An Online Linguistic Journalism Agency – Starting Up Project

Annibale Elia [1], *Ernesto D'Avanzo* [1], *Tsvi Kufik*[3], *Giovanni Catapano* [1], *Mara Gruber* [2]

[1] CLiCLab, Department of Communication Sciences, University of Salerno, Fisciano (SA), Italy
[2] Istituto Italiano di Scienze Umane, Napoli, Italy
[3]Management of Information Systems Department, University of Haifa, Haifa, Israel
aelia@unisa.it, edavanzo@unisa.it, tsvikak@mis.hevra.haifa.ac.il, gcatapano@unisa.it, gruber@fbk.eu

Abstract

The Web provides easy access from everywhere to every kind of information. It is becoming a substitute source for news, instead of traditional media such as newspapers, radio and television. However, with the ease of access and the tremendous amounts of information available online, finding relevant information is not an easy task. This paper reports on an under development joint research project which aims at the development of an online Journalism Agency that makes use of Natural Language techniques in order to provide trainee and professional journalists with topical summaries of information relevant to their interest on different channels (Web, PDAs, etc.). Our system obtained good results of the linguistic quality of the summaries in international summarization campaigns. With such a background we are quite optimistic for the future development of the project.

Index Terms: Automatic Text Summarization, Machine Learning, Linguistic Analysis, Web Mining

1. Motivation and Background

A report by Forrester [1] provoked many debates concerning the Web as substitute source for news, taking the place of traditional media as newspaper or television. This seems to be a growing trend as the report pointed out. New York Times' Web site for example counts about 20 million logs monthly meanwhile the printed edition is over one million daily. The huge number of news sources available online, and the easy, fast and free access to many news Web sites are only some reasons that motivate this trend.

However, the more information is available the harder it is to access the really relevant information. A reader interested in a given topic must read a large number of news items before he/she can satisfy his/her information need, facing the well known problem of the *information overload*.

To assist the *information consumer* in coping with this problem an out-and-out call to arms was raised, bringing together people from different disciplines in order to design and develop methods and techniques able to automatically, succinctly, and efficiently summarize these huge volumes of information available.

The "consumption" of information and accessing it is also a hard problem for information professionals such as journalist that are required on a daily basis, or even in real-time, to digest volumes of news and summarize them for a briefing with their editorial unit or to be published on the online or printed version of their newspaper.

The Department of Communication Sciences, at the University of Salerno, founded last year a Journalism School. Immediately, a lot of interests, both professional and scientific arose around the School.

One of the main goals of the Scientific Board of the School is the integration and the technological transfer from the Department of Communication Sciences. To this end this ongoing research project has been launched with the aim of building an online platform, able to support students of the school (that are journalist trainees), and the academic staff in their daily task of consumption and, at the same time, production of information. The Department developed linguistic resources as DELAS-DELAF and DELAC, two Electronic Dictionaries for single and compound terms containing about 500,000 lexical entries and local grammars. Based on these resources several Text Processing tools were built, that allow these resources to be easily exploited for Web Mining tasks.

During the past summer the Department of Communication Sciences launched CLiCLab (Computational Linguistics and Complexity Laboratory) an interdisciplinary lab with interests ranging from Computational Linguistics to Web Mining and Computational Models up to Cognitive Systems. The lab brings together people with different backgrounds, coming from different Departments and Universities worldwide.

CLiCLab works as a catalyst in order to start up this joined project between the Department of Communication Sciences and the Journalism School. Among its main activities CLiCLab developed Text and Web Mining tools acknowledged in international competitions where they were tested.

Summing up, we believe that the development of the Automatic Journalism Agency will bring the latest text mining and summarization research results and tools to the doorstep of future journalists, allowing them to better exploit the new opportunities offered by the Web.

2. Related Work

Among worldwide news Web services, *Google News* and *AltaVista News* are the most popular examples of news services that present clusters of related documents. However, none of these services is equipped with an overall summary of the whole cluster. And even if there are summaries for each document, the majority of these are created by extracting the top few sentences, with the risk of loosing important topics appearing later in the document.

Radev and co-workers developed NewsInEssence [2] a summarization system that, given a user's topic, acts as a users's agent in order to gather and summarize online news articles. The system can gather clusters of related stories from different sources on the Web and generates summaries of the whole cluster, emphasizing its most important content. NewsInEssence allows users to create personalized clusters of summaries. Cluster summaries created by NewsInEssence,

however, are not so readable and this is the major drawback of the system.

