar // pregunta, cuestión, interrogación, problema, investigación, indagación, encuesta, petición, examen 2. responder, reaccionar, atender, ser sensible, contestar, replicar // respuesta, reacción, responsorio, contestación, solución, réplica, resultado, explicación
2	Transmitir una idea (intelecto)	Carácter de la idea	1 Interpretar	1. interpretar, entender, traducir, explicar, comprender, servir de intérprete, descifrar // interpretación, explicación, reproducción, traducción, versión
3		Manera de la transmisión	1 Informar 2 Aceptar 3 Afirmar 4 Negar 5 Decir 6 verdad	1. informar, comunicar, informarse, avisar, denunciar, delatar, decir, contar, saber, informar, indicar, distinguir // información, datos, informes, informe, conocimientos, noticias 2. aceptar, admitir, consentir, convalidar,

			7 Mentir, engañar	tomar, llevar, asumir, permitir, dejar, autorizar, admitir, conceder // aceptación, aprobación, admisión 3, 5 decir, opinar, afirmar, expresar, dar, ratificar, jurar, declarar, anunciar, enunciar // afirmación, declaración, aseveración, ratificación, aserción, reivindicación 4. negar, denegar, rechazar, desmentir, privar, renegar de, denegar, rehusar, impugnar, privarse, anular, invalidar, anular, desaprobar, desmentir // negación, denegación, negativa, desmentido, abnegación, repudio, negación, desautorización, rechazo, negativa 6. // verdad, veras, verosimilitud 7. mentir, engañar, engañarse, burlar, defraudar, equivocarse, estacar, estafar, timar, trampear, trapacear // mentira, mentiras, posición, embustes, embuste, tapada, invención, invento, inventiva, ficción, mentira, fábula, engaño, decepción, fraude, mentira, ilusión, falsedad, error, equivocación, falta, descuido
4		Medios de transmisión	1 Pro-nunciar 2 Hablar 3 Charlar 4 Llamar 5 Platicar 6 Describir	1. pronunciar, declarar, dar su opinión, proferir, expresar, expender, dar a, publicar, entregar, llevar, pronunciar, lanzar, comunicar // pronunciamiento 2. hablar, conversar, charlar, platicar, cotorrear, parlotear // palabra, término, voz, vocablo, noticia, verbo, oración charlar, parlotear, castañetear, rajar, decir charlas, trapalear // parloteo 3. llamar, convocar, citar // llamado, convocatoria 4. hablar, conversar, charlar, chismear, platicar // charla, plática, chisme, palabra 5. describir, relatar, trazar, retratar, representar // descripción
5	Inter-individual (voluntad)	General	1 Ordenar	1 ordenar, mandar, comandar, dirigir, disponer, decretar // orden, comando, dirección, decreto, disposición
6		Especial	1 Permitir 2 Prohibir 3 Acordar	1 permitir, dejar, autorizar, admitir, conceder, aceptar, tolerar // licencia, permiso, autorización, libertad,

			4 Proponer 5 Rechazar 6 Pedir	2 prohibir, impedir, imposibilitar, privar, olvidar, suspender, ilegalizar, rechazar// restricción, prohibición 3 acordar, estar de acuerdo, aceptar, convenir, ponerse de acuerdo, coincidir, concordar, pactar // acuerdo, convenio, contrato, tratado, concordancia, pacto 4 proponer, plantear, proponerse // propuesta, oferta 5 rechazar, desestimar, denegar, desechar, descartar, excluir, rehusar // rechazo, rechazamiento, denegación, desestimación, exclusiva, negativa 6 pedir, preguntar, invitar, interrogar, solicitar, recabar, rogar // solicitud, petición, demanda, requerimiento, instancia, ruego, reclamación
7		Condicional	1 Prometer 2 Acordar 3 Discutir	1 prometer, presagiar, comprometerse, empeñar // promesa, esperanza, , garantía, misión, tarea 2 acordar, estar de acuerdo, aceptar, convenir, ponerse de acuerdo, coincidir, conceder, concordar, pactar // acuerdo, convenio, contrato, tratado, concordancia, pacto 3 discutir, debatir, comentar, estudiar, argumentar, argüir, razonar, reñir, abogar
8	Individual (voluntad)	Prospectivo	1 Advertir	1 advertir, avisar, prevenir, alarmar // advertencia, aviso, alarma, amonestación
9		Acción de voluntad	1 Aconsejar	1 asesorar, aconsejar, informar, avisar, recomendar, notificar, sugerir, indicar // consejo, asesoramiento, aviso, informe,
10	Individual (senti-mientos)	Personal	1 Bromear 2 Alabar 3 Regañar 4 Presumir	1 bromear, chancearse, contar chistes, burlarse burdamente, chunguearse // broma, chiste, burla, chanza, hazmerreír 2 alabar, elogiar, ensalzar, encomiar, enaltecer, loar // alabanza, alabanzas, elogio, encomio 3 pelear, reñir, regañar, quejarse, regañarse // regaño 4 presumir, pavonearse, hacer alarde de, hacer resaltar, // engreído, presumido, presuntuoso, envanecido, fatuo, pretencioso, vanidoso, engreído
11		Com-	1 Saludar	1 saludar, recibir, dar la bienvenida,

