Expert Systems with Applications 40 (2013) 638–645

Contents lists available at SciVerse ScienceDirect

Expert Systems with Applications

journal homepage: www.elsevier.com/locate/eswa

Mining interests for user profiling in electronic conversations

Matias Nicoletti *, Silvia Schiaffino, Daniela Godoy

ISISTAN Research Institute, CONICET-UNICEN, Campus Universitario, Paraje Arroyo Seco (B7001BBO) Tandil, Argentina

ARTICLE INFO

Keywords:
Topic identification
Text mining
Semantic analysis
Encyclopedia knowledge
User profiling

ABSTRACT

The increasing amount of Web-based tasks is currently requiring personalization strategies to improve the user experience. However, building user profiles is a hard task, since users do not usually give explicit information about their interests. Therefore, interests must be mined implicitly from electronic sources, such as chat and discussion forums. In this work, we present a novel method for topic detection from online informal conversations. Our approach combines: (i) Wikipedia, an extensive source of knowledge, (ii) a concept association strategy, and (iii) a variety of text-mining techniques, such as POS tagging and named entities recognition. We performed a comparative evaluation procedure for searching the optimal combination of techniques, achieving encouraging results.

1. Introduction

The search for user interests has become a matter of concern in the personalization of information contents. Generally, people tend to talk about the topics in which they are interested in. In fact, electronic conversations are one of the richest sources for applying semantic analysis. The current development of social tools for digital interaction makes this sort of information available on the Web. Through an automatic text processing system, relevant topics could be mined from textual conversations in order to create user profiles with interest information. Personal intelligent assistance (Schiaffino & Amandi, 2009), Information filtering (Yang, Nie, Shen, Yu, & Kou, 2011), and item recommendation (Christensen & Schiaffino, 2011; Kahng, Lee, & Lee, 2011) are some examples of the extensive applicability of these user profiles.

Semantic analysis of text is known as a prominent subarea of the Natural Language Processing research line. Even though several works can be found in the literature, topic identification is still an immature research area. Most related works aim to identify topics from static and general-purpose documents. But in this research, we target dynamic text sources in order to extract updated interest information about the users. Static documents do not usually reflect the current user interest as a dynamic source does. Also, in this work we study the use of data cleansing techniques, since they play an important role in the analysis when working with noisy text sources.

A variety of strategies have been applied for detecting topics from text, like frequent term vectors (Bengel, Gauch, Mittur, & Vijayaraghavan, 2004), named entities recognition (Clifton, Cooley, & Rennie, 2004) and concept association (Coursey, Mihalcea, & Moen, 2009; Schonhofen, 2006; Tiun, Abdullah, & Kong, 2001). For inferring the semantic meaning of word and phrases, different sources of knowledge have been proposed: Open Directory Project (Bengel et al., 2004), Wordnet (Clifton et al., 2004; Tiun et al., 2001), Wikipedia (Coursey et al., 2009; Csomai & Mihalcea, 2008; Egozi, Markovitch, & Gabrilovich, 2011; Kliegr, 2010), among others. Also, there have been approaches that use classification-based methods for the task (Medelyan, Witten, & Milne, 2008). However, using a classifier to assign topics to documents is not a scalable solution when the knowledge source is extended to the size of Wikipedia. One of the most relevant works we analyzed is Wikify! (Csomai & Mihalcea, 2008) and its extension. *Wikify!* is an algorithm that links Wikipedia concepts to documents using a keyword detection approach based on n-grams from Wikipedia article titles. Then, the extension of *Wikify!* (Coursey & Mihalcea, 2009) used the linked concepts to discover related topics with a graph centrality algorithm. Despite non mentioned concepts could be detected, this strategy may generate false positives and decrease the method effectiveness.

In this context, we propose a novel approach called *TopText* (TOPic identification from informal TEXT). It consists of an unsupervised method for the detection of topics from noisy text sources. The main idea is to associate Wikipedia articles, considered as concepts of human knowledge, to text messages in order to infer the topics of a textual conversation. Two strategies were proposed for the association of concepts to user messages: (i) using the raw text of messages for a search in the concept dictionary, and (ii) identifying entities from messages previous to the search for concepts. At this point, we formulated our first hypothesis (HYPOTHESIS A): the entity-based approach has a better effectiveness than the raw-text-based approach.

* Corresponding author. Tel.: +54 249 4439682x28.
E-mail addresses: matias.nicoletti@isistan.unicen.edu.ar (M. Nicoletti), silvia.schiaffino@isistan.unicen.edu.ar (S. Schiaffino), daniela.godoy@isistan.unicen.edu.ar (D. Godoy).

0957-4174/$ - see front matter © 2012 Elsevier Ltd. All rights reserved.
http://dx.doi.org/10.1016/j.eswa.2012.07.075

We realized that most of the conversations and thread of discussions on the Web focus on a few areas of human knowledge. Moreover, this study targets the Software Engineering and Computer Science area, in order to build user profiles in software development teams. Therefore, our second hypothesis (HYPOTHESIS B) was formulated: if we tailor the method to a specific area of knowledge, the effectiveness of the method should be better than using a general purpose version of the same method. Through empirical evaluation, the different combinations of pre-processing techniques with concept association strategies were compared. For the experiments, we defined a metric called *relevance score*, related to the relevance of the set of concepts linked to user messages. Promising results were obtained, which allowed us to detect a suitable combination of techniques for the problem. Also, the experiments provided reasonably good evidence for supporting our hypothesis.

The rest of the article is organized as follows. Section 2 presents a detailed description of the proposed approach, considering the dictionary, and the two different strategies for tackling the problem of topic identification. Section 3 describes the domain-tailored version of the algorithm. Section 4 describes the evaluation procedure and discusses the results obtained. Section 5 summarizes the related works found in the literature. Finally, in Section 6 we present our conclusions and future work.