McKeown and co-workers designed Newsblaster, another system for online news summarization [4]. The systems provides updates of the news daily, crawling news sites, filtering out news from non-news, grouping news into stories on the same event, and, finally, generating a summary of each event. The crawling made by Newsblaster considers a tunable number of news sites among which CNN, Reuters, Fox News, NY Post, etc. For each page considered if the amount of text is greater than a constant (usually about 500 characters) then it is assumed to be a news article.

Since its launch Newsblaster got some changes. The interface shows a "close up" frame where there are the most important news consisting of a cluster summary about a hot topic for which have been gathered news in the past few days. At the time of this writing the close up news is titled "Ahmadinejad: Annapolis failed, Israel doomed to collapse". The news belongs to *World* category, one of the six appearing on the home of the system (other five categories are *U.S., Finance, Entertainment, Science/Technology, Sports*) and is created summarizing 32 sources articles (Washington Post, L.A. Times, nytimes.com, etc.) that are listed at the end of the page containing the summary. The summary is completed with a list of keywords and an *event tracking* to record the story's development in time. At the core of Newsblaster there are two main components: the organizer of stories, the clusterer and the multidocument summarizer. The former component uses agglomerative clustering with a groupwise average similarity function and linguistic features, such as terms, noun phrase heads and proper nouns. The summarizer component is the Columbia Summarizer made of different summarization strategies chosen depending on the typology of document sets (i.e. *single-event* documents, *person-centered* documents and *multi-event* documents). For *single-event* document summarization Newsblaster uses MultiGen a system that makes use of machine learning and statistical techniques to extract similar sentences (a set of similar sentence is called *theme*). Afterward, an alignment of parse trees finds the intersection of similar phrases within sentences. Language generation techniques are then used to cut and paste together similar phrases from the theme sentences. A theme corresponds roughly to one sentence of the summary.

Biographical document are summarized using DEMS that uses frequencies of concepts (set of synonyms) combined with global information about what words are likely to appear in a lead sentence, to decide if an article sentence should be included in the summary

3. The Start Up Project: a Linguistic Journalist Agency

The project of our news agency is composed of several modules that can be grouped in three main components:

- Topical Web crawler able to gather related news on the Web
- Summarizer able to create multi-document summaries that gets the news clusters coming from the previous module and creates summaries using Natural Language Processing techniques
- A Web platform able to deliver summaries created to different channels (e.g., email, Web, PDA's)

The whole platform is equipped with an agent like behavior. The system infers users (journalists) preferences. Then, based on these it runs the topical crawler to gather new clusters of news. In the following we describe each of these components. Some of them are mature methodologies that we developed and evaluated as standalone applications. Others are under development and need to be evaluated. An overall evaluation of the agency, involving professional and trainee journalists, is planned after a first prototype will be released (September 2008).

3.1. Crawling

A web crawler/Spider is an automated script which automatically browses the World Wide Web. The crawler is mainly used by sites for providing updated data from web pages; it creates a copy of all the visited pages for later processing by a search engine that will index the downloaded pages to provide fast searches.

For our purposes in this early stage we immediately played with topical crawlers [5], a method and a system for directing Web crawling to a topic, using a focused search engine that produces a specialized collection of documents and document ranking. The method includes the following steps:

- receiving a user's request query that includes one or more words, phrases or documents, for defining the topic
- generating an affinity list which is a ranked list of terms, phrases, documents or set of documents related to the query that are derived from statistics about the document collection locating and retrieving seed documents, that includes relevant and irrelevant information
- training a binary classifier using seed documents to define documents
- causing a web spider to locate and retrieve documents related to the user's query
- ranking URLs associated with the documents

An important aspect to consider when applying crawlers, especially topical crawlers, is the nature of the crawl task. Crawl characteristics such as queries and/or keywords provided as input criteria to the crawler, user-profiles, and desired properties of the pages to be fetched (similar pages, popular pages, authoritative pages etc.) can lead to significant differences in crawler design and implementation.

In our experiments we have available profiles (as a set of keywords) acquired from trainee journalists. After a preliminary set of experiments where keywords were manually provided we installed cookies on the client (a client for each trainee) in order to automatically acquire user preferences during her/his navigation.

3.2. Document Summarization

This is the main components of the platform. In this preliminary stage to let us end the overall development cycle we are using LAKE (Linguistic Analysis based Keyphrase Extractor) a tool that worked as a single-document and multi-document summarization obtaining encouraging experimental results.

