Obtaining Knowledge from the Web using Fusion and Summarization Techniques

Sandra Escudero, Angel L. Garrido, Sergio Ilarri
IIS Department
University of Zaragoza
Zaragoza, Spain
Email: {***sandra.escudero, garrido, silarri***}***@unizar.es***

***Abstract*—Nowadays, information and knowledge are fundamental in our society. This has induced an information overload problem in the Internet. For this reason, we propose to create an automatic system to retrieve, select, and extract information from the Web whose methodology is based on fusion techniques. The system, called Diana, facilitates and improves the identification of interesting contents, and it allowsto extract the most relevant information about a certain topic from the Web as a summary. To do this, we have developed algorithms that use semantic tools, Natural Language Processing (NLP) techniques, statistics, a generic gazetteer, and fusion methods. The development of the system is undergoing, but the preliminary results that we have obtained so far are very promising and show the interest of our proposal.**

***Keywords*-Summaries; fusion; text mining; NLP.**

I. Introduction

Currently, the number of web pages keeps growing at impressive rates. The explosion of written information generated through the Internet, such as blogs, social networks, or news, leads to the availability of a huge amount of text information. This makes it difficult to find what is really needed, since most of this information is hidden, messy, and unsorted. For this reason, proposals related to improving information and knowledge retrieval from the Web are receiving a great interest in the last years.

In this paper, we propose a multilingual system called Diana, motivated by the interest to create a program that allows to collect all the results that a web search returns, gathering the most relevant information correctly and showing it to the user in a compact and easy-to-consume format. With Diana, the user is able to have a global vision of the general contents of the set of web pages that the search engine has returned, without the need to perform a time-consuming exploration of the contents of each one separately.

Data fusion is defined as a "multilevel process that deals with the automatic detection, the association, the correlation, the estimation, and the combination of data and information from single and multiple sources" [1]. Information fusion systems deal with large quantities of heterogeneous data. The data fusion of multiples sources implies a set of techniques, similar to the cognitive process that a human performs, to integrate data of multiple sensors/observers in order to make inferences about the outside world. Therefore, data fusion aims at obtaining a better and precise overview of the relevant world.

The automatic integration of information from multiple and diverse sources is a central topic for many data fusion applications. In our system, we will have a very specific type of data items, which are documents (web pages) that contain textual information. So, in this paper we perform a data fusion of several sources of text-based information with the goal to provide a general overview of their contents through a suitable summarization process.

In our approach, on the one hand, the set of web pages obtained from the search are gathered, and they are converted to plain text in order to process their contents using Natural Language Processing (NLP) tools, semantic techniques, and statistics. On the other hand, we also carry out the same text extraction process but taking into account the HTML labels of this aforementioned text. The relevance of each sentence in the text is evaluated by using *four different approaches*, obtaining different values/scores for each of them. Three of these values are obtained from the analysis of the plain text, and the other one from the analysis of the HTML marks. These scores are then combined by using classical fusion techniques based on probabilistic logics. The final combined score is used to decide which sentences are the most relevant. Those sentences will be used to compose a summary that synthesizes the most important knowledge collected in the different web pages. Finally, this information is shown to the user.

In order to build the summary, Diana carries out a double clustering approach in order to process the sentences extracted from the web pages. The first clustering process ensures that the content is not repetitive, and the second clustering process assures the selection of the most relevant content, the preservation of the general concepts of the texts, and the organization of the final summary.

To produce highly-informative summaries, the summarization process also includes a simplification step at the end of its processing pipeline. This text simplification aims at clarifying natural language texts, by simplifying their sentences structurally into shorter and simpler ones, while keeping the meaning and information that the sentences contain. Both the simplification and the duplication removal are typical steps in a data fusion process [2]. Moreover, it is

important to remark that the user can configure the search of the information using a set of filters as, for example, setting the number of pages that the search will have to return, or the size of the summary performed.

To sum up, Diana is an application that allows to perform a summary of several web pages and present it to a user to provide him/her with a general overview of the results. The summary is created through a search of similar content among the information collected in all the web pages found, using fusion techniques. Diana also uses relevant attributes of the web pages (such as links and heading tags) to obtain a better summary, as certain HTML tags may contain relevant content that should be especially assessed for potential consideration as part of the summary.

