Volume 3, Issue 1
May 2016
ISSN: 1390-9266

LAJC

LATIN-AMERICAN JOURNAL OF COMPUTING

FACULTAD DE INGENIERÍA DE SISTEMAS

QUITO - ECUADOR

http://lajc.epn.edu.ec/

ESCUELA POLITÉCNICA NACIONAL

FACULTAD DE INGENIERÍA DE SISTEMAS

LATIN AMERICAN JOURNAL OF COMPUTING

LAJC

Vol III, Issue 1, May 2016
ISSN: 1390-9266
e-ISSN: 1390-9134

Published by:
National Polytechnic School
Faculty of Systems Engineering
Department of Informatics and Computer Sciences

Quito-Ecuador

LATIN AMERICAN JOURNAL OF COMPUTING - LAJC

Published by:

National Polytechnic School
Faculty of Systems Engineering
Department of Informatics and Computer Sciences
Ecuador

Mailing address:

Escuela Politécnica Nacional, Facultad de Ingeniería de Sistemas
Ladrón de Guevara E11-253, La Floresta
Quito - Ecuador, Apartado Postal: 17-01-2759

Web address:

http://lajc.epn.edu.ec/

E-mail:

lajc@epn.edu.ec

Frequency:

2 issues per year

Circulation:

500

EDITORIAL

Dear readers,

Welcome to the first issue of the Latin American Journal of Computing – LAJC in 2016. For this issue we are pleased to inform you that our Journal was indexed in Latindex Catalogue. Latindex Catalogue is a subset of the Directory, which contains a selection of journals that fulfill a series of international quality criteria. Titles are classified according to a previously agreed international quality parameters such as peer review procedures, coverage in international databases, abstracts and keywords in more than one language, international editorial boards, among a total of 33 parameters for printed journals and 36 for electronic journals. To date, more than 3,500 journals have been rated and included in this catalogue.

As LAJC maintain the electronic and the printed version, both of them passed through this review process where they met 33 parameters of 36 required for the electronic journal and 26 parameters of 33 required for the printed version. For more information, you can visit the URL: http://www.latindex.org/latindex/ficha?folio=25216

We would like to thank for this achievement to all the authors, who contributed to the success of the Journal. Special thanks to the reviewers for their contributions to keeping the high quality of the selected papers. Cordial thanks are due to the Section Editors members for their efforts and the organizational work. Finally, we cordially thank National Polytechnic School for supporting and publishing the Journal.

This first issue in 2016, includes five articles covering different aspects of Information Systems, Intelligent Systems and Software Engineering. We hope that you enjoy reading this issue and find the articles informative and useful.

Once again, we highly encourage you to submit your work within the scope of LAJC. Please keep in mind that we have an open call for submissions twice a year. For detailed instructions on the preparation and submissions of manuscripts, please check the URL below:

http://lajc.epn.edu.ec/index.php/LAJC/pages/view/call-for-papers

We will be happy to receive your comments and feedback on our journal.

Best Regards,

PhD Jenny Torres
Editor in chief LAJC

Latin American Journal of Computing - LAJC
Reviewers

We are most grateful to the following individuals for their time and commitment to review manuscripts for Latin American Journal of Computing - LAJC for this edition.

Aguiar Pontes Josafá, PhD. Escuela Politécnica Nacional, Ecuador

Aguilar José, PhD. Universidad de Los Andes, Venezuela

Buitrago Alex, PhD. Universidad Externado de Colombia, Colombia

Ferreira Vera, PhD. Universidade Federal do Pampa, Brasil

Fonseca Efraín, PhD. Universidad de las Fuerzas Armadas ESPE, Ecuador

Loza Aguirre Edison, PhD. Université Grenoble Alpes, France

Lucio Naranjo José Francisco, Escuela Politécnica Nacional, Ecuador

Magreñán Alberto, PhD. Universidad Internacional de la Rioja, España

Parsons Rebecca, PhD. ThoughtWorks Inc, United States

Sicilia Juan Antonio, PhD. Universidad Internacional de la Rioja, España

TABLE OF CONTENTS

A Social Framework for Set Recommendation in Group Recommender Systems

Lorena Recalde

***Abstract*—This research article presents a study about the background in *Group Recommender Systems* and how *social factors* are directly related to these applications. Some important group recommender systems in academia are described to exemplify their contribution in different domains. Besides, a framework that is intended to improve group recommender systems is proposed. The main idea of the framework is to enhance social cognition to help the group members agree and make a decision. Its structure includes a process where an influential group is detected among the target groups of people to recommend to. Social influence detection uses the knowledge behind online social connections and interactions. Trying to understand human behavior and ties among groups in a social network and how to use this to improve group recommender systems is considered the main challenge for future research. Combining this with the kind of item recommendation which involves a temporal sequence of ordered elements will present a novel and original path in Group Recommender Systems design.**

***Index Terms*— group preferences, group recommender systems, information propagation, social factors, social attraction.**

I. Introduction

THE adaptive web provides information sites where the users can be benefited from high degrees of personalization. For instance, e-commerce offers certain products to a specific user who actually needs or likes them, and a video player website profiles the users to extract the list of videos to present particularly to each of them. Today, online social networks customize the user's contact updates board depending on which of his/her friends the user is interested in knowing more about. Most of these websites have as part of their implementation a *recommender system*. For example, websites like Amazon, Netflix, Last.fm, Pandora, YouTube, etc. incorporate a recommender engine. The recommender is responsible for building the user interests model and finding the item or ranked list of items that best fits their preferences and needs. Therefore, the level of personalization increases when the recommender system knows more about the user.

The target user might be a *single user* or a *group of people*. Thus, considering the type of target user, the recommenders are classified in *Recommender Systems* and *Group Recommender Systems* respectively. This classification has been proposed since modeling the interests of a person is not the same as modeling the interests of a family, a sport team, a group of friends or a group of people who are sharing a room.

Particularly, the present article focuses on the analysis of literature in group recommender systems and the social factors involved in order to propose a framework that models the research findings in a way that improvements may be added. Group recommenders must be able to identify items that the group of users will like so that their needs, explicit or not, are equally satisfied. It can be implemented considering three different components: i) the nature of the target group to recommend to, ii) the kind of recommendation made (one item, an ordered set of elements, a bunch of items put together), and iii) external factors (groups dynamics, social influence, personality, tie strength and emotions) that may be considered when formulating the recommendation techniques used to match group - items.

Group recommender systems are relevant because activities like watching movies, eating in restaurants and having holiday trips are usually done by groups. Their main aim is to augment social cognition. As a result, the group members use the recommender because it is easy for them to agree and make a decision, it enhances the members' participation, it provides them with a strong sense of belonging and it offers reliable suggestions.

With the wide development and expansive use of Online Social Networks, researchers have realized that the technical and visual requirements that satisfied people's needs are not sufficient any more. Human Computer Interaction studies have gone further in order to analyze, prototype and evaluate society or *community* needs. The Social Web does not imply people interacting with a machine. It represents people interacting with people thanks to machines. Consequently, sciences like psychology and sociology play an important role in socio-technical systems design [1].

Considering the statements mentioned above, and the fact that we interact with people more than we consume a service or a product, it is justified the study of groups dynamics and other social factors in group recommender systems. In consequence, a new - social - approach in group recommenders design will be introduced in this paper, aiming to improve the user experience, from algorithms to interface.

Questions to guide the research are:

• Can groups of people be influenced by other groups at the moment they are making a decision? If yes, how to implement this social factor in a Group Recommender System?
• Should susceptible groups' preferences be model in a different

Lorena Recalde is a predoctoral researcher in the Department of Information and Communication Technologies, Universitat Pompeu Fabra, Barcelona, 08018 Spain, e-mail: lorena.recalde@upf.edu

way from influential groups model?
• Is it important to let susceptible groups visualize the influential group members, their choice and the reasons why they made that decision? Does it help or manipulate them?

The article is structured as follows. Section II details the context of the article and the state of the art in group recommender systems, the kinds of domains and approaches implemented as well as the social factors studied in recommenders. The social framework proposed and the methodology to embed it in a group recommender system is presented in Section III. Section IV presents the current challenges by associating the concepts and techniques, as well as the fundamental issues in recommender systems to the research that is needed for future work. To finish, Section V presents the conclusions of the article.

II. Context and State of the Art

In the light of the above overview about the main function of a recommender system, we could say that the popularity of those systems has increased because of their usefulness. In fact, there are three relevant reasons that justify the importance of those systems in the people's daily lives: i) we make choices about every aspect that is part of our lives all the time [2]; ii)the quantity of available online information about different alternatives, services or products is constantly increasing, so we have to rely on others' opinions and recommendations to make good decisions; and iii) computational systems were created with the aim of augmenting human cognition, so people can remember, think and reason in better ways [3]. Therefore, it is important to have systems that support the decision-making process.

The term item is a general word used to make reference to the object that the recommender system suggests. Accordingly, an item to recommend would be a singer, a movie, a restaurant, a Twitter user to follow or a Facebook friend to add. However, the recommendation might be not only a ranked list of independent items, from which the user selects, buys or adopts any of the items presented. It may be an ordered set of elements, where one item recommendation signifies some elements provided in a specific order, or a bunch of items put together having the notion of better together, so a bundle of two or more objects conforms the item recommendation [4]. The nature of the recommendation or the type of item recommended is usually determined by the system domain. The domain guides the design of the recommender system because the approaches and techniques to implement may differ depending on whether the system recommends a recipe, a medical treatment or a car to rent.

The approaches applied in recommender systems have evolved since mid-1990's. Many improvements to the algorithms and techniques have been published as a result of academia and industry research. The main approaches are:

- Collaborative Filtering. The algorithms use historical rating information to compare how similar the users' preferences are. The search of neighbors of the current user allows to recommend him/her items with high ratings provided by his/her peers.
- Content-based techniques. The recommender bases its suggestions on the degree of high previous acceptance of items which have the same features or attributes as the ones unseen by the user. Therefore, because of their similarity they may be recommended.
- Knowledge-based techniques. In these systems, there are knowledge bases about users and items. Most of the time the needs are elicited through conversational interactions between the user and a recommender assistant until discovering the item that has the desired characteristics.

The approaches mentioned have different variations and may be combined as a hybrid recommender system [5] in order to minimize their individual drawbacks. In recent years Context-Aware, Social-Based and Trust-Aware Recommenders have also emerged to present paradigms that the recommender systems developers may analyze to find which of the approaches best suits the requirements of the system.

Decisions about the design of the recommender have to be made after knowing the item to recommend or, in other words, once the domain is defined. However, knowing which target user to recommend to has the same importance. Group preferences modeling is a demanding task and it differs from single users modeling process. This section presents the previous work on group recommender systems, different kinds of items recommendation: single object, bunch of elements as well as temporal sequence of items, and social factors in the recommendation process. A summary with relevant information is detailed in Table I.

A. *Group Recommender Systems*

In context aware recommendations the system has to evaluate the present condition of the user, taking into account, for example, the localization, time, weather and company [6]. For instance, the idea of such a recommender system is to consider the preferences of the user, but when he or she is in the company of friends it should change the context of the user to adapt the recommendation for a group of people. Nevertheless, it is not a group recommender system. A group recommender system supports the recommendation process by modeling the preferences of a group of people generally, by using aggregation methods. This is needed when there is an activity (domain) that can be done or enjoyed as a group.

The aggregation methods to extract the group's interests to build the group's model work on combining the previous ratings of the individuals into a single group rating. In [7], Masthoff presents the evaluation of aggregation methods in an Interactive Television recommender system. She chose that domain considering that watching TV is the most frequent activity done in family and that the group of people usually have heterogeneous preferences. Her work presents the results obtained after studying how some aggregation strategies work, such as multiplicative, approval voting, least misery, most pleasure, fairness, and so on. She found that average, average without misery, and least misery are good candidates for implementation. Some aggregation methods and other ways to generate group recommendations are described with examples in [8].

One of the first group recommender systems was presented in [9]. Here the authors developed MusicFX, a recommender

TABLE I
GROUP RECOMMENDER SYSTEMS APPLICATIONS: EXAMPLES IN DIVERSE DOMAINS.

Group Recommender System	Domain	Nature of Items Recommendation	Groups of Users	Interests Extraction Method	How to predict the Item
Interactive Television	tv programs	temporal sequence	family	framework: explicit elicitation	affective state after watching a program
MusicFX	music station	one item	people working out in a gym	explicit elicitation of music genre preferences	rating scale -2, -1, 0, 1, 2
Pollens	movies	ranked list	any group	aggregation of ratings/ collaborative filtering	5 stars scale
INTRIGUE	tourist destinations	temporal sequence	family	explicit elicitation of preferences + socio-demographic information	contextual information: geolocation and schedules of activities
Poker Restaurant Finder	restaurants	ranked list	any group	explicit elicitation of preferences and needs priority classification	context information: current place of group
I-Spy	web pages	ranked list	communities of searchers	implicit feedback: queries and links selected in the past + similarity in the community	ranking pages function adaptation

that presents a music station to play for the group of people working out in a gym. The article states that an intelligent environment (a fitness center, a restaurant or a store) should respond to the group of people who are the current inhabitants by recognizing who they are and what preferences they have.

Therefore, the system can play the music the clients like. MusicFX obtains the information about the clients' interests from a database that stores their music genre preferences (previously and explicitly specified). With the use of an authentication mechanism, the system controls who are the clients that are in the gym at a given time. After applying an aggregation algorithm, the system computes the group ranking for each music genre, so it randomly chooses which music station to play among the top n ranked stations.[1]

The research work presented in [10] shows another group recommender system, INTRIGUE. This recommender was designed to offer personalized suggestions about tourist environs in a specific geographical area to constrain the location for the tour. It recommends multiple destinations to visit and itineraries considering the preferences of groups, such as families with heterogeneous kinds of members like children and elderly. The aggregation method differs from the one used in Interactive Television and MusicFX where the aggregation is done by extracting the individual *preferences* to finally have the group's interests model to be able to compute the recommendation. On the other hand, in INTRIGUE the individual *recommendations* for each member or for homogeneous subgroups are computed, and then they are aggregated to have the entire group's recommendation.[2] The system applies a variation of the average aggregation strategy. The weights depend on the size of the homogeneous subgroups and their relevance. That is to say, if there is a subgroup of children, they are more relevant or their recommendation weighs more when computing the whole group recommendation.

Similarly, in [12], McCarthy proposes the Pocket RestaurantFinder, that recommends restaurants to groups of people considering their culinary preferences and location. Specifically, the recommender uses information like travel distance, expected facilities, cuisine desired and budget planned. When using the recommender system, the group members have to express explicitly and individually the desired values for the four features, and they also need to order the features in a level of priority. Then, Pocket RestaurantFinder computes the recommendation by applying an average preferences aggregation method. The restaurants are displayed in a ranked list that matches the group's likes.

The system in [13] recommends web pages by exploring the implicit behavior of communities of searchers, where a community is defined as a group of users with similar information needs. The authors argue that if there are users with very similar information requirements, they send similar queries to the search engine. For the system, named I-SPY, it is important to extract the user preferences by considering the query repetition and selection regularity (which pages they click among the retrieved ones) measures in web search. If the search activity is performed within a well-defined context, let's say in a specific website search box, the set or community of users are known to have specific information preferences. As the community uses the search engine, the system will gradually

[1]There are 91 stations and each one is associated with a music genre.
[2]Recommendation aggregation or merging was also considered to be used in the PolyLens recommender [11], but for the domain this method presented significant drawbacks.

adapt the ranking pages function considering the historical data about a given query and the clicked results for it.

In [14], Jameson *et al.* detail the prototype of the group recommender Travel Decision Forum. The web-based system is designed to recommend places for vacation to some friends who will perform an asynchronous communication through the system in order to agree. The prototype makes use of an animated character who shows the potential trip options and plays the role of mediator. Its role is to help the group make a decision. The preferences concerning the vacation need to be expressed individually by every person in the group. Then, the recommender uses a preferences aggregation method to define the group's preferences as a whole. However, an important issue to be solved in the Travel Decision Forum is to allow each member to be aware, so every group member can visualize the others' preferences in the interface.

I have presented some group recommender systems and how they perform the recommendations in different application domains. Research works like [8], [15] and [16] detail more group recommenders, kinds of target groups, methods to compute the group recommendations and some explanations about the group recommendation process design. The authors agree that the existing challenges in group recommender systems are not similar to those seen in common recommender systems.

In fact, the problems that arise are harder to solve. In their works, they emphasize how important it is to do more research in social issues such as influence among the members and their attitude when deciding about an alternative, the affective state of the members of the group while enjoying a set of activities, and the nature of groups formed. For example, they might be an established, an occasional or a random group.

As was mentioned before, it is necessary to know what the recommendation is going to be. Consequently, the next section reviews the literature about the nature of the items to recommend.

B. *Items Recommendation*

The state of the art in recommender systems is very broad. However, it usually addresses the analysis and improvement of approaches like Collaborative Filtering [17], Content-Based [18], Constraint-Based [19] and Hybrid Recommender Systems [20] considering individual item recommendation to single users. The recommendation of items is generally presented as a ranked list of individual objects and the user can choose one item or another because they are independent. The web-based recommender system Movielens (www.movielens.umn.edu) [21], uses the collaborative filtering approach to predict the user rating for a movie. A set of ratings on already seen movies has to be provided by the user. Consequently, the system can recommend movies whose rating value predictions are high for the user.

In [22], a content-based recommender system, called Informed, is explained. The system creates an ontology for the items based on the previous consumer reviews. Natural Language Processing techniques and Text mining are applied to extract the features or attributes of the item and identify each of them as good or bad, according to users' opinions. For example, a photo camera may have good resolution and bad battery life. The system will give a weight for the features depending not only on the quality feature classification, but also on the degree of relevance of the feature for the user. The Informed system uses the expertise information about the user to compute the weights and then produces a ranked list of items that best suits his/her needs.

The research done in [23], shows a content-based movie recommender called Cinemappy. The application works in mobile devices and uses data extracted from DBpedia about each of the movies, as well as contextual information related to the current time and location of the user. The computation of similarity between movies is done taking into account shared features like same director, same genre, same stars, for example. Similarity helps to identify other movies that the user will like because they have common features with previously seen movies which the user Liked. Google Places and Trovacinema are websites which use available information to extract the contextual data. For example, Cinemappy will recommend a list of movies, including information like their genres and the cinema name where the movie is showing. Additionally, it will let the user know the distance from his/her current geographical location to the cinema, which is a constraint to produce the recommendation. However, the system can also show other cinemas that play the movie chosen by the user.

Little work has been done when a single recommendation is composed by some units, ordered or not. For example, in [24] the system creates a playlist of songs for the user. It is not static, so if a new song appears the playlist is reorganized considering the user model and then the personal recommendation of the new arranged set of songs is made. In this kind of recommenders, when the suggestion is made up of a sequence of elements, most of the time their position depends on the user interests or other constraints. Consequently, at the moment the user chooses an item, he/she has access to a set of ordered units. In [25] another similar system, patented by Amazon Technologies, is detailed. The algorithm output presents three or more items that work well together, so they are recommended as a bundle (in this case, without a specific order), refining the idea of better together.

The systems previously mentioned were developed taking into account that the target user is a single one.[3] Nevertheless, two of the group recommender systems studied in the preceding section show interesting items recommendation. For instance, the Interactive Television system [7] presents an ordered sequence of TV programs. The order is assigned by considering the preferences of the group and their affective state after having seen a program. The idea is to balance the satisfaction of the group members by ordering the programs

[3]Cinemappi is a recommender that handles contextual information, so the user has the possibility to tell the system that he/she wants to watch a movie with a friend; however, the system does not model the preferences of groups.

correctly. INTRIGUE [10] recommends multiple attractions to be visited by the group during their trip. It considers a sequence of places in the recommendation without any other restriction than the visits schedule.

We have seen important research works related to the most relevant issues in recommender systems: users (single users versus groups) and items (lists of individual items, bundles of items and sequences). Table I presents a summary of that information considering group recommender system examples.

Next, studies about social factors in the recommendation process are detailed.

C. Social Factors in Recommender Systems

Understanding the user's interests and needs is not enough to design a recommender system that takes into account the user experience. Visualization techniques, human cognition, social behavior, choice theory, persuasion, information diffusion and community formation are some of the concerns when implementing an application for the Web. In fact, *social recommender systems* or recommenders for the Social Web have emerged with the aim of modeling the user's preferences by using the information he or she and their friends have published in online social networks [26].

In [27] the authors propose a framework to merge behavioral theory and social recommender systems design. They make their proposal based on the argument that social and psychological theories may be employed as sources of Information Systems design principles. Therefore, in the authors' studies it is proved that homophily, tie strength, and trustworthiness leverage the recommendation acceptance (sociological view).

The researchers in [28], also model the preferences of the user in a social recommender, but they take into account that some of the user's friends might have different interests. Their argument is that we always look for our friends' recommendations, so in their work they establish the difference between trust relationships and social friendships. In a social network, a trust relationship is understood as users who may not know each other and there are unilateral connections. However, a social friendship reflects real and mutual relationships, so this kind of links are the important ones when implementing a social recommender system. In this work, the authors represent the diversity of tastes among the user's social connections (matrix factorization) to improve the accuracy of the recommendations.

In [29], the authors propose an approach for group recommender systems by merging Collaborative Filtering and a Genetic Algorithm that learns from known group ratings. The authors state that important social effects like opinion leadership, influence in thoughts, feelings, and actions as well as kinds of interactions, are present in group decision-making. Therefore, they need to be considered in group recommender systems. Actually, the social factor included in the framework proposed is the preceding interaction among group members reflected in their past ratings as an individual and also as subgroups. An interesting metric for the system is influential personality, that can be inferred from the ratings.

In [30], Quijano et al. study individual behaviors, group personality composition and trust relationships among members of the group to make recommendations for them. They propose a set of methods that could be integrated into any social network and then make recommendations like movies, restaurants, trips, etc. The system is able to infer social characteristics about the group members. For example, the approach evaluates parameters like collaborating, consensuating, evasion, and may assume a permissiveness personality; another parameter may be closeness between friends. Having these values, the system improves the group decision-making process.

Other recent works in group recommender systems have tested the way the recommendations are presented in the interface in order to prove that showing members emotions about the item can influence the user adoption. For instance, in [31], Chen *et al.* show that the integration of emotion cues in GroupFun, a recommender mobile system that suggests songs for a group of friends, might make each of them be aware of the others' preferences. The system generates playlists considering the aggregation of the members' ratings, but mutual awareness may influence the rating values when a user sees how his/her friends feel about the song.

III. Social Framework for Group Recommenders

The wide and quick spread of the use of online social networks is an evidence that the users not only need to contact other users and establish social connections, but also that they look for information generated by other people around. Therefore, a way to keep the social links with others is being aware of what they do, think, feel, share and buy. On one hand, human factors influence a person's decision. In fact, literature about personalization improvement in the recommendation process has shown that including personality as well as cognitive and learning style has a positive impact on the extraction of long-term preferences to design the user preferences model [32]. On the other hand, a user who has his online profile is able to interact with a huge number of people, and social factors arise showing that only some of his contacts influence his decisions, enrich his reasoning skills and provide additional knowledge through their online activity. Actually, the degree of social connections influence does not depend on the user personality, but on his desire to belong to a community.

This paper proposes a new recommendation framework to include information about influential groups' decisions (in the preferences model adaptation and in the interface) in a system where the target is a group of people and the recommendation is a sequence of ordered elements.

The research questions presented in Section I plus the current state of research seen in Section II are considered as guidelines to define the main components in the framework proposed (Figure 1).

A. Influential Group Identification

The detection of the influential group among the target groups to which recommend needs to implement both: the recognition of the groups' members who are known because of

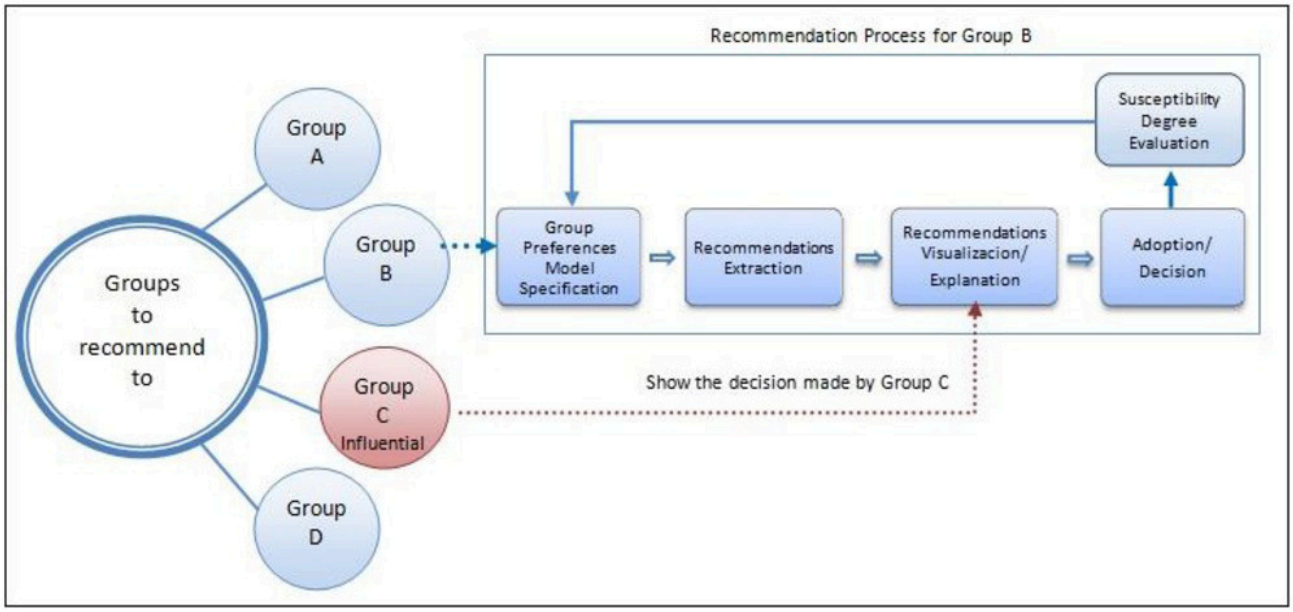


Fig. 1. After identifying the influential group/s and knowing its decision, the recommendation process for the Group B will let them know which was the choice of the influential group and why they chose that option. If group B decides to make the same choice, the model of Group B preferences will adapt his level of susceptibility.

their expertise background, sensibleness, trust and extroversion and information diffusion through communities by mining online social networks. Once having this information processed the extraction of main influential groups could be possible, presenting also the rest of groups that are susceptible to easily adopt a recommendation made by an influential group.