2. Proposed approach

In this work, we introduce *TopText*, an unsupervised method for topic identification. For our study, we defined a *topic* as a fraction of human knowledge with a certain level of abstraction. For example, the possible topics involved in the sentence *"We must try Google's new framework for web development"* are: *frameworks*, *Google*, *web development* (low-level) and *software development*, *software companies*, *systems design*, *computer sciences* (high-level), among others. Te goal of the technique is to build user profiles with these semantic units of information, which are more informative than term-based representations (Coursey et al., 2009).

The general schema of our method is presented in Fig. 1. The system inputs are mainly informal text logs from electronic conversations. In the first step of the process, the input is parsed in order to identify the involved users and their messages. As a result, messages are grouped by users and irrelevant content from the logs (e.g. log timestamps) is filtered out. Secondly, noisy texts are pre-processed. As it is expected, the text of chat logs is extremely dirty. Therefore, data cleansing techniques are used to prepare the user messages for future analysis.

The third step is the concept association. For this purpose, we use a semantic dictionary containing concepts of human knowledge. The result of this step is a set of concepts associated to each message. This information is expected to give a general idea of the semantic meaning of the messages.

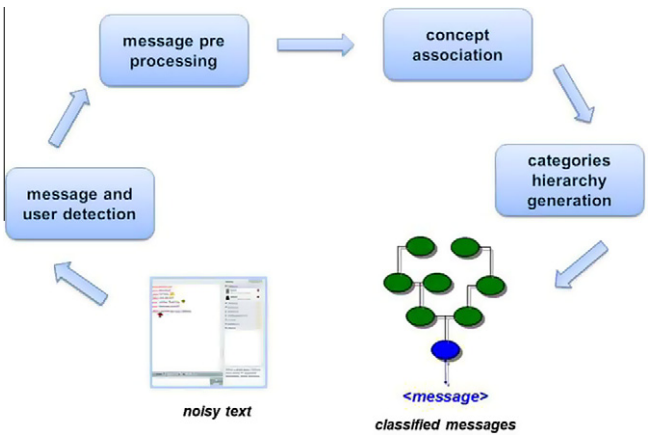

Fig. 1. Overview of our proposal.

Finally, a category hierarchy is built from each concept. The information about categories is also extracted from the dictionary. First, we connect each concept with its corresponding first level category. Next, the process is repeated for higher level categories. As a consequence, we are able to identify topics from text considering different levels of abstraction.

The final result of the process is a set of user profiles, each one containing (i) a ranking of the most relevant concepts, (ii) a ranking of the most relevant first level categories and (iii) a list of the most significant general categories of any level. Since this profile is based on topics that are regularly mentioned by the user, the information may be considered as the user interests or, at least, user-related concepts.

The rest of this section is organized as follows. Section 2.1 describes the dictionary used to perform the semantic analysis. Then, we propose two approaches based on the general schema in order to tackle the topic identification problem, which are described in Sections 2.2 and 2.3.

2.1. Wikipedia as the dictionary

One of the main sources of knowledge in the Web is Wikipedia,[1] which consists of a large set of articles (over 3M), multi-level categories and disambiguation information that describes a variety of concepts of human knowledge. Since Wikipedia contains an extremely high amount of unstructured information, performing any kind of analysis becomes a hard task. Therefore, we took advantage of DBpedia (Auer et al., 2008), which is a community project aiming at the extraction of structured information from Wikipedia. The DBpedia project allows users to download a complete snapshot of the encyclopedia in several parts, depending on their information needs. This feature allowed us to easily select only the information about article titles, extended abstracts, categories relations and term disambiguation.

For each article, we indexed with Lucene[2] the title and extended abstract in order to generate a *concept index*. Apache Lucene is an open source text search engine library that provides efficient indexing and searching functionality. We built two indexes for categories: one index relates articles with first level categories (*article-categories index*) and the other one relates first level categories with higher level categories (*category-categories index*). Also, a *disambiguation index* was developed to handle ambiguous concepts.

When using Lucene, text analyzers are commonly used tools for the pre-processing of text before indexing and searching. To evaluate the most suitable techniques, we defined four kinds of analyzers: (i) the standard*analyzer*, which executes the most common techniques like *stop-words* filtering, lower case conversion and URL detection, (ii) the stemming*analyzer*, which adds Porter's stemming algorithm (Porter, 1997) to the *standard* process, (iii) the *synonym analyzer*, which adds synonyms for each term using Wordnet to the *standard* analyzer, (iv) the *synonym*-stemming analyzer, which first adds synonyms and then, applies stemming to the *standard* process.

2.2. First approach: using the raw text of messages

Based on the general schema, we proposed a first approach for topic identification. The general structure of our method (Section 2) must be specified with concrete operations and techniques in each step. The first step remains unmodified, since the objective is to de-

[1] Wikipedia. http://www.wikipedia.org/. Accessed Feb-3-2012.
[2] Apache Lucene Project. http://lucene.apache.org/. Accessed Mar-20-2012.

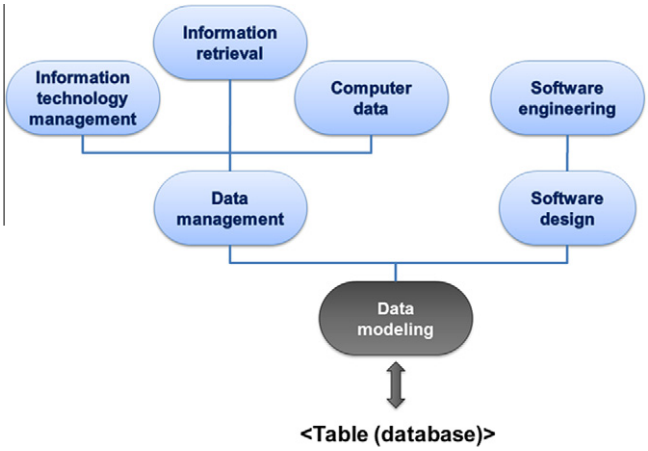

Fig. 2. A sample category hierarchy for the concept *Table* (*database*).

tect the users and messages from the logs. This step must be adapted to the specific type and representation of the input data. For example, it is not the same parsing the data from chat rooms as working with log from an instant messaging system.