Therefore, the main contribution of this paper is the development of a new methodology of information retrieval that uses NLP, semantics, statistics, and fusion techniques, to obtain informative summaries that contain the main knowledge stored in a set of web pages.

The rest of this paper is structured as follows. Section II is a brief description of the state of the art. Section III explains our proposed methodology. Section IV shows a working example. Finally, Section V provides our conclusions and explains our lines of future work on this topic.

II. Related Work

In this section, we briefly review some work related to information extraction from web pages and fusion for information retrieval.

A. Information Extraction from Web Pages

A web page usually contains several elements related to navigation, decoration, interaction and contact information, which are usually not relevant to a user's search. Therefore, detecting the content structure of a web page could potentially improve the performance of tasks related to information retrieval from web pages. HTML filtering tools allow us to extract only the relevant information. For example, the well-known PageRank [3] is a tool able to classify web pages objectively and effectively, in order to measure the interest of their content.

There are several proposals dedicated to information extraction from the Internet, such as for example [4], which presents a portal that provides a software agent configured to perform searches of web pages and summarize them. These summaries can be accessed by subscribers of the service. Users can specify their information needs by filling templates. The information obtained can be sent immediately to the subscriber or saved by the portal server for later retrieval.

It is interesting to mention that we can find many tools that automatically obtain summaries from web sources, such as for example: (i) summarization services of news in SMS (Short Message Service) messages or e-mail format for mobiles phones, (ii) searching services that obtain a summary automatically translated from a foreign language text, and (iii) search engines (like Google) that presents compressed descriptions of the search results.

Another interesting and important aspect is the readability in web summarization algorithms and the readability quality in real-time monitoring of search engines. In this sense, we can find proposals such as [5], which uses machine learning with a corpus of random queries and their corresponding search results. Then, the features of the summaries that are judged to potentially characterize readability are extracted. The authors try to obtain a model (gradient boosted decision tree) that predicts human judgments based on the features in order to avoid an alternative time-consuming solution where the quality of the summary is judged by a human. The model obtained can then be compared with other models of readability, such as the Collins-Thompson-Callan model [6].

Finally, we can find a work that also studies the use of abstracts in the context of the Web is [7]. This work uses hierarchies to provide a means of organizing, summarizing, and accessing correct information. This technique allows summarizing the documents retrieved by a search.

B. Fusion Applied to Information Retrieval

Data fusion has two main goals: increasing the integrity and boosting the conciseness of data. An enhance of integrity is achieved by adding more data sources to the system, and an increase in conciseness is reached by removing redundant data and ambiguities [2]. These goals are explained in works such as Motro [8] or Naumann [9]. In the context of data fusion, two main quality measures are considered: conciseness and completeness. These measures are analogous to the classical precision and recall measures used in the context of information retrieval. The results obtained should be complete, concise, and consistent. A result is complete if it contains all the relevant objects and all the relevant attributes present in the sources; it is concise if all the real-world objects and all the semantically-equivalent attributes present are described only once; finally, it is consistent if it contains all the tuples from the sources that are consistent regarding a specified set of integrity constraints (inclusion or functional dependencies) [10].

To conclude this section, it is interesting to mention two additional interesting systems that perform information retrieval over multiple documents. First, *SIMS* [11] is an information mediator that provides access and integration of multiples sources of information. It determines the relevant data sources and how to efficiently retrieve the desired information. Second, *Ariadne* [12] consists of wrappers for web sources and a central mediator which is responsible for query processing. The domain model and query language are the same as in SIMS, whereas Ariadne additionally includes binding patterns in the source descriptions.

Our methodology of information retrieval is different from the approaches mentioned previously, as we have merged NLP, semantics, statistics, and fusion techniques, to obtain summaries that contain the main knowledge stored in a set of web pages. Our proposal offers a configurable tool that allows to obtain a summary for each search engine selected from a query introduced by the user. With Diana the users can obtain easily a general idea about the topic searched.