B. *Group Preferences Model and Adaptation*

What defines a group of people is their similarities, so that they could recognize the social category they belong to, and also the social categories they do not. A group has its social identity established when the members see themselves as a group. Self categorization theory says that when a person sets the differential parameters with other individuals, he sees himself with his own identity; on the other hand, when he is aware that he has a membership in a group he maximizes perceptually his similarities with the rest of members reducing in this way, their individual differences [33]. This fact will be considered at the moment of formulating the preferences aggregation method: the extraction of individual interests has a lower impact than the rate of items experienced before for the group as a whole, its current expectations, present goals and needs. The model should define the group identity in contrast to other groups. Actually, an influential group has a preferences model that includes different parameters than the susceptible groups, which have a model that adapts the parameter of *susceptibility* every time that they choose the influential group recommendation.

C. *Sequence of items prediction*

The kind of recommendation is planned to be a sequence of elements order in a way that all the group members enjoy the social activity. For example, the recommendation for group A

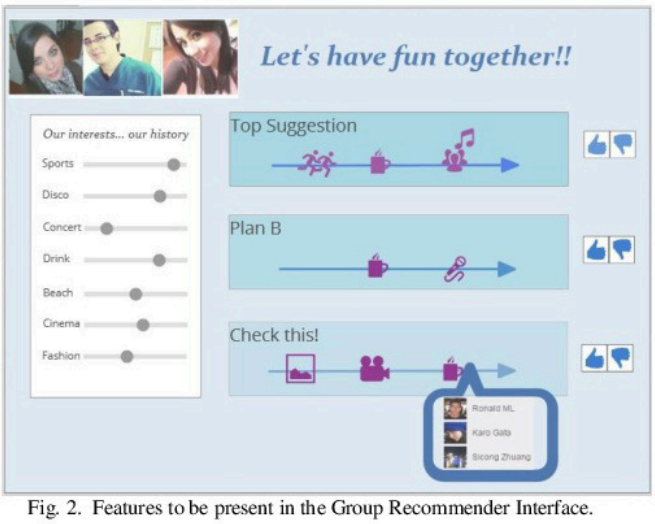


Fig. 2. Features to be present in the Group Recommender Interface.

could be: element p, then element q and them element r; while for group B it is: element p, element o and then element r. In fact, the preferences model of the group expresses the features needed or desired and the sequential integration of the elements recommended in a specific order should match those group needs. Generally, the approaches used depend on the domain of the recommendation: entertainment, content, e-commerce, service or social item. In the scope of the present research, the recommender system is thought to suggest leisure activities for a weekend with family or friends. That is to say, a sequence of leisure events which order is based not only in contextual information, but also in the preferences model and the estimated degree of acceptance of the recommendation of the influential group adoption.

TABLE II
METHODOLOGY ACCORDING TO THE FRAMEWORK COMPONENTS

Framework Component	Methodology
Influential Group Identification	Social Web Mining as well as Big Data and Information Diffusion Analysis
	Comparison Social Theories vs User studies and Log Data Analysis
Preferences Model and Adaptation	Social Web Mining and NLP (Sentiment Analysis)
	Preferences Elicitation Techniques
	IR techniques
	Preferences Aggregation and methods to include a Susceptibility Adaptation parameter
Sequence of Items Prediction	Contextual information extraction
	IR techniques
	User studies to evaluate order of elements
	Evaluation of the accuracy of the prediction
Recommendations Visualization	High-Fidelity Prototyping
	Usability tests
	User Centered Design Techniques

D. Recommendations Visualization

The goal of the Group Recommender System Interface is to support cooperative work in a way that the members of the group can be aware of one another needs but still they can see themselves as a whole, who have a common aim. Its design will be centered in characterizing the group interests and offer the option to see why one group they know (the influential one) chose a specific recommendation so that they could trust this is a good recommendation also for them (Figure 2). In this way, the interface pretends to implement a conflict resolution feature to help, in a non intrusive way, the group to make a decision faster. The recommender engine is half of the system; the other half is having groups using it to find the social activity that better matches their preferences.

The components of the framework discussed above require a methodology summarized in Table II.

IV. CHALLENGES AND FUTURE WORK

The state of the art review has described illustrative approaches, frameworks and systems that represent fundamental research in recommender systems. Nevertheless, it has been shown that there are social requirements that need to be addressed specifically in group recommender systems. Business logic as well as social factors have to be included in recommender algorithms and they also have to integrate ways to give importance to the user experience [34]. There are some challenges facing current concepts and techniques:

- Target User: group of people with heterogeneous interests, size evaluation.
- Nature of the Recommendation: algorithms to suggest a set of items in a sequential order depending on certain context to improve the process.
- Social Factors: Analysis about human behavior (emotions, personality, social identity, awareness) considering the inter-group level of interaction, as well as social behavior (influence, collaboration, curiosity) in the intra- group level.
- Group preferences model: scheme that evaluates preferences aggregation methods and inclusion of historical group ratings.
- User Experience: UX is designed in every phase of the recommendation, from algorithms to visualization.
- Interface: display of other groups' choices and explanations to facilitate agreement among the group.

The combination of those features is a novel approach that, if the pieces fit well, could improve group recommender systems. For future work, the implementation of the social framework shown in Section III in a group recommender system is planned. Its evaluation will be carried out assessing the four components: Influential Group Identification, Group Preferences Model and Adaptation, Sequence of items prediction and Recommendations Visualization.

V. CONCLUSIONS

Previously, I made evident the effort that has been invested in studying social factors to improve recommender systems for single users or groups. Nevertheless, group recommender systems research is scarce compared to the great improvements found in personal recommender systems. The assumptions that explain the reasons are:

- Recommender systems need users' information as input to build the user interests model. This information can be gathered by processing the explicit actions of the users (ratings, opinions, purchases) and/or their implicit feedback (search queries, item navigation through, clicks) [4]. There are datasets with this information for the Web activity of single users, but it may be a challenge to find datasets in academia about ratings given by groups as a whole [35].
- By mining online social networks it is possible to analyze the users' behavior and know who are influential persons [36]. On the other hand, it is needed to make studies about detecting influential communities in the Social Web but that can be applied in a group recommender system.
- Social factors have been investigated in group recommenders by taking into consideration the intra-group level (between the group members) but not in the inter-group level (among groups) because there has been a gap separating sociological theories and computer science until the Social Web arrival. Each framework component faces specific challenges and needs to be implemented by defining its own methodology and techniques. Some of them will have psychological and sociological information as input, others will need to be tested by applying more than one approach and algorithm combinations.

Understanding the nature of a group, their dynamics, how they are formed, size of influential groups and the ways they interact by using online social networks is the first issue to address.

REFERENCES

[1] B. Whitworth and A. Ahmad, *The Social Design of Technical Systems: Building technologies for communities*, 2nd ed. Aarhus C, Denmark: The Interaction Design Foundation, 2014.

[2] P. Resnick and H. R. Varian, "Recommender systems," *Commun. ACM*, vol. 40, no. 3, pp. 56–58, Mar. 1997.

[3] E. H. Chi, "The social web: Research and opportunities," *Computer*, vol. 41, no. 9, pp. 88–91, 2008.

[4] F. Ricci, L. Rokach, and B. Shapira, "Introduction to recommender systems handbook," in *Recommender Systems Handbook*, F. Ricci, L. Rokach, B. Shapira, and P. B. Kantor, Eds. Springer US, 2011, pp. 1–35.

[5] R. Burke, "Hybrid web recommender systems," in *The Adaptive Web*, ser. Lecture Notes in Computer Science, P. Brusilovsky, A. Kobsa, and W. Nejdl, Eds. Springer Berlin Heidelberg, 2007, vol. 4321, pp. 377– 408.

[6] G. Adomavicius and A. Tuzhilin, "Context-aware recommender systems," in *Recommender Systems Handbook*. Springer US, 2011, pp. 217–253.

[7] J. Masthoff, "Group modeling: Selecting a sequence of television items to suit a group of viewers," in *Personalized Digital Television*, ser. Human-Computer Interaction Series. Springer Netherlands, 2004, vol. 6, pp. 93–141.

[8] A. Jameson and B. Smyth, "Recommendation to groups," in *The Adaptive Web*, ser. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2007, vol. 4321, pp. 596–627.

[9] J. F. McCarthy and T. D. Anagnost, "Musicfx: An arbiter of group preferences for computer supported collaborative workouts," in *Proceedings of the 1998 ACM Conference on Computer Supported Cooperative Work*, ser. CSCW '98. New York, NY, USA: ACM, 1998, pp. 363–372.

[10] L. Ardissono, A. Goy, G. Petrone, M. Segnan, and P. Torasso, "Intrigue: Personalized recommendation of tourist attractions for desktop and hand held devices," *Applied Artificial Intelligence*, vol. 17, no. 8-9, pp. 687–714, 2003.

[11] M. OConnor, D. Cosley, J. Konstan, and J. Riedl, "Polylens: A recommender system for groups of users," in *ECSCW 2001*. Springer Netherlands, 2001, pp. 199–218.

[12] J. F. McCarthy, "Pocket restaurant finder: A situated recommender system for groups," in Proceeding of Workshop on Mobile Ad-Hoc Communication at the 2002 ACM Conference on Human Factors in Computer Systems, 2002.

[13] B. Smyth, E. Balfe, J. Freyne, P. Briggs, M. Coyle, and O. Boydell, "Exploiting query repetition and regularity in an adaptive community-based web search engine," *User Modeling and User-Adapted Interaction*, vol. 14, no. 5, pp. 383–423, Jan. 2005.

[14] A. Jameson, S. Baldes, and T. Kleinbauer, "Enhancing mutual awareness in group recommender systems," in *Proceedings of the IJCAI 2003 Workshop on Intelligent Techniques for Web Personalization*, B. Mobasher and S. S. Anand, Eds. Menlo Park, CA: AAAI, 2003.

[15] J. Masthoff, "Group recommender systems: Combining individual models," in *Recommender Systems Handbook*. Springer US, 2011, pp. 677–702.

[16] L. Boratto and S. Carta, "State-of-the-art in group recommendation and new approaches for automatic identification of groups," in *Information Retrieval and Mining in Distributed Environments*, ser. Studies in Computational Intelligence. Springer Berlin Heidelberg, 2011, vol. 324, pp. 1–20.

[17] B. Sarwar, K. G., K. J., and R. J., "Item-based collaborative filtering recommendation algorithms," 2001, pp. 285–295.

[18] M. Pazzani and D. Billsus, "Content-based recommendation systems," in *The Adaptive Web*, ser. Lecture Notes in Computer Science. Springer Berlin Heidelberg, 2007, vol. 4321, pp. 325–341.

[19] A. Felfernig and R. Burke, "Constraint-based recommender systems: Technologies and research issues," in *Proceedings of the 10th International Conference on Electronic Commerce*, ser. ICEC '08. New York, USA: ACM, 2008, pp. 1–10.

[20] R. Burke, "Hybrid recommender systems: Survey and experiments," *User Modeling and User-Adapted Interaction*, vol. 12, no. 4, pp. 331–370, 2002.

[21] B. J. Dahlen, J. Konstan, J. Herlocker, N. Good, A. Borchers, and J. Riedl, "Jump-starting movielens: User benefits of starting a collaborative filtering system with "dead date"," 1998.

[22] S. Aciar, D. Zhang, S. Simoff, and J. Debenham, "Informed recommender: Basing recommendations on consumer product reviews," *Intelligent Systems, IEEE*, vol. 22, no. 3, pp. 39–47, May 2007.

[23] V. C. Ostuni, T. D. Noia, R. Mirizzi, D. Romito, and E. D. Sciascio, "Cinemappy: a context-aware mobile app for movie recommendations boosted by dbpedia," in *Proceedings of the International Workshop on Semantic Technologies meet Recommender Systems & Big Data, Boston, USA, November 11, 2012*, 2012, pp. 37–48.

[24] S. Ward, "System and method for creating dynamic playlists," 2003, uS Patent 6,526,411.

[25] G. Chanda, B. Smith, and R. Whitman, "Automated selection of three of more items to recommend as a bundle," 2013, uS Patent 8,438,052.

[26] I. Guy and D. Carmel, "Social recommender systems," in *Proceedings of the 20th International Conference Companion on World Wide Web*, ser. WWW '11. New York, USA: ACM, 2011, pp. 283–284.

[27] O. Arazy, N. Kumar, and B. Shapira, "A theory-driven design framework for social recommender systems," *Journal of the Association for Information Systems*, vol. 11, no. 9, pp. 455–490, 2010.

[28] H. Ma, D. Zhou, C. Liu, M. Lyu, and I. King, "Recommender systems with social regularization," in *Proceedings of the Fourth ACM International Conference on Web Search and Data Mining*, ser. WSDM '11, New York, USA, 2011, pp. 287–296.

[29] Y. Chen, L. Cheng, and C. Chuang, "A group recommendation system with consideration of interactions among group members," *Expert Systems with Applications*, vol. 34, no. 3, pp. 2082 – 2090, 2008.

[30] L. Quijano-Sanchez, J. Recio-Garcia, B. Diaz-Agudo, and G. Jimenez-Diaz, "Social factors in group recommender systems," *ACM Transactions on Intelligent Systems and Technology, vol. 4, no. 1, pp. 8:1–8:30, Feb. 2013.*

[31] Y. Chen, X. Ma, A. Cerezo, and P. Pu, "Empatheticons: Designing emotion awareness tools for group recommenders," in *Proceedings of the XV International Conference on Human Computer Interaction*, ser. Interaccion '14. New York, NY, USA: ACM, 2014, pp. 16:1–16:8.

[32] R. Hu and P. Pu, "Enhancing collaborative filtering systems with personality information," in *Proceedings of the Fifth ACM Conference on Recommender Systems*, ser. RecSys '11. New York, USA: ACM, 2011, pp. 197–204.

[33] J. Turner, P. Oakes, S. Haslam, and C. McGarty, "Self and collective: Cognition and social context," *Personality and Social Psychology Bulletin*, vol. 20, no. 5, pp. 454–463, 1994.

[34] J. Konstanand, J. Riedl, "Recommender systems: from algorithms to user experience," *User Modeling and User-Adapted Interaction*, vol. 22, no. 1-2, pp. 101–123, 2012.

[35] S. Shang, Y. Hui, P. Hui, P. Cuff, and S. Kulkarni, "Beyond personalization and anonymity: Towards a group-based recommender system," in *Proceedings of the 29th Annual ACM Symposium on Applied Computing*, ser. SAC '14. New York, NY, USA: ACM, 2014, pp. 266–273.

[36] N. B. Ellison, J. Vitak, R. Gray, and C. Lampe, "Cultivating social resources on social network sites: Facebook relationship maintenance behaviors and their role in social capital processes," *Journal of Computer- Mediated Communication*, vol. 19, no. 4, pp. 855–870, 2014.

Lorena Recalde received the B.S. degree in computational systems engineering from National Polytechnic School (EPN), Quito, Ecuador, in 2008. From 2010 to 2012 she worked as teacher assistant at the EPN. She received the M.S. degree in computer science from Fribourg University, Fribourg, Switzerland, in 2014. In the same year she started her Ph.D. studies in information and communication technologies at the Pompeu Fabra University, Barcelona, Spain, where currently, she is doing research in the field of Group Recommender Systems.

Comparison of Clustering Algorithms for the Identification of Topics on *Twitter*

Marjori N. M. Klinczak and Celso A. A. Kaestner

***Abstract*—Topic Identification in Social Networks has become an important task when dealing with event detection, particularly when global communities are affected. In order to attack this problem, text processing techniques and machine learning algorithms have been extensively used. In this paper we compare four clustering algorithms – k-means, k-*medoids*, DBSCAN and NMF (Non-negative Matrix Factorization) – in order to detect topics related to textual messages obtained from *Twitter*. The algorithms were applied to a database initially composed by *tweets* having hashtags related to the recent Nepal earthquake as initial context. Obtained results suggest that the NMF clustering algorithm presents superior results, providing simpler clusters that are also easier to interpret.**

***Index Terms*—text processing; clustering algorithms; NMF algorithm; Twitter topics identification.**

I. Introduction

Social Networks are naively defined as a way for one person to meet up with other people on the Net. They constitute a global phenomenon, and are employed for several activities, such as work, entertainment and personal use [24]. They are also a huge source of information, reflecting peoples' opinions and desires, and serve as an almost instantaneous channel for communication and spreading news [23].

However, to extract useful information from the texts that appear in the social networks is not an easy task, due to the huge size of the data involved and the speed of their creation [25]. The problem is only recently being attacked, employing automatic procedures whose fundamentals include Text Processing (TP) [1, 2] and Machine Learning (ML) [3] techniques.

The *Twitter* is a microblog created in 2006 and widely used over the world. Nowadays it contains more than 465 million of accounts, and its messages form a textual database where discussions and opinions of several matters can be found [17]. Also, it contains information about on-time events of many types and scales [5]. The high connectivity and the almost instantaneous responses entails that this social network is the one where the information travels faster [23].

Therefore, the *Twitter* can be considered as a real-time source of information [6]; this is especially true in the case of global, catastrophic and/or big media events. In this paper we discuss the automatic extraction of topics - a set of cohesive terms related to a specific subject – that appear in the *tweets* obtained from a broad initial context given by a list of hashtags.

The topics are obtained as a result of a clustering procedure as follows:

- initially the tweets are converted in plain text (some metadata are also stored for additional use);
- the obtained texts are preprocessed using classical techniques such as case conversion, stop-words removal and stemming; *urls*, *retweets* and profile information are also removed, this steps are showed at figure 1.
- a (*tweet* x term) matrix – which corresponds to the (document x term) matrix in the Information Retrieval area – is obtained according to the well-known Vector Space Model [1];
- the clustering algorithms are applied to this matrix; each obtained cluster is associated to a topic;
- the quality of the obtained clusters is considered in two ways: using the intra / inter cluster measures and using a word cloud associated to each cluster.

The overall clustering procedures were tested using *Twitter* data related to the recent Nepal's Earthquake.

The rest of this paper is organized as follows: section 2 presents similar works that deal with event extraction from social networks; section 3 describes the text preprocessing techniques used, the obtained the (*tweet* x term) matrix, and the employed clustering algorithms; section 4 presents the testing cases and discusses the obtained results; finally, section 5 presents the conclusions and future work.

II. Similar Works and Related Research

In the literature, there are several works dealing with the identification of topics that appear on Social Networks. Related applications range from real time event detection, the impact of natural disasters, opinion mining and the identification of diseases for public health actions [6]. Some of these works are briefly summarized in the following.

Shamma, Kennedy and Churchill [7] use as research scenario the debate between Barack Obama e John McCain that occurs in September 26th, 2008, during the national campaign for the USA presidency. They investigate the practice of sharing short messages (microblogging) around live media events. A

Marjori N. M. Klinczak is with the Mosaic Web Company, and is currently a student at the Graduate Program in Applied Computer Science at the Federal University of Technology of Paraná (UTFPR), Curitiba, Paraná, Brazil (e-mail: mnmk.lvseg@gmail.com).

Celso A. A. Kaestner is a senior professor at the Graduate Program in Applied Computer Science at the Federal University of Technology of Paraná (UTFPR), Curitiba, Paraná, Brazil (e-mail: celsokaestner@utfpr.edu.br).

reactions' database was obtained from *Twitter*, considered as the act of live annotation of a broadcast media event. *Tweets* that contain the hashtags #*current*, #*tweetdebate* and #*debate08* were recorded, generating a database with 3,238 *tweets* from 1,160 different users. The traffic volume per minute was computed, as well as a network graph of all users and their tag relations as seen when clustered by tags; half of them used the #*current* tag when discussing the debates during air time. The authors hypothesized that frequent terms from *Twitter* traffic would reflect the topics being discussed; this was tested by breaking the debate into nine pieces and by computing the corresponding topic segments. They also show that the usage of twitter was not one of summarizing or even discussion about the debate on hand. Finally, they conclude that *Twitter* traffic can provide insights into segmentation and entity detection, however, the correlation between content leaves further questions to be investigated.

Sakaki, Okazaki and Matsuo [8] investigate if a real-time event can be detected only by monitoring *Twitter* activity. For example, when an earthquake occurs, there are many *tweets* related to this event, which enables its occurrence promptly, simply by observing these *tweets*. They use data extracted in Japan from this social media using the terms *earthquake*, *shaking* and *typhoon*. Then they use the size of the *tweet*, textual attributes given by a set of keywords and their context as attributes, and a Support Vector Machine (SVM) classifier. Subsequently, they produce a probabilistic spatiotemporal model for the center of the target event, by using the geographic location of the emitted tweets. In summary, they consider Twitter as a sensor network, and use *Kalman* filtering and particle filtering to provide location estimation, as widely employed in ubiquitous/pervasive computing. The paper contains detailed experimental results proving that their approach is feasible. They also propose a system that can detect an earthquake with high probability – 96% of earthquakes of Japan Meteorological Agency (JMA) seismic intensity scale 3 or more are detected – merely by monitoring tweets. This system also sends e-mails to registered users, and this notification is delivered much faster than the announcements that are broadcast by the JMA.

Becker et al. [5] argue that microblogging on social media have emerged as a powerful real-time mechanism to detect events. Due to its nature – short messages almost instantaneously propagated in the web – *Twitter* is particularly well suited as a source of theses contents. The authors focus their work in analyzing the stream of *tweets* to distinguish between messages about real world events and non-event messages. To do so they use an incremental, online clustering algorithm in order to effectively cluster a stream messages in real time. Employed features include temporal (e.g. traffic-volume), social (e.g. *retweets*), topical (e.g. term cohesion) and *Twitter* centered features (e.g. tag usage). They test a variety of classifier using the *Weka* platform[1] and the SVM (Support Vector Machines) algorithm. The conducted evaluation uses the macro-averaged F1 metric and precision for k clusters [9], obtaining and F1 score of 0.837 in the test set, and a precision superior of 80 % in the best clustering case (K=5).

Gupta and Kumaraguru [6] study the credibility of the information that is found in the *Twitter* messages. They use data related to 14 high impact global events of 2011, including for example the UK Riots, the Libya crisis, an earthquake in Virginia and the hurricane Irene. From the analyzed data, on average only 30% of the posted *tweets* related to an event really contain situational information about the event, while 14 % were merely spam. In addition, only 17 % of the total *tweets* posted contain situational awareness information that is credible. The authors use regression analysis to identify the important content and source-based features, in order to predict the credibility of a *tweet*. Employed features include the number of unique characters, swear words, pronouns, and emoticons in the text, and user based features like the number of followers and length of the username. A supervised machine learning procedure and the relevance feedback approach were used to rank *tweets* according to their credibility score. This performance evaluation has proved to significantly enhanced the results, allowing the automatic extraction of credible information from *Twitter*.

Godfrey [10] analyze the *Twitter* data during the FIFA World Cup. They employ cluster analysis and text-mining to extract underlying patterns from a database composed by large collections of text messages. A collection of about 30,000 *tweets* were extracted just before the 2014 World Cup started. To eliminate spurious tweets, unrelated to the main theme, they use an algorithm that combined the DBSCAN algorithm and a consensus matrix. Then the authors perform cluster analysis using k-means [3] and the Non-Negative Matrix Factorization (NMF) algorithm. Obtained results were very similar but, according to the authors, the NMF proved to be faster and provided results that are more easily to interpret. Result comparison in the paper is subjective, using graphics and figures from two visualization tools, *Gephi* [2] and *Wordle* [3].