For the second step, we defined the pre-processing strategy with the aim of handling the chat text issues. This involves (i) the deletion of references to users by their names, (ii) the filtering of invalid characters and (iii) the execution of analyzer operations, like *stop-words* filtering or the stemming algorithm.

As regards the third step, we defined a specific strategy for concept association. In this case, we used the concept index built from Wikipedia articles. The strategy is simple: the whole text of each message is used to perform a TF-IDF-based query in the index (Roelleke & Wang, 2008), similar to the one done in search engines. The first C items of the result sets are matched to each message (parameter C). Also, the position in the result set is used to define the relevance value of the concept to the user message.

Finally, the two category indexes are used for the categories hierarchy generation. Initially, we associated first level categories to concepts, and then, we established relations with the higher level categories in order to build the hierarchical structure. Since categories tend to become too general (e.g. 'living people') and the computing time grows exponentially, we decided to limit the hierarchy tree depth to three levels. An example of this process can be found in Fig. 2.

2.3. Second approach: identifying entities from messages

We proposed a second approach based on the detection of relevant entities from the messages. This strategy is supposed to be more effective than the first approach, as HYPOTHESIS A indicates. We made this assumption based on the fact that filtering the entities from the messages eliminates the noise, leaving only the important terms and increasing the probability of associating a correct concept. The new process presents some differences with the first one, as it is shown in Fig. 3. Also, we used two typical text mining tools: a named entities recognizer and a POS (Part-Of-Speech) tagger, both provided by the Stanford NLP Group.[3]

A named entities recognizer (NER) typically uses a classification-based approach to detect named entities, like persons, institutions, locations or any kind of proper nouns. A POS tagger is a tool that automatically assigns a grammatical label to every word in a sentence. In particular, we are interested in the analysis of nouns and their modifiers, like adjectives or adverbs (adverbs are actually adjective modifiers). The procedure of entities detection is divided in three steps: (i) identification of named entities using the NER, (ii) POS tagging each message in order to identify the nouns and their modifiers, and (iii) filtering of duplicated entities from the two first steps. As a result, a set of entities is associated with each message. For example, if we consider the message *"We must try Google's new framework for web development"*, we have the entities *"Google"* (identified by the NER), *"new framework"* and *"web development"*(by the POS tagger). In contrast to the first approach, the entities names are used as the search parameter in the concepts index. Consequently, we have a higher number of concepts in each profile.

We identified some inefficiencies in the dictionary. Since a significant number of articles in Wikipedia are related to ART, in particular to MOVIES and MUSIC, some of the titles are similar to common expressions used by people in conversations. *That is a good idea* or *it is OK* are examples of this situation, which we considered *irrelevant messages*. We noticed that art topics are frequently linked with a user profile, although the user is not referring to them. Therefore, an optimization of the indexes was performed in order to filter out this kind of information, which is not relevant to our domain of study.

Additionally, we realized that most irrelevant messages do not provide important information for topic identification. Thus, the pre-processing step was modified in order to add a filtering process of messages containing less than W words (parameter W). Although this is a simple approach that must be improved with a more sophisticated method, it is useful to handle the irrelevant messages issue.

Eventually, ambiguous concepts are linked to entities. For these cases, DBpedia provides a useful disambiguation data base extracted from Wikipedia articles. A concept is considered ambiguous if it is present in the disambiguation index. We used an adapted version of Michael Lesk's algorithm (Lesk, 1986), which considers a windows of the N nearest concepts to the ambiguous one (parameter N). Thus, the description of each disambiguation concept was compared to the nearest concepts descriptions using the cosine text comparison function. Finally, the most related concept replaced the ambiguous one.

3. Tailoring the domain of study

The use of a huge knowledge data base is a desirable feature of our method, since its semantic capability is incremented. However, in most of the electronic conversation methods (e.g., chat rooms, instant messaging, forum) each thread of discussion is usually limited to a certain area of knowledge, or only to a certain group of general topics. Therefore, we established HYPOTHESIS B: by tailoring the semantic database to a specific area of knowledge, we should be able to achieve better results than using a general purpose database. In particular, it would be useful to consider only concepts related to Software Engineering and Computer Sciences. Then, the method could be applied in software development environments to generate profiles of the team members for personalization strategies.

In order to implement the modifications, non-related concepts and categories were removed from the database. However, the general structure of the algorithm remained unmodified. The changes only affected the knowledge database, and required a filtering process on the set of indexes described in Section 2.1. Because of the large sizes of the indexes, we discarded the use of a manual procedure to adapt the database. Instead, we carefully designed a top-down algorithm for filtering the non-related concepts and categories.

[3] The Stanford NLP Group. http://nlp.stanford.edu/. Accessed Nov-11-2011.