		pasional	2 Felicitar 3 Bendecir 4 Maldecir 5 Amenazar 6 Agradecer 7 Perdonar	presentarse, aclamar // saludo, bienvenida, salutación 2 felicitar, congratular // felicitación 3 bendecir, // bendición 4 maldecir // palabrotas 5 amenazar, amagar, acechar // amenaza, espanto 6 agradecer, // gracias 7 perdonar, disculpar, indultar // perdón, indulto, misericordia
12		Moral	1 Aprobar 2 Debatir 3 Adular 4 Calumniar 5 Acusar 6 Justificar 7 Sentenciar	1 aprobar, autorizar, pasar, aprobar, superar, adelantar, aceptar, convenir, endosar, avalar, ratificar // ok, visto bueno, aprobación, beneplácito, ratificación, firma, confirmación 2 discutir, debatir, comentar, estudiar, tratar, reflexionar // debate, discusión, disputa 3 halagar, adular, lisonjear, favorecer, incensar // adulación, halago, lisonja, lisonjas, zalamería, piropo 5 calumniar, infamar, decir mal de, rajar, manchar, desprestigiar, mancharse, embarrar, difamar // calumnia, difamación 5 acusar, culpar, tachar de, incriminar, inculpar, acriminar // carga, cargo, costo, acusación, precio, cobro 6 justificar, alinear, establecer, disculpar, vindicar, // excusa, pretexto, disculpa, justificación, razón, defensa 7 condenar, sentenciar // sentencia, juicio, fallo, criterio, opinión, discernimiento

En la tabla 2 presentamos los mismos datos para la otra clasificación [5].

Tabla 2. Segunda clasificación junto con los marcadores léxicos

	Situación	**Actos de habla**	**Lexemas que permiten la detección del acta de habla**
1	Desde el exterior	1 Pronunciar 2 Nombrar	1 pronunciar, proferir, expresar, expender, dar a, publicar 2 nombrar, mencionar, llamar, denominar, designar, bautizar
2	Desde el contenido	1 Expresar 2 Charlar	1 expresar, decir, sonorizar, hacerse eco de, opinar, afirmar

		3 Bromear 4 Galantear 5 Engañar 6 Filosofar	2 charlar, parlotear, castañetear, rajar, decir charlas, trapalear, conversar 3 bromear, chancearse, contar chistes, burlarse, chunguearse, hablar en broma, ridiculizar, 4 galantear, // amable, amabilidad 5 engañar, embaucar, bromear, burlar 6 filosofar, razonar
3	Desde la comunicación	1 Transmitir mensaje 2 Prometer 3 Aconsejar 4 Explicar 5 Demostrar	1 transmitir, difundir, comunicar, llevar, transportar por, tener una comunicación 2 prometer, presagiar, jurar, prometer, hacer voto, dar un juramento 3 asesorar, aconsejar, informar, avisar, recomendar, notificar, sugerir, indicar 4 explicar, explicarse, exponer, aclarar, describir, ilustrar, interpretar, entender, traducir 5 demostrar, manifestar, hacer una manifestación, dar a ver, mostrar, presentar, indicar, enseñar, probar
4	Interacción verbal	1 Hablar 2 Acordar 3 Discutir 4 Debatir	1 hablar, conversar, charlar, platicar, hablar, decir, intervenir, sonar, comunicarse, comunicar 2 acordar, estar de acuerdo, aceptar, convenir, ponerse de acuerdo, coincidir 3 discutir, comentar, estudiar, tratar de, argumentar, argüir, razonar, reñir, abogar por, disuadir 4 debatir, reflexionar
5	Contacto	1 Preguntar 2 Responder 3 Acordar 4 Contradecir 5 Negar 6 Tratar	1 pedir, hacer, preguntar, invitar, interrogar, publicar, indagar, averiguar, informarse, inquirir, cuestionar, interrogar, dudar de, poner en duda 2 responder, reaccionar, atender, ser sensible, contestar, satisfacer, replicar 3 conceder, acordar, concordar, pactar, acordar, conceder, concertar, conformarse, convencionarse, deferir, prestar consentimiento, resolver, tomar el acuerdo 4 contradecir, desdecir, acusar 5negar, denegar, rechazar, desmentir, privar, renegar de, rehusar, impugnar, privarse de 6 asistir, atender, ir a, acompañar, servir, tratar
6	Motivación	1 Ordenar 2 Encargar 3 Incitar 4 Pedir 5 Invitar 6 Llamar	1 dictar, ordenar, requerir, decretar, mandar, disponer, comandar, // mandato, orden, requerimiento, consigna, mandato, disposición 2 cometer, confiar, dejar, delegar, encomendar, recomendar // cargo, comisión, encargo, encomienda, orden, tarea, cometido, comisión,

