Building a contextual dimension for OLAP using textual data from social networks

Karel Gutiérrez-Batista, Jesús R. Campaña, Maria-Amparo Vila, Maria J. Martin-Bautista
Department of Computer Science and Artificial Intelligence, University of Granada, Granada, Spain

A R T I C L E   I N F O

Article history:
Received 15 May 2017
Revised 19 September 2017
Accepted 4 October 2017
Available online 5 October 2017

Keywords:
OLAP
Data warehousing
Multidimensional model
Context detection
Text clustering

A B S T R A C T

Due to the continuous growth of social networks the textual information available has increased exponentially. Data warehouses (DW) and online analytical processing (OLAP) are some of the established technologies to process and analyze structured data. However, one of their main limitations is the lack of automatic processing and analysis of unstructured data (specifically, textual data), and its integration with structured data. This paper proposes the creation, integration and implementation of a new dimension called Contextual Dimension from texts obtained from social networks into a multidimensional model. Such a dimension is automatically created after applying hierarchical clustering algorithms and is fully independent from the language of the texts. This dimension allows the inclusion of multidimensional analysis of texts using contexts and topics integrated with conventional dimensions into business decisions. The experiments were carried out by means of a freeware OLAP system (Wonder 3.0) using real data from social networks.

1. Introduction

The popularity and dramatic increase of the use of social networks in the last ten years has led to the creation of huge amounts of textual data generated by tens of millions of users on a daily basis (Guille, Hacid, Favre, & Zighed, 2013). The automatic massive processing and analyzing capabilities of most current technologies and systems are not enough to deal with such a huge quantity of heterogeneous, semi-structured, and unstructured data. Another challenge is the integration of textual information with traditional data, so that organizations can use this new resource and the possibilities offered by social network data. Content in social media messages can be very relevant to queries and decision makers need to use this content in order to take into account the full extent of the information available.

The processing of massive data involves the summarization and clustering of the data. For such a purpose, DW and OLAP are presented as the most adequate technologies, as they base their success on the advantages of integration, storage, and the operations of a multidimensional model. DW and OLAP thus allow the development of aggregations through conventional and unconventional dimensions for heterogeneous data. For the specific case of textual data, these systems need to undergo some kind of transformation to bring data to a more structured format, thus facilitating the analysis. In order to apply DW and OLAP to analyze the textual information provided by social networks, it is necessary to detect the main terms of the domains for the contexts of the texts. This would allow the decision makers to segment each context found and to treat it by taking advantage of the features and capabilities provided by multidimensional analysis.

There are many papers devoted to the use of the DW and OLAP technologies for the study of textual data present in the different social networks. Most of these papers involve information retrieval, sentiment analysis, recommendation systems, etc. These approaches do not take into account the context to which textual data actually belong, or use a predefined context.

In a previous study a new representation for textual data and their associated operations for query definition were presented (Martin-Bautista, Martínez-Folgos, & Vila, 2015). Based on that representation, in Martin-Bautista, Molina, Tejeda-Avila, and Vila (2013) a new definition and implementation (Wonder 3.0) of a textual hierarchy (AP-dimension) is presented. Wonder is an OLAP system based on PostgreSQL that gives support to textual dimensions. We will use the definition of textual dimension and Wonder 3.0 to manage the domain hierarchy included in our proposal which is part of the context hierarchy. We have also previously carried out research about context detection and the influence of sentiment words in the process of context detection, which have been used in this proposal.
The findings from our previous research have been integrated into a new methodology which uses data from social networks and builds a contextual dimension. This contextual hierarchy of textual dimensions contains the semantics associated to texts, which allows the business analysts to implement a detailed study by means of the use of an OLAP system (Wonder 3.0) on the topics discussed by users in social networks.

This study offers a novel solution to the current difficulty of achieving the application of multidimensional analysis on heterogeneous data by integrating textual data from social networks. It also improves results from the automatic detection of the contexts included in texts.

The contributions of this paper are:

- A new methodology for the creation of a data warehouse with textual dimension organized by contexts (set of topics) named Contextual Dimension, and its implementation in a real OLAP system (Wonder 3.0). This dimension is created automatically using a system to process and organize data from social networks. The methodology uses tools such as Multilingual Central Repository 3.0 (MCR 3.0) in order to make the process multilingual. The Contextual Dimension is a data warehouse dimension extracted from texts formed of two components: a context hierarchy composed of groups of topics discussed in a text, and within each level and context of this hierarchy, a domain hierarchy including the terms included in the texts.
- The Contextual Dimension will allow the decision makers to analyze the data from social networks selecting a context that has been automatically extracted and organized from social network data. The originality of our proposal lies in the fact that decision makers do not need to know contexts in advance to perform queries. In this analysis, it is possible to combine textual information with other traditional attributes, e.g. time of the day, and day of the week when the comments are made.
- The procedure to perform the integration is automatic and independent from the language in which the text is written. This enables the analysis of textual data in social networks using aggregations involving conventional and unconventional dimensions on heterogeneous data.