LAKE is a keyphrase extraction system based on a supervised learning approach that applies linguistic processing on documents. In the past the system used Naïve Bayes algorithm [6] as the learning method and $TF \times IDF$ term weighting with the *position* of a phrase as features. For this year competition we have used a Support Vector Machine (SVM) as a learner [7]. Unlike other keyphrase exctraction systems LAKE chooses the candidate phrases using linguistic knowledge. The candidate phrases generated by LAKE are sequences of Part of Speech (PoS) containing Multiword Expressions (ME) and Named Entities (NE). Extraction is driven by a set of "patterns" which are stored in a pattern database; once there, the main work is done by the learner

device (i.e., the SVM). The linguistic database makes LAKE unique in its category.
LAKE is based on three main components: the Linguistic Pre-Processor, the candidate Phrase Extractor and the Candidate Phrase Scorer. In the following sections there is a brief description of the system. For a more detailed description the reader is referred to previous publications.

Linguistic Pre-Processor

Every document is analyzed by the Linguistic Pre-Processor following three consecutive steps: Part of Speech (PoS) analysis, Multiword Expressions (ME) recognition and Named Entities (NE) recognition.

Candidate Phrase Extractor

Syntactic patterns have a twofold objective:

- focusing on uni-grams and bi-grams (for instance Named Entity, noun, and sequences of adjective+noun, etc.) to describe a precise and well defined entity;
- considering longer sequences of PoS, often containing verbal forms (for instance noun+verb+adjective+noun) to describe concise events/situations.

Once all the uni-grams, bi-grams, tri-grams, and four-grams are extracted from the linguistic pre-processor, they are filtered with the patterns defined above. The result of this process is a set of keyphrases that may represent the current document.

Candidate Phrases Scorer

Candidates keyphrases identified in the previous step are scored in order to select the most appropriate phrases as representative of the original text. The score is based on a combination of *TF ×IDF* and *first occurrence*, i.e. the distance of the candidate phrase from the beginning of the document in which it appears.
However, since candidate phrases do not appear frequently enough in the collection, it has been decided to estimate the values of the TF×IDF using the head of the candidate phrase, instead of the whole phrase. According to the principle of headedness (Arampatzis et al., 2000), every phrase has a single word as head. The head is the main verb in the case of verb phrases, and a noun (last noun before any post-modifiers) in noun phrases. As learning algorithm, it has been used an SVM provided by the WEKA package (Witten and Frank, 1999).
The classifier was trained on a corpus with the available keyphrases. From the document collection we extracted all nouns and verbs. Each of them was marked as a positive example of a relevant keyphrase for a certain document if it was present in the assessor's judgment of that document; otherwise it was marked as a negative example. Then the two features (i.e. TF×IDF and first occurrence) were calculated for each word. The classifier was trained using this material and a ranked word list was returned. The system automatically looks in the candidate phrases for those phrases containing these words. The top candidate phrases matching the word output of the classifier are kept. The model obtained is reused in the subsequent steps. When a new document or corpus is ready we use the pre-processor module to prepare the candidate phrases. The model we got in the training is then used to score the phrases obtained. In this case the pre-processing part is the same. Using the model thus obtained, we extracted nouns and verbs from documents, and then we kept the candidate phrases containing them.

The Lake system uses two parameters for controlling its work: one is the maximum number of words allowed in a keyphrase and the second is the maximum number of keyphrases to be extracted from a document.
These parameters are used for creating from a set of documents a brief, well-organized, fluent summary addressing a need for information expressed in a specific topic, at a level of granularity specified in the user profile (DUC-2005 definition).
Lake is required to select the most representative keyphrases that have the highest *relevance* and *coverage* scores of a set of document, given the topic and profile.
The *relevance* of a keyphrase list kl_j with respect to a cluster C_j is computed considering the frequency of the keyphrases composing the list. The intuition is that keyphrases with higher frequency bring the more relevant information in the cluster:

$$relevance(kl_j) = \frac{\sum_{w=1}^{n} freq(w, kl_j)}{freq(w, C_j)}$$

where $freq(w,kl_j)$ is the count of a word w in a certain document and $freq(w,C_j)$ is the count of w in all the documents in the cluster C_j.
The *Coverage* of a keyphrase list kl_j is an indication of the amount of information that the keyphrase list contains with respect to the total amount of information included in a cluster of documents:

$$\mathrm{cov}erage(kl_j, C) = \frac{length(kl_j)}{\max length(kl_j, C)}$$

where $length(kl_j)$ is the number of keyphrases extracted from document j and $maxlength(kl_j,C)$ is the length of the longest keyphrase list extracted from a document belonging to cluster C_j. The intuition underlynes that the longer the keyphrase list, the more is its coverage for a certain cluster.
Relevance and *Coverage* are combined according to the following formula:

$$rep(kl_j) = relevance(kl_j, C) \times \mathrm{cov}erage(kl_j C)$$

which gives an overall measure of the representativeness of a keyphrase list for a certain document with respect to a cluster.
Finally, the keyphrase list which maximize the two parameters is selected as the most representative of the cluster and each keyphrase is substituted with the whole sentence in which it appears.