Another study involving FIFA was done by Klinczak and Kaestner [12]; this study is related to the recent corruption scandal in the FIFA federation. Differently from other works, they are compared directly the performance of several clustering algorithms (k-means, k-*medoids* and NMF) in the same data obtained from *Twitter*, using the hashtags #*fifa* and #*fifagate* as initial context; after the text preprocessing the dataset has 2,460 *tweets*. The employed algorithms present similar results, but the NMF algorithm presents the best results in most of the cases. This can be partially explained because in the NMF algorithm the same term can be appear in many clusters with different weights.

The above research works make clear the importance of social networks – and particularly micro-bloggers as *Twitter* – in the detection and analysis of real-time events. Cluster analysis is frequently employed, but a direct comparison of the available clustering algorithms cannot be easily found. Also, only the last two works employ the NMF algorithm, considered

[1] http://www.cs.waikato.ac.nz/ml/weka/
[2] https://gephi.org/
[3] http://www.wordle.net/

nowadays the best technique to perform clustering in text applications. In the paper we address this research task.

III. Employed Techniques: Text Preprocessing and Clustering

To use *Twitter* data for Information Extraction purposes a series of preprocessing steps must be followed. Initially, the *tweets* are recorded using a specific API using the R language[4]; it includes several filters, such as the presence of specific hashtags, the language employed in the message, geographic location restrictions, information about *retweets*, etc. Besides the text message itself, the obtained record includes meta-attributes such as the user-id, time/date of the *tweet*, geographical location of the emitter, and some metadata like links and images, that can be used for specific purposes.

A. Preprocessing and Text Model

Text preprocessing is a very important step to obtain the semantic elements related to the message. The use of techniques originally employed in Information Retrieval (IR) [1, 2] is convenient for this task.

In the case of *Twitter* messages additional elements appear: due to the small size of the message users extensively use abbreviations and emoticons, introducing some noise in the pure text model.

The classical text preprocessing steps are (se also Figure 1):

- text unit identification: in this case the *tweet* textual information is considered the basic unit;
- case-folding: to standardize the extracted characters; it can include additional conversions because many tweets have strange characters;
- stop-words removal: stop-words are very frequent textual elements that carry almost no semantics and can be eliminated; a stop-word list includes articles, prepositions and conjunctions; in some applications, like that, numbers are also eliminated. Also is eliminated the initial hashtags and the noise like the emoticons and abbreviations.
- stemming: is a procedure that aims to connect textual elements of similar semantics, by obtaining their *root*; suffixes and prefixes are eliminated, plurals and verbal variations of the same term are reduced to a unique form.

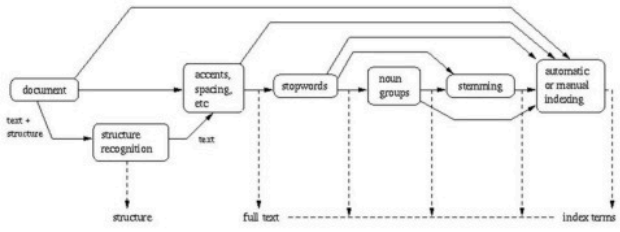


Fig. 1. *Text Preprocessing Steps [1]*

As result of the preprocessing step, each *tweet* is now a series of text elements, usually called indexing terms in the IR terminology. By computing the union of these terms, we have – after defining an order – a global list of terms of the database.

Then a text model must be used: the most employed model nowadays is the Vector Space Model (VSM) [1] where each term corresponds to a dimension in a huge *NT*-dimensional vector space, *NT* being the number of terms. Obviously, each tweet will contain only few terms: the message collection is therefore very sparse in this space. It is usual to view the text collection as a huge matrix *D* = (text unit X term), or *D* = (*tweet* x term) in this case.

Given a pair (*tweet*, term) the corresponding *D* = *(d, t)* entry is the weight of the term *t* in the *tweet d*. Several weighting schemes can be employed: the simplest is the Boolean model, where 0 is used for absence and 1 for presence of the term *t* in the *tweet d*; more employed schemes include the frequency model, where the weight if the frequency *tf* of the term *t* in the tweet *d*, or the *tf-idf* (term frequency - inverse document frequency) model, where the weight of the term *t* in the *tweet d* is given by:

$$tf\text{-}idf\,(d, t) = tf\,(d, t) * log(||ND|| / df\,(t)) \quad (1)$$

where $||ND||$ is the total number of documents, *tf* is the frequency of the term *t* in the *tweet d* and *df* stands for the number of documents in which the term *t* appears.

B. Clustering algorithms

After text processing the obtained (*tweets* x terms) *D* matrix form the base for cluster computations, following the classical scenario employed in ML. So, several clustering algorithms can be readily employed.

a. k-means

An oldest option is to use the well-known k-means algorithm [13]. Briefly, it works as follows: (a) a series of k initial points are randomly generated; (b) these points are considered as cluster centers (or means); (c) each text instance is used as input: it will be assigned to the cluster with closest center; (d) the value of the cluster mean is updated to consider this new cluster element; (e) steps (c) to (d) are repeated until no changes occur in the instance cluster labels (the cluster assigned to it) [13].

The employed metric for distance is very important: in the case of text documents and following the VSM, it is common to use one of two metrics: the classical Euclidian distance or the cosine similarity measure, given, for documents d_1 and d_2, by:

$$dist\,(d_1, d_2) = < d_1, d_2 > / || d_1||.|| d_2|| \quad (2)$$

where <, > stands for the dot vector product, and $||d||$ is the norm of the document *d*.

b. k-medoids

The k-*medoids* algorithm is similar to the k-means, the only difference being the fact that in this case the mean of each cluster is replaced by its "*medoid*", the most central existing data point. That is, in the k-means centers of the clusters are the means of theirs points, not necessary a data point, whereas in

[4] https://www.r-project.org/

the k-*medoids* algorithm these values are chosen to be existing data-points [13].

c. Density-Based Spatial Clustering of Applications with Noise (DBSCAN)

Another widely used cluster algorithm that uses a different principle is DBSCAN [13]. Its basic technique is to connect all the high density regions of the underlying space, the low density regions being the inter-cluster space.

In summary, DBSCAN works as follows: (a) a user-defined parameter $\varepsilon > 0$ is used to specify the radius of a neighborhood for every object; the ε-neighborhood of an object O is the space within a radius centered at O; (b) the density of a neighborhood can be measured simply by the number of objects in the neighborhood; to determine whether a neighborhood is dense or not, DBSCAN uses another parameter *MinPts*, which is a density threshold; (c) a core object is one where its ε-neighborhood contains at least *MinPts* objects; they are the pillars of dense regions; (d) after computing core objects, the clustering task is reduced to the use of core objects and their neighborhoods to form dense regions which are the clusters; (e) for a core object O and an object P, we say that P is directly density-reachable from O if it is within the ε-neighborhood of O; (f) using the directly density-reachable transitive relation, a core object can connect objects to form a dense region.

Here again, the metric employed to evaluate the distance between instances is crucial, the cosine distance being the most employed one for text applications.

d. Non-negative Matrix Factorization (NMF)

The NMF method was initially proposed by Lee e Seung [14] as an alternative for the Principal Component Analysis (PCA) method, which is classically used in matrix decompositions.

To remember, PCA is an orthogonal linear operator that transforms the data to new coordinates, such that the greatest variance by some projection of the data lies on the first coordinate (the first principal component), the second greatest variance lies on the second coordinate, and so on. That is, given an $(m \times n)$ matrix M, PCA computes $M = W \cdot \Sigma \cdot W^T$, where Σ is a diagonal matrix of the principal components (sorted by magnitude), and W is formed by the eigenvectors' coordinates. To perform data reduction the sub-matrix of size $(k \times k)$ of Σ, usually noted as Σ_k, is commonly employed [15].

PCA has been successfully used in text applications, but negative values that appear in the decomposition are difficult to interpret and sometimes contradict the reality. In NMF decomposition, on the other hand, non-negativity is preserved, making the resulting matrices easier to inspect, especially in applications such as text processing, where the non-negativity is inherent to the data being considered.

In NMF the original $(m \times n)$ matrix D is decomposed as $D \approx W \cdot H$, where W and H have dimensions $(m \times k)$ and $(k \times n)$, respectively, and k is a user-defined parameter that depends on the application; in our case, it is associated to the number of considered *tweet* topics.

The decomposition $D \approx W \cdot H$ for given k is not an exactly solvable problem in general, so it is commonly approximated numerically. Given the $(m \times n)$ matrix D and a positive integer $p < min$ (m, n), find two non-negative matrices H and W that minimize the functional:

$$f(W, H) = (1/2) \cdot ||D - W \cdot H||^2 \quad (3)$$

where $||\_||$ is a matrix Frobenius norm, and all the elements of W and H must be positive or zero, that is, w_{ij} , $h_{ij} \geq 0$.

Several procedures have been used to solve this optimization problem, such as multiplicative update algorithms, gradient descent algorithms and alternate least square algorithms (ALS). These algorithms are summarized by Berry et al. [16], that also deal with algorithm performance in large datasets.

The NMF clustering algorithm has been employed successfully, mainly because it can be adapted to specific applications [11, 16].

IV. Clustering Comparison

We did some experiments to compare the performance of the described clustering algorithms: k-means, k-*medoids*, DBSCAN and NMF. The employed database use *Twitter* data obtained in May 19, 2015, using the two hashtags "*#NepalEarthquake*" and "*#NepalQuake*" as initial context. We use the Twitter API [4] from its R Language interface, initially obtaining 10,000 tweets, restricted to the English language. Some of them were discarded because they were not in English or due to the presence of unknown characters; also, due to performance issues, we restrict the current analysis to dataset of 500 *tweets*.

Text preprocessing follow the steps described in the previous Section. Our basic text unit is the message part of each *tweet*; we perform case folding and characters standardization, and use the stop-words list obtained from the work of [18], with some additional element such as "*RT*". For stemming we use the well-known Porter's stemming algorithm [19], and we only considered terms with more than two characters. The final (*tweet* x term) matrix was constructed using term frequency; in our experiments its dimension is (500 x 1203).

All clustering algorithms were executed using the R Language version 3.0.2. The employed metric was the Euclidian distance.

In order to compare the clustering algorithms, we use the same value of k for the k-means and for the k dimension of the NMF. We present the results for k = 3, 5 and 7. DBSCAN results are not directly comparable, but are given for reference; it obtains 4 clusters.

Obtained results are summarized in the following. To compare the results, we use the clustering measures separation (BSS) and cohesion (WSS), given by the formulas [20]:

$$BSS = \Sigma_i \, ||C_i|| \cdot (m - m_i)^2 \quad (4) \text{ and}$$

$$WSS = \Sigma_i \; \Sigma_{x \, \epsilon \, C_i} (x - m_i)^2 \quad (5)$$

where $||C_i||$ and m_i stands for the size and the mean of the cluster i, and m is the global mean of the dataset and x is a data point that belongs to the cluster C_i.

Table I presents the results for the case of the k-means algorithm: the first column indicates the number of clusters; the second column gives the intra-cluster measure or cohesion WSS; the third one indicates the inter-cluster measure or separation BSS; and the last column gives the number of instances that occur in each cluster [13, 22].

Similarly, Table II presents the results obtained with the k-*medoids* algorithm.

Results for the DBSCAN algorithm are summarized in Table III. As it is a density-based algorithm, its performance cannot be adequately measured using intra and inter-cluster measures. So, we indicate in Table III the employed parameters and obtained cluster characteristics. The first column indicates the neighbor proximity parameter; the second column shows the number of obtained clusters; third and fourth ones present the number of seed and border points for each cluster respectively. In all the experiments the minimum number of points (*MinPts*) is set to 0.2.

TABLE I. RESULTS FOR K-MEANS ALGORITHM

K: #of clusters	*WSS*	*BSS*	*||Ci||*
3	487.25 3,109.20 675.48	731.34	8 1,172 23
5	2,851.20 487.25 291.44 21.17 337.64	1,014.59	1,162 8 16 6 11
7	184.00 0.00 2,420.00 256.35 272.85 131.14 340.73	1,397.99	6 1 1,126 17 20 7 26

TABLE II. RESULTS FOR K-*MEDOIDS* ALGORITHM

K: #of clusters	*WSS*	*BSS*	*||Ci||*
3	4,545.73 2,445.67 660.34	702.00	1,183 12 8
5	4,529.16 3,038.71 62.42 63.10 2,539.19	1,129.00	1,162 8 16 6 11
7	2,532.10 3,030.00 63.15 3,401.13 2,529.70 0.00	2,592.92	1,149 11 12 7 113 10 1

In the case of the NMF algorithm, we use the default multiplicative update algorithm; this is not a deterministic algorithm, so different executions can provide different results. We recall that the original (*tweet* x term) matrix D is decomposed in the matrices (tweet x topic) W and (topic x term) H. So, we can analyze clusters related to *tweets* and related to terms. We compute cohesion and separation for both options considering that each one of the k lines of the H matrix is the "center" of a cluster; similarly, each one of the columns of the W matrix is considered also the "center" of a cluster.

TABLE III. RESULTS FOR THE DBSCAN ALGORITHM

ε	*# of clusters*	*# of seed points*	*# of border points*
2	5	1,076 11 8 6 5	14 2 1 0 2
3	4	1,146 6 5 0	2 0 0 44
4	2	1,172 1	22 4

In the case of the NMF algorithm, we use the default multiplicative update algorithm; this is not a deterministic algorithm, so different executions can provide different results. We recall that the original (*tweet* x term) matrix D is decomposed in the matrices (tweet x topic) W and (topic x term) H. So, we can analyze clusters related to *tweets* and related to terms. We compute cohesion and separation for both options considering that each one of the k lines of the H matrix is the "center" of a cluster; similarly, each one of the columns of the W matrix is considered also the "center" of a cluster.

Tables IV and V indicate the values of cohesion, separation and cluster size for 3, 5 and 7 clusters. Results for NMF-*tweets* are comparable to the ones obtained for k-means and k-*medoids*; for NMF-terms they are difficult to interpret since several clusters are empty, or has few elements.

TABLE IV. RESULTS FOR THE NMF ALGORITHM FOR *TWEETS*

K: #of clusters	*WSS*	*BSS*	*||Ci||*
3	513.12 3,266.52 76.16	830.66	59 391 50
5	177.24 2,695.30 139.04 338.46 296.02	834.44	58 318 23 57 44
7	207.60 39.08 224.15 2,719.07 0.93 160.97 54.37	1,139.43	45 13 27 322 15 29 49

RESULTS FOR NMF ALGORITHM FOR TERMS

K: #of clusters	*WSS*	*BSS*	*||Ci||*
3	0.00 344.67 4,399.31	43.32	0 3 1,200
5	0.00 0.00 0.00 4,057.80 235.50	45,298.76	1 1 1 1,198 2

K: #of clusters	*WSS*	*BSS*	*‖Ci‖*
3	0.00 344.67 4,399.31	43.32	0 3 1,200
7	4,264.95 0.00 0.00 184.00 0.00 0.00 0.00	23.37	1,195 0 1 6 1 0 0

We also compute the word cloud of each cluster using Wordle for an empirical evaluation. Some of these results, for k = 3, are presented.

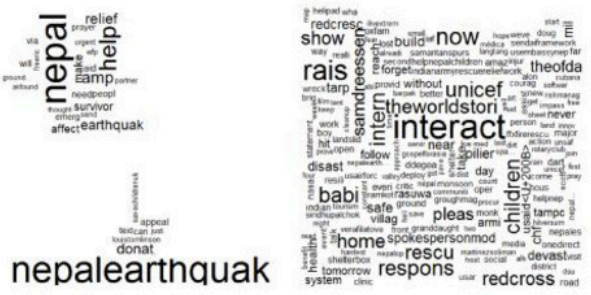


Fig 2. *Word clouds of terms for the clusters using k-means, k=3*

Figure 2 presents the terms – as word clouds – that appear in each of the clusters for the k-means algorithm. Figure 3 presents similar word clouds for the k-*medoids* algorithm. Figure 4 presents the word clouds obtained from DBSCAN with provides 4 clusters. Finally Figure 5 presents the word clouds of terms obtained from the NMF algorithm for 3 clusters.

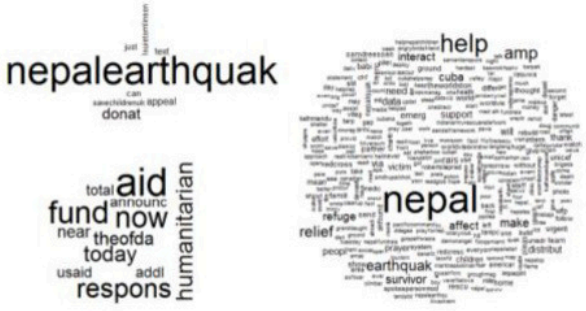


Fig. 3. *Word clouds of terms for the clusters using k-medoids algorithm, k=3*

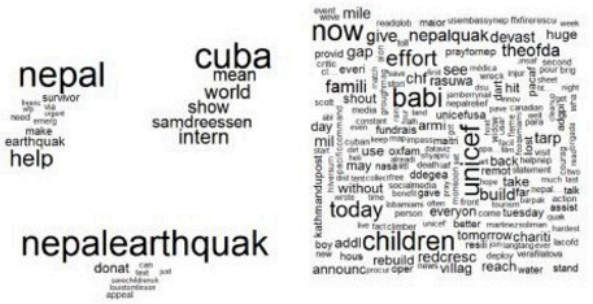


Fig. 4. *Word clouds of terms for the clusters using DBSCAN algorithm, ε=3*

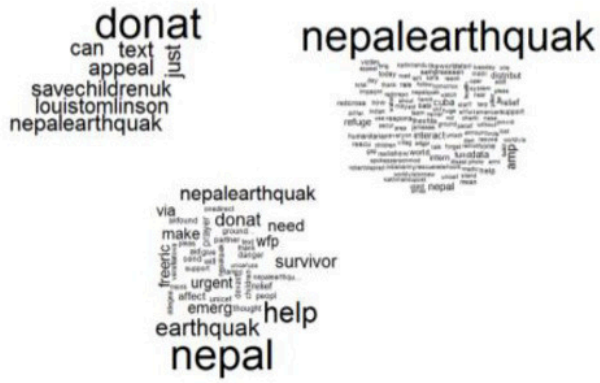


Fig. 5. *Word clouds of terms for the clusters using NMF, k=3*

V. Conclusions and Future Work

In this paper we analyze topics identification and clustering algorithms in the context of text messages obtained from the social microblog *Twitter*. To do so, we use text preprocessing techniques and Machine Learning clustering algorithms.

This study was tested using a dataset of *tweets* related to the recent Nepal's earthquake. Results show that this proposal is feasible for human analytical purposes: using word clouds is possible to obtain the main topics related to the given initial context.

Four different clustering algorithms are employed: k-means, k-*medoids*, DBSCAN and NMF. We numerically compare their results using the cohesion (WSS) and separation (BSS) measures. Results are presented on Tables I, II, III, IV and V; in general, these tables show similar results; in the case of the NMF algorithm, the clusters obtained from the (topic x term) matrix are difficult to compare to the others, since most if the terms are clustered together.

The word clouds associated to each cluster is used to show an empirical evaluation, in this case we argue that the NMF algorithm present the best results, since it seems that the associated topics are easy to understand; this result is also in accordance with the conclusion obtained by Godfrey [21].

We plan to extend this research work in several ways: (a) testing alternative text preprocessing techniques; (b) employing different term weighting schemes; (c) testing different

algorithmic options, particularly in the case of the NMF algorithm where several optimization options are available; and (d) testing human interpretations of the word clouds associated to each cluster.

We also plan to extend this work by using the geographic information – latitude and longitude of the *tweet* emitter – available on *Twitter*: using this information it will be possible to generate graphs associated to the emission and spreading of the specific topics that appear on this social microblog.

REFERENCES

[1] Ricardo Baeza-Yates and Berthier Ribeiro-Neto. Modern Information Retrieval. ACM Press, New York. 1999.

[2] C. D. Manning, P. Raghavan and H. Schütze. Introduction to Information Retrieval, Cambridge University Press. 2008.

[3] T. M. Mitchell. 1997. Machine Learning, McGraw-Hill.

[4] Twitter Documentation. https://dev.twitter.com/rest/public. Accessed at May 29, 2015.

[5] Hila Becker, Mor Naaman and Luis Gravano. Beyond Trending Topics: Real-World Event Identification on Twitter. Proceedings of the Fifth International AAAI Conference on Weblogs and Social Media, 438–441. 2011.

[6] Aditi Gupta and Ponnurangam Kumaraguru. Credibility Ranking of Tweets during High Impact Events. Proceedings of the 1st Workshop on Privacy and Security in Online Social Media, article 2. 2012.

[7] David A. Shamma, Lyndon Kennedy e Elizabeth Churchill, 2009. Tweet the Debates: Understanding Community Annotation of Uncollected Sources. *Proceedings of the first SIGMM Workshop on Social Media*, vol. 22(1), pp. 3-10.

[8] Takeshi Sakaki, Makoto Okazaki and Yutaka Matsuo. Earthquake Shakes Twitter Users: Real-Time Event Detection By Social Sensors. Proceedings of the 19th International conference on Word Wide Web, 851–860. 2010.

[9] C. D. Manning, P. Raghavan e H. Schütze. *Introduction to Information Retrieval*, Cambridge University Press. 2008.

[10] Daniel Godfrey, Caley Johns, Carol Sadek, Carl Meyer e Shaina Race, 2014. *A Case Study in Text Mining: Interpreting Twitter Data From World Cup Tweets*. Cornell University Library, http://arxiv.org/abs/1408.5427, acessado em 20 de maio de 2015.

[11] Moody Chu and Robert Plemmons. Nonnegative matrix factorization and applications. Bulletin of the International Linear Algebra Society, 34, 2–7. 2005.

[12] Klinczak, Marjori N. M. and Kaestner, Celso A. A. Identification of Topics on Twitter: Comparison of Clustering Algorithms and Case Study. LA-CCI. 2015.

[13] J. Han and M. Kamber. Data Mining Concepts and Techniques, Morgan Kaufmann. 2001.

[14] D. D. Lee and H. S. Seung. Unsupervised learning by convex and conic coding. Advances in Neural Information Processing Systems, 9(1):515–521, MIT Press. 1997.

[15] I.T. Jolliffe. Principal Component Analysis, Springer-Verlag. 2002.

[16] Michael W. Berry, Murray Browne, Amy N. Langville, V. Paul Pauca and Robert J. Plemmons. Algorithms and applications for approximate nonnegative matrix factorization. Computational Statistics and Data Analysis, 52. 2006.

[17] Twitter Team. 2012. Twitter turns six. http://blog.twitter.com/2012/03/twitter-turns-six.html . accessed at July 08, 2014.

[18] David D. Lewis, Yiming Yang, Tony G. Rose and Fan Li. RCV1: A New Benchmark Collection for Text Categorization Research. Journal of Machine Learning Research, 5(1):361–397. 2004.

[19] Martin F. Porter. An Algorithm for Suffix Stripping. Program, 14(3): 130–137. 1980.

[20] Anil K. Jain and Richard C. Dubes. Algorithms for Clustering Data, Prentice Hall. 1988.

[21] Daniel Godfrey, Caley Johns, Carol Sadek, Carl Meyer and Shaina Race. 2014. A Case Study in Text Mining: Interpreting Twitter Data From World Cup Tweets. Cornell University Library, http://arxiv.org/abs/1408.5427 , accessed in May 20th, 2015.

[22] Anil K. Jain and Richard C. Dubes. Algorithms for Clustering Data, Prentice Hall. 1988.

[23] Naaman, Boase, and Lai. *Is it really about me? Message content in social awareness streams*. CSCW10. 2010.

[24] Gupta, Aditi & Kumaraguru, Ponnurangam. *Credibility Ranking of Tweets during High Impact Events*. Proceedings of the 1st Workshop on Privacy and Security in Online Social Media. 2012.

[25] Ryaboy, Dmitriy & Lin, Jimmy. *Scaling Big Data Mining Infrastructure: The Twitter Experience*. ACM SIGKDD. 2012.

Celso A. A. Kaestner, has a BSc in Mathematics from the Pontifical Catholic University of Paraná and in Civil Engineering from the Federal University of Paraná. He has a MSc– Control Systems – and a PhD – Information Systems – in Electrical Engineering at the Federal University of Santa Catarina. He had post-doctoral positions at the École de Technologie Supérière the University of Quebec in Montreal, Canada, and at York University in Toronto, Canada. He was a Full Professor at the Informatics Department, of the Federal Technological University of Paraná in Curitiba, Brazil until his retirement in June 2015. He is currently a senior professor at the Graduate Program in Applied Computing at the same University.

Marjori Naiele Mocelin Klinczak, has a degree in Internet Systems by the Faculty of Administration and Economy of Paraná (FAE), graduate in Software Development in International Markets at the Federal University of Paraná (UFPR) and is currently studying at the Graduate Program in Applied Computer Science at the Federal University of Technology – Paraná (UTFPR). She is also the CEO of the Mosaic Web company and works with web and mobile development.

