International Journal of Information Management 37 (2017) 684–692

Contents lists available at ScienceDirect

International Journal of Information Management

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

Review

Textual aggregation approaches in OLAP context: A survey

Mustapha Bouakkaz[a,*], Youcef Ouinten[a], Sabine Loudcher[b], Yulia Strekalova[c]

[a] LIM Laboratory, University of Laghouat, Algeria
[b] ERIC Laboratory, University of Lyon 2, France
[c] JC College, University of Florida, USA

ARTICLE INFO

Keywords:
Aggregation
Data warehouse
OLAP
Textual data
Data mining

ABSTRACT

In the last decade, OnLine Analytical Processing (OLAP) has taken an increasingly important role as a research field. Solutions, techniques and tools have been provided for both databases and data warehouses to focus mainly on numerical data. however these solutions are not suitable for textual data. Therefore recently, there has been a huge need for new tools and approaches that treat and manipulate textual data and aggregate it as well. Textual aggregation techniques emerge as a key tool to perform textual data analysis in OLAP for decision support systems. This paper aims at providing a structured and comprehensive overview of the literature in the field of OLAP Textual Aggregation. We provide a new classification framework in which the existing textual aggregation approaches are grouped into two main classes, namely approaches based on cube structure and approaches based on text mining. We discuss and synthesize also the potential of textual similarity metrics, and we provide a recent classification of them.

1. Introduction

In many complex fields full of uncertainties, such as health, safety, security and transport, decision-makers rely on indicators and diagnostics tools to perform, validate, justify, evaluate and correct the decisions they face. Online Analytical Processing (OLAP) has emerged as a tool to assist users in the process of decision-making. The model building in OLAP is based on a multidimensional structure that facilitates the consultation and the aggregation of a given dataset. This structure represents both subjects to analyse facts and dimensions. To change the level of detail in dimensions, the OLAP process uses an aggregation function. According to Sullivan (2017), multidimensional analysis is a robust and mastered technique for numeric data warehouses. However, only 20% of corporate information system data is transactional (Tseng & Lin, 2006; Ravat & Teste, 2007), and the rest of useful information is non-additive data such as textual data that, are out of reach of OLAP processes, which makes tools and approaches proposed in OLAP unsuitable.

Data containing text have been growing very fast whereas existing aggregation functions focused mainly on numerical data and, thus, are not suitable for textual data. To use the OLAP with textual data, we need new approaches for textual aggregation, and text Mining provides the necessary techniques. text mining, which is more challenging than traditional data mining, aims to explore how users understand, organize, analyse and compare text documents. It combines the techniques from many disciplines, for example, data mining, natural language processing, artificial intelligence and, machine learning. Among the different functions addressed by researchers are information retrieval, information extraction, keywords aggregations, document categorization and, text summarization. The design and implementation of aggregation functions in text OLAP have been discussed from different perspectives, and they leverage scientific methods to assist users in the process of decision-making. In the design of an aggregation functions, many factors must be taken into account; some of them are independent of the structure of the data warehouse in which we plan to perform textual aggregation approaches. In its most general formulation, the problem of aggregation in the text OLAP context is hard because it is constrained by several requirements. The key challenges we can encounter in the textual aggregation approaches can be summarized as follows: First, textual aggregation approaches often require the help of human experts. This challenge consists of providing a high degree of automation by reducing human efforts as much as possible. Human feedback, however, may play an important role in raising the level of accuracy achieved by a textual aggregation approaches. Second, textual aggregation approaches should be able to process large volumes of data in relatively short time, because decision-makers need to perform timely analysis of environment conditions.

The goal of this paper is to provide a structured and comprehensive overview of the research in textual aggregation approaches in the OLAP context and to provide an overview of the most recent results reported

* Corresponding author.
E-mail addresses: m.bouakkaz@lagh-univ.dz (M. Bouakkaz), ouinteny@lagh-univ.dz (Y. Ouinten), sabine.loudcher@univ-lyon2.fr (S. Loudcher), yulias@ufl.ed (Y. Strekalova).

http://dx.doi.org/10.1016/j.ijinfomgt.2017.06.005
Received 9 February 2017; Received in revised form 14 April 2017; Accepted 15 June 2017

in the literature. We also propose a classification of ex-isting textual aggregation approaches to shed light on the various research directions in this field and to understand the extent to which the techniques initially applied in one particular application domain are later re-used in others. To the best of our knowledge, this is the first paper that deeply analyses textual aggregation approaches from a perspective of their application fields. We also provide a detailed discussion of techniques to perform textual aggregation approaches. We identify two main categories, i.e., approaches based on cube structure and approaches based on collection content. For each category, we first describe the basic employed techniques and then we illustrate their variants. We also show how each category addresses the problems of textual aggregation. After that, we focus on similarity metrics that exist in the literature that are strictly interconnected with textual aggregation tasks. We also discuss the potential of the assessment of the results obtained by the textual aggregation approaches, whether through human and automatic evaluation strategies.

This paper is organized into six sections. Section 1 illustrates the approaches exploited for the aggregation in a text OLAP context. Section 2 discusses the current state and provides a classification of aggregation approaches. Section 3 reviews the evaluation of textual aggregation approaches. Specifically, we identify two main ways to evaluate compared approaches. In Section 4 we describe a developed system for textual aggregation approaches and existing benchmarks to test it. Finally, in Section 5 we draw our synthesis and discuss potential applications of textual aggregation approaches that might arise in the future.

2. State of the art

In this section we adopt a different point of view, and we provide a new classification of the existing aggregation approaches. We classify contributions found in the literature into two major categories, approaches based on the data structure such as the proprieties of the data cube, and approaches that are not based on data structure. Approaches that belong to this latter category are further classified into four sub categories, approaches based on a linguistic knowledge, approaches based on external knowledge, approaches based on graph and approaches based on statistical information. Details concerning these approaches are developed next.

2.1. Approaches based on data structure

The X-OLAP (XML-OLAP) proposed by Park, Han, and Song (2005) is based on the text mining approach. In XML-OLAP, it is assumed that an XML document represents both the fact and dimension data. XML-OLAP based on the text mining technique that aggregates the text content of XML documents. This approach to analysing XML documents stored in a data warehouse, represented by a multidimensional model. In XML-OLAP, a query result returns a text cube; the content of this cube is a set of words, paragraphs or clusters. They introduce an expression called XPath to define the text segment of the XML document in order to analyse it. The approach uses for aggregation some text mining functions such as k-means and frequent pattern, the patterns are the top keywords. Therefore, XML-OLAP presents the benefit of text mining technology for Text OLAP analysis. There are several other works on the methods of efficiently performing text mining algorithms for the analysis of text documents (Ravat & Teste, 2007; Ravat, Teste, & Tournier, 2008).