III. Knowledge Fusion with Diana

The aim of our system Diana is to facilitate the identification of interesting contents available in the Internet. This system allows to extract the most relevant information about a certain topic in the Web in the form of a summary. We have developed an algorithm based on *the method of the three steps* proposed by F. Naumann [2]. First, we retrieve a set of web pages from the Internet using a search engine. Then, we extract from each web page a set of features that are considered useful for the data fusion process. Afterwards, we normalize the data obtained. Next, Diana proceeds to combine all the data obtaining a text summary of the contents as a final product. Finally, the summary goes through a simplification process, in which duplicates and ambiguities are eliminated, so that only the most relevant parts are extracted and the rest is discarded. In the rest of this section, we describe this process in detail.

A. Configuration and User Data Entry

Before performing a search, the user can configure the search engines on which the query is executed, as well as the number of web pages to be returned, when the default configuration is not desired. The default configuration in our prototype uses Google as the search engine, and as the number of web pages to return we have chosen 5 to simplify the results. After gathering all the settings, Diana shows a window where the user can introduce a set of keywords to perform the query. Diana launches the query over each of the selected search engines, searching in the Internet the aforementioned number of web pages with relevant content to the user's query. Finally, all those web pages are collected in a local repository for further analysis.

B. Extraction of Relevant Features

At this stage, Diana will extract all the features offered by the contents obtained in HTML format. First, the text in HTML must be transformed to legible text without tags. To do this, Diana uses a module that performs a filtering process which removes the tags and keeps the plain texts.

The text of each web page is divided into sentences to identify which of those sentences have more relevant information. Then, Diana scores each sentence. This score depends on the terms that compose the sentence. For this, Diana lemmatizes the text. The sentence scoring procedure defines the sentence relevance in the overall collection of sentences obtained from each input web page text. The overall score is computed as the sum of the `tf-idf` (Term Frequency - Inverse Document Frequency) scores [13] (it is obtained in turn by considering the word lemmas obtained previously) of each word of the sentences s ($tfidf_w$), smoothed by the number of words in the text of the Web page (*totw'*). With this metric, the relevance of a sentence not only depends on the frequency of the words present in it, but also on the number of texts in which the words appear. Equation 1 describes the computation of this score, where "w" refers to words that belong to the sentence "s":

$$score_s = \frac{\sum_w (tfidf_w)}{totw'} \quad (1)$$

The features that Diana obtains in this stage for each web page are: *named entities*, *keywords*, *synsets*, and *HTML content*. Named entities are fragments of text that represent names of persons, organizations, or locations [14]. For example, if we have the sentence "Peter drives a bus in New York", the named entities in this case are "Peter" and "New York", because the first one is a name of a person and the second one is a location. The keywords are the relevant words that are used to reveal the internal structure of a document. The synsets are provided by WordNet [15], that is a large lexical database. They are a set of cognitive synonyms expressing a distinct concept depending on the word given. The synsets are interlinked by means of conceptual, semantic and lexical relations. The synsets allow to reduce the number of relevant keywords of a sentence. An example of synset would be: *good*, *right* and *ripe*, where each word is most suitable or convenient than the other according to the context in which you are:

- "Now is a good time to go to the mountain"
- "Now is the right time to act"
- "The time is ripe to great changes"

Two analysis processes are performed to obtain the features mentioned above. The first one considers only the plain text of each web page. In this first analysis Diana detects named entities, synsets, and keywords. The second one uses the full information of the web pages, which includes both the plain text and the HTML tags referring to the format of those texts.

For each feature, a different scoring method is applied. The score for the keywords depends on the *tf-idf*, i.e., it depends on the frequency with which the term appears. The score for the named entities is computed in a similar way, but dividing the total score for the named entities by the total number of named entities. In the same way, the score obtained for the synsets is the sum of the score of each synset divided by the number of synsets contained in the text.