3.3. Experiments

Document Understanding Conferences (DUC) is a series of text summarization campaigns presenting text summarization competitions results. LAKE participated at DUC since 2004 while obtaining encouraging results every time. For brevity, we only report the linguistic quality of LAKE's summaries

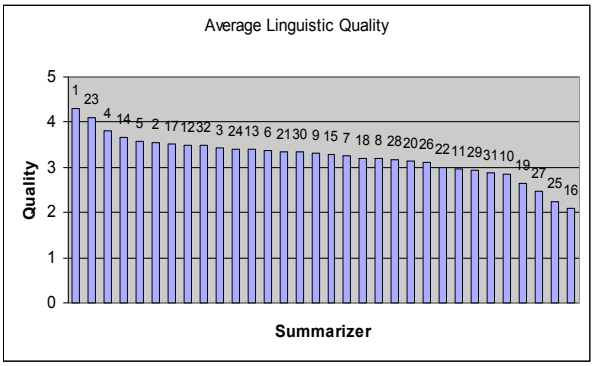

Figure. 1: Average linguistic Quality.

Linguistic quality assesses how readable and fluent the summaries are, without comparing them with a model summary. Five Quality Questions were used, and all questions were assessed on a five-point scale from"1" (very poor) to "5" (very good). Being a linguistically motivated summarizer, LAKE is expected to perform well at the manual evaluation with respect to language quality and responsiveness. Regarding language quality, as can be expected, LAKE scored relatively high – it was ranked 6th out of the 30 systems for average language quality (see figure 1), with an average value of 3.502 compared to 3.41 – the overall average - and 4.23 which was the highest score of the baseline system (no 1) and very close to the second baseline system (no 2) that scored 3.56. However, we should note that most of the systems scored between 3.0 and 4.0 for linguistic quality, so the differences were relatively small. Compared to 2006, Lake got a little lower score (3.5 compared to 3.7), and was ranked relatively lower (3rd in 2006).

Figure 2

3.4. Web Interface

Figure 2 shows the Web interface at very beginning stage. The output of the system will be more elaborated like Newsblaster's one, reporting the sources of the summaries and some more keywords of the final summary displayed on the PDA or mobile of the user.

4. Conclusions

We presented an initial idea for a prototype of an online news agency platform. The prototype will be the product of a joint research project between the Department of Communication Sciences and the Journalism School at the University of Salerno.

The prototype aims at assisting and supporting trainee journalists in their daily search of information avoiding/reducing the Information Overload problem. In fact, even if there are many popular news services available online (e.g. Google News) they all suffer from one major drawback - that the user has to open/read a lot of documents before she/he can find the information satisfying her/his information need. Moreover, current automatically generated snippets/summaries of document only represent the initial parts of them, with the risk of losing a lot of relevant information.

The methodology proposed allows the acquisition of user (journalist) preferences as a background process and gathering the information on an ongoing or daily baseis, based on these preferences.

Information gathered and summarized using machine learning and Natural Language Processing techniques will delivered using different channels: Web, email, PDAs, so users will be able to access it literally every time and everywhere

5. Acknowledgements

We thank Biagio Agnes, Lillo D'Agostino and Pino Blasi respectively Director, President of Scientific Board and Coordinator of the Journalism School for their collaboration and helpful advices in this start up process. We also thank Mario Monteleone that let us to play with DELAS-DELAF and DELAC dictionaries and other resources.

6. References

Kelley, C.M., DeMoulin, G., The Web cannibalizes media. Technical Report, The Forrester Group. May 2002

Radev, D., Otterbacher, J., Winkel, A., and Blair-Goldensohn, S., NewsInEssence: summarizing online news topics, Commun. ACM, Vol. 48, 2005, ACM, New York

Allen, L., Charron, C., and Roshan, S. Re-engineering the news business. Technical Report, The Forrester Group, June 2002.

McKeown, K. R., Barzilay, R., Evans, E., Vasileios, H., Klavans, J. L., Nenkova, A., Sable, C., Schiffman, B. Sigelman, S., Tracking and summarizing news on a daily basis with Columbia's Newsblaster, Proceedings of the second international conference on Human Language Technology Research, Morgan Kaufmann Publishers Inc., 2002, San Francisco, CA.

Pant, G., Srinivasan, P., Menczer, F., Crawling the Web. In M. Levene and A. Poulovassilis, eds.: Web Dynamics, Springer, 2004

Mitchell, T. 1997. Machine Learning. McGraw-Hill.

Cristianini, N., Shawe-Taylor, J. An Introduction to Support Vector Machines and Other Kernel-based Learning Methods, Cambridge University Press, 2000.