The rest of the article is organized as follows: Section 2 summarizes the main work developed in this research area. Section 3 explains the structure of the proposed dimension and presents the formal definitions supporting such a structure. Section 4 presents the creation process of the contextual dimension and its integration into a multidimensional model from the implementation point of view. The experimental results of our proposal and its discussion are shown in Section 5. Finally, Section 6 summarizes the conclusions which can be obtained from the implemented work and introduces some ideas regarding future research where we will further study the topics dealt within this research.

2. Related work

As previously mentioned, the main idea of our research is the automatic creation and integration of a contextual dimension into a multidimensional model, enabling users, organizations, and researchers to analyze data from social networks according to the main detected contexts and topics. Due to this reason, we study and analyze the main contributions related to the detection of contexts from texts in social networks, and the multidimensional analysis on heterogeneous data integrating both textual data and conventional data in social networks. In each case, the main characteristics of our approach which make it different from others are highlighted.

2.1. Context detection

Allan et al., in the first study on topic detection and tracking (Allan, Carbonell, & Doddington, 1998a; Allan, Papka, & Lavrenko, 1998b), focus on the exploration of techniques for the detection of the appearance of new topics and for tracking their reappearance and evolution. In Young-Woo and Sycara (2004), the authors mention the fact that topic extraction must deal with untagged data and that new subsets of events with similar contents are clustered together, therefore clustering algorithms are a good choice for discovering unknown events.

This latter approach is the closest to the contribution of this study, as it adjusts to our task of automatically detecting contexts from large volumes of textual data to achieve their integration as a new dimension in a multidimensional model. In order to achieve this goal, we will specifically focus on the application of hierarchical clustering methods.

There are many papers on the automatic categorization of texts, such as Chung-Hong (2012), where a mechanism which allows the clustering in real time of the content of the Twitter microblogs to detect events is established. Then, to assess the relationship amongst the different events, a method which combines the advantages of a clustering algorithm and a supervised learning model is developed. Skarmeta et al. present a study of the use of a semi-supervised agglomerative hierarchical clustering (ssAHC) algorithm, which allocates the texts to predefined categories (Skarmeta, Bensad, & Tazi, 2000). Zheng and Li (2011) propose a new approach for the semi-supervised hierarchical clustering based on the establishment of a connection between the dendrogram of the hierarchical clustering algorithm and the ultrametric distance matrix, where the restrictions have been introduced by means of the “Triple-wise relative constraints” method. The previous studies share the condition that the set of tags or categories in which the texts will be grouped is known beforehand.

Hierarchical clustering algorithms have been widely studied in the literature for different kind of situations, including textual data (Deshmukh, Kamble, & Dandekar, 2013; RaghavaRao, Sravankumar, & Madhu, 2012). In Voorhees (1986) and Willett (1988) a general overview of the types of traditional agglomerative hierarchical clustering algorithms and their operation in the context of textual data is presented.

Gao, Gao, He, Wang, and Sun (2013) propose a new topic detection algorithm from pieces of news published on the internet on large disasters, based on group average hierarchical clustering (GAHC). The main idea of such an algorithm consists of dividing large data into smaller clusters and then using hierarchical clustering on these groups to generate the final topics. A practical tool for helping journalists and news readers to find interesting topics in message streams without feeling overwhelmed is presented in Martin, Corney, and Goker (2013). In this case, a time dependent variation of the classical tf-idf approach is presented. Sentences are grouped into bursts, often appearing in the same messages, so as to identify the emerging topics in the same time window. The experiments were carried out with Twitter data related to sports and politics. The study in Xiaohui, Xiaofeng, Yuming, Shencheng, and Xutao (2013) shows a conceptual graph containing concepts as nodes. Nodes connected by an edge share the same topic terms. When the hierarchical clustering is executed in this conceptual graph, the behavior curves for highly correlated concepts are grouped as topics.

It is worth mentioning that these studies are mainly oriented to topic or event detection in social networks, where the data belong to a specific domain and happen during a given interval of time.
Besides, as happens in Xiaohui et al. (2013), the number of topics for each dataset used in the experiments is previously known.

Conversely, our approach considers automatic context detection in textual data from social networks without any kind of previous knowledge about the number of clusters or contexts, meaning it does not require the presence of any expert. Because of this, we have considered the use of hierarchical clustering methods for our purposes. Moreover, thanks to the hierarchical nature of these algorithms, we are able to obtain a context hierarchy as a final result, which allows the user to choose the level of detail and thus work within the user’s contextual dimension.