		7 Mandar 8 Inclinar 9 Permitir 10 Prohibir	delegación, embajada, mandato, requisitoria, requisitorio 3 impulsar, inducir, insinuar, incitación, aludir, // alusión, insinuación, reticencia 4 demandar, impetrar, pedir, peticionar, postular, pretender, recurrir, solicitar, suplicar, // alocución, alzada, dirección, manejo, pedida, recurso, ruego, solicitud, suplicación, trata, exhorto, instancia, pedida, pedido, pedimento, petición, postulación, requerimiento, ruego, solicitud, súplica 5 invitar, // invitación 6 llamarse, nombrar, // llamado 7 dictar, ordenar, requerir // mandato, orden, requerimiento, 8 disponer, colocar, decidir, determinar, inclinar, poner en orden, // inclinación 9 permitir, dejar, autorizar, admitir, conceder, aceptar 10 defender, desaprobar, interdecir, prohibir, proscribir, vedar
7	Actitud emocional	1 Ofender 2 Burlarse 3 Acciones de etiqueta verbal	1 insultar, ofender, injuriar, trapear, agraviar, denigrar 2 burlarse, mofarse de, mofar, zamparse 3 saludar, recibir, dar la bienvenida, presentarse a, despedirse, separarse, apartarse
8	Evaluación emocional	1 General 2 Positiva 3 Negativa	1 evaluar, valorar, // evaluación, valoración 2 alegrarse, regocijarse, regocijar, alegrar, causar alegría a // alegre, jovial, animado, contento, satisfecho, bueno, jubiloso, divertido, regocijado, alborozado, animado, vivo, alegre, lleno de vida, bullicioso, enérgico 3 entristecerse, enojarse, asustarse // triste, lamentable, doloroso, apagado, malísimo, afligido, lastimoso, lúgubre, deprimente, sombrío, tétrico, tenebroso, melancólico, oscuro, pesimista, tenebroso, lóbrego, amoratado, deprimente, miserable, desgraciado, triste, desdichado, abatido, mezquino, miserable, desgraciado, triste, desdichado, abatido, mezquino, enojado, enfadado, furioso, airado, colérico, malhumorado, transverso, repudio, desconocimiento, anulación, incumplimiento, cancelación, rechazamiento, susto, pánico, alarma, espanto, terror, sobresalto, horror,

			espantajo, ansiedad

En la tabla 3 presentamos nada más la clasificación de Roget (sin los marcadores léxicos), mientras que en la tabla 4, la segunda clasificación sin los marcadores léxicos.