People Recognition for Loja ECU911 applying artificial vision techniques

Diego Cale, Verónica Chimbo, Henry Paz-Arias and J. J. Barriga-Andrade

***Abstract*—This article presents a technological proposal based on artificial vision which aims to search people in an intelligent way by using IP video cameras. Currently, manual searching process is time and resource demanding in contrast to automated searching one, which means that it could be replaced. In order to obtain optimal results, three different techniques of artificial vision were analyzed (Eigenfaces, Fisherfaces, Local Binary Patterns Histograms). The selection process considered factors like lighting changes, image quality and changes in the angle of focus of the camera. Besides, a literature review was conducted to evaluate several points of view regarding artificial vision techniques.**

***Index Terms*—OPENCV, QT CREATOR, EIGENFACES, FISHERFACES, LBPH, ICONIX, FRAME, artificial vision, people recognition.**

I. Introduction

Because of its nature to optimize process, resources and time, intelligent systems could be used for monitor and control in several areas such as security, health, criminology among others.

The system presents a solution for people searching by using IP video cameras; its main objective is to contribute with such type of solution that is in boom [1]. The artificial vision system is based on algorithms fusion for detecting and identifying faces in an intelligent way. In order to identify an specific person, a list of pictures is stored; once the person has been found several alerts will be triggered by using a web page that relies on Google Maps for geolocation. Several tests on ATMs, supermarkets, buses have been conducted in order to the system feasibility. In addition, other type of test was performed in the technological area of Loja ECU911 using cameras such as the Loja's Fair in September 2015 which provided feasible and favorable results according to expectations.

Currently, ECU911 is in charge of handling the video surveillance system to respond to emergency situations across the Ecuadorian territory. Its aim is to contribute permanently to the achievement and maintenance of public comprehensive security, and presents the following services: Video surveillance, emergency hot line, community engagement and institutional coordination, which are described in the following Table I.

Nowadays, people searching processes are done manually by ECU911 and other institutions in charge of monitoring, controlling and securing citizens, which is time and resource demanding since it is required to have people performing such tasks 24 hours a day, 365 days a year. Therefore, an artificial vision system is proposed, which would monitor in an autonomous way all days of the year, by providing real time alerts, supporting the video surveillance service of ECU911.

TABLE I
ECU911 Services.

Service	Description
Video Surveillance	It uses the most advanced technology to monitor activities that might produce risk.
Emergency hot line	Emergency service attending calls dialed to 911, 24 hours a day, 7 days of the week, 365 days of the year.
Community engagement	Talks and training for children, youth and communities focused on the proper use of ECU911 service and the importance of citizen cooperation in comprehensive security.
Institutional coordination	All emergency institutions working together permit to attend the same situation in a complete perspective, allowing and effective and comprehensive response. It reduces times and manages to mobilize specialized units for specific emergencies.

At present, ECU911 of Loja owns technological infrastructure that allow to control certain locations of the city, their cameras provide a great quality of image and range. Indeed, the the only automated process handled by ECU911 is the storage of such information, but in order to find something it is necessary to completely review the whole video, which is time consuming; the technical proposal aims to replace such process with one that is able to process data in real time.

This paper is structured in the following way: Section II compares several vision artificial techniques, section III shows features of the tools, section IV contemplates system implementation, section V presents a case study which describes the operation of the system and the results obtained, section VI is about conclusions arrived and section VII covers considerations in terms of informatic security.

D. Cale and V. Chimbo are with the Universidad Nacional de Loja (e-mail: dacale, vpchimb@unl.edu.ec)

H. Paz and J. Barriga are with the Escuela Politécnica Nacional (e-mail:henry.paz, jhonattan.barriga@epn.edu.ec)

II. General Study of Artificial Vision Techniques

A. Importance of Artificial Vision

Artificial Intelligence computer system are currently used in several fields of investigation to make processes more autonomous and automatic, given the ability to make decisions by themselves according to [2]. Artificial vision is a field of Artificial Intelligence that aims to perform an abstraction of the real world to mathematically model processes of visual perception of living beings, generating computer programs through these simulation capabilities [3].

B. Artificial Vision Application

Several enterprises around the world are implementing this technology with good results.

- RCG Holdings Limited in Hong Kong is using it for security, monitoring and access control, they implemented an artificial vision system with a face recognition engine that adapts to several states of lighting and poses of a person according to [4].
- Ample Trails from India developed a real-time people recognition system to control access and record attendance with a success rate of 99 % as mentioned in [5].
- Toshiba with other computer companies, developed an artificial vision system adapted to computers for security and faster accessibility by replacing hibernation or standby states with the use of the face to enable access as compiled in [6].
- Also, Facebook DeepFace is able to recognize faces with a precision rate of 97.25 % which is very similar to the ability of a person as mentioned in [7].

C. Algorithms and methods

There are several algorithms for face detection within artificial vision which have considered lightning changes, image quality, several changes in the face such as beard, hair color or glasses which may affect its identification. Then, a review of the most used techniques and methods within artificial vision will be performed, which highlights the ones used for features extraction as it is one the most important phases in the process of people face detection and recognition. Such methods are divided in three groups which are: based on appearance, characteristic points of the face and hybrid ones.

Holistic methods based on appearance These methods use all face region as an input for the face recognition system. The image of the face is transformed to a space where statistical techniques are applied. However, using all face as the only feature, often presents limitations given by expression changes, pose or illumination. It implements the technique of Principal Component Analysis (PCA) which let to obtain vectors of lower dimensions without loss of important information as mentioned in [8].

Methods based on featured points of the face. These methods are based on extracting common features that make the face such as nose, eyes or mouth, to classify its geometric features and/or the appearance separately in the system.

Distance analysis to characteristic points. One the first recognition system its based on the technique of geometric points of the face. From the detection of different characteristic points, vectors that contain data of distances between them are created. The more points detected, the greater number of distances that could be calculated thus obtaining better results in the recognition as showed in Figure 1. [8].

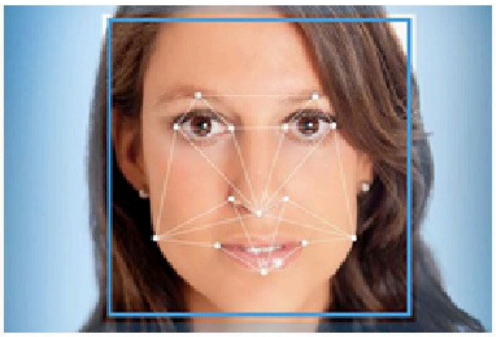


Figure. 1. Face Detection

Local Binary Patterns. Although LBP algorithm is simple it does provide robust information against lighting changes. It is based on taking neighbors about a central pixel which sets a threshold value as stated in [9]. The neighborhood is changed to binary depending on the value whether it is higher or lower than the threshold, and every found value is concatenated to build a unique binary number which later will be converted to a decimal value that will represent the new value of the pixel. The image is divide in regions where LBP is applied to obtain its histogram. These histograms are then concatenated to obtain a representation of the face.

EigenFaces. It that performs Principal Components Analysis (PCA) of the covariance matrix formed by the source images and the input image, which compares distances between the vector of the original image with the rest.

FisherFaces. Algorithm that uses FLDA to reduce dimension.

LocalBInaryProcess. Uses histogram of oriented gradients (HOG) to improve performance of detection.

The following process was used to compare the three algorithms (EigenFaces, FisherFaces and LocalBinaryProcess):

- **Image processing:** Size of images is normalized, plus contrast is equalized.
- **Facial detection:** Viola-Jones algorithm was implemented getting a 20ms time detection.
- **Image handling:** Once it is known that there is a face, it would be cut to stay with the information that really interests (the face).
- **Extraction and feature comparison:**
  - Use of FisherFaces with a response time of 10 ms.
  - Use of EigenFaces with a response time of 10 ms.
  - Use of LocalBinaryProces with a response time of 10 ms.
- **Properties:**
  - **EigenFaces:**
    - Eigenfaces number: 200.

  - Threshold: 10000.
  - **Advantages:** Small computation time and very good results in optimum condition.
  - **Disadvantages:** Bad results on adverse conditions (lighting, pose and orientation) and/or few training images.
- **FisherFaces:**
  - Threshold: 1500.
  - **Advantages:** Soften the problem of changes in posture or light and deformations of the face. It performs better than Eigenfaces when the number of poses is lower.
- **LocalBInaryProcess:**
  - Binary Patterns.
  - Images of 320x243 pixels.
  - 3500 images used to train.

In conclusion, the chosen algorithm was Local Binary Process because it is better suited to lighting conditions, the operator is an extension of the original LBP coding; hence, sometimes it is called LBP extended. If the points of the circle do not match image coordinates, the point has to be interpolated. Indeed, OpenCV [10] which is an open source library for computer vision and machine learning, implements a bi-linear interpolation as shown in equation 1:

$$f(x,y) \approx [1-xx] \begin{bmatrix} f(0,0) & f(0,1) \\ f(1,0) & f(1,1) \end{bmatrix} [\frac{1-y}{y}] \qquad (1)$$

By definition, LBP operator is more robust against changes in monotone gray scale. Finally, spacial information is included in the facial identification model.

D. Facial recognition process

Five phases have been considered (as shown in Figure 2) for the development of the artificial vision module and are listed as follows:

Figure. 2. Artificial Vision stages

1) **Acquisition**. Images were obtained with an IP camera, it is important to consider camera location, verify that the quality of images is not lower than 640x480 pixels of resolution, images should not be distorted. Indeed, the better quality of the image, the fastest is the process and the results are more effective.
2) **Detection**. This is the most critical part since it would impact the rest of the phases if an adequate detection and location has been performed. Detection comprises two parts:
   - **Detection of the region of the face**. The detection of the regions of interest in an image is done by Haar-Like features as shown in Figure 3, adapted by Viola and Jones from the use of Haar wavelets. This system considers rectangular regions in a window of detection, it add the intensities of the pixels in each region and calculates the difference between these amounts. The difference is used to classify subsections of the image. [11]
   - **Eye position detection**. To correctly align the image during pre-processing, it is mandatory to determine the coordinates of the eyes in it. Besides, there are several methods to detect eyes, the most direct is to use classifiers of Haar as in face detection, but trained with images of eyes according to [12].

Figure. 3. Face recognition

3) **Pre-processing and normalization**. Pre-processing stage according to [13] is carried out from the information obtained during detection. This stage performs a series of geometric transformations over the image, leaving it ready for proper feature extraction. During pre-processing four phases are used to normalize and align the image as mentioned in [9].
   (a) **Image Acquisition**. The first step is obtain the initial image that will be processed.
   (b) **Rotation**. One of the utilities to calculate eyes coordinates, lies in determining the angle of rotation of face in an image and offset it. Having faces with no rotation would produce better results during the recognition process.
   (c) **Scaling**. To ensure that all images have the same size, the distance between the centers of the eyes is used to get a radio by which the image will be increased or reduced. Indeed, this is required since several recognition techniques demands input data with the same size (In this case the matrix of pixels).
   (d) **Cutout**. Once the image has been rotated and scaled, next is to cut it to obtain its region of interest. The coordinate of the right eye is used to establish the region of interest. There are several standardized images by which the region of interest can be extracted according to the system needs. Such formats are documented in the standard ISO/IEC 19794-5 according to [14] which defines an area similar to a passport photograph, based on such standard a region that exclusively comprises the area of the face is defined. [11]

   A summary of the process is shown in Figure 4
4) **Feature Extraction**. It is used to obtain information that is relevant in undertaking a comparison as shown

Figure. 4. Pre-processing and Normalization process

in Figure 5. Local Binary Pattern Histograms (LBPH) method (createLBPHFaceRecognizer()) was used in this stage.

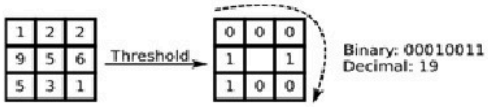


Figure. 5. Feature Extraction

5) **Comparing and Recognition**. There are several methods to obtain a score for making a decision. Such methods could be divided in distance and classifiers according to [15]:

- **Distance**
  - Euclidean Distance:
    It is represented in equation 2, and is one of the most basic measures to calculate distances. This distance is defined as the direct distance between two points on a plane. The clearest example is the distance between two points on a plane of two dimensions with coordinates x and y. If there were two points P1 and P2 with coordinates

    $$x_1, y_1$$

    and

    $$x_2, y_2$$

    respectively, the euclidean distance between them would be:

    $$d_E(P1P2) = \sqrt{(x_2 - x_1)^2 + (y_2 - y_1)^2} \quad (2)$$

  - Chi-Square:
    Chi-Square distance takes such name because the formula used to calculate is almost similar to the goodness of fit test that is used to compare discrete probability distributions. For biometric recognition, it is used to measure the distance from histograms. Distance calculation for two histograms S and M is represented in the following equation 3:

    $$x^2(S, M) = \sum_i \frac{(S_i + M_i)^2}{S_i + M_i} \quad (3)$$

- **Classifiers**
  - K-Nearest Neighbours:
    K-NN method is a non parametric object classification technique based on the samples that are closer to the space of features. The algorithm is based on finding k neighbors that are more near to the object, depending on the amount thereof; classify in the set having a greater number of nearby samples [16].
  - Support Vector Machines (SVM):
    SVMs are learning models for regression and classification. Its objective is to represent in a space a series of classes and try to find a hyperplane that divides them into zones in which when any input enters, it would be classified according to such zones [16].
    SVM classification is used to produce a model based on training data, which will be able to accurately predict the class of labels of test data. Figure 6 shows a visualization of the process of SVM classification,

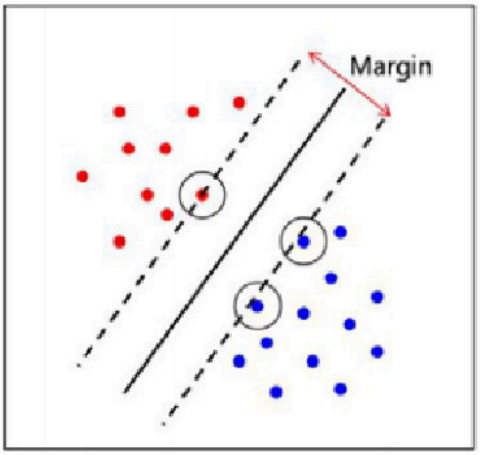


Figure. 6. SVM Classification support vector machines find the hyperplane that maximizes the margin between two different classes

For this reason, the method chosen for this study is SVM because it allow to establish a margin between two data sets.
  - Gaussian Mixture Models:
    GMMs are density parametric probability functions represented as a sum of Gaussian components. Biometric classification is usually used in conjunction with EM (Expectation Maximization) to estimate its parameters. Notably OpenCV, still has no implementation of this technique.

III. Developed Software Tools

Nowadays, ECU911 does not have tools to support people search and identification process. Therefore, an intelligent system which uses cameras and gives new security citizen services is presented. The technology used is described below.

(a) **Desktop Application (DVFACE-DETECTOR)**. The purpose of this system is to handle people, it functionality is described below:

- **People:** Allow to handle information of people to be searched.

- **Users:** In charge of register users and control access to the system.
- **Cameras:** Manage the cameras that are located in several places of Loja.
- **People Search:** Before searching a person, it is required to upload a list of pictures of the person.

(b) **Web Application (DVFACE-ALARM.)** Used to visualize camera location so that it would be easy to determine a person location. It has the following modules:

- **People:** Used to insert people information from a web environment.
- **Users:** Module used to register users and control access to desktop or web version.
- **Cameras:** Camera registration along with its geographic location.

IV. Implementation

The following describes the architecture of the system proposed.

1) The input data (Frames) arrives from ECU911 cameras.
2) Database is used to store the images obtained with the cameras and process data during detection.
3) The result is notified through the screen once the person has been found.

The following applications have been developed as part of the Artificial Vision System:

- **DVFACE-DETECTOR:** C++ was used together with QT-CREATOR framework because of the experience acquired and the fact that it is free with great amounts of documentation. GitHub was chosen as a repository since it allows to have the code available online to be accessed from everywhere, plus it is compatible with QT-CREATOR.

- **DVFACE-ALARM:** Javascript and EXTJS framework were used to build this web application on the client side. PHP was used on the server side to perform database queries. The tools mentioned before were chosen because of the experience. GitHub was also chosen as the code repository due to its benefits.

V. Case Study

The following describes the application of the system on a Case Study.

1) **Face Detection** Figure. 7 shows image processing in the people recognition module, the following shows all the images processed by OpenCV library.

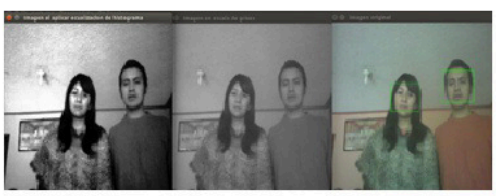


Figure. 7. Image Processing

2) **Face Processing** Figure 8 shows face detection which requires the activation of the camera used for face recognition.

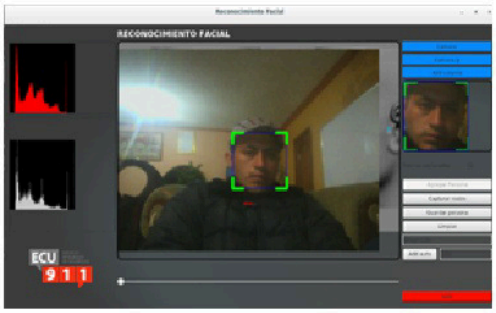


Figure. 8. Face Processing

3) **Face Recognition** In order to search a person, it is required to have all of his data as well as the cameras that will provide the frames to analyze during image processing; then, two process are executed (training and face identification).

(a) **Training** A list of photos is required so that the model of the face of a searched person could be registered, it is recommended to have a set of photos of different lighting sources since it benefits expected results as shown in Figure. 9.

Figure. 9. Face Training

(b) **Face Identification** Once the model has been trained with a specific person to search and having the camera activated, will allow the system to identify such face and raise and alert with the name of the person that has been identified as shown in Figure. 10 Then, web application will display an alert stating that the person has been found. Figure 11 shows information of the found person as well as the initial picture used to search such person and the picture containing the most prominent features. Also, the user will have the ability to mark a person as found if all information is correct otherwise the process could

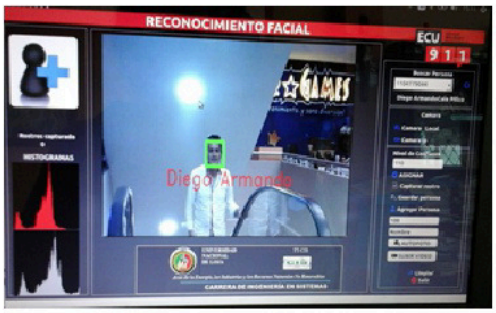


Figure. 10. Face Identification

be restarted if results were not as expected.

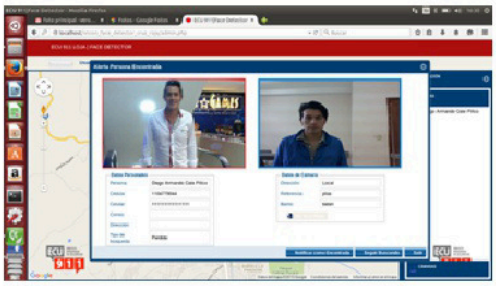


Figure. 11. Notification Process

VI. Conclusions

- FaceRecognizer class with HaarCascade Frontal Faces sorter for face detection and Local Binary Patterns Histograms (LBPH) for face identification, produced the intended results.
- Lighting changes should be monitored for the construction of artificial vision systems since it might affect the results.
- People recognition system is a technological proposal based on artificial vision for searching people in an intelligent way by using IP video cameras which replaces the current manual process that is time and resource demanding.

VII. Future Work

The following mechanisms are described in order to secure networks links and information that will be handled.

First of all, network links has to be secured in terms of using either dedicated channels or VPN so that ECU911 could securely connect to several sources of information (IP cameras) that are located across the city. Besides, it will help to secure providers as well since they will have to expose part of their critical infrastructure (particularly financial institutions). Indeed, a network security architecture has to be defined from scratch.

Second, secure protocols based on TLS must be in place throughout the application to encrypt information that will be traveling and stored in different places of the solution. Third, a hash verification signature needs to be added to prevent data tampering as it may be intercepted by malicious user aiming to corrupt or modify it. Indeed, such hash would help to verify that the searched people is the one selected by an employee of ECU911.

In addition, role based access control is required in the system to avoid unscrupulous users modifying or uploading undesired data. Moreover, audit trails has to be present across the system and database, although encryption is not a must to have it is required to apply it for sensitive information such as users, passwords and personal data. Also, a code review is required to guarantee that the application is free of vulnerabilities due to code obsolescence.

Finally, an Ethical Hacking has to be performed to find other vulnerabilities that might compromise the system, and to verify that the inclusion of new network links would not pose a threat to ECU911 and providers. Last but not least, high-availability needs to be reviewed depending on how critical is the service provided by ECU911.

In summary, several points of the solution has to be secured and examined to protect information integrity, availability and confidentiality.

References

[1] E. Aldabas-Rubira, "Introducción al reconocimiento de patrones mediante redes neuronales," *IX Jornades de Conferències d'Enginyeria Electrònica del Campus de Terrassa, Terrassa, España, del 9 al 16 de Diciembre del 2002*, 2002.

[2] T. Ahonen, A. Hadid, and M. Pietikäinen, "Face recognition with local binary patterns," in *Computer vision-eccv 2004*. Springer, 2004, pp. 469–481.

[3] N. J. Nilsson, R. M. Morales, J. T. P. Méndez, and E. P. Aris, *Inteligencia artificial: una nueva sintesis*. McGraw-Hill Boston, 2001, vol. 2.

[4] G. Hiebert, "Openal 1.1 specification and reference," 2005.

[5] Y. Xu and D. Zhang, "A new solution scheme of unsupervised locality preserving projection method for the sss problem," in *Structural, Syntactic, and Statistical Pattern Recognition*. Springer, 2008, pp. 775–781.

[6] BioBouncer. (2016, feb) Biobouncer. [Online]. Available: http://www.engadget.com/2006/02/28/biobouncer-facial-recognition-system-for-bars-clubs/

[7] P. I. Wilson and J. Fernandez, "Facial feature detection using haar classifiers," *Journal of Computing Sciences in Colleges*, vol. 21, no. 4, pp. 127–133, 2006.

[8] X. Lu, "Image analysis for face recognition," *personal notes, May*, vol. 5, 2003.

[9] W. Clarksburg, "Fbi announces full operational capability of the next generation identification system," *Criminal Justice Information Services Division*, 2014.

[10] Opencv. (2015, feb) Face recognition with opencv. [Online]. Available: http://docs.opencv.org/modules/contrib/doc/facerec/facerec_tutorial.html

[11] A. S. Abdallah, A. L. Abbott, and M. A. El-Nasr, "A new face detection technique using 2d dct and self organizing feature map," in *Proc. of World Academy of Science, Engineering and Technology*, vol. 21, 2007, pp. 15–19.

[12] C. R. Giardina and E. R. Dougherty, "Morphological methods in image and signal processing," *Engelwood Cliffs: Prentice Hall, 1988*, vol. 1, 1988.

[13] M. A. Turk and A. P. Pentland, "Face recognition using eigenfaces," in *Computer Vision and Pattern Recognition, 1991. Proceedings CVPR'91., IEEE Computer Society Conference on*. IEEE, 1991, pp. 586–591.

[14] J. Sang, Z. Lei, and S. Z. Li, "Face image quality evaluation for iso/iec standards 19794-5 and 29794-5," in *Advances in Biometrics*. Springer, 2009, pp. 229–238.

[15] W. Zhao, R. Chellappa, P. J. Phillips, and A. Rosenfeld, "Face recognition: A literature survey," *ACM computing surveys (CSUR)*, vol. 35, no. 4, pp. 399–458, 2003.

[16] J. KIM[1], B.-S. Kim, and S. Savarese, "Comparing image classification methods: K-nearest-neighbor and support-vector-machines," *Ann Arbor*, vol. 1001, pp. 48 109–2122, 2012.

Diego Cale Systems Engineer from the Universidad Nacional de Loja Ecuador in 2016, experience in Java, JavaScript, C++, Qt Creator, HTML5, PHP, Extjs, Information Technology. His interests are web, mobile and AI software development.

Verónica Chimbo She has an engineer degree in Systems by the Universidad Nacional de Loja Ecuador in 2016, software analysis and development with languages like Java, JavaScript, C++, Qt Creator, HTML5, PHP y Android. Her intests regarding investigation are development of intelligent applications for robotics.

Henry P. Paz Arias He has an engineer degree in Systems by the Universidad Nacional de Loja Ecuador (2010). In 2012 he obtained the Master in Computer Science in the Area of Artificial Intelligence at University of Hidalgo - México. Currently he is pursuing a PhD in computer science in the area of intelligent systems of the National Polytechnic School (EPN) of Ecuador. Ing. Paz is currently acting as computer science titular professor at the National Polytechnic School (EPN) of Ecuador. He has also acted as teacher int the Universidad Interglogal - Pachuca - México, National University of Loja - Ecuador teaching materials artificial intelligence.