2.2. Multidimensional analysis in social networks

Data warehouses and OLAP are technologies of paramount importance in systems related to decision making and have proven their competitiveness and advantages in several types of applications. One of the main advantages of the OLAP systems is the simplicity for summarizing and clustering large datasets in a multidimensional way, thus making it a highly useful tool in applications from different domains.

Nowadays, with the increase and growth of social networks, organizations, research groups, etc. have devoted time and resources to the study of the data stored in these networks. One of the alternatives which has appeared in response to the analysis of large volumes of data in social networks is their integration into a multidimensional model. As a result of this integration, two research lines have been established: Social network analysis and Social media analysis. This study is framed within the second of these research lines.

Social network analysis (SNA) is a strategy for the study of social structures by means of the use of network and graph theories (Otte & Rousseau, 2002), whereas social media analysis (SMA) is the process of obtaining information from conversations in a digital format and such information can be used in processes related to decision making, marketing, customer support, sales, etc.

Many papers related to SNA can be found in the literature, as for example Zhao, Li, Xin, and Han (2011), where a new DW model supporting OLAP queries on large multidimensional networks is introduced. On the other hand, Park, Yu, Park, and Kim (2013) propose a new model for the analysis of large volumes of traffic in data stored during a long period of time. These studies are undoubtedly oriented towards the study of the social structure of a given system, whereas we are more interested in studies related to the analysis and study of the data stored in social networks.

This is found in Brinay et al. (2011), where a data warehousing model to analyze large volumes of tweets by proposing measures relevant in the context of knowledge discovery is defined. The paper proposes two different models to manipulate cubes of tweets: without using a predefined hierarchy, and using an existing hierarchy (MeSH medical subject headings medical hierarchy). In both cases the basic premise is a known hierarchy, as in the first approach a hierarchy present in a traditional dimension is used (e.g. location) to implement the roll-up and drill-down operations. Our proposal, meanwhile, is able to construct a textual hierarchy independent of the text domains, therefore in principle, no hierarchy is necessary.

Another paper which takes into account texts from tweets is Liu et al. (2013). In this, the authors present a text cube to study the different kinds of human, social and cultural behavior (HSCB) present in the Twitter stream. The text cube allows data to be organized (Twitter text) in multiple dimensions and hierarchies to efficiently query and visualize data. The main approach is to implement sentiment analysis and data visualization. As has been mentioned, the main aim of our research is to allow a multidimensional analysis of textual data in social networks according to a given context. As a result, if instead of implementing a sentiment analysis of all texts, this task is carried out according to their context, the obtained results would substantially improve.

In Zhang, Zhai, and Han (2009) a data model known as Topic Cube is proposed. It combines OLAP with a topic probabilistic model, allowing the execution of OLAP in all textual dimensions of a multidimensional text database. The Topic Cube extends the traditional OLAP data cube to support a domain hierarchy and stores probabilistic measures of the text learnt by means of a topic probabilistic model. The main difference with our approach lies in the fact that the domain hierarchy must be specified by an analyst, whereas the new contextual dimension proposed in this article is automatically built.

Most of the studies oriented towards the integration of texts from social networks into a multidimensional model are based either on the extraction of new dimensions (contexts/topics, entities, sentiment, etc.) or measures from the texts. An example of this can be seen in Pérez, Berlanga, Aramburu, and Pedersen (2008), where a new framework to integrate opinions from social networks with a corporate data warehouse is proposed. The study in Moya, Kudama, Cabo, and Llavori (2011) proposes the integration of the result of the implementation of a sentiment analysis on opinions from social networks with a data warehouse, being able in such way to implement OLAP on these data. Rehman et al. try to extend the OLAP to allow multidimensional analysis of data from social networks by integrating text mining methods and opinions with a data warehousing system and using several techniques related to the knowledge discovery of semi-structured and unstructured data from social networks (Rehman, Weiler, & Scholl, 2013).

To the best of our knowledge, the idea underlying our approach is completely novel. Firstly, the main contexts of the texts are detected and are integrated automatically into a multidimensional model. Secondly, new dimensions and measures of the texts according to the contexts are extracted. This process provides analysts with better structured information from texts.

3. Contextual dimension structures

3.1. Concept definitions

We can find many papers in the literature which use the “topic” concept without any clear definition of this term. The term “topic” is even used indistinctly by various authors when referring to events, concepts, contexts, etc. (Allan et al., 1998a; Allan et al., 1998b; Chung-Hong, 2012; Gao et al., 2013; Martin et al., 2013; Xiaohui et al., 2013; Young-Woo & Sycara, 2004).

For a better understanding of this section we have included definitions of the terms topic and context which best fit our proposal.