The DocCube was introduced by Mothe, Chrisment, Dousset, and Alaux (2003). It is used to examine and envisage the whole document in a corpus using the classification approach. It treats several facts of a document as dimensions. These dimension tables are similar to the standard of OLAP systems. Nevertheless, the major characteristic of DocCube lies in the nature of the content of a fact table that contains links; A link is established between a document and a fact row. The document represented by the dimension values that serve as the document identifier or the document URL. These links are defined by their weights according to the degree of confidence of the association (Doc, Ref). The multidimensional visualization provides in DocCube gives a user a possibility to know the relatedness among documents and gives a direct access to explore document content. By exploring the dimensions, the user can view the distribution of the documents according to their URL and can manipulate the level of aggregation for visualization. At any moment, a user can have direct access via the links to the documents associated with selected dimension values.

The Topic Cube: The analysis of text using OLAP must support drill-down or roll-up if we want to analyse a text data on a topic dimension. Zhang, Zhai, and Han (2009) proposed an approach called Topic Cube, the main idea of a topic cube is to use the hierarchical topic tree as the hierarchy for the text dimension. This structure allows a user to drill-down and roll-up along this tree and discover the content of the text documents in order to view the different granularities and levels of topics in the cube. The first level in the tree contains the detail of topics, the second level is more general types and the last level contains the aggregation of all topics. A textual measure is needed to aggregate a textual data. The authors proposed two types of textual measures, word distribution and topic coverage. The topic coverage computes the probability that a document contains the topic. These measures allow users to know which topic is dominant in the set of documents by aggregating the coverage over the corpus. The perspectives of Zhang et al. (2009), are realized in a new extension called iNextCube (Information Network-Enhanced Text Cube) proposed by Yu, Lin, Sun, and Chen (2009). This extension (iNextCube) constructs the topic hierarchy automatically.

The Document Cube: Tseng and Chou (2006) proposed an approach for multidimensional analysis of scientific documents. Many data extracted from the scientific articles are used as dimensional data, such as, the keywords, names of authors, title, name of conference or journal and publication date. However, there is no clear explanation how the keywords and the metadata are structured in a hierarchical order. Tseng and Chou (2006) propose a textual measure, associate each document with an identifier and a number of similar documents in order to facilitate rolling up, drilling down and navigation in the different granularities and perspectives. A query results is a text cube, where cells contain the identifiers of corresponding documents stored in the corpus. A new extension for Document cube is proposed by Tseng and Lin (2006). provides a new query language specially designed for the document cube called MD2X (MultiDimensional Document eXpression).

The Text Cube: In order to introduce the semantic aspect in the textual aggregation Lin, Ding, Han, and Zhu, (2008), proposed an approach for data cube called text cube. The main idea is to give the user the possibility to make a semantic navigation in data dimension. To achieve that, two OLAP operations such as the pull-up and push-down. They proposed also two metrics based on information retrieval and which represent term frequency and inverted index. To specify the semantic level in the text cube, they proposed a hierarchy where the extracted keywords represent the nodes at the base level, the ancestor nodes at upper level are more general than children at lower level, and the nodes at toper level contain terms of the corpus. The use of textual measures pull-up or push down facilitates the navigation in the hierarchy. Thus the measures, term frequency and inverted index are used for aggregated text data.

The Tube: Lauw, Lim, and Pang (1998) proposed an approach called TUBE (Text-cUBE) to discover associations among entities. The model adopts a concept similar to data cube designed for relational databases which is applied to textual data, where cells contain keywords, and an interestingness value is attached to each keyword.

The R-Cube: Perez, Berlanga, and Aramburu (2007), Perez, Berlanga, and Aramburu (2008a), Perez, Berlanga, and Aramburu (2008b), Perez, Berlanga, and Aramburu (2008c); focus on the task of

integrating structured and textual data in the same data warehouse. The authors proposed an architecture for a decision support system called contextualized warehouse that, allows a user to obtain knowledge from heterogeneous data and documents by analyzing data under different contexts. A collection of text documents is considered as a context, which can be used to analyse and exploit keywords extracted from the content to facilitate decision making tasks. Based on the variability of data, users might specify the analytical context by providing a list of keywords, and then an R-cube (Relevance Cube) retrieves the documents and the facts related to the selected context. In R-cube the fact is linked to the contexts, and has a dimension value corresponding to the relevance with respect to the specified context. The construction of an R-cube starts with the evaluation of a document warehouse and the result is a set of documents. Second, selected facts are described by each document according to their frequency. Then, each document is assigned to those facts of the corporate data warehouse whose dimension values can be rolled-up or drilled-down. Finally, the relevance value of each fact is calculated.

The Cube Index: Azabou, Khrouf, Soule-Dupuy, and Valles (2015) proposed a model called Cube Index based on a hierarchical description of each document. This hierarchy specifies relationships between words with respect to one document. It is used for the analysis of words in various levels of abstraction in a document. They introduce two operations scroll up and scroll down (inverse operator of scroll up.) It supports Tf*Idf (Term Frequency-Inverse Document Frequency) to facilitate information retrieval techniques.

2.2. Approaches based on content

The approaches that, describe document warehousing through the most representative keywords without using the structure of data or the proprieties of cube, found in the literature can be classified into four categories. The first one is based on linguistic knowledge, the second one is based on the use of external knowledge, the third one is based on graphs, and the last uses statistical methods.

The approaches based on linguistic knowledge consider a corpus as a set of the vocabulary mentioned in the documents, but the results in this case are sometimes ambiguous. To overcome this obstacle, techniques based on lexical knowledge and syntactic knowledge previews have been introduced. Kohomban and Lee (2007), Poudat, Cleuziou, and Clavier (2006) described a classification of textual documents based on scientific lexical variables of discourse. Among these lexical variables, they chose nouns because they are more likely to emphasize the scientific concepts, rather than adverbs, verbs or adjectives.

The approaches based on the use of external knowledge select certain keywords that represent a domain. These approaches often use models of knowledge such as ontology. Ravat and Teste (2007) proposed an aggregation function that takes as input a set of keywords extracted from documents of a corpus and outputs another set of aggregated keywords. They assumed that both the ontology and the corpus of documents belong to the same domain. Oukid, Asfari, and Bentayeb (2015) proposed an aggregation operator Orank (OLAP rank) that aggregates a set of documents by ranking them in a descending order using a vector space representation. Mukherjee and Joshi (2014) propose a textual aggregation model using ontology. They propose an approach to construct keywords Ontology Tree and aggregate keywords by their ancestors.