Jhonattan J. Barriga Andrade is a teacher of Escuela Politécnica Nacional, Quito - Ecuador. Systems and Informatic Engineer at Army Polytechnic School (ESPE), MSc. (Distinction) Computer Forensics and Systems Security at University of Greenwich – England. Currently, PhD student of Informatics Doctorate at Systems Engineering School from Escuela Politécnica Nacional with focus on CyberSecurity. His interests are: Malware, Penetration Testing, Remote Access Trojans, Secure Coding, Security Architecture, and Computer Forensics.

Annotated Corpus for Citation Context Analysis

M. Hernández-Álvarez, José Gómez Soriano and Patricio Martínez-Barco

***Abstract*—In this paper, we present a corpus composed of 85 scientific articles annotated with 2092 citations analyzed using context analysis. We obtained a high Inter-annotator agreement; therefore, we assure reliability and reproducibility of the annotation performed by three coders in an independent way. We applied this corpus to classify citations according to qualitative criteria using a medium granularity categorization scheme enriched by annotated keywords and labels to obtain high granularity. The annotation schema handle three dimensions: PURPOSE: POLARITY: ASPECTS. Citation purpose define functions classification: use, critique, comparison and background with more specific classes stablished using keywords: Based on, Supply; Useful; Contrast; Acknowledge, Corroboration, Debate; Weakness and Hedges. Citation aspects complement the citation characterization: concept, method, data, tool, task, among others. Polarity has three levels: Positive, Negative and Neutral. We developed the schema and annotated the corpus focusing in applications for citation influence assessment, but we suggest that applications as summary generation and information retrieval also could use this annotated corpus because of the organization of the scheme in clearly defined general dimensions.**

***Index Terms*— Corpus, annotation, methodology, machine-learning, function, polarity, aspects, schema, keywords, labels, classification.**

I. Introduction

IT is necessary to overcome the absence of a common framework to facilitate research progress in collaborative conditions for citation context analysis. This framework should include a standard annotation scheme, and an annotated corpus according to such scheme. In fact, [1] suggested that the biggest problem facing researchers in this field is that there is not a public available annotated corpus that responds to a medium or high granularity scheme that can be used on a shared basis for scholars. The few annotated corpus available present some of the following problems: different ad hoc classification schemes are developed for each application; corpus are not publicly available for shared work; or, they are not presented in a standard format that other researchers could understand and use. Moreover, most of the previous citation work do not take into account citation context but only the sentence that contains the citation, method that results in loss of information that difficult achieving better classification results [2].

Different annotation approaches present diverse levels of granularity in citation function definitions. These schemes define from three to 35 different classes. Less granularity often refers to polarity (positive, negative, or neutral/objective). Schemes that are more complete correspond to diverse approaches and applications.

In [3], they categorize annotation schemes in two classes according if they have acceptable results using manual or automatic methods. In that study, we observe that manual classification schemes have medium granularity, while automatic processed schemes have low granularity. Annotated corpora with medium or high granularity provide valuable information indispensable to citation context analysis, but its annotation is a complex task, even for human coders, because even people have problems to achieve a good Inter-annotator agreement. Of course, challenges for automatic annotation are even greater [4]. The schemes with medium or high granularity need to be manually labeled by their authors; because attempting automatic labeling of this kind of corpora until now generates poor and not reliable results [1]. Even manual labeling without an adequate methodology results in a poor Inter-annotator agreement [5].

We could not find a classification scheme for citation function that combines a sufficient granularity with a simple structure, in a way that allows it to be useful in Citation Context Analysis, also having the necessary clarity to yield good Inter-annotator agreement; index that is indispensable to assure reproducibility and reliability.

We had three objectives to fulfill in the present study. First, to define a simple but complete structure scheme with enough information about purpose which is defined as aim and intention of the reference; and, citation polarity defined as author's disposition towards a reference that could be favorable or positive, unfavorable or negative and neutral [6]. Second, to annotate a corpus using this scheme obtaining a good Inter-annotator agreement, and make it available for collaborative work in the University of Alicante digital repository[1] and in the LRE map. Third, to apply in the previously mentioned citation corpus a machine-learning algorithm to classify automatically function and polarity with an acceptable outcome. In further work, we intent to identify influence levels of the citations applying in the developed corpus a machine-learning algorithm taking as inputs: function, polarity, and features related to position of the references.

M. Hernández-Álvarez, *Escuela Politécnica Nacional, Facultad de Ingeniería de Sistemas, Quito, Ecuador,* myriam.hernandez@epn.edu.ec

José Gómez Soriano, *Universidad de Alicante, Departamento de Lenguajes Informáticos, Alicante, España,* jmgomez@dlsi.ua.es

Patricio Martínez-Barco, *Universidad de Alicante, Departamento de Lenguajes Informáticos, Alicante, España,* patricio@dlsi.ua.es

[1] http://hdl.handle.net/10045/47416

II. Description of the Classification Scheme

As mentioned, we designed a classification scheme in order to maintain a simple structure with two dimensions: function that is associated to purpose, and polarity related to the disposition of the citing author towards the cited paper (Figure 1).

Function defines purpose; therefore, they have to do with categories such as use, comparison, critique and background. In some of these categories, we have classes that are more specific. USE: The functions *Based on, Supply* correspond to citation content that the citing author use in the same paper. Detecting related aspects separate this grouped class. *Based on* have to do with concept, methods and similar aspects. *Supply* has aspects such as tools, data, task and so on. The function *Useful* corresponds to a citation mentioned as used in other work, but that the citing paper does not apply.

COMPARISON: The function *Contrast* performs a comparison between aspects of different studies with positive, negative and neutral outcome. Frequently positive outcome results from a comparison with citing author's work.

CRITIQUE: This type of purpose corresponds to the functions *Weakness* and *Hedges*. *Weakness* is a direct criticism, *Hedges* is a concealed critique as defined by Hyland (1998).

BACKGROUND: This type of purpose relates with work that the citing paper nor other mentioned studies applied. It corresponds to functions *Acknowledge, Corroboration* and *Debate*. These grouped functions are separated using aspects. *Acknowledge* is a simply recognition of previous work. In *Corroboration,* there are aspects that determine agreement with the cited paper. *Debate* involves aspects that express difference of opinion with some of the content of a citation.

Polarity could be Positive, Negative and Neutral according to a favorable, unfavorable or neutral disposition from the author of the citing paper. Polarity definition relates to sentiment analysis.

We combined the two-dimension structure PURPOSE: POLARITY, with keywords and labels that indicate citation aspects: concept, method, data, tool, task, etc.; and positive, negative or neutral features. This more complete combination PURPOSE: POLARITY: ASPECTS yields high granularity, comparable with exhaustive ontologies as CiTO[2]. In [4], it is noticed that ontologies like CiTO present difficulty for annotation and obtain a low Inter-annotator agreement due to their complexity. In contrast, our proposed scheme facilitates understanding and application in the annotation process. The keywords and labels work both ways: to clarify function and polarity for the annotators, and later, they will serve as inputs for the automatic classification of function and polarity of the corpus.

[2] http://purl.org/spar/cito

Function	Description
Based on, Supply	Citing paper uses work from the citation. *Based on* refers to aspects such as concept, method and similar. Aspects of Supply function are data, tool, task, etc.
Useful	Citing paper does not use work from the citation, but it mentions citation as used in other studies. Aspects of this function are concept and method, but also data, tool, task, etc.
Acknowledge, Corroboration, Debate	Citation is mentioned as background to recognize prior work. Aspects separate the grouped functions. Paper could be mentioned just in passing (*Acknowledge*); to agree with cited paper (*Corroboration*); or to discuss cited paper (*Debate*). Citing paper does not use cited work. Other paper mentioned in citing paper does not use cited work.
Contrast	Citation is compared to citing paper or other work. Result can be a criterion positive, negative or neutral.
Weakness	Citing paper notes an error or weakness from cited paper.
Hedges	Citing paper uses careful language to disguise a criticism directed to the reference.

Table 1: Function classification scheme

Figure 1 shows classification dimensions, while Table 1 presents the function classification scheme.
Results for Inter-annotator agreement will demonstrate that our scheme is easy to apply. Annotators are able to take advantage of all possibilities of classification, because they need to understand and remember only six functions clearly defined and three levels for polarity; as opposed to what happen with complex ontologies as CiTO, where coders have to apply 92 object properties.

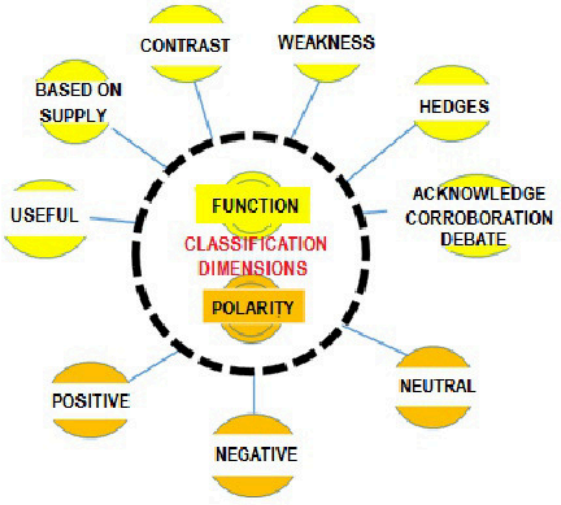


Figure 1: Function and polarity classification levels.

III. Dataset and Methodology

We applied the proposed scheme in a citation corpus composed by 85 articles taken randomly from ACL Anthology[3] with 2092 citations. We developed a program for converting text to XML, labeling paper titles, authors, sections, paragraphs and citations. After this initial pre-processing, we annotate citation function and polarity according to the suggested scheme using a methodology that includes a step of pre-annotation in which keywords and semantic tags are marked to clarify and standardize an internal representation that a coder or annotator creates about citation context. Using this method, the mental model is more likely to coincide with the ones produced for other coders, and consequently we obtain a good rate of Inter-annotator agreement in function and polarity classification. Experimentally we observed that with this pre-annotation step, we dramatically improve the agreement among annotators, which is indispensable to validate reliability and reproducibility of the annotation scheme.

Reliability and reproducibility of a classification scheme show whether it is possible to generalize results obtained in the annotation test to the complete process, in which probably are going to participate new annotators and not only the ones that codify the sample [8].

According to [9], annotation reliability and reproducibility is achieved if annotation process comply three conditions: a clear scheme with detailed instructions, specific criteria to choose annotators; and, the process must have at least three annotators working in an independent way. In our experiment, we fulfilled with these three requirements. We proposed a guide with a clear scheme, very detailed and with enough application examples; annotators are familiar with computational linguistics and with our guide, they revised the scheme carefully; and, we had three annotators working separately.

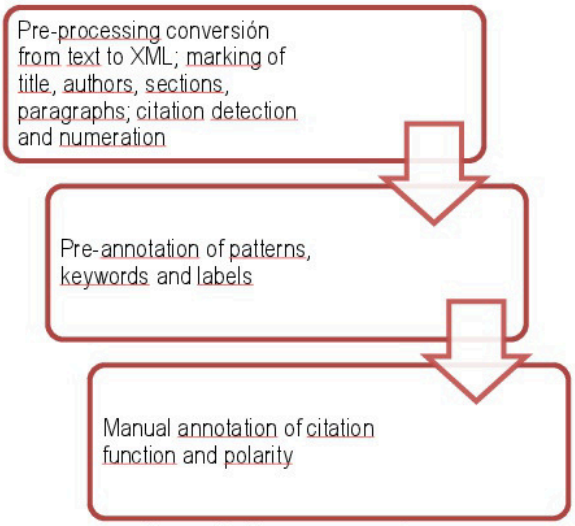


Figure 2: Corpus annotation process

Annotators chose keywords and labels from a list that corresponds to the most used words and phrases for each function and polarity classification; during the annotation process, we created new entries to this list as necessary. Annotators recognized relevant citation context inside a paragraph in which a citation is located. The keywords and labels list was refined while annotating the corpus and was part of the annotation guide. Figure 2 shows the corpus annotation process.

Illustrative keywords associated to polarity are "robustly" for *Positive*; "however" mostly for *Negative*; "previous work" for *Neutral*. Examples for keywords related to function: "build on" for *Based on, Supply*; "available" for *Useful*; "approach is not very satisfactory" for *Weakness*; "similarly" for *Contrast*; "another possibility for" for *Acknowledge, Corroboration, Debate*. Examples for labels associated with aspects of the citation function are "cited work", "author", "method", "theory", "task", "tool", "result", "feature", "positive feature", and "negative feature". Annotators can take these words or sequences of words from a specialized lexicon, but for our experiments, we defined these keywords and labels during the design of the coded corpus and through the course of the annotation process. In later experiments, we plan to annotate automatically keywords and labels, detecting those using bag_of_words and n-gram techniques from the lexicon we developed in the manual annotation.

For instance, if we have an original citation sentence: "Our classifier is built on the detailed previous work by Dong and Schäfer, 2011". Resulting XML with annotation will be "<author>Our</author> <tool>classifier</tool> <kw>is built on </kw> the <posfeature>detailed</posfeature> previous work by <cite id='citation_number_ identification' function='based on, supply' polarity='pos'>Dong and Schäfer, 2011</cited>". The pattern is "AUTHOR TOOL *is built on* POSFEATURE CITE", the different features of this pattern will be the input for the classification both manual and automatic. In this example, the classification is *Supply, Positive*. We improve Inter-annotator agreement marking first keywords and labels, but we also used these patterns to improve the granularity of the corpus in combination with function and polarity to disaggregate grouped functions and to define citation aspects. In this example, we classified the citation as *Supply* because it refers to a tool used by the author, and it is *Positive* for the kind of feature associated to it. Keywords were important to clarify the classification. To illustrate the role of the keyword, if the aspect were a method and not a tool, the classification for function would be *Based on*.

A special treatment is required for the recognition of the *Hedges* function. For instance, the classification should recognize the combination of a positive feature followed by a negative one.

For example if we have the quote: "The only recent work on citation sentiment detection using a relatively *large corpus* is by Athar (2011). However, this work *does not handle* citation context".

In this example, the author intention is to make a disguised criticism softened with a prior recognition of a positive characteristic. The result is a *Hedges* function because the real intention is criticism (Hyland, 1998). Here, the positive feature is "large corpus"; the negative feature is "doesn´t handle".

[3] Released Dec. 2013 http://clair.eecs.umich.edu/aan/index.php

Another case for the detection of the *Hedges* function involves not expressing categorically a negative expression (Hyland, 1998).

For example in the citation: "The first experiments in Argumentative Zoning used Naïve Bayes (NB) classifiers Kupiec et al., 1995; Teufel, (1999), which assume conditional independence of the features. However, this assumption ***is rarely true*** for the types of rich feature representations we want to use for most NLP tasks".

The negative opinion is softening by the words "rarely true" to avoid making a more categorical affirmation but the intention is again criticism, and therefore the function is *Hedges*.

Our scheme is simple but powerful because of the three dimensions used for classification: function, polarity and annotated patterns formed by keywords and labels: FUNCTION: POLARITY: ASPECTS. The combination of the three criteria produce high granularity without a complex structure.

In Figure 3, we present an example of the high granularity achieved using these three dimensions. A citation function classified as useful can refer to different aspects as tool, data, task, method; also, it can be mentioned with positive, negative or neutral features that facilitate definition of polarity, also it can be defined with its name. With all these elements, we obtained a complete citation description.

For instance, a citation could be referred as a specific tool, which is reported as useful because it is applied in other study and not in citing paper, and have positive reports that are detected by a positive feature annotated as a label. In this case, the function is *Useful*; polarity is *Positive*; and its aspect is that it is a tool. In general terms, the aspect is a third very important dimension that will specify if the citation refers to a tool, data, task or method or other; besides it will tell if it has positive, negative or neutral features which will define polarity.

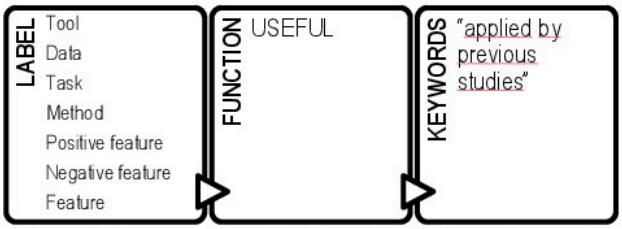


Figure 3: Improved granularity using labels and keywords.

IV. Annotation Results: Inter-Annotator Agreement

We validate Inter-annotator agreement and show results for function in Table 2, and for polarity in Table 3. We can see that the values of Fleiss' Kappa are as high as 0.862 for function and 0.912 for polarity. These values correspond to an almost perfect agreement in accordance to the scale of [10].

Using keywords and labels, we obtain a considerable improvement, because without this step, with the same annotators, there were low results for this index: 0.386 and 0.259 for function and polarity respectively, because of the difficulty to form coincident mental models among different coders.

The pre-annotation step allows forming these matching mental models and in addition, it provides information to feed as input to classifiers. Therefore keywords and labels added in the pre-annotation step, help both manual and automatic classification. Other studies [5] showed that it is very difficult to obtain a Kappa value for Inter-annotator agreement higher than 0.75 for a scheme with more than three classes.

Fleiss	**Krippendorff**	**Pairwise avg.**
A_obs=0.911	D_obs = 0.089	% agr = 91.1
A_esp=0.354	D_esp = 0.648	Kappa=0.862
Kappa=0.862	Alpha = 0.862	

Table 2: Inter-annotator agreement for function annotation

Fleiss	**Krippendorff**	**Pairwise avg.**
A_obs = 0.98	D_obs = 0.02	% agr = 98
A_exp=0.776	D_exp = 0.225	Kappa=0.913
Kappa=0.912	Alpha = 0.912	

Table 3: Inter-annotator agreement for polarity annotation

Regarding the context length for classification, in the annotation results, we noticed that the context length chosen by coders largely corresponds to just one statement: the one with the citation. With less frequency appears a length context of two or three sentences. It is probable that the context should not include more than three sentences to cover all the necessary information about the reference.

Context length	**Number of occurrences**
One sentence	1502
Two sentences	377
Three sentences	127
Four sentences	56
Five or more sentences	30

Table 4: Citation context length chose for annotators

Table 4 shows the number of sentences chose for annotators for citation context. In 95.6% of cases, the context length refers to one, two and three sentences including the one that contains the citation.

We evaluate performance indexes for function and polarity classification that uses the annotated keywords and labels as inputs. Results rated high for F-Measure, which demonstrate suitability of the chosen features for those classifications. We chose algorithms after the recommendations of our initial study [1]. In our results, SVM with SMO training has the best values; we show our experiment outcomes for function classification in Table 5; and, for polarity classification in Table 6. Used relation between tests vs. training datasets was 10% - 90%.

In Table 7, we present the relationship between function and its polarity.

Class	F-Measure
Useful	0.89
Weakness	0.94
Acknowledge, Corroboration, Debate	0.92
Based on, Supply	0.86
Contrast	0.89
Hedges	0.67
Weighted Avg.	**0.896**

Table 5: Function classification performance with SVM - SMO algorithm.

Previous studies presented results not as good for similar o less granularity. In [11], they used the model of [12], with four facets and their F1 scores varied from 0.68 for discriminating idea from a tool, to 0.51 for conformational / negational facets (similar to polarity), with scores between this minimum and this maximum for the other two classes. In [13], they classified fundamental idea /technical basis /comparison with F1 values of 0.66. In [5], they achieved F1 of 0.71 but just for polarity classification. In [14], they classified two function: corroborate and contrast with a recall of 0.83 for the first and 0.67 for the other. In [15], it was implemented a citation-classification algorithm through pattern matching, with a highest Recall of 0.49. In [16], they classified 10 citation functions to discover only 6 of them and a very variable F1 scores that go from 0.05 to 0.802 with an average of 0,49. In [17], they used a six-function scheme to obtain an average F macro of 0.58.

Class	F-Measure
Positive	0.94
Negative	1
Neutral	0.96
Weighted Avg.	**0.957**

Table 6: Polarity classification performance with SVM – SMO algorithm.

	Positive	Neutral	Negative
Useful	226	479	0
Weakness	0	0	123
Acknowledge, Corroboration, Debate	62	708	12
Based on, Supply	280	57	0
Contrast	14	69	25
Hedges	0	0	37

Table 7: Relationship between function and polarity classification

We noted that certain functions do not have results for some polarities. *Useful* do not appear as Negative; *Weakness* and *Hedges* are always with Negative polarity; and, Based on, Supply do not have occurrences with Negative polarity. All of that make sense from function and polarity definitions.

V. CONCLUSIONS

The developed scheme are consistent to citation purpose and citing author's disposition towards references. In further work, we intent to use this scheme and corpus for citation analysis to obtain influence levels of a citation in a paper. With this scheme, we annotated 85 ACL articles obtained randomly with 2092 citations. We suggest that this scheme and developed corpus could also be applied for summary generation and information retrieval, because of the clear organization of the scheme in general dimensions: PURPOSE: POLARITY: ASPECTS.

Annotation results are high with an Inter-Annotator agreement of 0.862 and 0.912 for citation function and polarity classification respectively. This kind of results we could not have obtained without our annotation methodology that has a pre-process of labeling patterns formed by keywords and labels that clarify the scheme dimensions. Later we also use these patterns as input features for the machine-learning algorithm for function and polarity classification.

We use the annotated corpus to perform automatic classification of citation function and polarity and we obtained an F1 weighted average of 0.896, which are higher than results in other studies. However, it is important to notice that annotated data in our corpus is relevant and delivers a sufficient amount of information to feed classifiers to yield optimal results; marked keywords and labels define what we called Aspects. For some other corpus, automatic annotation generally is performed just in a lexical and / or syntactic level and have lots of not pertinent information (noise). When other studies use these noisy annotations, they achieve low algorithm performance.

In contrast, we manually annotated our corpus, using an annotation scheme with relevant features organized according the scheme, that take into account citation context (inside a paragraph). These criteria form the basis for building a good model for automatic citation classification. Aspects annotated in a variable context length, give a great amount of information and allow achieving satisfactory results for function and polarity citation classification. According our results optimal context length could be from one to three sentences around a citation.

Classification results in our experiments confirm the validity of our classification scheme. If an application requires a trusty classification, it is important to define relevant features that should be included in any annotation effort, manual or automatic; they give information that is indispensable for good results.

In summary, in the present work, we intent to contribute with the following:

- A proposed annotation scheme simple in its structure, but with high granularity thanks to the combination of

information from function, polarity, keywords and semantic labels, organized in three dimensions: PURPOSE: POLARITY: ASPECTS.

- The annotation methodology, particularly regarding to the pre-annotation process to detect keywords and labels that are useful to create mental models in the annotators. These characteristics also serve as input features in classification algorithms. Therefore, we used keywords and labels to improve Inter-annotator agreement, but also we applied those to increase the granularity of the corpus.
- An annotated corpus with a sufficient size that contains those relevant features and is accessible for collaborative work. The XML files for our annotated corpus is available in the University of Alicante digital repository [18].
- The experimental finding that the significant context around a citation usually takes no more than three sentences including the one with the mention.

As future work, we will continue populating the corpus with new annotated documents and new collaborative tools for manual annotation.

There are controversies regarding counting approaches to measure citation impact, because they consider all citations as equal regardless of the purpose or the polarity with which they were mentioned. In [19], it was showed that incomplete, erroneous, or controversial papers have higher citation counts. Therefore, we plan to use the corpus to obtain citation influence in a paper using a machine-learning algorithm using as features the same dimensions: PURPOSE: POLARITY: ASPECTS, with additional information: citation position in an IMRAD paper structure, and frequency of the citation in the different sections of the paper. For this new challenge, we are labeling the training dataset with answers of authors of citing papers that will state influence of the works they cited. We are sending a survey with this request to the authors of the articles in our corpus and we are in the process of receiving and tabulating answers. We will use this information to measure precision in our influence classification.

Due to the reliability that is obtained in our manual corpus annotation, we suggest that, in the near future, the data continue to be annotated manually using our methodology. We state that it is necessary to improve current automatic annotation techniques marking relevant information for obtaining reliable results when applied to an annotation scheme with medium or high granularity.

Regarding to automatic annotation, as future work, we consider that our scheme and detected features determine a clearer path for the development of automatic annotation techniques, because we divide a complex task in ones that are more manageable. It would be easier for an automatic classifier to recognize characteristic patterns for each of our defined dimensions. From the lexicon created for this study, we intent to develop an automatic annotation process for marking keywords and labels using simple techniques as bag_of_words and n-gram detection.

REFERENCES

[1] Hernández Álvarez, M., & Gómez Soriano, J. (2015b). Survey about citation context analysis: Tasks, techniques, and resources. *Natural Language Engineering*. Cambridge University Press. Available on CJO 2015 doi: 10.1017/S1351324915000388

[2] Athar, A. (2014). Sentiment analysis of scientific citations. Technical Report, University of Cambridge. (UCAM-CL-TR-856).

[3] Mandya, A. A. (2012). Enhancing Citation Context based Information Services through Sentence Context Identification. Doctoral dissertation, University of Otago. Retrieved from: http://hdl.handle.net/10523/2520

[4] Ciancarini, P., Di Iorio, A., Nuzzolese, A. G., Peroni, S., & Vitali, F. (2014). Evaluating citation functions in CiTO: cognitive issues. In *The Semantic Web: Trends and Challenges* pp. 580-94. Springer International Publishing.