We consider \( W = \{w_1, w_2, \ldots, w_n\} \) as any vocabulary (in our case a series of vocabularies associated with a group of texts) and we define it as follows:

**Definition 1** (Topic). A topic is a subgroup of the vocabulary terms \( T \subseteq W \), which have a semantic link associated with a lexicon resource (WordNet, Wikipedia, etc.).

**Example 1.** internet = \{LAN, WIFI, WEBSITE\}, programming = \{JAVA, PHP, WEBSITE\}, laptop = \{ACER, TOSHIBA, WIFI\}

**Definition 2** (Context). A context is a group of topics which are semantically linked \( C = \{T_1, T_2, \ldots, T_n\} \). Obviously all the context can be considered as a wider topic, in other words as a more general element in the topic hierarchy.

**Example 2.** Computer Science = \{internet, programming, laptop\}

In other words, a context would be a general theme, whilst topics would be those themes related to a principal theme (e.g. the
topics internet, programming, laptop, etc. would be part of the context Computer Science).

The new dimension (contextual dimension), reflects or establishes the topics present in the different detected contexts in social networks. When a multidimensional model supporting a contextual dimension is created, it would allow the user to use this dimension together with the traditional dimensions. In this way it is possible to do OLAP operations on data extracted from social networks in the form of the main topics mentioned on a given context.

In the following section, the structure of contextual dimensions is explained in detail, introducing the formal definitions presented in previous studies which will act as the base for their creation (Martin-Bautista, Martinez-Folgoso, & Vila, 2008; Martin-Bautista, Molina, Tejeda, & Vila, 2010). As it can be seen in Fig. 1, the two components which make up the contextual dimension are the context hierarchy and the domain hierarchy.

The context hierarchy is a hierarchy of tagged clusters obtained from the application of hierarchical clustering algorithms. For each cluster belonging to the context hierarchy, a domain hierarchy is obtained containing the most relevant terms mentioned in such contexts and which will act as the base of the OLAP searches and operations defined in our multidimensional model. In Sections 3.2 and 3.3, the structure of the hierarchies is described. Finally, Section 3.4 shows the integration process of the contextual dimension in the obtained multidimensional model.

### 3.2. Context hierarchy

One of the components of the contextual dimension is the context hierarchy. It is important to point out that the creation process of both the context hierarchy and the domain hierarchy associated to each context in each level is fully automatic and that the implementation details will be explained in Section 4. The context hierarchy is obtained as a result of the application of a hierarchical clustering algorithm (Complete Linkage, Single Linkage, Centroid Linkage, Average Linkage, etc.) to the previously syntactically and semantically preprocessed textual data. In this way, when the data is grouped, a set of groups which represent the main detected contexts is obtained, allowing the use of this new dimension as part of OLAP analysis.

As it can be seen in Fig. 1, the context hierarchy presents a tree structure, where each level results from a cutting in the tree obtained after the execution of a hierarchical clustering algorithm on the preprocessed texts for a given number of clusters (17, 25, 40, 60, 80, 100, 120, etc.). These cuttings are implemented taking into account an unsupervised clustering measure, the Silhouette Coefficient (Rousseeuw, 1987). Each level consists of the clusters (main contexts) in which the analyzed texts were grouped. Moreover, it is important to point out the fact that the relationships among the different levels are due to the nature of the hierarchical clustering algorithms, where a cluster or a context from an upper level can be divided into two or more clusters or contexts in the immediate lower level.

![Context dimension components](image-url)
Obtaining the context hierarchy is a complex automatic process which includes several necessary stages for its correct operation. The main elements for understanding the process are explained below.

The process begins by syntactically and semantically preprocessing the texts. During these subprocesses, some tasks such as the part-of-speech tagging, entity tagging, tokenization, and removal of stop words are carried out. Terms which are not nouns are discarded, leaving in the texts just those terms which add relevant information for the detection of contexts. Then, during the semantic preprocessing, the terms are disambiguated to establish their meaning, for this we use the description of the words in Multilingual Central Repository 3.0 (MCR 3.0) (Agirre, Laparra, & Rigau, 2012) and BabelNet (Navigli & Ponzetto, 2012). Disambiguation is a really important task, as if it is correctly executed, it enables a correct identification of the meaning of the terms and the tags by which they will be replaced. These tags belong to the WordNet Domains taxonomy (Magnini & Cavaglia, 2000) included in MCR 3.0 and are used for obtaining the clusters or contexts by means of a hierarchical clustering algorithm. Tags pertaining to the two first levels in the taxonomy are discarded due to the fact that they are generic domains. It is worth mentioning that at this stage of the process, the sentiment related words are discarded by mean of lexical resources such as SentiWordNet 3.0 (Andrea & Fabrizio, 2006) and WordNet Affect (Valitutti, 2004). However, some sentiment terms are not useful information for the context detection task. For this reason, it is possible that some texts are not taken into account for the following stages, since all words present in a given document can be removed if they are not relevant for the topic identification.