The approaches based on graphs use keywords to construct graphs, where each node represents a keyword obtained after preprocessing and candidate selection. An edge represents the strength or relatedness (or semantic relatedness) between two keywords. After the graph representation step, different types of keywords-ranking approaches have been tried. The first proposed is an approach called TextRank (Mihalcea & Tarau, 2017), where, the edges represent the cooccurrence relations between the keywords. The idea of this approach is that if a keyword is linked to a large number of other keywords, it is considered as important (Mihalcea & Tarau, 2017). It constructs the term graph, in which the links between terms reflect their semantic relatedness and are calculated by the term co-occurrences in the corpus. It is based on the PageRank algorithm to obtain the PageRank score for each two terms to rank a candidate. TextRank, extracts high-frequency terms as keywords because these terms have more opportunities to get linked with other terms and obtain higher PageRank scores. Moreover, TextRank usually constructs a term graph using term co-occurrences as an approximation of the semantic relations between words. The graph introduces much noise due to the connection of unrelated words which influence the extraction performance. Other approaches were based on TextRank in order to improve it, such as ExpandRank (Wan & Xiao, 2008) which uses a small number of neighbour documents to provide more information of term relatedness for the building of term graphs. Another potential approach to alleviate vocabulary gap is latent topic models. Latent topic models learn topics from a collection of documents. Using a topic model, we can represent both documents and terms as the distributions over latent topics. The semantic relatedness between a term and a document can be estimated using the similarities of their topic distributions. The similarity scores can be used as the ranking criterion for keywords extraction (Blei & Lafferty, 2006).

Textual aggregation by graph (TAG): Bouakkaz, Loudcher, and Ouinten (2014) proposed a method which performs aggregation of keywords of documents based on the construction of graph using the affinities between keywords, and the construction of cycles on the graph. This function produces the main aggregated keywords out of a set of terms representing a corpus. Their aggregation approach is called TAG (Textual Aggregation by Graph). It aims at extracting from a set of terms a set of the most representative keywords for the corpus of textual document using a graph. The function takes as input the set of all extracted terms from a corpus, and outputs an ordered set, containing the aggregated keywords. The process of aggregation goes through three steps: (1) Extraction of keywords with their frequencies, (2) Construction of the affinity matrix and the affinity graph, and (3) Cycle construction and aggregated keywords selection. Also Bouakkaz, Loudcher, and Ouinten (2016) proposed a method based on a data mining technique called k-means with a new distance measure which is the Google Similarity Distance, in order to find the semantic aggregation of the keywords.

The approaches based on statistical methods use the occurrence frequencies of terms and the correlation between terms. Landauer, Foltz, and Laham (1998) proposed a method called the Latent Semantic Analysis (LSA) in which the corpus is represented by a matrix where the rows represent the documents and the columns represent the keywords. An element of the matrix represents the number of occurrences of a word in a document. After decomposition and reduction, this method provides a set of keywords that represent the corpus. Bringay, Bchet, Bouillot, and Poncelet (2011) proposed two aggregation functions. The first one is based on a new adaptive measure of Tf.Idf (Term Frequency-Inverse Document Frequency) which takes into account the hierarchies associated to the dimensions. The second one is build dynamically and is based on clustering. Wartena and Brussee (2008) used the k-bisecting clustering algorithm based on the Jensen-Shannon divergence of probability distributions described in Fuglede and Topsoe (2004). Their method starts by selecting two elements that are far apart as the seeds of the two first clusters. Each one of the other elements is then assigned to the cluster of the closest seed. Once all the elements have been assigned to clusters, the centres of both clusters are computed. The new centres are used as new seeds for finding two new clusters and the process is repeated until each of the two new centres converge up to some predefined precision. If the diameter of a cluster is larger than a specified threshold value, the whole procedure is applied recursively to that cluster. Ravat et al. (2008) proposed a second aggregation function called TOP-Keywords to aggregate keywords. They computed the

frequencies of terms using the *T f.Idf* function, and then they selected the first k most frequent terms. The C-Value algorithm, which creates a ranking for potential keywords uses the length of the phrases which contain keywords and their frequencies (Frantzi, Ananiadou, & Mima, 2004). El-Ghannam and El-Shishtawy (2014) proposed a technique for extracting summary sentences for a set of documents using the weight of the sentences and the documents.

3. Evaluation

Evaluating aggregated keywords is a difficult task because there is no ideal tool or metric to measure the quality of results obtained by aggregation functions. From papers surveyed in the previous sections and elsewhere in the literature, it has been found that agreement between human keywords aggregation is quite low, both for evaluating and generating aggregated keywords. Another important problem in aggregated keywords evaluation is the widespread use of disparate metrics. The absence of a standard human or automatic evaluation metric makes it very hard to compare different approaches and establish a baseline. further, a manual evaluation is too expensive. Hence, an evaluation metric having high correlation with human scores would obviate the process of manual evaluation. In this section, we look at some important recent papers that have been able to create standards in the evaluation of keywords aggregation.

3.1. Human evaluation

Human evaluation is subject to specific guidelines given to the human judgement when performing the evaluation task, the human judgement can easily decide whether the returned aggregated keywords are good representatives of a corpus' content or not. Thus, manual evaluation is not restricted to exact matches between gold standard aggregated keywords and aggregated keywords returned by an approach. Human evaluation has been suggested as a possibility (Matsuo & Ishizuka, 2004) but is time consuming and expensive, as stated by Lin (2004). As an example, large-scale manual evaluation of keywords as in the DUC (Document Understanding Conference) would require over 3000 h of human effort (Lin, 2004). Manual evaluation of aggregated keywords is very costly and time-consuming. In particular, it is not suited for any kind of parameter tuning, as the output of each new system configuration involves manual re-evaluation.

3.2. Automatic evaluation

Automatic evaluation is assessing the performance of the aggregated keywords using metrics or tools. Many type of metrics have been proposed in (Jones, 1997; Tague-Sutcliffe, 1992; Voorhees & Harman, 2005), but the most used are the recall, the precision, and the F-measure. The recall is the ratio of the number of documents to the total number of retrieved documents. The precision is the ratio of the number of relevant documents to the total number of retrieved documents. The F-measure or balanced F-score, which combines precision and recall, is the harmonic mean of precision and recall. The performance of aggregated keywords is measured in terms of the overall performance of the application. However, this entails the influence of parameters besides the keywords aggregation algorithm to be tested. For example, Bracewell, Ren, and Kuriowa (2005) use the information retrieval task of keyword search to determine the effectiveness of keywords by describing the document from which they were extracted. However, this method might extract aggregated keyword sets that are good indicators for relevant documents but that are not acceptable when presented to humans. Hulth (2004) use a summary-based evaluation, where an aggregated keyword is used as a gold standard and indicator if it appears in the document and in the summary.