Tabla 3. Primera clasificación (basada en Roget) sin los marcadores léxicos

	Situación general	**Situación particular**	**Acto de habla**
1	Formar una idea (intelecto)	Preliminar	1 Preguntar 2 Responder
2	Transmitir una idea (intelecto)	Carácter de la idea	1 Interpretar
3		Manera de la transmisión	1 Informar 2 Aceptar 3 Afirmar 4 Negar 5 Decir 6 verdad 7 Mentir, engañar
4		Medios de transmisión	1 Pronunciar 2 Hablar 3 Charlar 4 Llamar 5 Platicar 6 Describir
5	Inter-individual (voluntad)	General	1 Ordenar
6		Especial	1 Permitir 2 Prohibir 3 Acordar 4 Proponer 5 Rechazar 6 Pedir
7		Condicional	1 Prometer 2 Acordar 3 Discutir
8	Individual (voluntad)	Prospectivo	1 Advertir
9		Acción de voluntad	1 Aconsejar
10	Individual (senti-mientos)	Personal	1 Bromear 2 Alabar 3 Regañar 4 Presumir
11		Com-pasional	1 Saludar 2 Felicitar 3 Bendecir

			4 Maldecir 5 Amenazar 6 Agradecer 7 Perdonar
12		Moral	1 Aprobar 2 Debatir 3 Adular 4 Calumniar 5 Acusar 6 Justificar 7 Sentenciar

Tabla 4. Segunda clasificación sin los marcadores léxicos

	Situación	**Actos de habla**
1	Desde el exterior	1 Pronunciar 2 Nombrar
2	Desde el contenido	1 Expresar 2 Charlar 3 Bromear 4 Galantear 5 Engañar 6 Filosofar
3	Desde la comunicación	1 Transmitir mensaje 2 Prometer 3 Aconsejar 4 Explicar 5 Demostrar
4	Interacción verbal	1 Hablar 2 Acordar 3 Discutir 4 Debatir
5	Contacto	1 Preguntar 2 Responder 3 Acordar 4 Contradecir 5 Negar 6 Tratar
6	Motivación	1 Ordenar 2 Encargar 3 Incitar 4 Pedir 5 Invitar 6 Llamar 7 Mandar

		8 Inclinar 9 Permitir 10 Prohibir
7	Actitud emocional	1 Ofender 2 Burlarse 3 Acciones de etiqueta verbal
8	Evaluación emocional	1 General 2 Positiva 3 Negativa

4. Comparación de las clasificaciones de verbos de habla

Vamos a proceder a realizar la comparación de las dos clasificaciones consideradas anteriormente utilizando el siguiente procedimiento: vamos a empezar con la primera clasificación y tomaremos las clases más detalladas una por una; para cada identificador de la clase vamos a buscarlo tanto entre los actos de habla como entre los lexemas que le corresponden (sus sinónimos) de la segunda. En caso de encontrar un sinónimo, vamos a anotar en la columna correspondiente el nombre del acta de habla de la segunda clasificación (no es sinónimo). Después realizamos el mismo procedimiento empezando con la segunda clasificación.

Los resultados del análisis se presentan en la tabla 5.

Tabla 5. Comparación de las clasificaciones de verbos de habla partiendo de la primera clasificación

	Situación general (Roget)	**Situación particular (Roget)**	**Acto de habla (Roget)**	**Acto de habla (en otra clasificación)**	**Situación (en otra clasificación)**
1	Formar una idea (intelecto)	Preliminar	1 Preguntar 2 Responder	1 Preguntar 2 Responder	1 Contacto 2 Contacto
2	Transmitir una idea (intelecto)	Carácter de la idea	1 Interpretar	1 Explicar	1 Comunicación
3		Manera de la transmisión	1 Informar 2 Aceptar 3 Afirmar 4 Negar 5 Decir 6 verdad 7 Mentir, engañar	1 Aconsejar 2 Acordar 3 Expresar 4 Negar 5. Expresar 6 -- 7 Engañar	1 Comunicación 2 Interacción 3 Contenido 4 Contacto 5 Contenido 6 -- 7 Contenido
4		Medios de	1 Pronunciar	1 Pronunciar	1 Exterior