Once a hierarchical clustering algorithm has been used, it is necessary to label the clusters. Cluster labels are selected from the most representative tags of each cluster, which play an essential role in applications where the final user needs to know which label or tag was associated to each text (Manning, Raghavan, & Schütze, 2008). This is our case, where the selected labels for each cluster represent or identify the main contexts extracted from the texts. The two main cluster labeling methods are Cluster-internal labeling and Differential cluster labeling.

In this study, the method used is Differential cluster labelling (Manning et al., 2008) with a Chi-Square ($X^2$) distribution, as depicted in Eq. (1). This is one of the most widely used techniques in Differential cluster labelling and one of which shows the best results. This method determines the label of a cluster, taking into account both the information of such a cluster and that of the rest of the clusters, thus being perfectly suited to the requirements of our study. The value for ($X^2$) is calculated by means of the following formula:

$$X^2 = \sum_{a \in A} \sum_{b \in B} \frac{(O_{a,b} - E_{a,b})^2}{E_{a,b}}$$

(1)

where $O_{a,b}$ is the observed frequency of $a$ and $b$ happening simultaneously and $E_{a,b}$ is the frequency with which they are expected to happen. For the specific case of the labelling of clusters, $A$ is associated to the fact that a document belongs to a group and $B$ is associated to the presence of a term in a document. Both variables $a$ and $b$ can take the values $[0,1]$.

As was previously mentioned, each level of the context hierarchy is obtained when a cut is done for a given number of clusters. For such a purpose, the Silhouette Coefficient (Rousseeuw, 1987) is used as goodness value Eq. (2), where $d(i)$ represents the mean of the distance of the element $i$ to the rest of elements of the same cluster (cohesion), and $b(i)$, the distance to the centroid of the closest cluster (separation). The values of this coefficient are in the interval $[-1,1]$, being 1 the best value.

Fig. 2 shows the obtained context hierarchy for a set of real data from Twitter using Complete Linkage as the hierarchical clustering algorithm and the cosine as the distance measure (in this case, the cuts were made at 17, 25, and 40 clusters). Each square in the hierarchy corresponds to a context, which is described by means of the set of labels it contains.

To assign labels to each context, we start from the most specific level of the context hierarchy and select the corresponding cluster labels. The contexts of the less specific levels (upper levels) in the context hierarchy are labeled using the union of the labels of their children contexts in the hierarchy. It is worth mentioning that the contexts of the hierarchy are labeled with the most representative labels, bearing in mind that the texts are clustered in each context. Therefore it is possible that some labels found in more specific contexts do not appear in less specific contexts (e.g. in Fig. 2, the label “GRAMMAR” from cluster 6 does not appear in cluster 1).

$$S(i) = \frac{a(i) - b(i)}{\max[a(i), b(i)]}$$

(2)

The context hierarchy is an essentially significant component for the creation of the contextual dimension, as it is used to create the domain hierarchy associated to a level and a cluster or context. This domain hierarchy allows us to treat the new dimension just as a classic dimension, as it supports the operations of the multi-dimensional model. The main definitions which enable the establishment of such a domain hierarchy are introduced below.
3.3. Domain hierarchy

The other component of the contextual dimension is the domain hierarchy. For each context of the context hierarchy, a domain hierarchy is created. For such a purpose, the data from a given context are processed and a lattice structure is created as an intermediate representation form. Using this representation form it is possible to carry out queries containing the main terms included in the context. It is worth mentioning that only the frequent term sets which meet a given predefined support value will appear in this structure. This structure is known as the AP-Structure, and it is formed by AP-sets. More about obtaining the AP-structures from texts is explained in Martin-Bautista, Prados, Vila, and Martinez-Folgoso (2006). The main definitions and properties of AP-sets and AP-structures are briefly explained below. A more detailed study of these structures and of their use in the textual analysis is presented in Martin-Bautista et al. (2006).

3.3.1. AP-Set

Informally speaking, an AP-Set is formed by a set of terms and the lattice of its subsets. All AP-sets must meet the following conditions, the first condition is that any AP-set verifies the Apriori property, and the second one assures the existence of a unique set called \( Y \) of maximal cardinality, which characterizes the AP-Set.

The Apriori Property is a characteristic belonging to the Apriori Algorithm proposed in Agrawal and Srikant (1994), which states that any subset of frequent itemsets must also be frequent. This considerably minimizes the execution time needed for the algorithm.