4. Textual benchmarks and systems

4.1. The software systems

The software system developed in this domain consists of two main components: text pre-processor and features extractor. Text pre-processor offers learning and inference functionalities. The learning functionality pre-processes a document collection by exploiting a stop words list and a keywords list to obtain the word-document matrix according to the bag-of-words model. The user can choose the number of words to be used for document indexing. The inference functionality processes a document to obtain the following bag-of-words representations; binary, term frequencies and the inverse term document frequency.

OpAC (Operator for Aggregation by Clustering) is proposed for online analysis (BenMessaoud & Rabasda, 2004). The developers think that the fact of coupling OLAP and data mining will achieves interesting results. The main idea of OpAC consists in using the agglomerative hierarchical clustering to achieve a semantic aggregation on the attributes of a data cube dimension.

Topic extractor implements a customized version of the Latent Dirichlet Allocation (LDA) model (Blei & Jordan, 2003). The solution of the LDA learning is obtained by using the Expected Maximization and the Gibbs Sampling algorithms which have been implemented in the C + + programming language on a single processor machine. Each topic is summarized through the estimate of its prior probability, a sorted list of its most frequent words together with the estimate of their conditional probabilities is produced.

Semantria[1] is a text analytical tool that offers an API that performs sentiment analysis and analytic text. Users can be integrated in the service to quickly yield actionable data from their unstructured text data, from review sites, blogs, or other sources.

Bhide, Chakravarthy, Gupta, and Gupta (2008), Chakaravarthy, Gupta, and Roy (2006), proposed a system called **EROCS** (Entity RecOgnition in Context of Structured data) which links structured data in a data warehouse with a given text document. Their system considers the structured data in a data warehouse as a group of entities, and selects the pertinent entities that is best related to the given document. EROCS establishes the links between entities and a segment within the given documents. It can also, predict a link even when the entity is not mentioned in the document. EROCS takes as input a corpus of documents, and applies a text process to filter and retain only the pertinent keyword. These keywords are considered as the entities stocked in a data base, and they are associated with their context information. The context information is used to specify the location of each entity in each document.

KEEL: Knowledge Extraction based on Evolutionary Learning, is an open source Java software tool proposed by Alcala-Fdez, Sanchez, and Garcia (2009), Alcal, Fernndez, Luengo, and Derrac (2010) assess evolutionary algorithms for Data Mining problems including extraction, regression, classification, clustering, pattern mining and so on. It contains a big collection of classical knowledge extraction algorithms, preprocessing techniques (training set selection, feature selection and extraction, imputation methods for missing values), computational intelligence based learning algorithms and hybrid models such as genetic fuzzy systems. It allows users to perform a complete analysis of any learning model in comparison to existing ones, including a statistical test module for comparison.

Other tools have been created for searching, classifying and retrieving textual information. Examples include Signature (LET Centre, 2017), Word Cruncher (AtlasTi, 2017), Word Smith Tools (Scott, 2015), Intext (Intelligent Systems, 2015) and WoW (Keith, Kaser, & Lemire, 2017). Steven Keith et al. proposed the creation of user-driven tools to interface with a Data Warehouse of Words (WoW). A WoW is built by

[1] https://semantria.com/.

an Extraction, Transformation, and Loading (ETL) procedure, which processes the text and aggregates data from different sources. A WoW stores its data in data cubes and allows to the evaluation of the aggregation of queries across several dimensions and at different level of granularity. These queries generally take advantage of the hierarchical nature of cube dimensions.

4.2. Existing textual benchmarks

There are several publicly available benchmarks for testing textual aggregation approaches as follows:

WordNet: (Miller, 1995) is a large lexical database of English. Nouns, verbs, adjectives and adverbs are grouped into sets of cognitive synonyms (synsets), each expressing a distinct concept. Synsets are interlinked by means of conceptual-semantic and lexical relations. The resulting network of meaningfully related words and concepts can be navigated with the browser. WordNet is also freely and publicly available for download. WordNet's structure makes it a useful tool for computational linguistics and natural language processing. WordNet superficially resembles a thesaurus, in that it groups words together based on their meanings. However, there are some important distinctions. First, WordNet interlinks specific senses of words rather then just word forms. Second, unlike thesaurus which do not follow any explicit pattern other than meaning similarity in the grouping of words, WordNet labels the semantic relations among words.

DUC: (Wan & Xiao, 2008) The Document Understanding Conference (DUC) is a series of summarization evaluations that have been conducted by the National Institute of Standards and Technology (NIST) since 2001. Its goal is to further progress in automatic text summarization and aggregation. It enables researchers to participate in large-scale experiments in both the development and evaluation of summarization and aggregation systems.

TREC: (Wan & Xiao, 2008) contain a series of textual benchmarks published in the Text REtrieval Conference (TREC) co-sponsored by the National Institute of Standards and Technology (NIST) and U.S. Department of Defense. It was started in 1992 as part of the TIPSTER Text program. Its purpose was to support research within the information retrieval community by providing the infrastructure necessary for large-scale evaluation of text retrieval methodologies.

MeSH: (Wan & Xiao, 2008) is an on line vocabulary look-up aid available for use with (Medical Subject Headings). It is designed to quickly locate descriptors of possible interest and to show the hierarchy in which descriptors of interest appear. Virtually complete MeSH records are available, including the scope notes, annotations, entry vocabulary, history notes, allowable qualifiers, etc.

Tweets: Twitter provided identifiers for approximately 16 million tweets sampled between January 23rd and February 8th, 2011. The corpus is designed to be a reusable, representative sample of the twittersphere in which both important and spam tweets are included. The tweets corpus is unusual in that what you get is a list of tweet identifiers, and the actual tweets are downloaded directly from Twitter, using the open-source twitter-tools.

Reuters: Currently the most widely used test collection for text analyses research, though it is likely to be superseded over the next few years by RCV1. The data were originally collected and labelled by Carnegie Group, Inc. and Reuters, Ltd. in the course of developing the CONSTRUE text categorization system. The Reuters collection is distributed in 22 files. Each of the first 21 files contains 1000 documents, while the last contains 578 documents.

The British National Corpus (BNC) is a 100 million word collection of samples from a wide range of sources, designed to represent a wide cross-section of British English from the later part of the 20th century. The BNC includes words extracted from regional and national newspapers, specialist periodicals and journals for all ages and interests, academic books and popular fiction, published and unpublished letters and memoranda and school and university essays, among many other kinds of text.