		transmisión	2 Hablar 3 Charlar 4 Llamar 5 Platicar 6 Describir	2 Hablar 3 Charlar 4. Llamar 5 Hablar 6 Explicar	2 Interacción 3 Contenido 4 Motivación 5 Interacción 6 Comunicación
5	Inter-individual (voluntad)	General	1 Ordenar	1 Ordenar	1 Motivación
6		Especial	1 Permitir 2 Prohibir 3 Acordar 4 Proponer 5 Rechazar 6 Pedir	1 Permitir 2 Prohibir 3 Acordar 4 --- 5 Negar 6 Pedir	1 Motivación 2 Motivación 3 Interacción / Contacto 4 --- 5 Contacto 6 Motivación
7		Condicional	1 Prometer 2 Acordar 3 Discutir	1 Prometer 2 Acordar 3 Discutir	1 Comunicación 2 Interacción / Contacto 3 Interacción
8	Individual (voluntad)	Prospectivo	1 Advertir	1 ---	1 ---
9		Acción de voluntad	1 Aconsejar	1 Aconsejar	1 Comunicación
10	Individual (senti-mientos)	Personal	1 Bromear 2 Alabar 3 Regañar 4 Presumir	1 Bromear 2 --- 3 --- 4 ---	1 Contenido 2 --- 3 --- 4 ---
11		Com-pasional	1 Saludar 2 Felicitar 3 Bendecir 4 Maldecir 5 Amenazar 6 Agradecer 7 Perdonar	1 Etiqueta 2 --- 3 --- 4 --- 5 --- 6 --- 7 ---	1 Emocional 2 --- 3 --- 4 --- 5 --- 6 --- 7 ---
12		Moral	1 Aprobar 2 Debatir 3 Adular 4 Calumniar 5 Acusar 6 Justificar 7 Sentenciar	1 --- 2 Debatir 3 --- 4 --- 5 Contradecir 6 --- 7 ---	1 --- 2 Interacción 3 --- 4 --- 5 Contacto 6 --- 7 ---

Las diferencias en caso de que partamos de la primera clasificación están marcadas en la tabla 5 como ausencia de algunos actos de habla.

En caso que partamos de la segunda clasificación, las diferencias son: *invitar* corresponde a *preguntar*, *mandar* corresponde a *ordenar*. No están presentes, ni con sus sinónimos, los siguientes verbos: *nombrar, galantear, filosofar, demostrar, encargar, incitar, inclinar, ofender, burlarse*.

Se puede observar que la primera clasificación (Roget) tiene más actos de habla, especialmente los que describen expresión de los sentimientos. En este sentido consideramos que conviene considerar que la segunda clasificación es simplemente incompleta en este aspecto. En los demás aspectos las clasificaciones parecen mucho entre sí.

Nótese que las clasificaciones usan diferentes clases en el nivel más alto, sin embargo sus clases a nivel más bajo parecen mucho. Consideramos que eso habla a favor de nuestra idea de que existe una clasificación de los actos de habla reflejada en los verbos de habla en los lenguajes. Como una solución práctica, nuestra recomendación es tomar la primera clasificación, y considerar si es necesario completarla con unos pocos verbos que no estén presentes en ella, pero sí están presentes en la segunda clasificación.

Se puede observar que las clasificaciones presentes en los lenguajes, más aspectos que pueden ser relevantes en los actos de habla que las clasificaciones basadas en pura lógica.

5. Conclusiones

En este artículo hemos presentado varias consideraciones relacionadas con la selección de los actos de habla. Consideramos que las clasificaciones tradicionales de los actos de habla pueden ser incompletas, estamos proponiendo desarrollar una clasificación de los actos de habla basada en la clasificación de los verbos de habla que existen en los lenguajes.

Para verificar esa idea hemos considerado dos diferentes clasificaciones de los verbos de habla tomados junto con las unidades léxicas que corresponden a los actos de habla y realizamos la comparación entre ellas. A pesar de que las clasificaciones tienen niveles altos diferentes, de clasificación más bajos tienen una correspondencia muy clara entre sí. Consideramos que eso habla a favor de la idea que se puede utilizar los verbos de habla como base de la clasificación de los actos de habla.

En comparación con las clasificaciones tradicionales de los actos de habla, la clasificación basada en los verbos de habla presenta mayor diversidad y toma en cuenta aspectos adicionales importantes (y por lo mismo reflejados en el lenguaje).

El artículo contiene los lexemas que corresponden a los actos de habla y en este sentido puede ser utilizado en los sistemas prácticos de manejo de diálogo.

Agradecimientos. Trabajo realizado con el apoyo de gobierno de la Ciudad de México (proyecto ICYT-DF PICCO10-120), el apoyo parcial del gobierno de México (proyectos CONACYT 50206-H y 83270, SNI) e Instituto Politécnico Nacional, México (proyectos SIP 20120418, 20131441, 20131702, COFAA).