**Example 3.** Let be \( X = \{\text{ACCESS, INTERNET, TWITTER, \ldots, WEBSITE}\} \) and \( \mathcal{R} = \{\{\text{ACCESS}\}, \{\text{INTERNET}\}, \{\text{WEBSITE}\}, \{\text{ACCESS, INTERNET}\}, \{\text{ACCESS, WEBSITE}\}, \{\text{INTERNET, WEBSITE}\}, \{\text{ACCESS, INTERNET, WEBSITE}\}\} \), then the spanning set of \( \mathcal{R} \) is \( Y = \{\text{ACCESS, INTERNET, WEBSITE}\} \).

Fig. 3 shows the inclusion lattice of Example 3. The spanning set \( Y \) is the root of the lattice, in the leaves, the single elements of \( Y \) are to be found, and in the intermediate nodes, the different combinations of \( Y \) elements with a cardinal 2 are included.

AP-sets define a set of operations that allow us to establish a formal mechanism for querying, and operations associated to their representation (Martin-Bautista et al., 2008).

3.3.2. AP-Structure

An AP-structure is a lattice of subsets whose upper extremes are their spanning sets. An example of the AP-Structure \( \mathcal{T} = g(\{\text{PAPER, WEBSITE}\}, \{\text{ACCESS, INTERNET, WEBSITE}\}) \) is offered in Fig. 4. It should be remarked that the two spanning sets have common elements and that these elements also appear as a common part in the underlying lattice, concretely, as a sub AP-Set with cardinal equal to one \( \mathcal{R} = g(\{\text{WEBSITE}\}) \).

One of the most important operations which can be defined on the AP-Structure is the Induced AP-Substructure operation. This operation is particularly significant due to the fact that it will enable us to find the representation of the AP-Structure, which corresponds to a given text within a cluster. It is just the AP-Structure resulting from intersecting any given AP-Structure with a given search term set.

Once the AP-Structure and induced AP-substructure have been designed, the following section discusses some necessary operations to be able to determine whether a given set appears or not within a given AP-Structure. These operations will be the base of the search implemented by our model.

**Matching of sets with AP-Structures**

Taking into account that the AP-Structure is obtained from the relevant terms which appear in the processed texts, we can say that it will contain most of the relevant terms appearing in such texts. As a result, we can state that the AP-Structure is the active domain of the domain hierarchy which was obtained for a given context and level within the context hierarchy.

The idea is that the user will express their requirements as sets of terms which will be launched as a query on the corresponding domain hierarchy with a given level and context within the context hierarchy.

3.4. OLAP system description

Before entering into the process of integrating the contextual dimension into an OLAP cube we are going to briefly have a look at the multidimensional model implemented in our system. For a better understanding see Fig. 5 which shows the multidimensional model created by Wonder 3.0 for the example presented in Section 4.3.

The OLAP model implemented by Wonder 3.0 is based on the Snowflake scheme, where the dimensions are normalized (each table represents a level in contextual dimension hierarchy) in order to save storage space by eliminating redundant data. In this case a table of the events has been created, which contains references to the respective dimensions (date and context), as well as the quantitative variable to be analyzed (i.e. the number of published tweets). For the example we are dealing with, each dimension has a hierarchy which allows the analysis of the data at different levels of granularity.

In the case of the contextual dimension it is necessary that the most specific level of the hierarchy be level 80 in the context hierarchy and that it is composed of the context which belongs to the tweets and their associated ADT (abstract data type) (tda_tweet). The latter is the semantic representation of the texts via which searches can be carried out by matching (the domain hierarchy).

The integration and management of this new contextual dimension in a multidimensional model are performed by taking the concept of AP-dimension presented in Martin-Bautista et al. (2010) as the starting point. It is precisely such an AP-dimension and its associated structures and operations which allows us to integrate the contextual dimension into
the multidimensional model. Such integration takes place by combining the following two actions:

1. Creation of a fact table for the level and context to be studied within the context hierarchy combined with traditional dimensions.
2. For contextual dimensions of the fact table, the domain hierarchy is added and treated as an AP-dimension.

For this reason, the domain of the domain hierarchy will be defined in the same way as that of an AP-dimension, therefore, the domain is the set of all the AP-substructures of the global AP-Structure associated with an attribute, as these are the possible values which this attribute may have.

**Dice, Roll-Up, and Drill-Down operations on Domain Hierarchy**

Due to the fact that the domain hierarchy is defined as a dimension, the dice, roll-up, and drill-down operations for the multidimensional model work in the same way as in an AP-dimension. In the example in Fig. 6, the functionality of these operations is shown. Due to space constraints, just a part of the hierarchy which could be formed is shown. The example corresponds to the COMPUTER SCIENCE context of level 40 cluster of the context hierarchy. As it can be seen in the example, the data can be analyzed by their initial sentences C₁, as well as by more or less detailed sentences, sets C² and C³ respectively. The details of the definitions for these operations can be seen in Martin-Bautista et al. (2010).