The corpora listed above represent the most popular benchmarks for testing textual aggregation used to compare the performance of different approaches proposed in the literature.

5. Experimental study

5.1. Textual benchmark

In this work we compiled two corpora, the first corpus is from the *IIT* conference[2] (conference and workshop papers) for the years 2008–2014. It consists of 700 papers ranging from 7 to 8 pages in IEEE format, including tables and figures. The second corpus called Ohsumed collection[3] which includes medical reports from the MeSH categories and It consists of 20,000 documents.

The keywords are extracted from the full words using Microsoft Academic Search[4] keywords. The keywords extraction function is based on the Microsoft Academic Search web site (*MAS*). MAS classifies scientific articles according to fifteen scientific fields by extracting the scientific keywords from articles and ordering them according to their frequencies. We use the lists of keywords produced by MAS and we choose 2000 most frequent keywords form each field as shown in Fig. 1.

The extraction of keywords from the two corpora is performed according to these chosen lists. The output of this process is the two fold matrix of *Documents* x *Keywords*, which is used to compare GOTA approach and the other textual aggregation approaches.

For the evaluation task of the keywords aggregation, many type of measures have been proposed in (Gomaa & Fahmy, 2013; Gupta et al., 2009). But the most used are the recall, the precision, and the F-measure. The recall is the ratio of the number of documents to the total number of retrieved documents.

$$\text{Recall} = \frac{|\{RelevantDoc\} \cap \{RetrievedDoc\}|}{|\{RelevantDoc\}|} \quad (1)$$

The precision is the ratio of the number of relevant documents to the total number of retrieved documents.

$$\text{Precision} = \frac{|\{RelevantDoc\} \cap \{RetrievedDoc\}|}{|\{RetrievedDoc\}|} \quad (2)$$

The F-measure or balanced F-score, which combines precision and recall, is the harmonic mean of precision and recall.

5.2. Results

In this section, we report an empirical study to evaluate the aggregated keyword functions (Bouakkaz et al., 2016; Bringay et al., 2011; Lauw et al., 1998; Ravat et al., 2008; Wartena & Brussee, 2008) using two real corpora. We also compare there performances.

The experimentation has been performed on a PC running the Microsoft Windows 7 Edition operating system, with a 2.62 GHz Pentium Dual-core CPU, 1.0 GB main memory, and a 300 GB hard disk. To test and compare the different approaches we have compiled two real corpora as mentioned in Section 4.1, with 600 articles, 800,000 words and 2182 keywords extracted for the first corpus and 20,000 articles, 1,300,000 words and 985 keywords extracted for the second corpus.

To perform this comparison, we use four evaluation metrics: recall, precision, F-measure and the run time for different values of k (number of aggregated keywords). We also give a comparison of the complexity for the five algorithms. The results are summarized in Figs. 2–9.

Overall, the approach GOTA produces highest values of the recall,

[2] http://www.it-innovations.ae.
[3] ftp://medir.ohsu.edu/pub/ohsumed.
[4] academic.research.microsoft.com/.

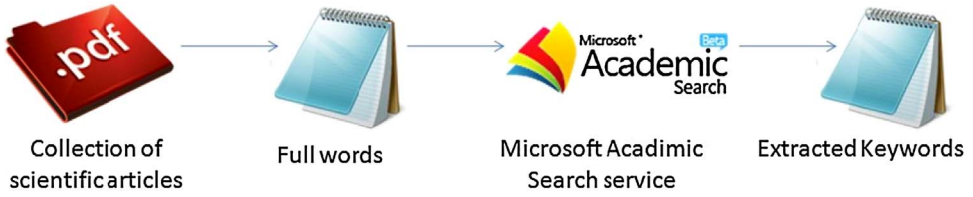

Fig. 1. Steps of keywords' extraction.

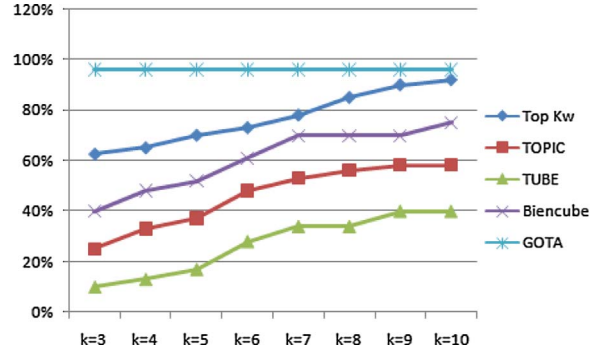

Fig. 2. Comparison between the recall – First corpus.

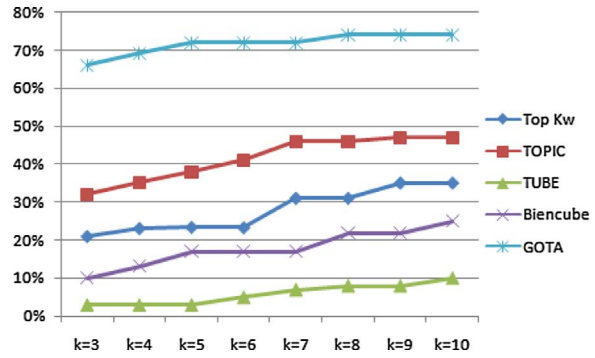

Fig. 3. Comparison between the Precession – First corpus.

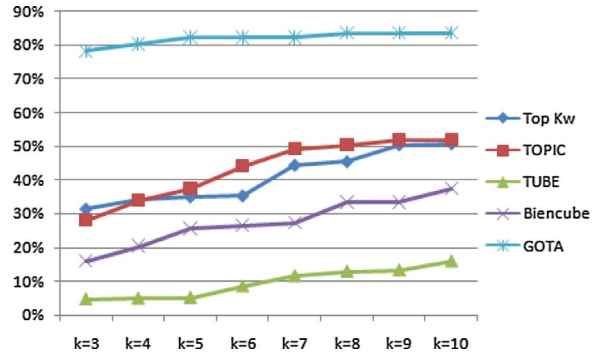

Fig. 4. Comparison between the F-mesure – First corpus.

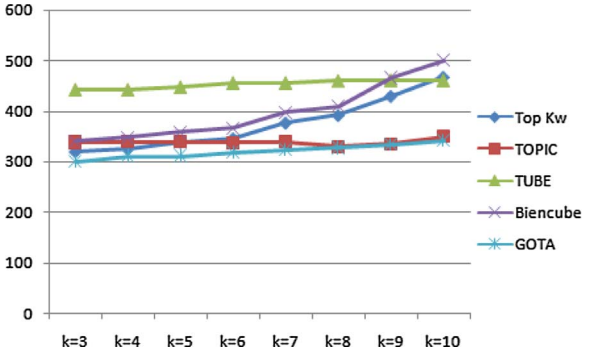

Fig. 5. Comparison between the Runtime – First corpus.