Referencias

1. Grice, P.: Logic and conversation. In: Syntax and Semantics, 3: Speech Acts, ed. P. Cole & J. Morgan, New York: Academic Press (1975) [Reprinted in Studies in the Way of Words, ed. H. P. Grice, pp. 22–40. Cambridge, MA: Harvard University Press (1989)]
2. Gelbukh, A., Sidorov, G.: Procesamiento automático del español con enfoque en recursos léxicos grandes. IPN, 307 p. (2010)
3. Pakray, P., Barman, U., Bandyopadhyay, S., Gelbukh, A.: A Statistics-Based Semantic Textual Entailment System. MICAI 2011, Lecture Notes in Artificial Intelligence N 7094, Springer, pp. 267–276 (2011)
4. Roget, P.M.: Roget's Thesaurus of Englsih words and phrases (1875, 1979)
5. Kobozeva, I. Semántica lingüística (en ruso). Moscú, Editorial URSS 352 p. (2004)
6. Levinson, S.C. Pragmatics, Cambridge University Press, Cambridge, MA, USA (1983)
7. Searle, J.: Indirect speech acts. In: Syntax and Semantics, 3: Speech Acts, ed. P. Cole & J. L. Morgan, pp. 59–82. New York: Academic Press (1975) [Reprinted in Pragmatics: A Reader, ed. S. Davis, pp. 265–277. Oxford: Oxford University Press (1991)]
8. Grigori Sidorov. Desarrollo de una aplicación para el diálogo en lenguaje natural con un robot móvil. Research in computing science, Vol. 62, 2013, pp. 95-108.
9. Dialog with Robots. AAAI 2010. Fall Symposium. November 2010, Arlington VA. Available: http://hci.cs.wisc.edu/aaai10/?page_id=169 (2010)
10. Lemaignan, S. and Ros, R. and Alami R.: Dialogue in situated environments: A symbolic approach to perspective-aware grounding, clarification and reasoning for robot. In: Proc. Robotics, Science and Systems, Grounding Human-Robot Dialog for Spatial Tasks workshop (2011)
11. Levin, E., Pieraccini, R., Eckert, W.: A stochastic model of human-machine interaction for learning dialog strategies. IEEE Transactions on Speech and Audio Processing. vol. 8 (1), pp. 11–23 (2000)
12. Marge, M., Pappu, A., Frisch, B., Harris, Th. K., Rudnicky, A.: Exploring Spoken Dialog Interaction in Human-Robot Teams. In: Proceedings of Robots, Games, and Research: Success stories in USARSim IROS Workshop, October 2009, St. Louis, MO, USA (2009)
13. Kruijff, G.J.M., Lison, P., Benjamin, T., Jacobsson, H., Zender, H., Kruijff-Korbayova, I., Hawes, N. Situated dialogue processing for human-robot interaction. Cognitive Systems, pp. 311–364 (2010)
14. Allen, J.F., Core, M.G.: DAMSL: Dialog Act Markup in Several Layers (Draft 2.1), Multiparty Discourse Group, Discourse Research Initiative (1997)
15. Jurafsky, D., Martin, J.: Speech and Language Processing. Prentice Hall (2009)

Reviewing Committee

(Comité de revisión del volumen)

Fernando Aldana
Sivaji Bandyopadhyay
Bert Bredeweg
Hiram Calvo
Niladri Chatterjee
Erik Cuevas
Iria da Cunha
Asif Ekbal
Andrea Formisano
Claude Frasson
Sofia N. Galicia-Haro
Alexander Gelbukh
Pilar Gomez-Gil
Monique Grandbastien
Christian Grimme
Yasunari Harada
Donato Hernandez Fusilier
Dieter Hutter
Laetitia Jourdan
Pinar Karagoz
Bruno Lara
Mikhail Mikhailov
Raul Monroy
Masaki Murata
Juan Carlos Nieves
Juan Arturo Nolazco Flores
Felipe Orihuela-Espina
Héctor Pérez-Urbina
David Pinto
Carlos Adolfo Piña-García
Natalia Ponomareva
Marco Antonio Ramos Corchado
Horacio Rodriguez
Abraham Sánchez López
Jose Santos
Manuel Vilares Ferro

Additional Reviewers

Roberto Alonso
Miguel Ballesteros
Santiago E. Conant-Pablos
Victor Darriba
Esteban Guerrero
Alev Mutlu
Hugo Terashima
Santanu Pal
Soujanya Poria
Daisuke Yokomori

Impreso en los Talleres Gráficos
de la Dirección de Publicaciones
del Instituto Politécnico Nacional
Tresguerras 27, Centro Histórico, México, D.F.
noviembre de 2013
Printing 500 / Edición 500 ejemplares