The scores based on the HTML tags are computed quite differently than the ones obtained for the other features. The

main idea is to give a score to each sentence according to the HTML format of its text. Headers, underscores, and italics receive a higher score than plain text. The size of the font used is also taken into account, as it is assumed that text shown more prominently on the web page is considered more important by the authors of the page. The weights of every text class can be configured by the user, allowing for more appropriate settings according to the context.
The extraction of each feature is performed as follows:

1) *Keywords* are identified by performing a morphological analysis to extract a list of important words from each text. Later, these words allow to identify which parts of the text "provide more knowledge" than others.
2) *Named Entities* are extracted using a morphological analysis with natural language processing tools and a gazetteer (to know if an entity is a place). We only take into account these kinds of elements in every sentence to obtain the score using the aforementioned statistical method `tf-idf`.
3) *Synsets* are obtained through a semantic analysis using a lexical database. So, if we know the synset of the words that compose the sentences, we can reduce the number of relevant keywords in order to perform a statistical analysis. This is due to the fact that some sets of words (for example: way, path, road, track and trail) correspond to the same synset.
4) *HTML Tags* such as images, tags, footers and headers, semantic tags (i.e., markup tags contained in an HTML file, to reinforce the semantics or meaning of data instead of just defining their presentation), hyperlinks, and comments, are extracted from each HTML file collected. These features are useful to know if the corresponding web page is related or not with the topic the user is interested in, as we rely on the assumption that everything that appears somehow highlighted in the web page is more relevant and therefore deserves a greater weight in its score.

C. Schema mapping

Once Diana has collected all the features of the web pages found, it proceeds to adjust the received data to a common measure. For this, the system normalizes all the scores obtained from the different features between 0 and 1. In this way the scores can be easily compared in the data fusion stage.

D. Data Fusion

In this stage, the fusion of all the data collected is performed, i.e., Diana unifies the multiple heterogeneous results obtained in the previous stage. The integration system responsible for carrying out this task combines the results, and then the combined output is shown to the user.

At this point, we use techniques related to data fusion based on statistics. We can see in Figure 1 a diagram that shows the process. To perform data fusion, each feature receives a modifier (weight) for its corresponding score before obtaining the unified final value.

Given our features, which are keywords (K), named entities (N), synsets (S) and HTML content (H), and their weights (wk, wn, ws, and wh, respectively), we can consider the general expression 2 to calculate the unified final value (V) of each sentence found.

$$V = \frac{(K * wk) + (N * wn) + (S * ws) + (H * wh)}{4} \quad (2)$$

Suitable values for the weights were obtained experimentally for our prototype, after testing it, so the context definitely plays an important role in order to tune the system. For example, if the scope of the search are news, the headlines are very important, because they capture much of the information contained in the text. In that case, the scores of the content of the HTML tags that belong to the header (h1, h2, ...) will be increased. In our tests, we have seen that when the scope is news, the scores of the headers can be multiplied up to 5, in order to acquire good results. As another example, in the scope of news we would also give significant weight to named entities that appear in the text, like important people or places.

Although the expression 2 shows an aggregation of values, it is important to remark that the aggregation is not the only data fusion action performed, as other aspects must also be considered, such as the data pre-processing, simplification and duplication removal, data alignment, and semantic interpretation.

When all the data have been combined, we have a set of sentences with a consolidated score. Then, we use an algorithm of automatic summarization [16] to obtain the final summary that contains the most relevant information found in all the web pages. This method consist of identifying which sentences are the most relevant ones according to their overall score.

Now, the aim is to identify relevant information in the collection of all the sentences contained in the web pages selected, and this is performed in two steps: a similarity clustering and a keyword clustering.

1) *Similarity clustering*. The similarity between two sentences is calculated considering the lemmas of the words. We obtain a similarity value taking account the subsequences of sentences and the overlapping words. If the similarity value is higher than the similarity threshold, then the sentences are grouped in the same cluster, to identify sentences conveying the same information. Finally, the sentence with the highest score from each cluster is selected to be used in the next step.