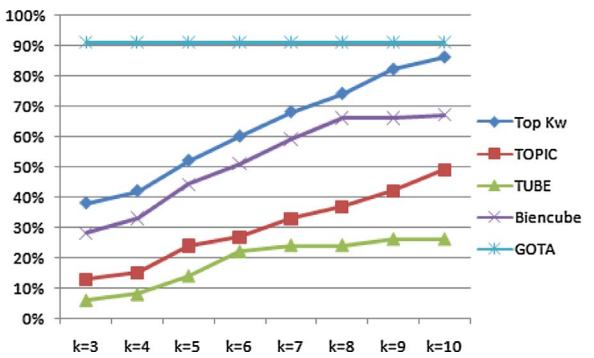

Fig. 6. Comparison between the recall – Second corpus.

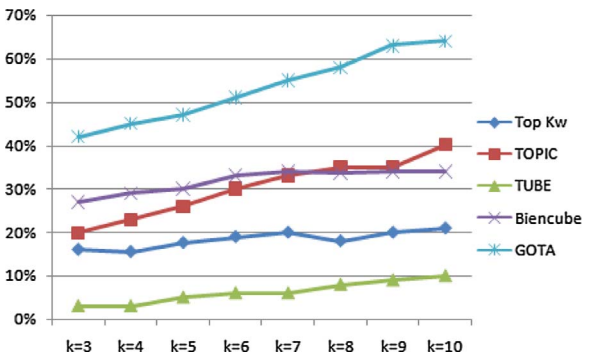

Fig. 7. Comparison between the Precession – Second corpus.

the precision and F-measure. For instance, in the case of k = 3, we obtained a recall of 96% compared with 63%, 25%, 40% and 10% obtained by Topkeyword, TOPIC, BienCube and TuBE respectively. We also obtained a precession of 66% compared with 21%, 32%, 10% and 3% obtained by Topkeyword, TOPIC, BienCube and TuBE respectively. As for the F-measure, we obtained for GOTA a value of 78% compared with 31%, 28%, 16% and 5% obtained by Topkeyword, TOPIC, BienCube and TuBE respectively. In the case of k = 10, the value we obtained a recall of 96% is to be compared with 92%, 58%, 75% and 40% obtained by Topkeyword, TOPIC, BienCube and TuBE respectively. The precession obtained of 74% for GOTA is compared with 35%, 47%, 25% and 10% obtained by Topkeyword, TOPIC, BienCube and TuBE respectively. As for the F-measure, the value of 84% obtained by GOTA is compared with 51%, 52%, 38% and 16% obtained by Topkeyword, TOPIC, BienCube and TuBE respectively. In order to determine the runtime for each approach, we carried out 10 executions of each approach and for each value of *k*.

The results obtained from the second test using a larger corpus confirm the results obtained in the first test. we note that the approach GOTA achieves better performance compared to the other approaches.

The difference between the five approaches is highly noticeable in (Figs. 6–9). This is due to the difference in the complexities of the five approaches. The approach GOTA is based on k-means which has a complexity of $O(N)$. the same thing with Topkeyword and BienCube which have a complexity of $O(N)$ (Bringay et al., 2011; Ravat et al., 2008). On the other hand TOPIC is based on the k-bisecting clustering

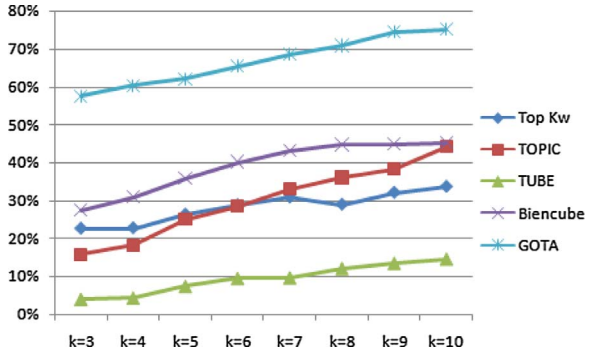

Fig. 8. Comparison between the F-mesure – Second corpus.

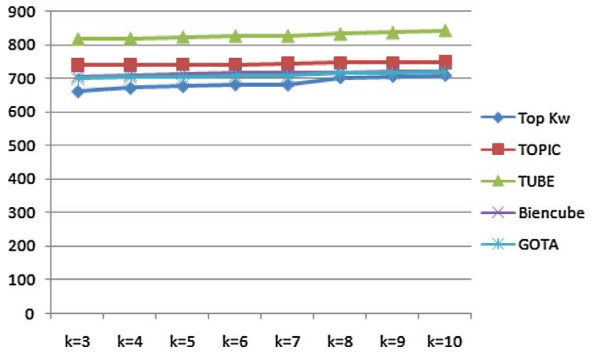

Fig. 9. Comparison between the Runtime – Second corpus.

which has a complexity of $O((k-1)kN)$. where k is the number of clusters and N the number of terms (Wartena & Brussee, 2008). for TUBE the complexity is $O(N^2)$ (Lauw et al., 1998).

6. Synthesis and future directions

A data warehouse contains a large amount of unstructured data such as textual information. The need for tools to exploit this information urged researchers to develop and implement various approaches to accomplish the task of an automatic aggregation of data in text OLAP in order to assist users in the process of decision making. Textual aggregation approaches have had (and continue to have) a wide range of applications in a number of fields, ranging from commercial to social web applications. The main goal of this survey was to classify existing approaches for textual aggregation in terms of the applications for which they have been employed. In the first part of this survey, we provided a classification of approaches used to aggregate textual data from a collection of documents and present several basic techniques. Then, we focused on how to measure the similarity between aggregated keywords and how to evaluate each approach. We provided different perspectives to classify textual aggregation approaches (like the use of a cube or the possibility of using linguistic knowledge or data mining techniques). Tables 1 and 2 summarize the 23 discussed textual aggregation approaches presented in this paper.

In conclusion we present some possible future applications of textual aggregation approaches.

6.1. Medical and scientific computing

the application of textual aggregation in the medical field is growing. There are many data warehouses for medical sources, in particular textual documents such as analyses, diagnosis in bio-chemistry and genetics. The application of textual aggregation can highly contribute to the extraction of knowledge from these data warehouses.

Table 1
Comparison of the reviewed works according to the tool and benchmark used.