A real example of the structure of the contextual dimension which can be obtained following all the definitions discussed so far is shown below. In the following sections, the operations of the model on these structures will be used to answer questions asked by the user.

Fig. 7 shows the Context Hierarchy of Fig. 2 including the Domain Hierarchy. Due to space constraints, just a segment of the domain hierarchy obtained for the contexts labeled as COMPUTER SCIENCE and PUBLISHING is shown. It is worth mentioning that with the integration of the contextual dimension and using the traditional operations of the multidimensional model and those introduced by our model for textual dimensions jointly, the user will be able to search for more or less specific sentences which may be interesting in each context belonging to a given level.

This will allow the user to study the contexts which have been automatically obtained, thus being able to integrate and treat jointly and in the same way both textual dimensions and traditional dimensions such as dates, places, etc.

Once the definitions and main operations which enable the integration of the contextual dimension in our multidimensional model have been commented, some details of the implementation process will be discussed in Section 4.

### 4. Methodology for creation and integration of the contextual dimension with multidimensional model

For a better understanding of the concept we have divided our methodology into three principal steps, which correspond to the modules in Fig. 8. This figure shows the processes which lead to the generation of the Contextual Dimension from a group of textual data.
1. Creation of the Context Hierarchy from texts from social networks: the texts from the social networks are syntactically and semantically processed. Then, via a hierarchical clustering algorithm, a hierarchy of the contexts is created and each context is labeled with the most representative labels for each context (labels which correspond to the original terms).

2. Creation of the Domain Hierarchy from a context in a level selected from the Context Hierarchy: for each context of the hierarchy contexts a domain hierarchy is created from the preprocessed texts via the Apriori algorithm.

3. Integration of the Contextual Dimension in a multidimensional model: the computational integration of this new dimension in such a model is carried out with the Wonder OLAP Server 3.0 tool, hereon Wonder 3.0, which is a freeware OLAP server. Implementation of each component of this module can be checked in more detail in previous papers (Martin-Bautista et al., 2015; Martin-Bautista et al., 2013). It is worth mentioning that it is necessary to implement new functionalities, as after creating an OLAP cube containing the contextual dimension, it is necessary to previously select the level and the context in the context hierarchy of the data to be analyzed.

4.1. Context hierarchy module

We now go on to explain the steps in the context hierarchy module, mentioning the input and exit data, as well as giving a detailed description of the process.

4.1.1. Syntactic preprocessing

The input data used by our methodology are texts extracted from social networks. These texts are syntactically processed and filters for enabling their automatic processing are applied. The processes to be executed are part-of-speech tagging and entity recognition. These are implemented using the tools Stanford POS (Toutanova, Klein, Manning, & Singer, 2003) and Stanford NER (Finkel, Grenager, & Manning, 2005) respectively. The first of these tools allows the allocation of grammatical categories to each word as a noun, adjective, verb, etc. from a text. The second labels those words within a text which represent names or things, such as individuals, companies, places, etc., from several models provided in the distribution.

The tokenization filter is then applied, removing all punctuation marks which hinder the automatic processing of the texts. Then, the necessary filters to remove all empty words are executed, removing those which are not identified as nouns by the POS tagger, those identified as proper nouns by the entity identifier, as well as those which are not included in the external knowledge base MCR 3.0, as none of them provide any useful information for detecting topics. This phase allows the original, clean texts to be obtained, in other words, they only contain the terms which provide useful information for the detection of the contexts.

4.1.2. Semantic preprocessing

Once the previous filters have been applied, the texts are ready to be semantically processed. The aim of the semantic preprocessing of the methodology is to homogenize the syntactic representation of the concepts of the text. To achieve this, each term is replaced by its corresponding labels in the WordNet Domains (Magnini & Cavaglia, 2000) taxonomy of the MCR knowledge base. To determine the set of labels of a term, it is necessary to know the meaning of such a term, as each meaning has a WordNet Domains label set associated to it. Due to this, before the replacement takes place, it is necessary to execute the disambiguation of all the terms to find out their actual meaning.

**Disambiguation Process**

In this study, the disambiguation is executed by means of the Lesk algorithm (Lesk, 1986), in which the meaning with a higher
overlap between its description and the current context where the word has been used is selected. To determine the highest overlap meaning, the common terms between the current context where the term has been used and the descriptions of the meanings of such a term in MCR 3.0 and BabelNet are counted. If two or more meanings present the same overlap, the most frequent term among them will be selected.

During the disambiguation process, one more filter is applied for removing those terms that have sentiment orientation according to the lexical resources SentiWordNet 3.0 and WordNet Affect (Valitutti, 2004). Finally, the texts can be grouped by contexts, as they just have those terms which actually contain a significant value for context detection.