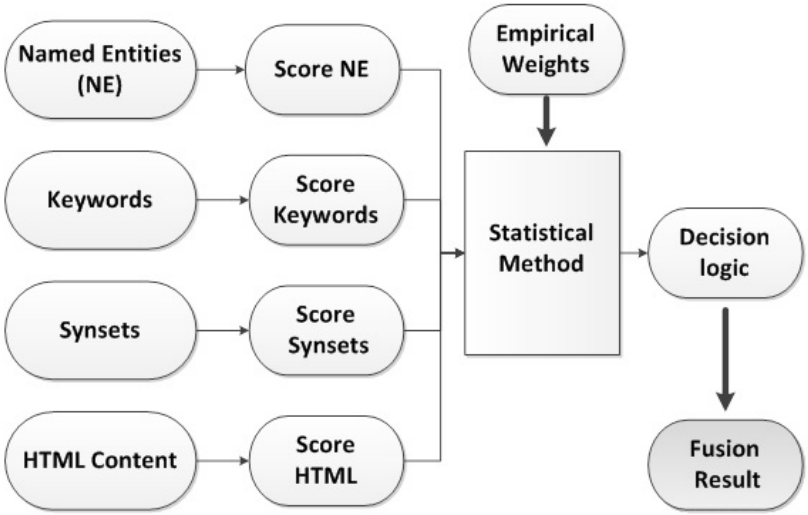

Figure 1. Process diagram

2) *Keyword clustering*. In this step, we have used an adapted version of the K-means algorithm to cluster sentences by keywords. As mentioned before, the input of this clustering process is composed by the sentences with the highest score selected in the previous clustering process. The keywords extracted previously represent the output clusters. A sentence is added to the cluster represented by the keyword that occurs more often in it. The sentences that do not contain any keyword are ignored.

After the two-step clustering process described above, Diana sorts the sentences based on their score, from the most relevant one to the less relevant one. Sentences at the top of the list are considered more significant than the others, since they address the main topics conveyed by the collection of sentences. At this point, the set of the most significant sentences constitutes already a summary of the text. Then, a syntactic and grammatical analysis of each sentence is performed, in order to identify the function of each word in the sentence and its type. For this purpose, we use an NLP tool. Afterwards, we carry on a simplification step in which duplicates and ambiguities are eliminated, so that only the most relevant parts are extracted and the rest is discarded. In this simplification step, we use an NLP tool too. Finally, we have a summary that it is composed by the most significant sentences.

The algorithm used in this section is motivated by its good performance in other applications such as TM-GEN [17], SOLE-R [18], and TMR [19]. For instance, in the first case, the algorithm was used to obtain a summary of multiple documents, and then used it to graphically represent knowledge through topic maps; in the second and in the third case, the applications are based on the use of TM-GEN, but in these cases the concept maps represent items to recommend. The concept maps are used too to model user profiles, and finally they are compared between them.

E. Summary of the Process

Summing up, the first thing that the user must do is to configure the system. Then, he/she launches the query. Next, Diana extracts all the features of the web documents obtained in the results using the selected search engines. Afterwards, it computes the score of each relevant feature, and the scores are normalized. Then, a data fusion is performed to combine the scores. Finally, it obtains a summary by selecting the most relevant sentences (the ones with the higher scores) of the web pages.

IV. A Complete Illustrative Example

The complete example we explain below has been performed with real users who have tested the system (see Figure 2 for a snapshot of the user interface of Diana). In our experiments, we have used Freeling [20] as NLP tool, both for morphological analysis as for syntactic analysis of the texts. As a lexical database we have used EuroWordNet [21].

To perform our experiments, we used a specific set of searches for our tests, which have helped us to prepare the system. This set is obtained by repeating the same questions all the time in the search engines and then analyzing the results (the summaries). 25% of the results of these searches have been used to train the system concerning the data fusion modifiers and the remaining 75% was used for real testing.

To carry out our examples, we have used several queries using different search engines. In the first test, the user introduced in Diana the search string “new mobile iphone 5”. The search engine found several web pages with relevant content. An example of the results obtained can be seen in Figure 3.

Figure 2. Diana interface

Many of the results obtained belong to the official website of Apple and the mobile company T-MOBILE. Diana allows us to choose among the selected search engines, and it shows us a first summary with the data obtained for each search engine. Then, we can access the results through a button whose name is "Summary".