Ref.	Tool	Benchmark	comparative study
Frantzi et al. (2004)	C/NC-value	Eye-pathology Corpus	Yes
Mothe et al. (2003)	DocCube	MeSH	
Mihalcea and Tarau (2017)	TextRank	Inspec/DUC	Yes
Park et al. (2005)	XML-MDX	U.S. Patent XML	
Tseng and Chou (2006)	Document Cube	E-mails	
Poudat et al. (2006)	SVM	LING-corpus	Yes
Perez et al. (2007)	R-Cube	News papers	
Kohomban and Lee (2007)	Word Sense	Euro WordNet	
Ravat and Teste (2007)			
Lauw et al. (1998)	IdentiFinder	TKBs site files	
Lin et al. (2008)	GreedySelect	Dell	Yes
Wan and Xiao (2008)	ExpandRank	DUC/TREC	Yes
Wartena and Brussee (2008)	Topic	Wikipedia	Yes
Ravat et al. (2008)			
Zhang et al. (2009)	Topic Cube	ASRS	Yes
Yu et al. (2009)	iNextCube	DBLP	
Bringay et al. (2011)	PostgreSQL	Tweets	
El-Ghannam and El-Shishtawy (2014)	ROUGE-S	TAC2011	Yes
Mukherjee and Joshi (2014)	PASOT	IMDB movie review	Yes
Bouakkaz et al. (2014)	OLAP-SKEA	IT-Innovation Articles	Yes
Oukid et al. (2015)	Orank		
Azabou et al. (2015)	CubeIndex		
Bouakkaz et al. (2016)	GOTA	MeSH	Yes

6.2. Customer care

Frequently companies that offer customers support handle large volumes of complex data including text. Relevant examples are emails, forum discussions, documentation and credit card transfer reports. The ability to analyse these documents and aggregate textual content is associated with several advantages. First, documents can be classified in more effective categories to make their retrieval easier. In addition, once the concepts present in a collection of documents have been aggregated, it is possible to identify relevant associations between documents on the basis of the concepts they share.

6.3. Opinion mining

This is related to opinion sharing and its evolution among users. In this case, users express opinions on products, experiences, services they enjoyed, etc. The most common form of opinion sharing is represented by blogs, comments, tags, polls, charts, etc. This information is often unstructured and aggregation is a challenge.

6.4. Social networks

Social Web platforms emerged as one of the most noticeable phenomenon on the Web. These platforms are built around users, offering them the possibility to create virtual links. User interactions generate a textual content that can be used to answer questions like: How human relationships are created according to their content? How can novel ideas be aggregated and spread through out the social network?

All in all, this survey explains in depth the main techniques based on both cube structure and text mining that exist in the OLAP textual aggregation literature. We tried to give a full overview of the standard ways to evaluate the performance of the proposed methods, as well as, we end with a detailed comparison of the previous existing approaches.

Table 2
Comparison of the reviewed works according to the used type of approach.

Ref.	Approach	Cube	RI/Statistic	DM	Linguistic	E-Knowledge	Graph
Frantzi et al. (2004)	C/NC-value		Frequancy		Tagging		
Mothe et al. (2003)	DocCube	X		Classification			
Mukherjee and Joshi (2014)	TextRank		Weight graph				Graph Ranking
Park et al. (2005)	XML-OLAP	X		k-means			
Tseng and Chou (2006)	Document Cube	X	D-Tree				
Azabou et al. (2015)	Descriptors				Lexical		
Lauw et al. (1998)	R-Cube	X	Relevance value				
Poudat et al. (2006)	WSD				Disambiguation		
Ravat and Teste (2007)	OLAP Ontology					Ontology	
Landauer et al. (1998)	TUBE		Association				
Lin et al. (2008)	Text Cube	X	Term hierarchy				
Mihalcea and Tarau (2017)	ExpandRank						Neighborhood
Wartena and Brussee (2008)	Topic		Jensen-Shan divergence	Bisecting k-means			
Ravat et al. (2008)	Top-Keywords		Tf-Idf				
Zhang et al. (2009)	TopicCube	X	Probability				
Yu et al. (2009)	iNextCube	X	Probability				
Bringay et al. (2011)	Biencube	X	Tf-Idf				
El-Ghannam and El-Shishtawy (2014)	Sen-Rich		weigh documents				
Oukid et al., (2015)	PASOT					Ontology	
Blei and Lafferty (2006)	TAG		Affinity				Cycles
Kohomban and Lee (2007)	CXT-Cube	X				Ontology	
Perez et al. (2008b)	Diamond	X	Tf-Idf				
Bouakkaz et al. (2016)	GOTA		Google Dist	k-means			

References

Alcal, J., Fernndez, A., Luengo, J., & Derrac, J. (2010). Keel: A software tool to assess evolutionary algorithms for data mining problems. *Journal of Multiple-Valued Logic and Soft Computing*, 255–287.

Alcala-Fdez, J., Sanchez, L., & Garcia, S. (2009). Enhanced business intelligence using erocs. *Journal of Multiple-Valued Logic and Soft Computing*, 307–318.

AtlasTi, Scientific software development, http://www.atlasti.de/.

Azabou, M., Khrouf, F. J., Soule-Dupuy, K. C., & Valles, N. (2015). Diamon multi-dimensional model and aggregation operators for document olap proceedings. *Decision Support Systems*, 363–373.

BenMessaoud, R., & Rabasda, S. (2004). Opac: A new olap operator based on a data mining method. *Databases and Information Systems*, 417–420.

Bhide, M., Chakravarthy, V., Gupta, A., & Gupta, H. (2008). Enhanced business intelligence using erocs. *IEEE 24th international conference on data engineering*, 1616–1619.

Blei, M., & Jordan, M. (2003). Latent dirichlet allocation. *Journal of Machine Learning Research*, 993–1022.

Blei, D., & Lafferty, J. (2006). Dynamic topic models. *23rd international conference on Machine learning*, 113–120.

Bouakkaz, M., Loudcher, S., & Ouinten, Y. (2014). Automatic textual aggregation approach of scientific articles in olap context. *Innovations in Information Technology*, 30–35.

Bouakkaz, M., Loudcher, S., & Ouinten, Y. (2016). Olap textual aggregation approach using the google similarity distance. *International Journal of Business Intelligence and Data Mining*, *11*(1), 31–48.

Bracewell, D., Ren, F., & Kuriowa, S. (2005). *Multilingual single document keyword extraction for information retrieval. Natural language processing and knowledge engineering*, 517–522.

Bringay, S., Bchet, N., Bouillot, F., & Poncelet, P. (2011). Towards an on-line analysis of tweets processing. *Database and Expert Systems Applications*, 154–161.