[5] Teufel, S., Siddharthan, A., & Tidhar, D. (2006, July). Automatic classification of citation function. In *Proceedings of the 2006 conference on empirical methods in natural language processing* (pp. 103-110). Association for Computational Linguistics.

[6] Hernández Álvarez, M., & Gómez Soriano, J. (2015a). Esquema de anotación para categorización de citas en bibliografía científica. *Procesamiento del Lenguaje Natural*, 54, 45-52.

[7] Hyland, K. 1998. *Hedging in Scientific Research Articles*, Vol. 54. Amsterdam: John Benjamins Publishing.

[8] Artstein, R., & Poesio, M. 2008. Inter-coder agreement for computational linguistics. *Computational Linguistics*, 34(4), 555-96.

[9] Krippendorff, Klaus. 2004. Reliability in content analysis: Some common misconceptions and recommendations. *Human Communication Research*, 30(3):411–33.

[10] Landis, J. R., & Koch, G. G. (1977). The measurement of observer agreement for categorical data. *Biometrics*, 159-74.

[11] Jochim, C., and Schütze, H. 2012. Towards a generic and flexible citation classifier based on a faceted classification scheme. In *Procedings of COLING'12* (pp. 1343–58).

[12] Moravcsik, M. J., & Murugesan, P. (1975). Some results on the function and quality of citations. *Social studies of science*, 5(1), 86-92.

[13] Dong, C., and Schäfer, U. 2011. Ensemble-style self-training on citation classification. In *Proceedings of 5th International Joint Conference on Natural Language Processing*, pp. 623–31. Chiang Mai, Thailand: Asian Federation of Natural Language Processing.

[14] Meyers, A. 2013. Contrasting and corroborating citations in journal articles. In *Proceedings of the International Conference Recent Advances in Natural Language Processing RANLP 2013*, Hissar, Bulgaria, pp. 460–6.

[15] Iorio, A., Di, Nuzzolese, A. G., and Peroni, S. 2013. Towards the automatic identification of the nature of citations. In *SePublica*, Montpellier, France, pp. 63–74.

[16] Li, X., He, Y., Meyers, A., and Grishman, R. 2013. Towards fine-grained citation function classification. In *Proceedings of Recent Advances in Natural Language Processing*, Hissar, Bulgaria, pp. 402–7.

[17] Abu-Jbara, A., Ezra, J., and Radev, D. 2013. Purpose and polarity of citation: Towards NLP-based bibliometrics. In *Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. ACL. Atlanta, GA. pp. 596–606.

[18] Concit – Corpus (2015), Universidad de Alicante digital repository. http://hdl.handle.net/10045/47416.

[19] Radicchi, F. 2012. **In science "there is no bad publicity": Papers criticized** in comments have high scientific impact. *Nature Scientific Reports* 2: 815.

Myriam Hernández-Álvarez received the Electronic and Telecomunications Engineering degree from Escuela Politécnica Nacional, Quito, Ecuador (1982); the Master of Science degree in Computer Science from Ohio University at Athens, Ohio, USA (1987); a Specialist degree in Business Management, Universidad Andina Simón Bolivar (1998); PhD Degree in Informatic Applications, Universidad de Alicante, Alicante, España (2015). Currently, she is doing research in the field of Natural Language Engineering and is Dean of the Systems Engineering Faculty of the Escuela Politécnica Nacional.

Patricio Martínez Barco, received his Master Degree in Computer Science Engineering by the Universidad de Alicante (1994) and his PhD in Computer Science (2001). He is member of the Language Processing and Information System Research Group (GPLSI) at the Universidad de Alicante; Member of the Natural Language Processing InterUniversity Group (Universidad Politécnica de Valencia and Universidad de Alicante) and Vicepresident of the Spanish Society for Natural Language Processing - SEPLN.

José Manuel Gómez Soriano, received his PhD degree by the Universidad Politécnica de Valencia in 2007. Currently he works as a researcher at the Universidad de Alicante, Alicante, Spain, as a member of the Natural Language Processing and Information System Group. Project: Intelligent, Interactive and Multilingual Text Mining based on Human Language Technologies.

Un enfoque Multi-Objetivo a la optimización del Alineamiento Múltiple de Secuencias (MSA)

Cristian Zambrano-Vega, Miriam Cárdenas-Zea y Ricardo Aguirre-Pérez

***Resumen*—El Alineamiento Múltiple de Secuencias (MSA por sus siglas en inglés) es uno de los principales tópicos de interés en el campo de la BioInformática, consiste en encontrar un alineamiento óptimo para tres o más secuencias biológicas en el que exista la mayor cantidad de zonas conservadas o columnas de caracteres totalmente alineadas. Diferentes métricas para evaluar la calidad de los alineamientos han sido definidas en la literatura, lo que hace preciso que el problema MSA sea formulado y resuelto como un Problema de Optimización MultiObjetivo (MOP). Es por ello que hemos realizado un estudio de la resolución del problema bajo un enfoque MultiObjetivo considerando, como objetivos a optimizar simultáneamente, dos de los criterios de calidad más usados: la Suma Ponderada de Pares (the weighted Sum-Of-Pairs with affine gap penalties -wSOP) y el Porcentaje de Columnas Totalmente Alineadas (TC), realizando un análisis de rendimiento de tres algoritmos de mayor referencia en el área de la Optimización MultiObjectivo: NSGAII, SPEA2 y MOCell, resolviendo un conjunto de problemas del benchmark BAliBASE (V3.0). Los resultados revelan el alto grado de competividad del algoritmo NSGAII generando los mejores resultados, tanto desde una perspectiva multi-objetivo como bajo un aspecto biológico.**

***Index Terms*—Alineamiento Múltiple de Secuencias, Metaheurísticas de Optimización MultiObjetivo, Optimización, BioInformática.**

***Abstract*—Multiple Sequence Alignment (MSA) is one of the main topics in the bioinformatics domain, consists in finding an optimal alignment for three or more biological sequences with the number maximum of conserved zones or totally aligned columns. Different scores to assess the quality of the alignments have been proposed, so the problem can be formulated and resolved as a Multi-Objective Optimization Problem (MOP). For this reason we have carried out a perfomanced study resolving the MSA problem under a multi-objective approach, considering two popular metrics as objectives to be optimized: The weighted Sum-Of-Pairs with affine gap penalties (wSOP) and the Totally Aligned Columns (TC), with three algorithms from the state-of-the-art of Multi-Objective Optimization: NSGAII, SPEA2 and MOCell. Our experiments reveals that the classic metaheuristic NSGA-II provides the best overall performance resolving some problems provided by the benchmark BAliBASE (v3.0), under a multi-objective and biological approach.**

***Index Terms*—Multiple Sequence Alignment, MultiObjective Optimization Metaheuristics, optimization, BioInformatics.**

I. Introducción

EL alineamiento múltiple de secuencias biológicas, sea ADN, ARN o estructuras primarias proteicas (proteinas), es uno de los principales tópicos de interés dentro del campo de la BioInformática [1]. Su objetivo principal es la representar y comparar más de dos secuencias de aminoácidos o nucleótidos para resaltar la mayor cantidad de zonas de similitud entre ellas, las cuales podrían indicar relaciones funcionales o evolutivas entre los genes o proteínas consultadas. Su importancia radica en que de la calidad de los alineamientos depende la exactitud y presición de otros procesos bioinformáticos que se realizan a partir de tales secuencias alineadas, como son la Inferencia Filognética y la predicción estructural y funcional de proteínas.

El alineamiento de un par de secuencias se puede realizar mediante el uso de técnicas de Programación Dinámica [2]. Sin embargo, estas estrategias no pueden ser aplicadas cuando se consideran más de dos secuencias en el proceso, debido a que el espacio de búsqueda crece de forma exponencial según el número y el tamaño de las secuencias consideradas [3]. Por estas razones, cada vez más, se considera importante y necesario el uso de metaheurísticas de optimización en la resolución del problema.

El procedimiento de alineación básica se basa principalmente en la inserción de espacios o huecos (**gaps**) representados por el caracter "-"dentro del conjunto de caracteres de las secuencias, para hacer que todas ellas tengan la misma longitud y para lograr la alineación del mayor número de sus columnas. Es importante llevar a cabo la manipulación de las operaciones con los gaps (inserción, eliminación, desplazamiento, agrupamiento, etc.) con el fin de ir generando nuevas alternativas de alineaciones para mejorar la presición y calidad del alineamiento final, ya que el número de gaps y sus ubicaciones determinan finalmente la calidad del mismo.

Se han propuesto una serie de métricas diferentes para medir la precisión y calidad de los alineamientos, tales como: el porcentaje de columnas totalmente alineadas (TC), el porcentaje de caracteres -No espacios- (NonGapsP), la Suma de pares (Sum-of-Pairs, SOP), la suma ponderada de pares con penalidad de gaps afines (weighted Sum-of-Pairs, wSOP), Strike [4], Entropy [5], BAliScore [5] o MetAl [6]. Sin embargo, todavía no existe un consenso acerca de qué métrica es la más apropiada o la más precisa para medir la calidad de los alineamientos. Por esta razón, es necesario considerar un enfoque MultiObjetivo para optimizar el problema, que permita obtener de forma simultánea alineamientos optimizados bajo dos o más criteriores de evaluación, a fin de que los biólogos puedan disponer, no de una, sino de un conjunto de soluciones que les brinde la posibilidad de escoger una mejor

Los autores Cristian Zambrano-Vega (czambrano@uteq.edu.ec), Miriam Cárdenas-Zea (mcardenas@uteq.edu.ec) y Ricardo Aguirre-Pérez (gaguirre@uteq.edu.ec), son docentes de la Carrera de Ingeniería en Sistemas de la Unidad de Estudios a Distancia de la Universidad Técnica Estatal de Quevedo. Quevedo - Los Ríos - Ecuador.

C. Zambrano-Vega se encuentra cursando sus estudios de Doctorado en Ingeniería del Software e Inteligencia Artificial de la Universidad de Málaga - España gracias al programa de Becas Doctorales de la Secretaría Nacional de Educación Superior, Ciencia y Tecnología - SENESCYT.

solución disyuntiva.

Es por esto que el objetivo principal de este artículo es brindar un estudio de rendimiento multiobjetivo al problema del MSA, empleando tres de las principales metaheurístcias de Optimización multiobjetivo: la técnica mayormente conocica NSGA-II [7], el algoritmo clásico SPEA2 [8] y el algoritmo celular MOCell [9], considerando como funciones objetivo a optimizar de forma silmultánea dos de las métricas más comunes y usadas en el problema MSA la Suma Ponderada de Pares con penalidad de gaps afines (wSOP) y el porcentaje de columnas totalmente alineadas (TC).

El resto del trabajo se organiza de la siguiente manera. En la Sección II se presenta una revisión de los trabajos relacionados a la optimización multiobjetivo aplicada al MSA, una descripción formal y una formulación multiobjetivo del problema se describen en la Sección III. Los detalles de la metodología empleada en la experimentación se presentan en la Sección IV, el análisis y discusión de los resultados obtenidos se realiza en la Sección V. Finalmente, las conclusiones y líneas de trabajo futuro se comentan en la Sección VI.

II. Trabajos Relacionados

En esta sección se revisan brevemente algunas propuestas publicadas en el estado del arte para resolver el problema del MSA utilizando técnicas de optimización multiobjetivo, entre ellas:

Ortuño *et al.* [10] implementaron un algoritmo evolutivo multiobjetivo basado en la técnica clásica NSGA-II y se establecieron tres objetivos a optimizar: la métrica *STRIKE*, porcentaje de NONGaps y porcentaje de columnas totalmente alineadas (TC).

El algoritmo Parallel Niche Pareto AlineaGA (*PNPAlineaGA*) fue propuesto por da Silva *et al.* [11] y está basado en un modelo de islas paralelizado, emplea una formulación bi-objetivo del problema MSA usando las métricas de calidad SOP y el número total de columnas alineadas.

Soto y Becerra [5], propusieron un algoritmo evolutivo multiobjetivo, también inspirado en NSGA-II, para optimizar secuencias previamente alineadas por otras técnicas. Utilizaron dos funciones objetivo para comparar la calidad de los alineamientos: Entropy y la métrica *MetAl*.

Un algoritmo genético multiobjetivo también basado en NSGA-II (*MSAGMOGA*) se describe en [12], donde se consideran tres objetivos: Similarity, Affine Gap Penalty y Support.

Recientemente, Abbasi *et al.* [13] presentaron una Búsqueda Local (LocalSearch) al enfoque multi-objetivo, en el que los objetivos a optimizar son SOP y reducir al mínimo el número de espacios en los alineamientos.

Los dos primeros trabajos descritos usan una estrategia de inicialización basada en generar los primeros descendientes a partir de alineaciones pre-computadas con herramientas clásicas como ClustalW, MUSCLE, Kalign, Mafft, RetAlign, TCOFFEE, ProbCons y FSA.

III. Alineamiento Múltiple de Secuencias (MSA)

En esta sección, se proporciona una definición formal y una formulación bi-objetivo al problema MSA.

III-A. Definición del problema

Dado un alfabeto finito Σ y un conjunto de k secuencias biológicas $S = (s_1, s_2, ..., s_k)$ de longitudes variables l_1 to l_k compuestas de caracteres $s_i = s_{i1}s_{i2}, ..., s_{il_i} (1 \leq i \leq k)$, donde para las secuencias de ADN, Σ consiste de cuatro nuclétidos representados por los caracteres {A, T, G, C} y para las secuencias de proteinas, Σ consiste de 20 amino ácidos representados por los caracteres {A, C, D, E, F, G, H, I, K, L, M, N, P, Q, R, S, T, V, W, Y}; S' es una matriz que representa el alineamiento óptimo de S, la cual está definida formalmente por la siguiente ecuación 1:

$$S' = (s'_{ij}),\ con\ 1 \leq i \leq k, 1 \leq j \leq l, max(l_i) \leq l \leq \sum_{i=1}^{k} l_i \tag{1}$$

Cumpliendo con:

1. $s'_{ij} \in \Sigma \cup \{-\}$, donde "-"denota el carácter de espacios o "gaps";
2. cada fila $s'_i = s'_{i1}s'_{i2}, ..., s'_{il} (1 \leq i \leq k)$ de S' es exactamente igual a la secuencia correspondiente s_i si eliminanos todos los gaps;
3. La longitud de todas las k secuencias es exactamente la misma;
4. S' no tiene columnas conformada solo por gaps.

Un ejemplo de alineamiento se muestra a continuación, en el se representan cuatro secuencias con seis columnas alineadas las cuales están marcadas con un asterisco (*).

```
APPSVFAEVPJQKTM-AQPVMKLJ
AKRS-V-E-PJFKTMR-IKMK---
-LISKRA-YPJ-KTM-I---MALP
-SASTIGVEPJCK-M-RA-P--KL
*         ** ***
```

III-B. Formulación bi-Objetivo del problema MSA

Hemos seleccionado dos de las funciones objetivo más utilizadas en el estado del arte: *(obj 1)* la Suma ponderada de pares con penalidad a los gaps afines (wSOP) y *(obj 2)* el TC. Ambas deben ser optimizadas, maximizando su valor, de forma simultánea. A continuación se presenta la formulación bi-objetivo del problema en la ecuación 2:

$$maximize\ F(S) = \{f_1(S'), f_2(S')\} \tag{2}$$

donde $f_1(S')$ y $f_2(S')$ son las funciones de las métricas wSOP y TC respectivamente, y S' es el alineamiento a evaluar.

III-C. La suma ponderada de pares

La suma ponderada de pares (wSOP) se calcula restando el puntaje de suma de pares (comparaciones entre pares de cada uno de los caracteres amino-ácidos o nucleótidos) de cada una de las columnas del alineamiento menos el puntaje de penalidad a los gaps afines de cada una de las secuencias. La wSOP está representada por la ecuación 3:

$$wSOP(S') = \sum_{l=1}^{L} SP(l) - \sum_{i=1}^{k} AGP(s'_i) \tag{3}$$

Donde L representa el tamaño o longitud del alineamiento, $SP(l)$ representa al puntaje de la suma de pares de la columna l el cuál está denifido como (ecuación 4):

$$SP(l) = \sum_{i=1}^{k-1} \sum_{j=i+1}^{k} W_{i,j}\ x\ \delta(s'_{i,l}, s'_{j,l}) \quad (4)$$

Note que en la ecuación 4, $W_{i,j}$ representa la ponderación (pesos) entre las sequencias s'_i y s'_j, para lo cual hemos definido la siguiente ecuación:

$$W_{i,j} = 1 - \frac{LD(s_i, s_j)}{max(|s_i|, |s_j|)} \quad (5)$$

Donde LD representa la distancia de *Levenshtein* entre dos secuencias no alineadas (s_i y s_j).

Y δ, en la ecuación 4, representa la matrix de sustitución (*Pointed Accepted Mutation*, - PAM [14] o *Block Substitution Matrix*, - BLOSUM [15]), la cual proporciona los costos de alineamientos de pares para cada uno de los aminoácidos y el valor de penalidad que se tiene al alinear un caracter con un gap. En el presente trabajo hemos usado los valores de la matriz de sustitución Blosum62 sin penalidad de alineamiento con gaps.

Finalmente $AGP(s'_i)$ en (3) representa la penalidad por gaps afines de la secuencia s'_i, definda por la siguiente ecuación:

$$AGP(s'_i) = (g_{open}\ x\ \#gaps) + (g_{extend}\ x\ \#spaces) \quad (6)$$

Donde g_{open} es el peso por empezar con un gap y g_{extend} es el peso por extender el gap con uno o mas espacios. En este trabajo hemos usado los siguientes valores para estos parámetros: $g_o = 6$ y $g_e = 0{,}85$.

III-D. El porcentaje de columnas totalmente alineadas

El porcentaje de columnas totalmente alineadas (TC) se refiere al número de columnas que están compuestas totalmente del mismo caracter en cada una de sus filas (amino ácidos o nucleótidos). Esta función objetivo necesita ser maximizada para asegurar la mayor cantidad de regiones conservadas dentro del alineamiento.

Cabe resaltar que estas dos funciones objetivo entran en conflicto entre ellas, ya que la mejora del TC implica añadir gaps al alineamiento, los cuales son penalizados por la matriz de sustitución utilizada para calcular el score SOP.

IV. Materiales y Métodos

En esta sección, describimos brevemente los algoritmos que hemos seleccionado, el conjunto de problemas de prueba escogido y la metodología de la experimentación.

IV-A. Metaheurísticas de optimización multiObjetivo

Los algoritmos que hemos considerado para este enfoque multiobjetivo son los siguientes:

- **NSGA-II** [7] es un algoritmo genético generacional basado en la creación de nuevos individuos a partir de la población original mediante la aplicación de los operadores genéticos de selección, cruce y mutación. Se aplica un procedimiento de rankeo para promover la convergencia, mientras que un estimador de densidad (La distancia de *Crowding*) se utiliza para mejorar la diversidad del conjunto de soluciones encontradas.
- **SPEA2** [8] es, al igual que NSGA-II, un algoritmo clásico en el campo de optimización multiObjetivo y ampliamente utilizado. Se caracteriza por el uso de una población y de un archivo de soluciones; así como también considera la aplicación de un riguroso control de la función fitness y la distancia al *k*-ésimo vecino más cercano para fomentar la convergencia y la diversidad de las soluciones, respectivamente.
- **MOCell** [16] es un algoritmo evolutivo multiobjetivo celular que utiliza un archivo externo para almacenar las soluciones no dominadas encontradas durante la búsqueda. Aplica un estimador de densidad (la misma distancia de *Crowding* utilizada en NSGA-II) para seleccionar qué solución debe ser removida cuando el tamaño del archivo supera su capacidad máxima.

IV-B. Indicadores de calidad multiObjetivo

En este estudio se han utilizado las siguientes métricas para evaluar la calidad de los resultados multiobjetivo: El **Hipervolumen ($\mathbf{I}_{HV}$)** [17], diseñada para medir los aspectos de convergencia y diversidad en un frente de Pareto. Esta métrica calcula el volumen (en el espacio de objetivos) cubierto por miembros de un conjunto Q, de soluciones no dominadas para problemas donde todos los objetivos han de ser minimizados. Los algoritmos que alcanzan mayores valores para I_{HV} son los mejores. Adicionalmente se usó el indicador **Épsilon ($\mathbf{I}_{E^+}$)** [18], que mide la convergencia al determinar la distancia mínima (en cualquier objetivo) que habría que desplazar cada solución para ser no dominada con respecto a otro frente (cuanto más pequeño mejor); y el indicador Dispersión (**Spread ($\mathbf{I}_{\Delta}$)**) el cual mide la distribución de las soluciones y la distancia con los extremos del verdadero frente de Pareto (su valor ideal es cero).

IV-C. Problemas de prueba

En la actualidad, una gran variedad de benchmarks han sido publicados para evaluar la calidad de los alineamientos generados por las técnicas presentadas, entre ellos está: OX-Bench [19], HOMSTRAD [20] y Prefab [21]. Para nuestro estudio se ha escogido el de mayor referencia en la literatura el dataset BAliBASE (v3.0) [22], un conjunto de secuencias extraídas manualmente desde el Protein Data Bank (PDB) [23], en particular, hemos elegido las secuencias del subconjunto RV11, el cual contiene 38 conjuntos de secuencias PDB las cuales comparten menos del 20 % de similitud entre cada par de secuencias y menos de 35 inserciones.

IV-D. Adaptación de los algoritmos al problema MSA

La codificación de las soluciones (alineamientos) están representados por una matriz de carácteres. Como se describe

en la sub-sección III-B las funciones objetivo a optimizar son la suma ponderada de pares (wSOP) y el TC. Se empleó la matriz de sustitución Blosum62 con valores de $g_o = 6$ y $g_e = 0{,}85$.

Todas las metaheurísticas usan los mismos operadores genéticos de reproducción, selección de individuos por Torneo Binario, cruce con el operador *Single-Point Crossover*, el cual corta en un punto aleatorio a los dos padres y para crear los dos nuevos descendientes mezcla ambas partes de ambos padres, y el operador de mutación *Shift-Closed-Gaps*, el cual mueve un conjunto de gaps cercanos a una posición aleatoria dentro de la misma secuencia. Las implementaciones de estos dos operadores, cruce y mutación, está basada en [10] la cual se describe en la sección II.

La estrategia de creación de la población inicial es similar a la usada por los trabajos presentados por da Silva *et al.* [11] y Ortuño *et al.* [10], descritos en la sección II, se crea a partir de un subconjunto de alineamientos previamente generados por las herramientas: ClustalW, MUSCLE, Kalign, Mafft, RetAlign, TCOFFEE, ProbCons y FSA, y aplicando el operador de Cruce *Single-Point Crossover* para generar los nuevos descendientes que la conforman.

Todos los algoritmos y el problema de MSA fueron implementados en jMetal versión 5 [24].

IV-E. Experimentación

Todos los algoritmos fueron configurados con los mismos valores con el objetivo de hacer una comparativa exacta entre ellos.

Cada ejecución independiente de cada uno de los algoritmos dura hasta 50000 evaluaciones computadas, el tamaño de la población es de 100 individuos, la probabilidad de Cruce es del 80 % y la de mutación del 20 %. El tamaño de archivo del algoritmo MOCell es de 100 soluciones.

Se llevaron a cabo 10 ejecuciones independientes de cada algoritmo resolviendo cada uno de los 38 problemas del grupo RV11 del BAliBASE (v3.0). Se han obtenido las medianas y rangos intercuartílicos (IQR) de los resultados de los indicadores de calidad descritos en la sub-sección IV-B, como medidas de localización y dispersión, respectivamente. En las tablas de resultados, los mejores de cada experimento son marcados con un fondo gris más oscuro para ser resaltados y los segundos mejores resultados son marcados con tono de gris más claro.

Además para comprobar si las diferencias obtenidas entre los algoritmos son estadísticamente significativas, se ha aplicado el test de Wilcoxon Rank-Sum con un nivel de confianza del 95 % para cada par de algoritmos, primero entre NSGAII y MOCell, y luego MOCell y SPEA2. Para ilustrar estos resultados, en las tablas se usa la siguiente simbología: el carácter '-' indica que no hay diferencias significativas entre los algoritmos de la fila y columna, el símbolo ▲ significa que el algoritmo de la fila ha producido mejores resultados que el algoritmo de la columna con significancia estadística, y el símbolo ▽ se utiliza cuando el algoritmo en la columna es estadísticamente mejor que el de la fila en el problema considerado.

Como no se conoce el frente de pareto óptimo de los problemas del BAliBASE, se ha generado un Frente de Pareto Referencial para cada problema, uniendo todos los frentes de pareto aproximados generados por todas las ejecuciones independientes de los tres algoritmos, descartando las soluciones dominadas.

Tabla I: Mediana y Rangos Intercuartílicos de los valores del indicador I_{HV}.