In this figure, we can see the URL found, and part of the content of the web page. Moreover, we can access a little window where we can see the content of the page found, such as the title, the number of words that compose the page, the links, the anchor text along with its link, the keywords, and the matches, as can be seen in Figure 4. Of course, the user can access more related information by clicking on the corresponding link (for instance, in this case it points to a web shop). This information can be about the topic of the page (in this case, technology and communications). Moreover, if the user wants, he/she can export the result obtained.

If we launch the query "new mobile iphone 5", Diana returns the following summary:

The new iPhone 5 is in black or white with 16GB or 32GB. iPhone 5 is the thinnest, and lightest. The iPhone is so simple and intuitive, you don't need a manual. iPhone 5 features a 4-inch Retina display, ultrafast wireless and the powerful A6 chip.

In the second example, we have used Yahoo as the only search engine. The search introduced in Diana has been "stores new iphone". Now, the results also belong to the official web of Apple as well as to other web pages with information about the high demand for these mobile phones. For this query, Diana returns this summary:

The Apple Stores will start taking reservations for the iPhone 5s and 5c on September 17. Apple has announced. iPhone 5s went on sale in 10 countries on Friday, September 20. The new Iphone was initially available in the U.S. with shipping estimates of 1-3 days, however the gold iPhone 5s was quickly sold out, and the shipping estimates of all the other models slipped to 7-10 business days.

In the third example, the only search engine used is Bing. In this case, the string introduced has been "iphone new launch". The results obtained for this query are a little different from the other cases. For example, we can find web pages discussing the characteristics of the new iPhone, web pages with videos about its release, and also news about this. The summary obtained for this query is:

Apple is expected to unveil its next iPhone on Sept. 10. The WSJ's Deborah Kan and Yun-Hee Kim discuss what to expect at Apple's big event. The hoopla with the new Iphone will be big. On September 10, 2013, Apple unveiled two new iPhone models. As thousands stood in long-lines outside Apple stores, iPhone users eagerly switched to the new operating system. The iPhone 5S has a brand new A7 processor, that's 40 per cent faster. With its new A7 chip, iPhone 5s is the first smartphone with 64-bit technology, providing blazing-fast performance when launching apps, editing photos. The iPhone 5S, which starts at $199.

