Semantic Enrichment of Geographical Databases

Sami Faïz
National Institute of Applied Sciences and Technology, Tunisia

Khaoula Mahmoudi
High School of Communications-Tunis (SUPCOM), Tunisia

INTRODUCTION

The distributed Web-based multi-document summarization system is conceived to enrich semantically Geographic Databases (GDB) (Faïz, 1999; Scholl et al., 1996). In fact, in a traditional database, for instance, a city is described by its alphanumeric features: name, population count, and so forth; however, in a GDB, it is further described by spatial attributes which indicate its position (coordinates) in the space and its shape (point, line, polygon, etc.). Although the use of this myriad of information (alphanumeric and spatial data), the GDB suffers from the lack of an exhaustive set of information describing in a quasi-complete way the entities handled by it (Faïz, 2001). Hence, Geographic Information System (GIS) is not able to provide the end-user with information not fed into the GDB and that is not inherent to the application for which the GIS is designed (Bâazaoui, Faïz & Ben Ghezala, 2001, 2003; Faïz, Abbassi & Boursier, 1998). For instance, given a map displayed on the screen, it is not possible to get economic or historical information about cities for a given country whenever the GIS is concerned only with administrative boundaries. Having this idea in mind, our intention is to profit from the huge mass of information available online to enrich semantically a GDB. To fulfill this purpose and to manage the great amount of documents retrieved from the Web in a quick and convenient fashion, we adopted the Text Mining techniques (Tan 1999; Weiss, Apte & Damerau, 1999) and more precisely the summarization. Indeed, with the fast growth in the amount of textual information available online and the multitude of documents reporting almost the same thing, there is clearly a strong need for automatic summarization that copes with not only one document at one time but a set of topically similar ones.

Such systems are referred to as Multi-Document Summarization systems (MDS). While building such summarizers, there are many user requirements that have to be satisfied, in essence, the minimization of the redundancy and the coverage of all the information reported by the set of documents.

In fact, with the continuous growth in an astounding rate of information and the capacity of reading and analyzing that remains constant, the users are becoming amassed with a huge amount of information. Building up a distributed system is time consuming. To fulfill this objective, we used the Multi-Agent Systems (MAS). The distribution is also justified because MDS can be seen as a naturally-distributed problem owing to the fact that more than one entity is involved. Our intention is to speed up the summarization process while holding the main issues inherent to the problem.

BACKGROUND

Some MDS systems will be outlined and the MAS paradigm will be presented.

Multi-Document Summarization (MDS)

MDS (Lin & Hovy, 2002; Gees et al., 2000; Mani & Bloedoran, 1999; Regina, Kathleen & Michael, 2000) consists of condensing the content of a corpus of documents while coping with some issues, essentially, the coverage and the redundancy. The former concerns dealing with all the information conveyed by the whole collection. For the redundancy, one has to summarize the corpus of documents while not retrieving portions of text reporting the same information already included in the summary. In what follows, we outline some MDS approaches.

Gees et al. (2000) developed a summarizer that first creates individual summaries for all the documents in the set. Afterwards, these document summaries are grouped into clusters according to the similarity of their topics. Finally, for each cluster, a representative summary is selected. In the case of a summary based on a query submitted by the user, the representative summary is the one that has the most similarity with the concerned topic. In the case of a generic summary, the representative will be the one having words that occur frequently across all the summaries of the cluster.
In Radev, Jing, and Budzikowska (2000), the summarization process begins by clustering the articles into groups relative to similar events. Then, for each cluster, the relevance of the sentences is computed. In fact, the degree of relevance (from 0 to 10) for a given sentence to the general topic of the entire cluster is determined by picking out the most frequent words across the cluster. An utility of 0 means that the sentence is not relevant to the cluster, and a 10 marks an essential sentence. Then, each sentence ranked first according to this degree and beyond a fixed threshold is retained. Because many sentences have similar contents, one has to minimize the redundancy by filtering out the resulting sentences. This is based on the notion cross-sentence informational subsumption. It states that a sentence $S_2$ subsumes $S_1$ if it has overlapping words with $S_1$ and presents additional information. Hence, we have to include only one sentence of each class, according to the level of details desired. At the end of this process, each cluster is described by a set of relevant and non-redundant sentences.

Another method belonging to this field is the Maximal Marginal Relevance (MMR) Multi-Document Summarization (Goldstein et al., 2000). This method begins by segmenting the documents into passages. These may be sentences, sequences of sentences, or paragraphs. Then, the relevant passages to a given query are identified, using the cosine similarity metric. Thus, the passages under a fixed threshold are discarded. For the remainder of the passages (above the threshold), the MMR metric is applied. According to this measure, text passage has high marginal relevance if it is relevant to the query, while having minimal similarity to previously selected passages. The selected passages constitute the summary of the documents corpus.

### Multi-Agent Systems (MAS)

MAS (Barbuceanu, 1998; Briot & Demazeau, 2001; Mahmoudi & Ghédira, 2000) are generally regarded as being systems in which a group of autonomous agents interact to perform some set of tasks or satisfy some set of goals. According to Ferber, an agent is a hardware or software entity able to act on itself and on its environment. It has a partial representation of its environment and is able to communicate with other agents. It aims at an individual goal, and its behavior is the result of its observations, knowledge, abilities, and interactions it can have with other agents and with the environment (Ferber, 1997). These agents have several concerns to cope with, and they inhabit and interact within dynamic and not entirely predictable environments. Relevant examples of MAS include a group of agents moving a heavy object, telecommunication networks, and so forth.

These systems have proven their efficiency owing to the large number of publications which report their applications in heterogeneous domains.

### MAIN THRUST OF THE ARTICLE

The overall process and the main modules of the system will be detailed.

### Overall Enrichment Process

The semantic enrichment occurs when a user is looking for information about geographic entities. In fact, the GDB is queried at first to retrieve the data stored. Whenever the user is not satisfied by the response, the user starts up the summarization system. Thus, a mining of the online documents is triggered and will be over by returning the generated summaries.

The system relies on two kinds of agents: interface agent and summarizer ones. Every agent enjoys a simple structure independently of its type: acquaintances (the agents it knows), a local memory gathering its knowledge, a mailbox, storing the received messages that it will process later on.

The interface agent is responsible of launching the overall summarization process. It gets the set of documents resulting from an information retrieval task triggered by the user handling a map. Afterwards, it creates the summarizer agents whose number is equal to the one of the documents. These latter execute simultaneously a segmentation algorithm in order to determine the subtopics discussed along through the texts of their documents. Thus, each document is seen as nothing but a set of subtopics related to the main topic. Then, the interface agent carried a delegation task which aims to allocate to each subtopic a delegate (one of the summarizers) responsible for condensing the segments relative to a given subtopic. In fact, the summary of the corpus of documents is the merging of the partial summaries generated by the delegates. Thus, each delegate agent will extract the most salient pieces of texts retrieved from the set of segments under its jurisdiction. It considers one or all the segments and builds its, or their, Rhetorical Structure-Trees (RS-tree). The summary of a given subtopic under the responsibility of a delegate is generated by gathering the most relevant sentences.

### Segmentation

Each summarizer agent executes a segmentation algorithm in order to detect the subtopics discussed in its