	NSGAII	MOCell	SPEA2
BB11001	$5{,}16e{-}01_{7{,}3e-02}$	$5{,}14e{-}01_{3{,}4e-02}$	$5{,}56e{-}01_{7{,}2e-02}$
BB11002	$0{,}00e{+}00_{5{,}9e-02}$	$4{,}71e{-}02_{1{,}4e-01}$	$8{,}48e{-}02_{1{,}4e-01}$
BB11003	$6{,}74e{-}01_{3{,}4e-02}$	$6{,}37e{-}01_{2{,}5e-02}$	$6{,}23e{-}01_{4{,}2e-02}$
BB11004	$5{,}96e{-}01_{5{,}9e-02}$	$5{,}74e{-}01_{2{,}6e-02}$	$5{,}56e{-}01_{5{,}1e-02}$
BB11005	$0{,}00e{+}00_{0{,}0e+00}$	$0{,}00e{+}00_{0{,}0e+00}$	$0{,}00e{+}00_{0{,}0e+00}$
BB11006	$6{,}64e{-}03_{2{,}1e-02}$	$0{,}00e{+}00_{4{,}6e-04}$	$1{,}21e{-}02_{6{,}8e-02}$
BB11007	$4{,}00e{-}01_{6{,}4e-02}$	$4{,}22e{-}01_{4{,}6e-02}$	$4{,}20e{-}01_{7{,}1e-02}$
BB11008	$0{,}00e{+}00_{5{,}9e-02}$	$4{,}09e{-}02_{1{,}1e-01}$	$6{,}26e{-}02_{1{,}5e-01}$
BB11009	$4{,}73e{-}01_{6{,}0e-02}$	$4{,}33e{-}01_{9{,}9e-02}$	$4{,}58e{-}01_{8{,}5e-02}$
BB11010	$6{,}66e{-}01_{6{,}6e-02}$	$6{,}39e{-}01_{1{,}1e-01}$	$6{,}93e{-}01_{7{,}8e-02}$
BB11011	$3{,}00e{-}01_{9{,}9e-02}$	$3{,}16e{-}01_{1{,}2e-01}$	$2{,}43e{-}01_{1{,}3e-01}$
BB11012	$5{,}02e{-}01_{1{,}5e-01}$	$5{,}17e{-}01_{1{,}4e-01}$	$5{,}61e{-}01_{1{,}1e-01}$
BB11013	$5{,}75e{-}02_{2{,}5e-01}$	$0{,}00e{+}00_{6{,}5e-03}$	$0{,}00e{+}00_{1{,}6e-02}$
BB11014	$8{,}02e{-}03_{3{,}2e-02}$	$7{,}99e{-}03_{1{,}5e-02}$	$1{,}47e{-}02_{5{,}7e-02}$
BB11015	$4{,}67e{-}01_{1{,}1e-01}$	$4{,}08e{-}01_{1{,}6e-01}$	$4{,}15e{-}01_{9{,}6e-02}$
BB11016	$4{,}72e{-}01_{7{,}2e-02}$	$5{,}28e{-}01_{1{,}1e-01}$	$4{,}69e{-}01_{1{,}3e-01}$
BB11017	$5{,}84e{-}01_{6{,}4e-02}$	$5{,}36e{-}01_{4{,}9e-02}$	$5{,}27e{-}01_{8{,}5e-02}$
BB11018	$1{,}35e{-}01_{1{,}9e-01}$	$3{,}64e{-}02_{1{,}7e-01}$	$8{,}15e{-}02_{2{,}8e-01}$
BB11019	$4{,}87e{-}01_{2{,}8e-01}$	$2{,}81e{-}01_{2{,}1e-01}$	$4{,}10e{-}01_{2{,}3e-01}$
BB11020	$4{,}58e{-}01_{8{,}3e-02}$	$4{,}03e{-}01_{1{,}4e-01}$	$4{,}26e{-}01_{5{,}8e-02}$
BB11021	$3{,}76e{-}01_{1{,}4e-01}$	$4{,}63e{-}01_{9{,}1e-02}$	$4{,}42e{-}01_{7{,}0e-02}$
BB11022	$2{,}40e{-}01_{5{,}2e-02}$	$2{,}47e{-}01_{6{,}7e-02}$	$2{,}05e{-}01_{1{,}2e-01}$
BB11023	$2{,}10e{-}01_{3{,}2e-01}$	$0{,}00e{+}00_{1{,}3e-01}$	$2{,}39e{-}01_{2{,}3e-01}$
BB11024	$2{,}62e{-}01_{1{,}6e-01}$	$3{,}00e{-}01_{1{,}9e-01}$	$3{,}06e{-}01_{2{,}0e-01}$
BB11025	$5{,}18e{-}01_{9{,}1e-02}$	$4{,}45e{-}01_{1{,}4e-01}$	$4{,}73e{-}01_{1{,}2e-01}$
BB11026	$0{,}00e{+}00_{0{,}0e+00}$	$0{,}00e{+}00_{0{,}0e+00}$	$0{,}00e{+}00_{0{,}0e+00}$
BB11027	$4{,}06e{-}01_{1{,}1e-01}$	$3{,}34e{-}01_{1{,}6e-01}$	$4{,}00e{-}01_{9{,}4e-02}$
BB11028	$2{,}89e{-}02_{9{,}1e-02}$	$2{,}53e{-}03_{4{,}7e-02}$	$0{,}00e{+}00_{1{,}0e-02}$
BB11029	$5{,}97e{-}01_{7{,}9e-02}$	$5{,}87e{-}01_{8{,}3e-02}$	$4{,}86e{-}01_{1{,}5e-01}$
BB11030	$2{,}22e{-}01_{3{,}5e-01}$	$1{,}86e{-}01_{2{,}6e-01}$	$3{,}18e{-}02_{2{,}3e-01}$
BB11031	$6{,}12e{-}01_{2{,}3e-01}$	$4{,}07e{-}01_{3{,}3e-01}$	$4{,}48e{-}01_{1{,}8e-01}$
BB11032	$2{,}69e{-}01_{1{,}2e-01}$	$1{,}10e{-}01_{1{,}3e-01}$	$2{,}69e{-}01_{1{,}2e-01}$
BB11033	$0{,}00e{+}00_{4{,}5e-03}$	$0{,}00e{+}00_{0{,}0e+00}$	$1{,}08e{-}01_{8{,}4e-02}$
BB11034	$2{,}13e{-}01_{6{,}2e-02}$	$1{,}62e{-}01_{1{,}1e-01}$	$2{,}73e{-}01_{7{,}1e-02}$
BB11035	$2{,}51e{-}01_{1{,}6e-01}$	$3{,}09e{-}01_{2{,}8e-01}$	$2{,}76e{-}01_{1{,}2e-01}$
BB11036	$5{,}68e{-}01_{2{,}0e-01}$	$5{,}32e{-}01_{1{,}9e-01}$	$6{,}36e{-}01_{1{,}4e-01}$
BB11037	$5{,}77e{-}01_{1{,}2e-01}$	$4{,}99e{-}01_{9{,}1e-02}$	$5{,}95e{-}01_{6{,}1e-02}$
BB11038	$4{,}37e{-}02_{4{,}0e-01}$	$0{,}00e{+}00_{0{,}0e+00}$	$0{,}00e{+}00_{0{,}0e+00}$

V. Resultados y Discusión

A continuación se detallan los resultados del rendimiento de los tres algoritmos.

V-A. Resultados multi-objetivo

La mediana y los rangos intercuartílicos IQRs de los valores de los indicadores de calidad HIPERVOLUMEN (I_{HV}), EPSILON (I_{E^+}) y SPREAD (I_Δ) se ilustran en las tablas I, II y III respectivamente. Hay que considerar que en el caso del indicador I_{HV} los mayores valores son los mejores, y de forma contraria (los menores valores) para los indicadores I_{E^+} y I_Δ.

Los resultados obtenidos por el indicador de calidad Hypervolumen (I_{HV}), el cual mide la convergencia y la diversidad de los frentes de Pareto generados por los algoritmos indican que, bajo un enfoque multiobjetivo, NSGAII es el algoritmo que mejores resultados genera para la mitad de los problemas del benchmark; 19 de los 38 problemas (50 %) son resueltos de mejor manera por NSGAII resaltando su alta competitividad frente a MOCell y SPEA2, los cuales logran ser mejores en 6 y 13 problemas, respectivamente.

Además con el objetivo de medir de forma individual la convergencia y la diversidad de los resultados multiobjetivo,

Tabla II: Mediana y Rangos Intercuartílicos de los valores del indicador I_{E^+}.

	NSGAII	MOCell	SPEA2
BB11001	$3{,}06e{-}01_{1{,}6e-01}$	$3{,}06e{-}01_{4{,}4e-02}$	$1{,}87e{-}01_{1{,}5e-01}$
BB11002	$7{,}84e{-}01_{1{,}3e-01}$	$6{,}20e{-}01_{2{,}2e-01}$	$6{,}67e{-}01_{2{,}9e-01}$
BB11003	$1{,}32e{-}01_{3{,}4e-02}$	$1{,}28e{-}01_{3{,}7e-02}$	$1{,}92e{-}01_{5{,}9e-02}$
BB11004	$1{,}44e{-}01_{4{,}5e-02}$	$1{,}53e{-}01_{4{,}6e-02}$	$1{,}49e{-}01_{4{,}2e-02}$
BB11005	$1{,}27e{+}01_{9{,}6e+00}$	$1{,}27e{+}01_{2{,}8e+00}$	$1{,}25e{+}01_{6{,}6e+00}$
BB11006	$7{,}32e{-}01_{5{,}0e-01}$	$1{,}01e{+}00_{4{,}2e-01}$	$7{,}40e{-}01_{5{,}1e-01}$
BB11007	$4{,}46e{-}01_{1{,}8e-01}$	$5{,}01e{-}01_{2{,}4e-02}$	$3{,}07e{-}01_{1{,}6e-01}$
BB11008	$8{,}20e{-}01_{3{,}0e-01}$	$7{,}56e{-}01_{4{,}1e-01}$	$6{,}69e{-}01_{2{,}4e-01}$
BB11009	$2{,}09e{-}01_{7{,}7e-02}$	$2{,}22e{-}01_{9{,}0e-02}$	$2{,}37e{-}01_{4{,}3e-02}$
BB11010	$1{,}24e{-}01_{5{,}1e-02}$	$1{,}53e{-}01_{8{,}3e-02}$	$1{,}10e{-}01_{5{,}4e-02}$
BB11011	$4{,}63e{-}01_{8{,}0e-02}$	$3{,}99e{-}01_{1{,}8e-01}$	$4{,}77e{-}01_{8{,}5e-02}$
BB11012	$3{,}40e{-}01_{2{,}5e-01}$	$3{,}22e{-}01_{2{,}3e-01}$	$1{,}84e{-}01_{2{,}5e-01}$
BB11013	$5{,}64e{-}01_{3{,}5e-01}$	$9{,}77e{-}01_{2{,}8e-01}$	$8{,}34e{-}01_{4{,}2e-01}$
BB11014	$9{,}86e{-}01_{1{,}2e-01}$	$9{,}86e{-}01_{2{,}1e-02}$	$9{,}72e{-}01_{3{,}5e-01}$
BB11015	$2{,}27e{-}01_{5{,}2e-02}$	$2{,}36e{-}01_{5{,}2e-02}$	$2{,}68e{-}01_{1{,}0e-01}$
BB11016	$2{,}44e{-}01_{6{,}0e-02}$	$2{,}36e{-}01_{1{,}2e-01}$	$2{,}53e{-}01_{5{,}6e-02}$
BB11017	$1{,}80e{-}01_{1{,}8e-02}$	$1{,}82e{-}01_{1{,}3e-02}$	$1{,}87e{-}01_{2{,}1e-02}$
BB11018	$6{,}65e{-}01_{3{,}5e-01}$	$7{,}93e{-}01_{3{,}1e-01}$	$7{,}48e{-}01_{3{,}9e-01}$
BB11019	$4{,}40e{-}01_{3{,}5e-01}$	$5{,}87e{-}01_{2{,}3e-01}$	$4{,}71e{-}01_{2{,}9e-01}$
BB11020	$3{,}51e{-}01_{2{,}2e-02}$	$3{,}54e{-}01_{6{,}5e-02}$	$3{,}37e{-}01_{3{,}0e-02}$
BB11021	$2{,}73e{-}01_{1{,}6e-01}$	$1{,}83e{-}01_{4{,}3e-02}$	$1{,}96e{-}01_{8{,}0e-02}$
BB11022	$3{,}06e{-}01_{1{,}2e-01}$	$2{,}98e{-}01_{1{,}4e-01}$	$3{,}21e{-}01_{2{,}7e-01}$
BB11023	$6{,}21e{-}01_{5{,}1e-01}$	$1{,}03e{+}00_{3{,}9e-01}$	$6{,}08e{-}01_{3{,}0e-01}$
BB11024	$4{,}08e{-}01_{2{,}1e-01}$	$3{,}82e{-}01_{1{,}9e-01}$	$3{,}38e{-}01_{1{,}3e-01}$
BB11025	$1{,}97e{-}01_{7{,}5e-02}$	$2{,}32e{-}01_{6{,}8e-02}$	$2{,}14e{-}01_{9{,}3e-02}$
BB11026	$9{,}32e{+}00_{7{,}1e+00}$	$6{,}64e{+}00_{9{,}9e+00}$	$1{,}00e{+}01_{1{,}2e+01}$
BB11027	$3{,}31e{-}01_{2{,}6e-02}$	$3{,}32e{-}01_{3{,}8e-03}$	$3{,}31e{-}01_{1{,}3e-02}$
BB11028	$4{,}49e{-}01_{4{,}6e-01}$	$5{,}50e{-}01_{4{,}9e-01}$	$4{,}96e{-}01_{1{,}7e-01}$
BB11029	$1{,}49e{-}01_{5{,}2e-02}$	$1{,}60e{-}01_{1{,}1e-01}$	$2{,}75e{-}01_{1{,}5e-01}$
BB11030	$5{,}41e{-}01_{2{,}7e-01}$	$6{,}45e{-}01_{4{,}9e-01}$	$7{,}21e{-}01_{3{,}6e-01}$
BB11031	$2{,}01e{-}01_{1{,}2e-01}$	$3{,}31e{-}01_{1{,}9e-01}$	$3{,}23e{-}01_{2{,}1e-01}$
BB11032	$4{,}76e{-}01_{3{,}2e-02}$	$5{,}42e{-}01_{1{,}0e-01}$	$4{,}67e{-}01_{6{,}5e-02}$
BB11033	$1{,}00e{+}00_{3{,}2e-02}$	$1{,}70e{+}00_{9{,}6e-01}$	$7{,}44e{-}01_{9{,}7e-02}$
BB11034	$3{,}29e{-}01_{1{,}2e-01}$	$4{,}06e{-}01_{1{,}3e-01}$	$2{,}89e{-}01_{2{,}4e-02}$
BB11035	$3{,}87e{-}01_{2{,}0e-01}$	$2{,}77e{-}01_{2{,}5e-01}$	$3{,}15e{-}01_{2{,}0e-01}$
BB11036	$2{,}67e{-}01_{2{,}2e-01}$	$3{,}16e{-}01_{9{,}5e-02}$	$2{,}38e{-}01_{9{,}1e-02}$
BB11037	$2{,}21e{-}01_{7{,}1e-02}$	$2{,}30e{-}01_{4{,}9e-02}$	$2{,}14e{-}01_{5{,}6e-02}$
BB11038	$8{,}92e{-}01_{9{,}0e-01}$	$1{,}84e{+}00_{5{,}9e-01}$	$1{,}52e{+}00_{6{,}7e-01}$

Tabla III: Mediana y Rangos Intercuartílicos de los valores del indicador I_{Δ}.

	NSGAII	MOCell	SPEA2
BB11001	$1{,}47e{+}00_{1{,}2e-01}$	$6{,}12e{-}01_{1{,}7e-03}$	$1{,}63e{+}00_{2{,}1e-01}$
BB11002	$1{,}43e{+}00_{8{,}0e-02}$	$7{,}65e{-}01_{1{,}3e-01}$	$1{,}48e{+}00_{1{,}5e-01}$
BB11003	$1{,}26e{+}00_{5{,}3e-02}$	$6{,}79e{-}01_{9{,}3e-02}$	$1{,}32e{+}00_{8{,}8e-02}$
BB11004	$1{,}52e{+}00_{1{,}2e-01}$	$7{,}85e{-}01_{1{,}7e-01}$	$1{,}50e{+}00_{7{,}6e-02}$
BB11005	$1{,}23e{+}00_{5{,}5e-02}$	$8{,}46e{-}01_{9{,}2e-02}$	$1{,}21e{+}00_{9{,}3e-02}$
BB11006	$1{,}44e{+}00_{2{,}0e-01}$	$6{,}72e{-}01_{3{,}3e-01}$	$1{,}49e{+}00_{1{,}5e-01}$
BB11007	$1{,}44e{+}00_{2{,}0e-01}$	$8{,}49e{-}01_{1{,}8e-01}$	$1{,}47e{+}00_{1{,}1e-01}$
BB11008	$1{,}41e{+}00_{5{,}5e-02}$	$7{,}83e{-}01_{2{,}2e-01}$	$1{,}43e{+}00_{6{,}3e-02}$
BB11009	$1{,}71e{+}00_{5{,}0e-02}$	$4{,}10e{-}01_{1{,}7e-01}$	$1{,}65e{+}00_{7{,}3e-02}$
BB11010	$1{,}49e{+}00_{1{,}2e-01}$	$6{,}42e{-}01_{1{,}8e-01}$	$1{,}51e{+}00_{1{,}2e-01}$
BB11011	$1{,}44e{+}00_{2{,}5e-01}$	$6{,}28e{-}01_{1{,}6e-01}$	$1{,}51e{+}00_{1{,}0e-01}$
BB11012	$1{,}47e{+}00_{2{,}2e-01}$	$6{,}29e{-}01_{8{,}1e-02}$	$1{,}54e{+}00_{2{,}3e-01}$
BB11013	$1{,}46e{+}00_{1{,}5e-01}$	$6{,}64e{-}01_{1{,}8e-01}$	$1{,}45e{+}00_{5{,}7e-02}$
BB11014	$1{,}14e{+}00_{2{,}0e-01}$	$9{,}32e{-}01_{4{,}5e-02}$	$1{,}12e{+}00_{3{,}4e-01}$
BB11015	$1{,}57e{+}00_{8{,}9e-02}$	$5{,}69e{-}01_{1{,}1e-01}$	$1{,}54e{+}00_{1{,}1e-01}$
BB11016	$1{,}50e{+}00_{1{,}3e-01}$	$8{,}39e{-}01_{2{,}8e-01}$	$1{,}58e{+}00_{1{,}3e-01}$
BB11017	$1{,}51e{+}00_{6{,}9e-02}$	$5{,}74e{-}01_{2{,}3e-01}$	$1{,}57e{+}00_{1{,}0e-01}$
BB11018	$1{,}33e{+}00_{2{,}0e-01}$	$8{,}00e{-}01_{3{,}1e-01}$	$1{,}37e{+}00_{2{,}2e-01}$
BB11019	$1{,}43e{+}00_{1{,}6e-01}$	$9{,}27e{-}01_{9{,}1e-02}$	$1{,}40e{+}00_{1{,}8e-01}$
BB11020	$1{,}45e{+}00_{3{,}3e-01}$	$7{,}05e{-}01_{1{,}7e-01}$	$1{,}49e{+}00_{1{,}2e-01}$
BB11021	$1{,}62e{+}00_{1{,}2e-01}$	$7{,}37e{-}01_{1{,}3e-01}$	$1{,}60e{+}00_{1{,}5e-01}$
BB11022	$1{,}70e{+}00_{1{,}5e-01}$	$4{,}47e{-}01_{4{,}9e-01}$	$1{,}73e{+}00_{1{,}4e-01}$
BB11023	$1{,}33e{+}00_{7{,}8e-02}$	$1{,}03e{+}00_{1{,}3e-01}$	$1{,}35e{+}00_{1{,}6e-01}$
BB11024	$1{,}49e{+}00_{8{,}2e-02}$	$7{,}42e{-}01_{2{,}7e-01}$	$1{,}50e{+}00_{1{,}1e-01}$
BB11025	$1{,}66e{+}00_{1{,}0e-01}$	$4{,}99e{-}01_{3{,}2e-01}$	$1{,}67e{+}00_{1{,}1e-01}$
BB11026	$1{,}00e{+}00_{2{,}1e-01}$	$1{,}00e{+}00_{0{,}0e+00}$	$1{,}00e{+}00_{1{,}6e-01}$
BB11027	$1{,}60e{+}00_{5{,}7e-02}$	$8{,}32e{-}01_{1{,}5e-01}$	$1{,}62e{+}00_{8{,}1e-02}$
BB11028	$1{,}57e{+}00_{9{,}8e-02}$	$6{,}33e{-}01_{4{,}3e-01}$	$1{,}63e{+}00_{1{,}1e-01}$
BB11029	$1{,}59e{+}00_{1{,}4e-01}$	$5{,}20e{-}01_{7{,}6e-02}$	$1{,}57e{+}00_{7{,}9e-02}$
BB11030	$1{,}38e{+}00_{2{,}0e-01}$	$8{,}95e{-}01_{2{,}2e-01}$	$1{,}36e{+}00_{3{,}2e-01}$
BB11031	$1{,}48e{+}00_{2{,}0e-01}$	$8{,}67e{-}01_{2{,}0e-01}$	$1{,}43e{+}00_{1{,}6e-01}$
BB11032	$1{,}50e{+}00_{6{,}6e-02}$	$8{,}65e{-}01_{2{,}1e-01}$	$1{,}48e{+}00_{7{,}1e-02}$
BB11033	$1{,}02e{+}00_{3{,}4e-01}$	$6{,}61e{-}01_{4{,}5e-01}$	$1{,}27e{+}00_{2{,}1e-01}$
BB11034	$1{,}38e{+}00_{1{,}2e-01}$	$7{,}44e{-}01_{1{,}3e-01}$	$1{,}45e{+}00_{1{,}2e-01}$
BB11035	$1{,}62e{+}00_{1{,}0e-01}$	$4{,}68e{-}01_{2{,}1e-01}$	$1{,}62e{+}00_{7{,}0e-02}$
BB11036	$1{,}49e{+}00_{7{,}3e-02}$	$9{,}51e{-}01_{2{,}2e-01}$	$1{,}49e{+}00_{2{,}5e-01}$
BB11037	$1{,}48e{+}00_{9{,}5e-02}$	$5{,}28e{-}01_{1{,}5e-01}$	$1{,}45e{+}00_{1{,}4e-01}$
BB11038	$1{,}46e{+}00_{1{,}5e-01}$	$8{,}71e{-}01_{1{,}4e-01}$	$1{,}41e{+}00_{7{,}5e-02}$

se han aplicado los indicadores de calidad Épsilon y Spread. Según los resultados del indicador Épsilon (I_{E^+}) en la tabla II, existe una alta relación de competitividad entre NSGA-II y SPEA2, ambos generan frentes de pareto con un mejor nivel de convergencia para casi el mismo número de problemas (14 y 16 respectivamente), mas del doble de los problemas en los que MOCell logra superarlos (8 de 38). Finalmente los resultados del indicador Spread (I_{Δ}) en la tabla III indican que los frentes generados por el algoritmo MOCell tienen una mayor diversidad de soluciones para el 100 % de los problemas del grupo RV11 del benchmark.

V-B. Test estadístico de Wilcoxon Rank-Sum

Dado la alta competitividad que existe entre los tres algoritmos, para confirmar estos resultados, hemos realizado el test estadístico de Wilcoxon Rank-Sum para conocer si existen o no diferencias significativas entre ellos. Se consideran 7 problemas seleccionados del BAliBASE (BB11007, BB11013, BB11019, BB11031, BB11032, BB11033 y BB11038) y los resultados se detallan en las tablas IV, V y VI para los indicadores Hypervolumen (I_{HV}), Épsilon (I_{E^+}) y Spread (I_{Δ}) respectivamente.

Tabla IV: Resultado del Test de Wilcoxon rank-sum del indicador I_{HV}

	MOCell							SPEA2						
NSGAII	–	▲	▲	▲	▲	–	▲	–	–	–	▲	–	▽	▲
MOCell								–	–	–	–	▽	▽	–

Tabla V: Resultado del Test de Wilcoxon rank-sum del indicador I_{E^+}

	MOCell							SPEA2						
NSGAII	▲	▲	▲	▲	▲	▲	▲	–	–	–	▲	–	▽	▲
MOCell								▽	–	–	–	▽	▽	–

Tabla VI: Resultado del Test de Wilcoxon rank-sum del indicador I_{Δ}.

	MOCell							SPEA2						
NSGAII	▽	▽	▽	▽	▽	▽	▽	–	–	–	–	–	–	–
MOCell								▲	▲	▲	▲	▲	▲	▲

Los resultados de los indicadores I_{HV} y I_{E^+}, revelan diferencias estadísticamente significativas a favor del algoritmo NSGAII frentre al algoritmo MOCell, en la mayoría de los problemas seleccionados evaluando el Hipervolumen, y en todos los problemas seleccionados evaluando la convergencia de sus frentes. Además ambos análisis estadísticos indican que entre los resultados generados por los algoritmos NSGAII y SPEA2, no existe mayor número de diferencias significativas entre ellos; aunque para ciertos problemas, cada uno logra tener resultados significativos a su favor. Y finalmente, confirmando los resultados por el indicador I_{Δ} en la tabla III, se resaltan las diferencias signifitivas favorables de los resultados del algoritmo MOCell frente a NSGAII y SPEA2.