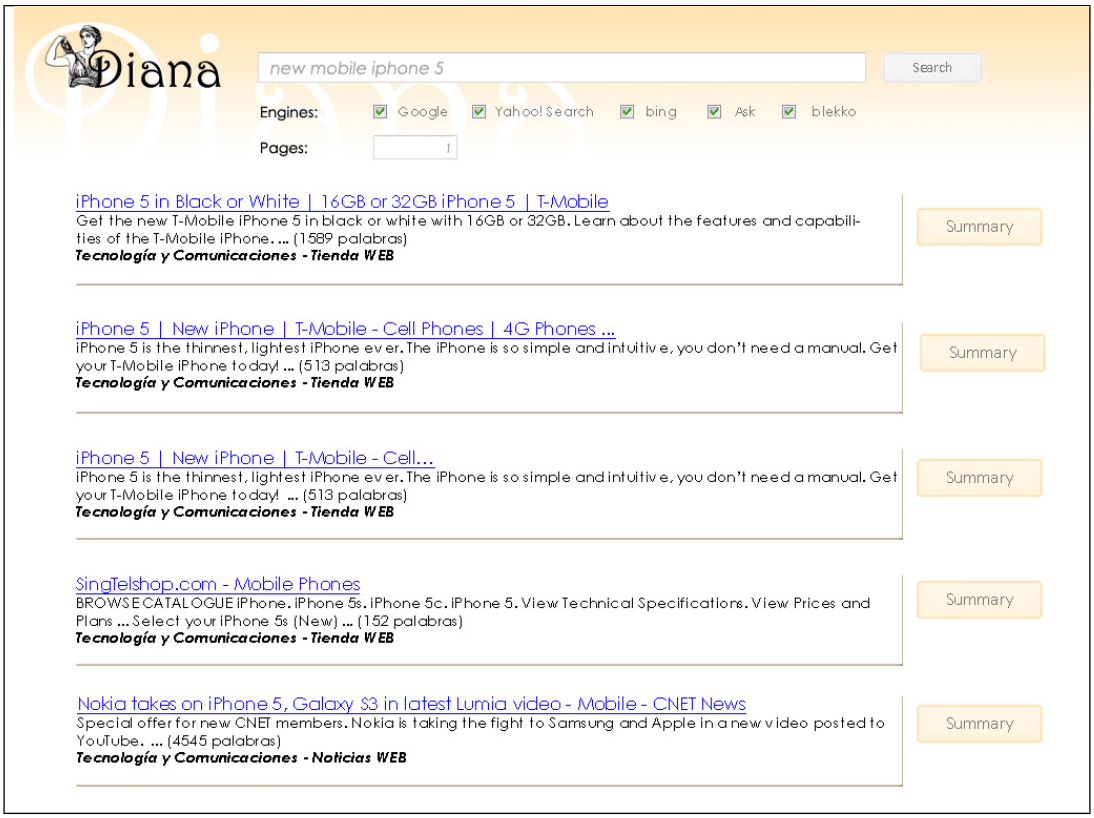

Figure 3. Results of the search

V. Conclusions and Future Work

In this paper, we have presented a multilingual system called Diana, that allows to obtain knowledge from a set of web pages using data fusion techniques with the help of a morphological analyzer, a lexical database, semantic tools, a syntactic parser, statistics methods, and a generic gazetteer. The proposal has been tested using well-known search engines.

Our main contributions are:

- The development of an algorithm to automatically obtain a summary from a set of web pages.
- The introduction of a new method to build summaries from a set of web documents using fusion techniques.

The possible scenarios where Diana can be used are diverse. For example, to improve the documental search engines, to facilitate tasks related with SEO (Search Engine Optimization), to increment the performance of Web browsers, or to automate and digitize the massive information. Finally, it may also be useful for end users.

The system is currently being tested and the results obtained so far are promising. However, this is still a seminal work and our next step will be to perform a more rigorous user testing to quantify the effectiveness of the solution in different contexts.

Another interesting idea for future work is to improve the interface showing the final summary, to look like a "report", combining texts and images of different websites found. In other words, the interface should be more friendly and intuitive for the user.

Acknowledgment

This research work has been supported by the CICYT project TIN2010-21387-C02-02 and DGA-FSE.

References

[1] E. Waltz, J. Llinas *et al.*, *Multisensor data fusion*. Artech house Boston, 1990, vol. 685.

[2] J. Bleiholder and F. Naumann, "Data fusion," *ACM Computing Surveys (CSUR)*, vol. 41, no. 1, p. 1, 2008.

[3] L. Page, S. Brin, R. Motwani, and T. Winograd, "The PageRank citation ranking: Bringing order to the web." *Stanford InfoLab*, 1999.

[4] S. K. Inala, P. V. Rangan, R. Satyavolu, and S. P. Rajan, "Method and apparatus for obtaining and presenting web summaries to users," Dec. 14 2000, U.S. Patent App. 09/737,404.