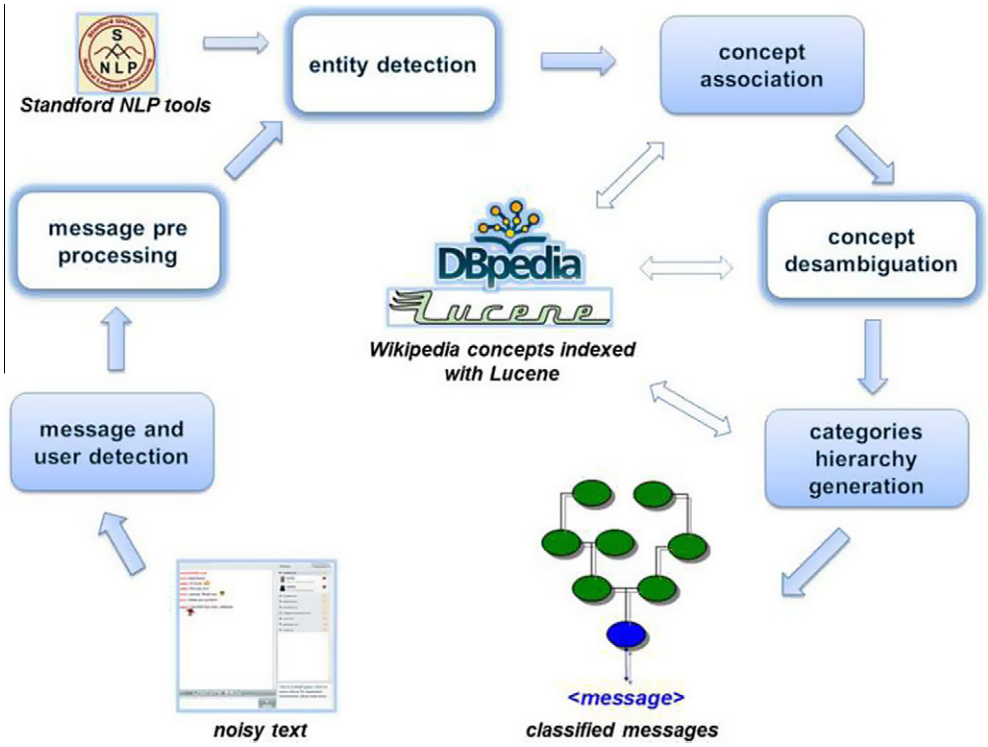

Fig. 3. Overview of the entity-based approach (modified components in white).

First, we selected 11 software-related top categories from Wikipedia: *Computing*, *Software engineering*, *Systems engineering*, *Information technology*, *Information science*, *Internet*, *Robotics*, *Systems architecture*, *Software architecture*, *Software design*, *Web design*. Second, we processed the *category-categories index*with an incremental strategy, filtering all the categories that were not related to any of the central topics. From 1,000,000 entries, only 11,500 remained. Third, we processed the *article-categories index* by leaving only the relations that refers to a category from the new *category-categories index*. The entries in this index decreased from 10,000,000 to 250,000. Fourth, a similar process was performed on the concept index, by leaving only the concepts that have a relation to any of the new *article-categories index*. On the new *concept index*, only 137,000 entries remained from the original 3,000,000.

4. Experimental evaluation

The empirical evaluation of the proposed approaches was divided in three stages. In Section 4.1, we describe the test procedure used for trying to determine the most suitable Lucene analyzer according to the domain characteristics. In Section 4.2, we compare the effectiveness between the first and the second approach. Finally, in Section 4.3 we carry out a comparison similar to one done in Section 4.2, but between the general purpose version and the specific purpose version of the method.

4.1. Evaluation of pre-processing analyzers

In the first step of the test, we selected a chat log from the Society of Genealogist[4] as the sample input data. The log contains 323 messages from 17 different users, in which topics related to overseas resources in Australia are discussed. Secondly, we executed an algorithm implementing the first approach on the input. In fact, we ran the Algorithm 4 times, using in each execution one of the analyzers defined in Section 2.1. As a result, we obtained 4 sets of user profiles: each set produced with one different analyzer. As regards the parameters of the algorithm, we use $C = 3$ as the number of concepts that are linked to a user message.

Next, we manually assigned one relevance score to each message, which represents the level of relation between the 3 associated concepts and the message. The tagging process was performed by mutual agreement between the authors of the paper. The score can have 3 possible values: [0], if none of the 3 concepts is relevant to the message; [0.5], if at least one concept is moderately relevant to the message; [1], if at least one concept is completely related to the message. Also, for each message we recorded the position of the most related concept, which is a measure commonly known as *hit position*.It must be noticed that when the relevance score is 0, the hit position is not defined.

Since the main goal was to evaluate and compare the effectiveness of each variant, we defined 4 metrics based on the data collected from the test. First, the *relevance score TOP 1* is the average of the relevance scores for each message, but just considering the first associated concept. Similarly, *relevance score TOP 2* is the average of the relevance scores, but considering the 2 first concepts. In the *relevance score TOP 3,* the 3 associated concepts are considered. Finally, we defined *AVG hit position*, which is the average of the hit positions for each message. The results of the test case execution for each analyzer are summarized in Figs. 4 and 5.

The results show that both *synonym* and *syn-stem* analyzers have a considerable lower relevance scores than the others. Since both analyzers use a synonym-based technique, we may assume that the use of synonyms increases the total number of concepts that are associated, but reduces the quality of the first ones. Because the objective of the system is to associate at least one relevant concept, these analyzers seem to be not suitable for topic identification. On the other hand, *standard* analyzer has the highest *TOP 1* and *TOP 2* relevance scores, as well as its *AVG hit position* is the closest to 1. Therefore, relevant concepts are frequently detected in the first 2 positions. However, the analyzer with the

[4] Society of Genealogists: Overseas in Australasia. http://www.sog.org.uk/prc/australasia.shtml. Accessed Dec-20-2011.