V-C. Frentes de Pareto aproximados

Los resultados de los indicadores de calidad multiobjetivo, son gráficamente confirmados por las figuras 1 y 2, en la que se ilustran los frentes de Pareto aproximados generados por los tres algoritmos junto a un frente de Pareto referencial generado en base a todas las ejecuciones independientes de los tres algoritmos, para los problemas BB11003, BB11010, BB11015, BB11021, BB11017, BB11019, BB11031 y BB11037 del BAliBASE.

Podemos observar que los cuatros frentes de Pareto de referencia de la figura 1 están en su mayoría conformados por las soluciones (alineamientos) de los frentes aproximados generados por el algoritmo NSGAII, y con pocas soluciones (alineamientos) de los algoritmos MOCell y SPEA2 en sus bordes. Para el caso de los problemas BB11031 y BB11037, cuyos resultados se ilustran en las figuras 1c y 1d, respectivamente, donde todo el frente de referencia está construido únicamente por el algoritmo NSGAII, en uno de los casos MOCell no brinda ninguna solución representativa en relación al frente de pareto de referencia, y en el otro, sucede lo mismo con SPEA2.

Así mismo en la figura 2 se muestran los frentes de Pareto de referencia y los frentes de Pareto de los tres algoritmos sobre los problemas BB11003, BB11010, BB11015 y BB11021, ilustrados en las figuras 2a, 2b, 2c y 2d, respectivamente. En estos cuatros problemas seleccionados podemos darnos cuenta la alta competitividad que existe entre los tres algoritmos en estudio, ya que los tres generan soluciones no-dominadas de referencia, pero que al igual que en la figura 1, las soluciones de los frentes de pareto aproximdos generados por NSGAII representan la mayoría de las soluciones del frente de Referencia en cada problema. Finalmente podemos resaltar que de manera similar al grupo de frentes de Pareto de la figura 1, hay problemas en que los algoritmos MOCell o SPEA2 no generan ninguna solución no-dominada frente a las soluciones generadas por el algoritmo NSGAII.

V-D. Resultados Biológicos

En esta subsección de resultados biológicos hemos agregado la métrica BaliScore [22] (**Q**), la cual evalúa la precisión que existe entre un alineamiento candidato frente a otro alineamiento de referencia, para nuestro estudio hemos usado como referencia los alineamientos generados para tal efecto por el benchmark BAliBASE (v3.0) para cada uno de los 38 problemas del grupo RV11.

Las puntuaciones de las métricas wSOP, TC y Q de los alineamientos obtenidos del experimento se muestran en la tabla VII, se detallan los mejores scores (máximo valores en 10 ejecuciones independientes) de los tres algoritmos resolviendo cada uno de los 38 problemas del grupo RV11 del BAliBASE.

Según estos resultados el algoritmo NSGAII logra obtener los mejores resultados de la métrica wSOP en el 42 % de los problemas del RV11 (16 de los 38 problemas) frente al 24 % y 34 % de MOCell y SPEA2 respectivamente; lo que indica que NSGAII presenta ser ligeramente el mejor de los tres algoritmos en estudio con respecto a este indicador de calidad biológico.

Alineando totalmente el mayor número de columnas de las secuencias (métrica TC), NSGAII presenta ser el mejor algoritmo en el 50 % de los problemas del RV11, a diferencia del 24 % y 26 % de los problemas resueltos de mejor manera por MOCell y SPEA2 respectivamente;

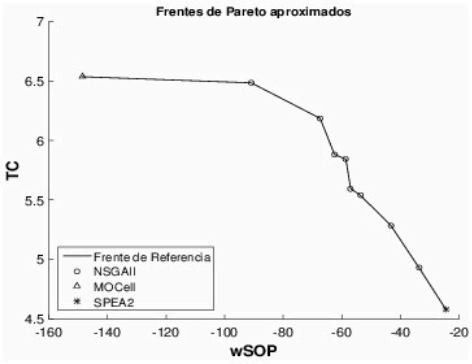
Y en el caso de la métrica Q existe un rendimiento muy competitivo y similar entre los tres algoritmos, NSGAII logra obtener los mejores puntajes en el 34 % de los problemas del RV11, MOCell en el 32 % y SPEA2 en el 34 %, lo que nos indica que los tres algoritmos generan alineamientos muy similares entre sí comparándolos con los alineamientos de referencia generados por BAliBASE.

VI. Conclusiones y Trabajo Futuro

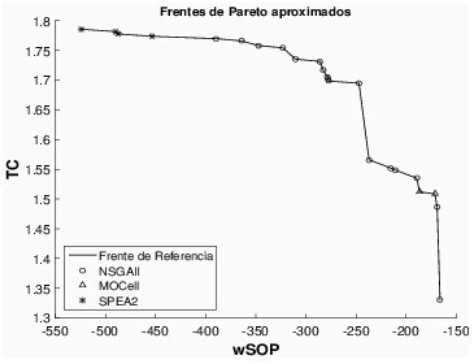
El objetivo del presente estudio fue optimizar el problema del alineamiento múltiple de secuencias (MSA) bajo un enfoque muitiobjetivo empleando metaheurísticas clásicas como NSGAII, MOCell y SPEA2. Se realizaron dos tipos de análisis en la experimientación, el primero bajo una perspectiva de optimizacion multiobjetivo, en el que se emplearon tres indicadores de calidad multiobjetivo, y el otro bajo la perspectiva biológica, en el que se definieron tres métricas para evaluar la calidad de los alineamientos wSOP, TC y BaliScore; este último para medir la presición de los resultados obtenidos frente a los alineamientos de referencia generados por el benchmark BAliBASE (v3.0). En base a los resultados obtenidos podemos concluir que, a pesar de la alta competitividad que existe entre los tres algoritmos en estudio, NSGAII y SPEA2 resultaron ser mejor que MOCell, pero que al competir entre ellos, NSGAII presentó generar un mejor rendimiento bajo ambas perspectivas (enfoque multi-objetivo y biológico). Sus frentes de Pareto aproximados presentan un mejor Hypervolumen (mayor convergencia y diversidad) que los generados por SPEA2, además sus puntajes de wSOP y la generación de zonas conservadas (TC) son mayores a los resultados de SPEA2. Esto nos permite considerar a NSGAII como un algoritmo base para implementar una futura propuesta mucho más competitiva e incluso definir otros objetivos a optimizar, como STRIKE que considera información estructural de las proteínas, para mejorar la calidad de los alineamientos.

Agradecimiento

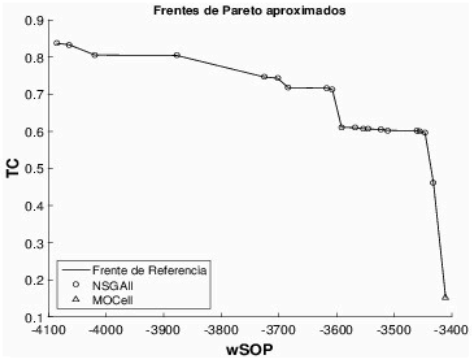
Este trabajo ha sido financiado por la Secretaría Nacional de Educación Superior Ciencia y Tecnología SENESCYT. Los autores agradecen el apoyo de las autoridades de la Universidad Técnica Estatal de Quevedo por incrementar la calidad de la producción Científica en la Unidad de Estudios a Distancia.



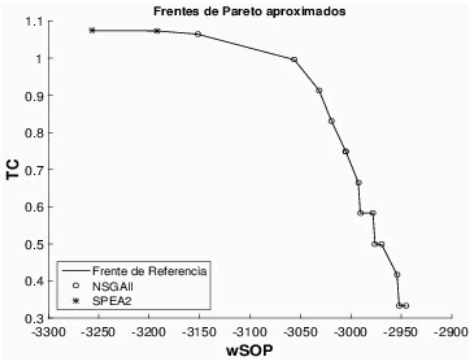
(a) Dataset BAliBASE BB11017



(b) Dataset BAliBASE BB11019

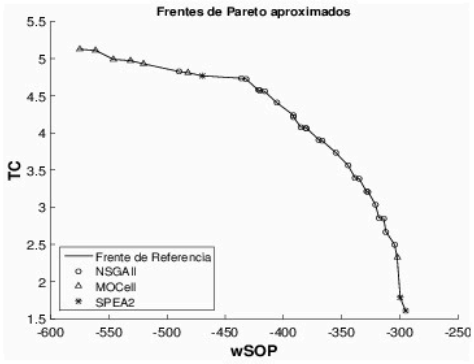


(c) Dataset BAliBASE BB11031

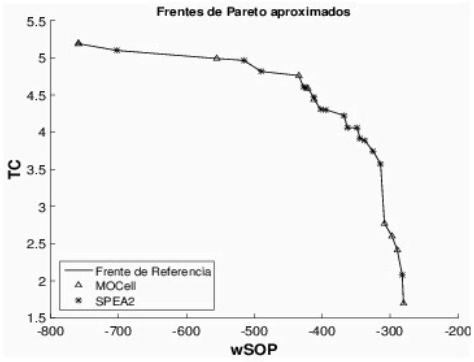


(d) Dataset BAliBASE BB11037

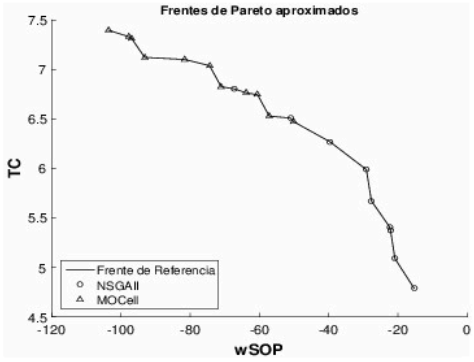
Figure 1: Frentes de Pareto aproximados de los algoritmos NSGAII, MOCell y SPEA2 a partir de todas sus 10 ejecuciones independientes y Frente de Pareto de Referencia generado en base a todas las ejecuciones independientes de los tres algoritmos, para los problemas del BAliBASE (v3.0): a) BB11017, b) BB11019, c) BB11031 y d) BB11037.



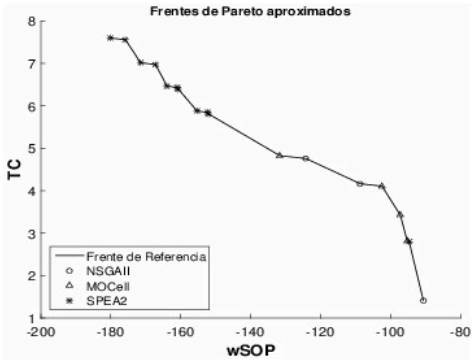
(a) Dataset BAliBASE BB11003



(b) Dataset BAliBASE BB11010



(c) Dataset BAliBASE BB11015



(d) Dataset BAliBASE BB11021

Figure 2: Frentes de Pareto aproximados de los algoritmos NSGAII, MOCell y SPEA2 a partir de todas sus 10 ejecuciones independientes y Frente de Pareto de Referencia generado en base a todas las ejecuciones independientes de los tres algoritmos, para los problemas del BAliBASE (v3.0): a) BB11003, b) BB11010, c) BB11015 y d) BB11021.

Tabla VII: Mejores Scores (suma ponderada de pares wSOP, columnas totalmente alineadas TC y BaliScore Q) generados por los algoritmos NSGAII, MOCell y SPEA2 en 10 ejecuciones independientes resolviendo todo el dataset de 38 problemas del grupo RV11 del BAliBASE (V3.0).

Instance	**NSGAII**			**MOCell**			**SPEA2**		
	wSOP	**TC**	**Q**	**wSOP**	**TC**	**Q**	**wSOP**	**TC**	**Q**
BB11001	**29.88**	5.88	**0.92**	28.66	6.48	0.92	29.88	**6.67**	0.92
BB11002	-803.23	**1.48**	0.49	-783.00	1.48	0.41	**-756.28**	1.48	**0.49**
BB11003	-304.43	4.83	0.67	-302.35	**5.13**	**0.69**	**-295.09**	4.77	0.68
BB11004	-399.73	3.49	0.60	**-389.04**	**3.65**	**0.60**	-410.03	3.59	0.59
BB11005	-2715.11	0.54	**0.50**	-2867.00	0.52	0.49	**-2695.12**	**0.66**	0.50
BB11006	**-824.19**	1.14	0.38	-866.19	1.11	0.39	-853.54	**1.15**	**0.42**
BB11007	**-96.25**	1.79	0.66	-107.69	1.72	0.68	-130.89	**1.96**	**0.68**
BB11008	-853.10	1.79	0.59	-850.69	**1.82**	**0.61**	**-849.63**	1.79	0.61
BB11009	**-474.27**	2.20	**0.37**	-475.31	**2.41**	0.35	-482.95	2.39	0.35
BB11010	-293.21	5.07	0.31	**-280.25**	**5.20**	0.30	-282.23	5.14	**0.32**
BB11011	-358.64	**2.54**	0.27	-368.39	2.28	**0.28**	**-355.22**	2.24	0.28
BB11012	165.03	**6.25**	0.88	162.79	6.10	**0.89**	**166.59**	6.25	0.88
BB11013	-179.86	**1.96**	0.11	-189.27	1.71	0.12	**-179.83**	1.85	**0.15**
BB11014	116.61	2.91	0.75	**139.12**	2.91	**0.75**	123.94	**3.06**	0.74
BB11015	**-15.32**	7.10	0.73	-24.22	**7.40**	0.74	-28.87	6.85	**0.75**
BB11016	-2321.15	0.97	**0.45**	**-2287.56**	**1.10**	0.43	-2305.29	1.01	0.43
BB11017	-32.79	6.48	0.73	-36.16	**6.54**	**0.75**	**-24.42**	6.48	0.73
BB11018	**-4500.18**	**1.05**	0.50	-4572.96	1.01	**0.52**	-4564.68	1.05	0.52
BB11019	**-166.27**	1.77	**0.65**	-171.50	1.72	0.63	-170.17	**1.79**	0.64
BB11020	**-288.92**	**2.47**	0.70	-310.87	2.46	0.69	-309.12	2.43	**0.72**
BB11021	**-90.63**	7.06	0.60	-91.67	7.56	**0.62**	-94.67	**7.60**	0.59
BB11022	-330.49	**3.81**	**0.12**	**-329.05**	3.81	0.12	-330.45	3.81	0.12
BB11023	-703.25	**2.22**	**0.45**	-681.06	2.10	0.40	**-676.98**	2.11	0.43
BB11024	-472.09	**2.91**	0.21	**-455.26**	2.72	0.19	-461.68	2.88	**0.22**
BB11025	**-122.97**	**4.92**	0.22	-126.78	4.92	0.18	-130.26	4.92	**0.23**
BB11026	-3992.68	**0.11**	**0.20**	**-3989.91**	0.11	0.19	-3989.97	0.11	0.17
BB11027	-1186.64	**1.14**	0.36	-1180.95	1.14	0.36	**-1180.70**	1.14	**0.37**
BB11028	**-606.22**	**0.44**	0.48	-613.32	0.44	**0.49**	-616.31	0.44	0.48
BB11029	-132.16	**7.89**	0.50	-134.09	7.33	0.50	**-132.13**	7.33	**0.51**
BB11030	-2111.00	0.42	**0.58**	**-2043.73**	0.42	0.57	-2092.90	**0.42**	0.58
BB11031	-3431.83	**0.84**	0.46	**-3410.62**	0.79	**0.46**	-3472.21	0.80	0.44
BB11032	-1339.50	1.18	**0.64**	-1359.47	**1.31**	0.63	**-1338.28**	1.15	0.61
BB11033	**-1093.50**	0.37	0.50	-1099.85	0.37	0.48	-1103.30	**0.39**	**0.51**
BB11034	-2044.50	**1.54**	0.44	-2081.76	1.50	0.42	**-2010.19**	1.49	**0.44**
BB11035	**-143.86**	**5.48**	0.54	-147.05	5.48	**0.54**	-147.64	5.44	0.54
BB11036	**-772.76**	1.68	**0.60**	-812.35	1.56	0.60	-783.83	**1.71**	0.59
BB11037	**-2945.67**	**1.08**	**0.48**	-2948.39	1.07	0.46	-2946.70	1.07	0.42
BB11038	**-1533.82**	**1.93**	**0.71**	-1607.61	1.92	0.70	-1558.03	1.91	0.70

REFERENCIAS

[1] J. Pei, "Multiple protein sequence alignment," *Current Opinion in Structural Biology*, vol. 18, no. 3, pp. 382 – 386, 2008, nucleic acids / Sequences and topology. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0959440X08000407

[2] S. Needleman and C. Wunsch, "A general method applicable to the search for similarities in the amino acid sequence of two proteins," *Journal of Molecular Biology*, vol. 48, no. 3, pp. 443 – 453, 1970. [Online]. Available: http://www.sciencedirect.com/science/article/pii/0022283670900574

[3] I. Elias, "Settling the intractability of multiple alignment," *Journal of Computational Biology*, vol. 13, no. 7, pp. 1323 – 1339, 2016.

[4] C. Kemena, J. Taly, J. Kleinjung, and C. Notredame, "Strike: evaluation of protein msas using a single 3d structure," *Bioinformatics*, vol. 27, no. 24, pp. 3385–3391, 2011.

[5] W. Soto and D. Becerra, "A multi-objective evolutionary algorithm for improving multiple sequence alignments," in *Advances in Bioinformatics and Computational Biology*, ser. Lecture Notes in Computer Science, S. Campos, Ed. Springer International Publishing, 2014, vol. 8826, pp. 73–82.

[6] B. Blackburne and S. Whelan, "Measuring the distance between multiple sequence alignments," *Bioinformatics*, vol. 28, no. 4, pp. 495–502, 2012. [Online]. Available: http://bioinformatics.oxfordjournals.org/content/28/4/495.abstract

[7] K. Deb, A. Pratap, S. Agarwal, and T. Meyarivan, "A fast and elitist multiobjective genetic algorithm: NSGA-II," *IEEE Transactions on Evolutionary Computation*, vol. 6, no. 2, pp. 182–197, 2002.

[8] E. Zitzler, M. Laumanns, and L. Thiele, "SPEA2: Improving the strength pareto evolutionary algorithm," in *EUROGEN 2001. Evolutionary Methods for Design, Optimization and Control with Applications to Industrial Problems*, K. Giannakoglou, D. Tsahalis, J. Periaux, P. Papailou, and T. Fogarty, Eds., Athens, Greece, 2002, pp. 95–100.

[9] A. Nebro, J. Durillo, F. Luna, B. Dorronsoro, and E. Alba, "Design issues in a multiobjective cellular genetic algorithm," in *Evolutionary Multi-Criterion Optimization. 4th International Conference, EMO 2007*, ser. Lecture Notes in Computer Science, S. Obayashi, K. Deb, C. Poloni, T. Hiroyasu, and T. Murata, Eds., vol. 4403. Springer, 2007, pp. 126–140.

[10] F. Ortuño, O. Valenzuela, F. Rojas, H. Pomares, J. Florido, J. Urquiza, and I. Rojas, "Optimizing multiple sequence alignments using a genetic algorithm based on three objectives: structural information, non-gaps percentage and totally conserved columns." *Bioinformatics (Oxford, England)*, vol. 29, no. 17, pp. 2112–21, Sep. 2013. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/23793754

[11] F. J. M. da Silva, J. M. S. Pérez, J. A. G. Pulido, and M. a. V. Rodríguez, "Parallel Niche Pareto AlineaGA–an evolutionary multiobjective approach on multiple sequence alignment." *Journal of integrative bioinformatics*, vol. 8, no. 3, p. 174, 2011. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/21926437

[12] M. Kaya, A. Sarhan, and R. Abdullah, "Multiple sequence alignment with affine gap by using multi-objective genetic algorithm." *Computer methods and programs in biomedicine*, vol. 114, no. 1, pp. 38–49, Apr. 2014. [Online]. Available: http://www.ncbi.nlm.nih.gov/pubmed/24534604

[13] F. P. M. Abbasi, L. Paquete, "Local search for multiobjective multiple sequence alignment," in *Bioinformatics and Biomedical Engineering*, ser. Lecture Notes in Computer Science, F. Ortuño and I. Rojas, Eds. Springer International Publishing, 2015, vol. 9044, pp. 175–182.

[14] M. Dayhoff, R. Schwartz, and B. B.C. Orcutt, "A model of evolutionary change in proteins," *In Atlas of Protein Sequences and Structure*, vol. 5, pp. 345–352, 1978.

[15] S. Henikoff and J. Henikoff, "Amino acid substitution matrices from protein blocks," *Proceedings of the National Academy of Sciences*, vol. 89, no. 22, pp. 10 915–10 919, 1992.

[16] A. Nebro, J. Durillo, F. Luna, B. Dorronsoro, and E. Alba, "Mocell: A cellular genetic algorithm for multiobjective optimization," *International Journal of Intelligent Systems*, vol. 24, no. 7, pp. 723 – 725, 2009.

[17] L. Bradstreet, *The hypervolume indicator for multi-objective optimisation: calculation and use*. University of Western Australia, 2011.

[18] E. Zitzler, L. Thiele, M. L. anb C.M. Fonseca, and V. da Fonseca, "Performance assessment ofmultiobjective optimizers: An analysis and review," *IEEE Trans. on Evolutionary Computation*, vol. 7, no. 2, pp. 117 – 132, 2003.

[19] G. Raghava, S. M. Searle, P. C. Audley, J. D. Barber, and G. J. Barton, "Oxbench: A benchmark for evaluation of protein multiple sequence alignment accuracy," *BMC Bioinformatics*, vol. 4, no. 1, pp. 1–23, 2003. [Online]. Available: http://dx.doi.org/10.1186/1471-2105-4-47

[20] P. I. W. de Bakker, A. Bateman, D. F. Burke, R. N. Miguel, K. Mizuguchi, J. Shi, H. Shirai, and T. L. Blundell, "Homstrad: adding sequence information to structure-based alignments of homologous protein families," *Bioinformatics*, vol. 17, no. 8, pp. 748–749, 2001. [Online]. Available: http://bioinformatics.oxfordjournals.org/content/17/8/748.abstract

[21] R. Edgar, "Muscle: multiple sequence alignment with high accuracy and high throughput," *Nucleic Acids Research*, vol. 32, no. 5, pp. 1792–1797, 2004. [Online]. Available: http://nar.oxfordjournals.org/content/32/5/1792.abstract

[22] J. Thompson, P. Koehl, and O. Poch, "Balibase 3.0: latest developments of the multiple sequence alignment benchmark," *Proteins*, vol. 61, pp. 127–136, 2005.

[23] H. Berman, J. Westbrook, Z. Feng, G. Gilliland, T. Bhat, H. Weissig, I. Shindyalov, and P. Bourne, "The protein data bank," *Nucleic Acids Research*, vol. 28, no. 1, pp. 235–242, 2000. [Online]. Available: http://nar.oxfordjournals.org/content/28/1/235

[24] A. Nebro, J. J. Durillo, and M. Vergne, "Redesigning the jmetal multi-objective optimization framework," in *Proceedings of the Companion Publication of the 2015 Annual Conference on Genetic and Evolutionary Computation*, ser. GECCO Companion '15. New York, NY, USA: ACM, 2015, pp. 1093–1100. [Online]. Available: http://doi.acm.org/10.1145/2739482.2768462

Cristian Zambrano-Vega Es estudiante de Doctorado en Ingeniería del Software e Inteligencia Artificial de la Universidad de Málaga. Su línea de investigación abarca las técnicas de optimización multiobjetivo aplicadas a la Inferencia Filogenética y al Alineamiento Múltiple de Secuencias. Docente en la carrera de Ingeniería en Sistemas de la Unidad de Estudios a Distancia de la Universidad Técnica Estatal de Quevedo - Ecuador. Email czambrano@uteq.edu.ec

Miriam Cárdenas-Zea Es Coordinadora de la Carrera de Ingeniería en Sistemas de la Unidad de Estudios a Distancia de la Universidad Técnica Estatal de Quevedo - Ecuador. Es estudiante de Doctorado en Ciencias Informáticas de la Universidad de Gramma - Cuba. Licenciada en Ciencias de la Educación en la Especialidad de Informática Educativa y Magister en Educación a Distancia y Abierta. Email mcardenas@uteq.edu.ec

Ricardo Aguirre-Pérez Es docente en la carrera de Ingeniería en Sistemas de la Unidad de Estudios a Distancia de la Universidad Técnica Estatal de Quevedo - Ecuador. Ingeniero en Sistemas e Informática, Magister en Educación a Distancia y Abierta y Especialista en Redes de Comunicación de Datos. Email gaguirre@uteq.edu.ec

Published by:

Escuela Politécnica Nacional
Facultad de Ingeniería de Sistemas
Departamento de Informática y Ciencias de la Computación
Ecuador

http://lajc.epn.edu.ec/
lajc@epn.edu.ec

May 2016