Chakaravarthy, V., Gupta, H., & Roy, P. (2006). Efficiently linking text documents with relevant structured information. *32nd international conference on very large data bases*, 667–678.

El-Ghannam, F., El-Shishtawy, T., Multi-topic multi-document summarizer, arXiv preprint (2014) 1401–0640.

Frantzi, K., Ananiadou, S., & Mima, H. (2004). Automatic recognition of multi-word terms: The c-value/nc-value method. *IEEE international symposium on information theory*, 31–38.

Fuglede, B., & Topsoe, F. (2004). Jensen-shannon divergence and hilbert space embedding. *International Journal on Digital Libraries*, 31–38.

Gomaa, W. H., & Fahmy, A. A. (2013). A survey of text similarity approaches. *International Journal of Computer Applications*, *68*(13).

Gupta, V., Lehal, G. S., et al. (2009). A survey of text mining techniques and applications. *Journal of Emerging Technologies in Web Intelligence*, *1*(1), 60–76.

Hulth, A. (2004). Enhancing linguistically oriented automatic keyword extraction. *Proceedings of HLT-NAACL*, 17–20.

Intelligent Systems (2015). *Intelligent internet tools*. Oxford University Presshttp://www.intext.com/.

Jones, Karen Sparck (Ed.), (1997). *Readings in information retrieval*. Morgan Kaufmann.

Keith, S., Kaser, O., Lemire, D., Analyzing large collections of electronic text using olap, arXiv preprint cs/0605127.

Kohomban, U., & Lee, W. (2007). Optimizing classifier performance in word sense disambiguation by redefining word sense classes. *International Joint Conference on Artificial Intelligence*, 1635–1640.

Landauer, T., Foltz, P., & Laham, D. (1998). An introduction to latent semantic analysis. *Discourse Processes*, 259–284.

Lauw, H., Lim, E., & Pang, H. (1998). Tube (text-cube) for discovering documentary evidence of associations among entities. *2007 ACM symposium on applied computing*, 259–284.

LET Centre, The signature textual analysis system, http://www.etext.leeds.ac.uk/signature/.

Lin, C., Ding, B., Han, J., & Zhu, F. (2008). *Text cube: Computing ir measures for multi-dimensional text database analysis*. ICDM905–910.

Lin, C. (2004). Rouge: A package for automatic evaluation of summaries. *Text summarization branches out: Proceedings of the ACL-04 workshop*, 74–81.

Matsuo, Y., & Ishizuka, M. (2004). Keyword extraction from a single document using word co-occurrence statistical information. *International Journal on Artificial Intelligence Tools*, 157–169.

Mihalcea, R., Tarau, P., *Textrank: Bringing order into texts*, Association for Computational Linguistics.

Miller, G. (1995). Wordnet: A lexical database for english. *AAAI*, 39–41.

Mothe, J., Chrisment, C., Dousset, B., & Alaux, J. (2003). Doccube: Multidimensional visualisation and exploration of large document sets. *Journal of the American Society for Information Science and Technology*, 650–659.

Mukherjee, S., & Joshi, S. (2014). Author-specific sentiment aggregation for polarity prediction of reviews. *Ninth international conference on language resources and evaluation*, 3092–3099.

Oukid, L., Asfari, F., & Bentayeb, N. (2015). Contextualized text olap based on information retrieval. *International Journal of Data Warehousing and Mining*, 1–21.

Park, B., Han, H., & Song, I. (2005). *Xml-olap: A multidimensional analysis framework for xml warehouses in data warehousing and knowledge discovery*. Berlin, Heidelberg: Springer, 32–42.

Perez, J., Berlanga, R., & Aramburu, M. (2007). *R-cubes: Olap cubes contextualized with documents*. ICDE1477–1478.

Perez, J., Berlanga, R., & Aramburu, M. (2008a). Integrating data warehouses with web data: A survey. *Knowledge and Data Engineering*, 940–955.

Perez, J., Berlanga, R., & Aramburu, M. (2008b). Contextualizing data warehouses with documents. *Decision Support Systems*, 77–94.

Perez, J., Berlanga, R., & Aramburu, M. (2008c). Towards a data warehouse contextualized with web opinions. *e-Business Engineering*, 697–702.

Poudat, C., Cleuziou, G., & Clavier, V. (2006). Catgorisation de textes en domaines et genres. *Document Numrique*, 61–76.

Ravat, F., & Teste, O. (2007). Olap aggregation function for textual data warehouse. *International conference on enterprise information systems*, 151–156.

Ravat, F., Teste, O., & Tournier, R. (2008). *Top keyword: An aggregation function for textual document olap. Data warehousing and knowledge discovery*, 55–64.

Scott, M. (2015). *Oxford wordsmith tools*. http://www.lexically.net/wordsmith/ Last Accessed 06 January 2015.

Sullivan, D., *Document warehousing and text mining: Techniques for improving business operations, marketing, and sales*, John Wiley Sons Inc.

Tague-Sutcliffe, J. (1992). Measuring the informativeness of a retrieval process. *15th annual international ACM SIGIR conference on Research and development in information retrieval*, 23–36.

Tseng, F., & Chou, A. (2006). The concept of document warehousing for multi-dimensional modeling of textual-based business intelligence. *Decision Support Systems*, 727–744.

Tseng, F., & Lin, W. (2006). D-tree: A multi-dimensional indexing structure for constructing document warehouses. *Journal of Information Science and Engineering*, 819–842.
Voorhees, E., & Harman, D. (2005). *Common evaluation measures. The twelfth text retrieval conference (TREC 2003)*, 500–555.
Wan, X., & Xiao, J. (2008). *Single document keyphrase extraction using neighborhood knowledge*. AAAI855–860.
Wartena, C., & Brussee, R. (2008). Topic detection by clustering keywords. *Database and Expert Systems Application*, 54–58.
Yu, Y., Lin, C., Sun, Y., & Chen, C. (2009). Inextcube: Information network-enhanced text cube. *Proceedings of the VLDB endowment*, 1622–1625.
Zhang, D., Zhai, C., & Han, J. (2009). *Topic cube: Topic modeling for olap on multi-dimensional text databases*. ISD1124–1135.

Mustapha Bouakkaz received the Engineer degree and Magister in Computer science from the Informatics and Mathematics Department, Laghouat University, Algeria, in 2008 and 2011, respectively. He is a PhD candidate in the Department of Informatics and Mathematics. His research interests include, data warehousing, data mining, and analyzing social networks. He is a member of the LIM laboratory. More details about his research and background can be found at: http://googlescholar.com/~mustaphabouakkaz.