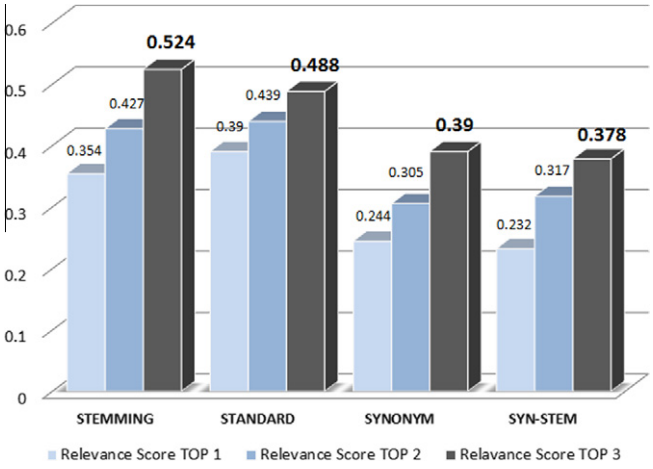

Fig. 4. Comparison of the relevance scores (first-approach-based method).

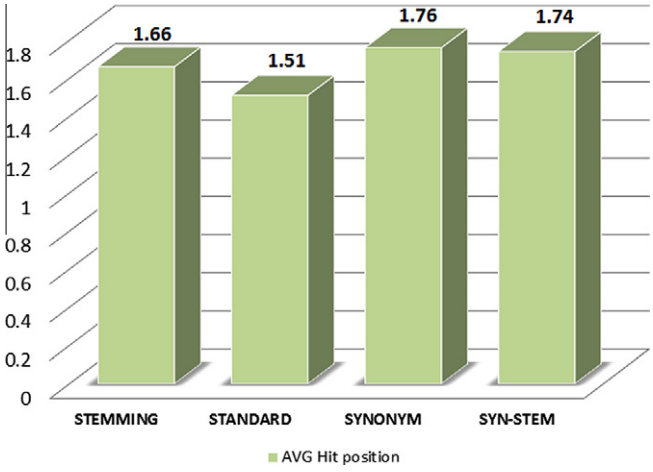

Fig. 5. Comparison of the AVG hit positions (first-approach-based method).

highest *relevance score TOP 3* value is *stemming*, which exhibits the best general effectiveness. Thus, we decided to continue the tests with these two analyzers, discarding the synonym-based ones.

4.2. Comparison of both approaches

The second stage of the evaluation has the objective of comparing the first and the second approach. Therefore, we designed a new test that uses an implementation of the entity-based method. The procedure is similar to the first evaluation, as it has the same general structure, the same input and uses the same metrics. The parameters of the entity-based algorithm were $C = 3$ (number of concepts per message), $W = 4$ (number of words to classify a message as irrelevant) and $N = 2$ (disambiguation window size).

The main difference with the first algorithm is that both relevance score and hit position were not assigned to each message. Instead, we manually assigned one relevance score and one hit position for each entity detected in each message (Fig. 6), called *entity relevance score* and *entity hit position*. Then, we computed the *message relevance score* as the average of the entity relevance scores for each message, and the *message hit position* is the average of the entity hit positions. Finally, *relevance score TOP X* metrics were calculated in the same way as the first test, but using the *message relevance scores*. Similarly, the *AVG hit position* is the average of the *message hit positions*. In Fig. 7 we present the results only for the *standard* and *stemming* analyzers, which have previously shown the best results.

In order to clarify how the entity-based method works, we present some examples of messages from the test data set, detected entities and linked concepts. Example 1: for the message *"The Old Bailey Trials are interesting and easily available from the Mitchell Library in NSW."*, the system identified the entities: *Mitchell Library* (concepts: Mitchell Library, David Scott Mitchell, Sydney B. Mitchell), *NSW* (concepts: Baseball NSW, Football NSW, NSW Volunteer of the Year) and *Old Bailey Trials* (concepts: Old Bailey, Trial of the century, F. Lee Bailey). Example 2: for the message "*The International Genealogical Index IGI of marriages, christenings & Births is an invaluable reference source for all genealogists*", the system identified the entities: *Births* (concepts: Men at Birth, Birth, Place of birth), *International Genealogical Index IGI*(concepts: IGI Global, International Genealogical Index, Internati), *genealogists* (concepts: The Master Genealogist, Society of Genealogists, Board for Certification of Genealogists) and *marriages*(concepts: Marriage problem, Group marriage, So This Is Marriage?).

The second experiment shows that, although the *AVG hit position* is still slightly lower, the *stemming* analyzer has higher relevance scores than *standard*. Additionally, the entity-based method combined with the *stemming* analyzer achieves the highest *relevance score TOP 3* of the whole evaluation procedure, with 0.656. This evaluation experience show us that the second approach seems to achieve higher *relevance scores* than the first approach. Particularly, for the *relevance score TOP 3* the increase was 0.132 for *stemming* and 0.085 for *standard*. From these results, we believe that HYPOTHESIS A could be confirmed with more empirical evaluations.

4.3. General purpose vs. specific purpose method

The last stage of the evaluation procedure has the objective of verifying if the tailored version of the algorithm (*software* version) exhibits better results than the original version (*general* version). For this purpose, we used a specific data set where only topics related to Software Engineering were discussed. The source of data was Stack Overflow,[5] which is a popular web forum of Software Development topics. We decided to use data logs from forums because of the availability of topics from an specific domain (Software Development in this case) and its similarity with chat data logs.

The evaluation procedure was executed as follows. We selected the 3 users who have made the richest and the majority of the contributions in several threads of discussion about Software Architecture. This means that the set of messages for each user may not necessarily belong to the same thread. Next, we ran the *software* version using the forum messages as the input. The evaluation criteria was exactly the same used in Section 4.2. The results for this test are shown in Table 1, where for each user we describe the amount of messages, amount of entities detected, the *AVG relevance score TOP X* metrics and the *AVG hit position*. We can infer that the results obtained are consistent, since the measures are strongly similar for each user.