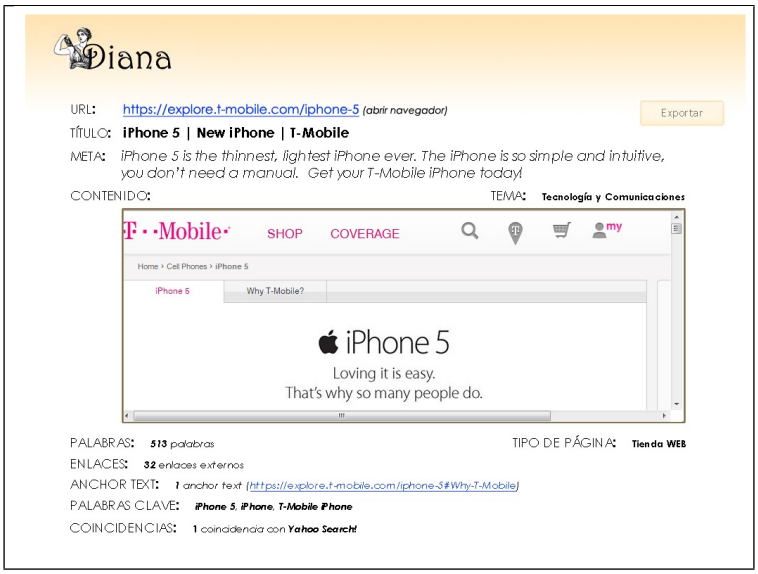

Figure 4. Diana Result

[5] T. Kanungo and D. Orr, "Predicting the readability of short web summaries," in *Second ACM International Conference on Web Search and Data Mining*. ACM, 2009, pp. 202–211.

[6] M. Heilman, K. Collins-Thompson, and M. Eskenazi, "An analysis of statistical models and features for reading difficulty prediction," in *Third Workshop on Innovative Use of NLP for Building Educational Applications*. Association for Computational Linguistics, 2008, pp. 71–79.

[7] D. J. Lawrie and W. B. Croft, "Generating hierarchical summaries for web searches," in *Annual International ACM SIGIR Conference on Research and Development in Informaion Retrieval*. ACM, 2003, pp. 457–458.

[8] A. Motro, "Completeness information and its application to query processing." in *Very Large Data Bases Conference*, vol. 86, 1986, pp. 170–178.

[9] F. Naumann, J.-C. Freytag, and U. Leser, "Completeness of integrated information sources," *Information Systems*, vol. 29, no. 7, pp. 583–615, 2004.

[10] A. Fuxman, E. Fazli, and R. J. Miller, "Conquer: Efficient management of inconsistent databases," in *Proceedings of the 2005 ACM SIGMOD International Conference on Management of Data*. ACM, 2005, pp. 155–166.

[11] Y. Arens, C.-N. Hsu, and C. A. Knoblock, "Query processing in the sims information mediator," in *ARPA/Rome Laboratory Knowledge-Based Planning and Scheduling Initiative Workshop*, 1996.

[12] C. A. Knoblock, S. Minton, J. L. Ambite, N. Ashish, I. Muslea, A. G. Philpot, and S. Tejada, "The Ariadne approach to web-based information integration," *International Journal of Cooperative Information Systems*, vol. 10, no. 01n02, pp. 145–169, 2001.

[13] G. Salton and C. Buckley, "Term-weighting approaches in automatic text retrieval," *Information Processing and Management*, vol. 24, no. 5, pp. 513–523, 1988.

[14] S. Sekine and E. Ranchhod, *Named Entities: Recognition, Classification and Use*. John Benjamins, 2009.

[15] G. A. Miller, "WordNet: a lexical database for English," *Communications of ACM*, vol. 38, no. 11, pp. 39–41, 1995.

[16] S. B. Silveira and A. Branco, "Extracting multi-document summaries with a double clustering approach," in *Natural Language Processing and Information Systems*. Springer, 2012, pp. 70–81.

[17] A. L. Garrido, M. G. Buey, S. Escudero, S. Ilarri, E. Mena, and S. B. Silveira, "TM-gen: A topic map generator from text documents," in *25th International Conference on Tools with Artificial Intelligence*. IEEE Computer Society, ISBN 978-1-4799-2971-9, ISSN 1082-3409, November 2013, pp. 735–740.

[18] A. L. Garrido, M. S. Pera, and S. Ilarri, "SOLER-R, a semantic and linguistic approach for Book Recommendations," in *14th IEEE International Conference on Advanced Learning Technologies - ICALT*, To be published in July 2014.

[19] A. L. Garrido and S. Ilarri, "TMR: A Semantic recommender system using topic maps on the items descriptions," in *11th European Conference of Web Semantic*, To be published in May 2014.

[20] X. Carreras, I. Chao, L. Padró, and M. Padró, "FreeLing: An open-source suite of language analyzers," in *Fourth International Conference on Language Resources and Evaluation (LREC'04)*. European Language Resources Association, 2004, pp. 239–242.

[21] P. Vossen, *EuroWordNet: A multilingual database with lexical semantic networks*. Kluwer Academic Boston, 1998.