The following step was to contrast these results with the ones of the second-approach-based method. Therefore, we computed the average measures for all the current users. As it is reflected in Fig. 8, the *software* version achieved better results in all the metrics considered. In particular, the *software* version has an increase of 0.054 *in relevance score TOP 3*, concluding with a final value of 0.71. These results, although preliminary, presents significant evidence for the confirmation of the HYPOTHESIS B.

[5] Stack Overflow web forum. http://stackoverflow.com/. Accessed Mar-9-2012.

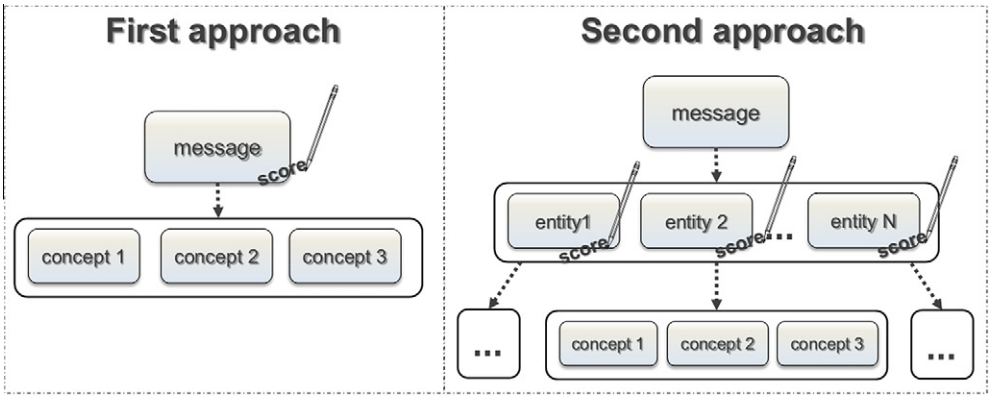

Fig. 6. Difference between both approaches (regarding to tagging).

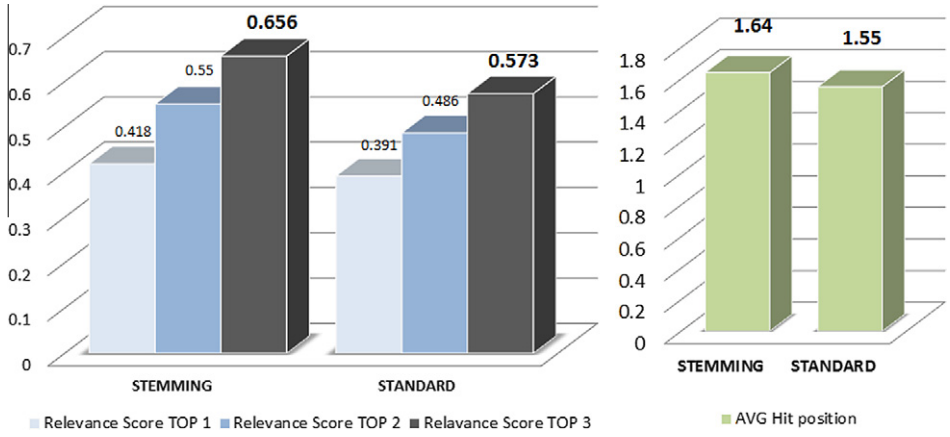

Fig. 7. Comparison of relevance scores and AVG hit positions (second-approach-based method).

Table 1
Results for the specific-purpose method evaluation.

User	No. of messages	No. of entities	Avg R-Score TOP 1	Avg R-Score TOP 2	Avg R-Score TOP 3	AVG hit position
User-1	56	130	0.5	0.63	0.72	1.49
User-2	35	78	0.53	0.61	0.71	1.44
User-3	48	112	0.48	0.58	0.7	1.53

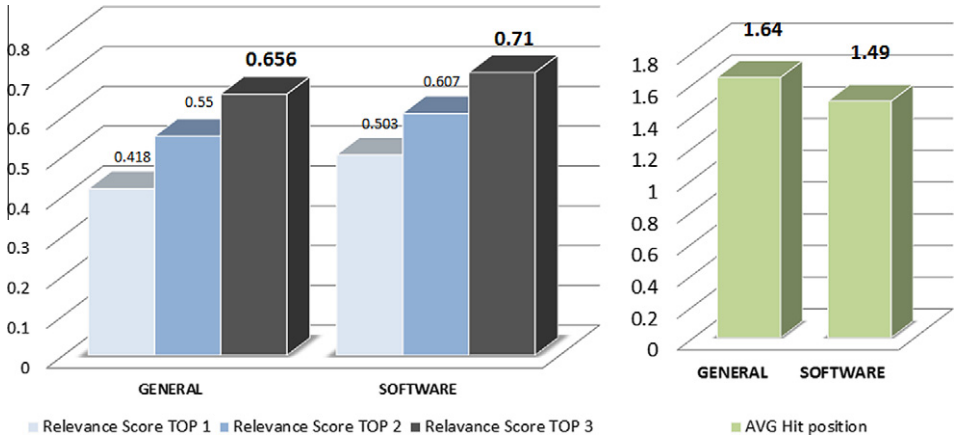

Fig. 8. Comparison of relevance scores and AVG hit positions (software vs.general version).

For this final stage, we decided to use a data set with chat logs from a forum, even though the evaluation of the general approach was made with chat room data. The reason for this is that there is no available and public data logs from chat rooms or instant messaging system about Software-related topics. Stack Overflow's discussion threads are in essence the same as a dialogue among multiple users but in an asynchronous way. Despite these minor differences, we believe we were able to make an accurate comparison.

5. Related work

We have detected certain common aspects among most of the works related to our research. As regards the general methodology, we found (i) approaches based on frequent term vectors, commonly known as *bag-of-words* (BOW) (Bengel et al., 2004), (ii) approaches that only use the detection of named entities (Clifton et al., 2004), (iii) techniques using concepts rather than just terms (Coursey et al., 2009; Schonhofen, 2006; Tiun et al., 2001). With respect to the general knowledge source, Wikipedia has become the most popular alternative (Coursey et al., 2009; Csomai & Mihalcea, 2008; Egozi et al., 2011; Kliegr, 2010). This rich encyclopedia has been used as a concepts and/or categories dictionary for semantic analysis. Other alternatives used for this purpose are the manually created Yahoo! directory[6] (Tiun et al., 2001), Open Directory Project[7](Bengel et al., 2004) or Wordnet (considering the hypernyms and hyponyms relations) (Clifton et al., 2004; Fellbaum, 1998; Tiun et al., 2001).

To the best of our knowledge, there are no previous researches on topic identification in noisy texts (e.g. chat logs) using Wikipedia as the general knowledge source. The most similar work is *Wikify!* (Csomai & Mihalcea, 2008), in which Wikipedia concepts are associated to sets of documents using a keyword detection algorithm. Nevertheless, some differences are identified: (i) the keyword extraction algorithm is based on the ranking of possible n-grams extracted from the dictionary, while our similar entity-based approach uses a combination of tools like a POS tagger and a named entities detector; (ii) just the Wikipedia article titles are considered, whereas our method also considers the extended abstracts; (iii) it is a general purpose document approach, while our approach aims for informal text from electronic conversations.

Unfortunately we were not able to compare our results with the ones of *Wikify!*. In that research, the authors performed a Turing-like test to determine weather a person could distinguish between a manual tagging (made by Wikipedia users) and an automatic tagging (made by the system). But this kind of test is not useful to quantify how accurate is the tagging (or concept association) as we did in this work. We believe that the comparison must be made with a quantitative method, in order to allow future researchers to compare their approaches to topic identification.

Another closely related study is presented in Coursey et al. (2009). Initially, *Wikify!* is used to detect concepts and then, a Wikipedia-graph centrality algorithm is applied to discover related topics. An advantage of this method is that non mentioned topics could be detected. However, this strategy may reflect negatively on the accuracy as irrelevant concepts may be associated, depending on the quality of the links between the topics in the graph. In Syed, Finin, and Joshi (2008) an approach with the same objective as ours is introduced. In this case, the whole text of the input documents is matched with Wikipedia articles texts using the cosine similarity function. A similar situation is reported in the study in Schonhofen (2006), where the presented approach uses the complete text of the input document and only considers the Wikipedia articles titles in the dictionary.

The disadvantage of using keyword-based methods, as the approaches mentioned above, is that the context (and even the actual meaning) of each term is lost. These method works fine for concepts that are described with one word but, in general, two or more words are needed to describe concepts of human knowledge. Therefore, by identifying entities we are able to model these kind of concepts, augmenting the semantic power of the extraction method. In addition, in this work we introduced another novel idea, which is the use of the extended abstracts in the matching of Wikipedia concepts, while all the related works just uses the title of the articles. In this way, we are able to associate not only concepts with the same name, but also strongly related concepts that are mentioned in articles abstracts.

In contrast with our unsupervised approach, the technique presented in Medelyan et al. (2008) requires previously annotated data to infer the topics from an input document. Similarly, Gabrilovich and Markovitch (2006) introduced a text classification system that calculates the most relevant Wikipedia concepts to a given input document. The classification approach works fine when only a limited number of topic and/or categories are considered. However, serious difficulties will be met if all the possible topics in Wikipedia are used. Also, in Clifton et al. (2004) clustering is performed over detected named entities in order to find representative topics in documents. In this case, Wordnet is used to generate a small set of keywords in order to improve the entity recognition. Nevertheless, this approach is only tested on a limited knowledge source that is considerable smaller than Wikipedia.

6. Conclusions and future work

In this paper we presented a novel technique for automatic topic identification from noisy text. Initially, the system aims for text logs from chat rooms and forums, but it could be easily extended to instant messaging, micro-blogging, or any kind of social media. Our approach takes advantage of some of text mining technologies, like POS tagging and named entities recognition, as well as it exploits the semantic power of the current hugest knowledge source in the Web, Wikipedia. A comparative evaluation of potential text analyzers was carried out. The results show that the appropriate analyzer for the task was *stemming*. We could also conclude that the use of strategies like synonyms, increase the number of associated concepts, which is related to the recall metric used in Information Retrieval. Indeed, recall was not an important factor to be targeted in this study.

In this study, two different approaches for topic identification were proposed. By an empirical evaluation, we found that the entity-based alternative is potentially superior to the one using the complete texts of messages (HYPOTHESIS A). In fact, the method combining the*stemming* analyzer with the entity-based approach has a general relevance score of 0.656. This means that in the 65.6% of the times, the system associates relevant concepts to messages from an extensive dictionary of about 3M possibilities. However, the effectiveness of the method seemed to be improved even more by tailoring the knowledge database to Software Engineering topics. The *software* version of the method achieved a general relevance score of 0.71, reflecting an improvement of 0.054 on the *original* version (HYPOTHESIS B). Although we cannot strongly confirm both hypotheses from these experiences and further testing is needed, we realize that the results are reasonably good to support both of them.

Since encouraging results were obtained, we propose future work to continue with this research. On the one hand, we are plan-

[6] Yahoo! Directory. http://dir.yahoo.com/. Accessed Nov-10-2011.
[7] Open Directory Project. http://www.dmoz.org/. Accessed Dec-02-2011.

ning to perform further testing using different and bigger data sets, specially with Software-related conversation logs. On the other hand, in order to improve the general effectiveness of the algorithm, we could evaluate the use of techniques for handling (i) writing mistakes (specially mistakes that represent valid words), (ii) not relevant messages (replacing the current strategy for a more sophisticated one), and (iii) abbreviations (for instance, considering if an abbreviation refers to a term already used in the context).

Acknowledgments

This work has been partially supported by ANPCyT (Argentina) through PICT Project 2010 No. 2247 and PICT Project 2011 No. 0366, and by CONICET (Argentina) through PIP Project No. 114-200901-00381.

References

Auer, S., Bizer, C., Kobilarov, G., Lehmann, J., Cyganiak, R., & Ives, Z. (2008). Dbpedia: A nucleus for a web of open data. In *Proceedings of the sixth international semantic web conference (ISWC). Lecture notes in computer science* (Vol. 4825, pp. 722–735). Springer.

Bengel, J., Gauch, S., Mittur, E., & Vijayaraghavan, R. (2004). Chattrack: Chat room topic detection using classification. In H. Chen, R. Moore, D. D. Zeng, & J. Leavitt (Eds.), *Intelligence and security informatics. Lecture notes in computer science* (Vol. 3073, pp. 266–277). Berlin/Heidelberg: Springer.

Christensen, I. A., & Schiaffino, S. (2011). Entertainment recommender systems for group of users. *Expert Systems with Applications, 38*(11), 14127–14135.

Clifton, C., Cooley, R., & Rennie, J. (2004). Topcat: Data mining for topic identification in a text corpus. *IEEE Transactions on Knowledge and Data Engineering, 16*, 949–964.

Coursey, K., & Mihalcea, R. (2009). Topic identification using wikipedia graph centrality. In *Proceedings of human language technologies: The 2009 annual conference of the north american chapter of the association for computational linguistics, Companion Volume: Short Papers. NAACL-Short '09. Association for Computational Linguistics* (pp. 117–120). Stroudsburg, PA, USA.

Coursey, K., Mihalcea, R., & Moen, W. (2009). Using encyclopedic knowledge for automatic topic identification. In *Proceedings of the 13th conference on computational natural language learning. CoNLL '09. Association for computational linguistics* (pp. 210–218). Stroudsburg, PA, USA.

Csomai, A., & Mihalcea, R. (2008). Linking documents to encyclopedic knowledge. *IEEE Intelligent Systems, 23*, 34–41.

Egozi, O., Markovitch, S., & Gabrilovich, E. (2011). Concept-based information retrieval using explicit semantic analysis. *ACM Transactions on Information Systems, 29*, 8:1–8:34.

Fellbaum, C. D. (1998). *WordNet: An electronic lexical database*. Cambridge, MA: MIT Press.

Gabrilovich, E., & Markovitch, S. (2006). Overcoming the brittleness bottleneck using wikipedia: Enhancing text categorization with encyclopedic knowledge. *Proceedings of the 21st national conference on artificial intelligence* (Vol. 2, pp. 1301–1306). AAAI Press.

Kahng, M., Lee, S., & Lee, S.-g. (2011). Ranking in context-aware recommender systems. In *Proceedings of the 20th international conference companion on world wide web. WWW '11* (pp. 65–66). New York, NY, USA: ACM.

Kliegr, T. (2010). Entity classification by bag of wikipedia articles. In *Proceedings of the third workshop on Ph.D. students in information and knowledge management. PIKM '10* (pp. 67–74). New York, NY, USA: ACM.

Lesk, M. (1986). Automatic sense disambiguation using machine readable dictionaries: How to tell a pine cone from an ice cream cone. In *Proceedings of the fifth annual international conference on systems documentation. SIGDOC '86* (pp. 24–26). New York, NY, USA: ACM.

Medelyan, O., Witten, I. H., & Milne, D. (2008). *Topic indexing with wikipedia* (Vol. 1). AAAI Press, pp. 19–24.

Porter, M. F. (1997). *An algorithm for suffix stripping*. San Francisco, CA, USA: Morgan Kaufmann Publishers Inc., pp. 313–316.

Roelleke, T., & Wang, J. (2008). Tf-idf uncovered: A study of theories and probabilities. In *Proceedings of the 31st annual international ACM SIGIR conference on Research and development in information retrieval. SIGIR '08* (pp. 435–442). New York, NY, USA: ACM.

Schiaffino, S., & Amandi, A. (2009). Intelligent user profiling. In M. Bramer (Ed.), *Artificial intelligence* (pp. 193–216). Berlin, Heidelberg: Springer-Verlag. Ch. Intelligent user profiling.

Schonhofen, P. (2006). Identifying document topics using the wikipedia category network. In *Proceedings of the 2006 IEEE/WIC/ACM international conference on web intelligence. WI '06* (pp. 456–462). Washington, DC, USA: IEEE Computer Society.

Syed, Z. S., Finin, T., & Joshi, A. (2008). Wikipedia as an ontology for describing documents. In E. Adar, M. Hurst, T. Finin, N. S. Glance, N. Nicolov, & B. L. Tseng (Eds.), *ICWSM*. The AAAI Press.

Tiun, S., Abdullah, R., & Kong, T. E. (2001). Automatic topic identification using ontology hierarchy. In *Proceedings of the second international conference on computational linguistics and intelligent text processing. CICLing '01* (pp. 444–453). London, UK: Springer-Verlag.

Yang, D., Nie, T., Shen, D., Yu, G., & Kou, Y. (2011). Personalized web search with user geographic and temporal preferences. In *Proceedings of the 13th Asia-Pacific web conference on Web technologies and applications. APWeb'11* (pp. 95–106). Berlin, Heidelberg: Springer-Verlag.