As was previously mentioned, the main aim of this process is to homogenize the syntactic representation of the concepts present in the texts, thus ensuring the improvement of the quality of the hierarchical clustering algorithms. This problem is fully solved, as after replacing each term by the set of corresponding labels, the highest number of different labels will be the number of nodes of the WordNet Domains taxonomy. In this way the exit data of the Semantic Preprocessing is the text of the input data where each term has been substituted with a label from WordNet Domains corresponding to the meaning of this term in the analysed text.

4.1.3. Hierarchical clustering

Once the texts have been homogenized, the hierarchical clustering of the texts from the WordNet Domains labels is carried out. For such a purpose, it is necessary to create a weight matrix with the relation between labels and documents from the texts obtained in the previous phase. The process for the creation of the weight matrix is explained below. For each term of a text:

1. The term is replaced by the label or labels with which the selected sense (synset) during the disambiguation process has been annotated.
2. Each label \( e \) will be allocated a value taking into account Eq. (3), where \( C_e \) is the number of labels by which the given term has been replaced and \( C_t \), the number of terms of the analysed text after being syntactically preprocessed. As mentioned before, if a term is discarded due to the fact that all its senses have been annotated with labels which belong to the first two
levels of WordNet Domains, then this term will not be taken into account to determine the value of the rest of labels in the weight matrix.

3. If the label e appears more than once in the text, then its final value will be the sum of each individual value.

$$\nu_e = \frac{1}{C_e + C_t}$$  \hspace{1cm} (3)

After the creation of the weight matrix, the hierarchical clustering algorithm for the creation of the groups of texts from the semantics provided by WordNet Domains is executed. Each text is associated to a group or cluster, therefore the output of this process is a cluster hierarchy.

4.1.4. Cluster labeling

Lastly, the clusters are labeled and each of them is associated to the most relevant label or labels from that cluster (context). As was previously mentioned, the technique $X^2$ was used for the selection of these labels. The result of this module of the system is a context hierarchy which can be defined as a shortlist between the syntactically processed text and the context to which it belongs in a given level of the hierarchy.

4.2. Domain hierarchy module

In this section we explain the steps which make up the Domain Hierarchy Module, as well as the input and output data of each process. From the context hierarchy, it is possible to create the domain hierarchy from the preprocessed texts belonging to a given context and level of the context hierarchy. The second module of the system is responsible for creating such a hierarchy using the output of the first module as input data.

4.2.1. Global AP-Structure obtaining

For this purpose, the Apriori algorithm is executed on the preprocessed textual data of a particular context. Through the implementation of the Apriori algorithm, the two main knowledge structures which contain the semantics of the textual data are obtained. These structures are the AP-sets and the AP-Structure and the Text Mining Tool V1.0 (Martin-Bautista et al., 2015) is used to obtain them.

The implementation of the Apriori algorithm allows us to obtain those itemsets which meet the support specified as a parameter and the AP-sets with their corresponding supports. Then, the global AP-Structure is obtained from the maximal AP-sets. The implemented tool enables the creation of a file for each of the cardinalities of the generated itemsets or the storage in just one file of all obtained itemsets. The structure of these files ensures the later optimization of the intersecting algorithm to obtain the abstract data type (ADT). In these files, the terms are alphabetically ordered within each tuple and as a result, the intersecting algorithm can search for a word inside a tuple, whilst the cardinal of the first character of each looked for word is lower or equal to the cardinal of the first character of the word which is being read from the itemset file.

4.2.2. AP-attribute obtaining

Up to this point, we already have the global AP-Structure and the sets of which it consists. From these data and from the preprocessed texts, the induced AP-Structure is obtained for each tuple of the textual attribute being processed. The induced AP-Structure is obtained after intersecting the global AP-Structure and the preprocessed textual attribute and the result is stored in the database as a new column attribute, known as ADT. In order to optimize this process, the implemented algorithm begins by searching for the intersection of each tuple with the AP-Set with a higher cardinality. In the same way, it calculates the intersection with each of the sets of the different cardinalities of the AP-Structure in a decreasing way. Thus the output of the ADT always guarantees that it matches with the highest cardinality sentence of the AP-Structure.

As was mentioned before, the output of this second module of the system is just the domain hierarchy for a group of preprocessed texts, of a given context and level. This domain hierarchy consists of the preprocessed textual attribute and of the ADT associated to such a text. This structure stores the semantics associated to the main terms present in the highlighted contexts from the texts. Due to this, the fact of integrating this domain hierarchy as a dimension in a multidimensional system is highly relevant, allowing the execution of OLAP with data from social networks, taking into account the topics mentioned in textual attributes of such networks.

4.3. Integration module

The third and last module of the system enables the integration of this contextual dimension into a multidimensional model. With this aim, the Wonder 3.0 tool, which was presented in Martin-Bautista et al. (2013), is used. Amongst its functionalities, it can support textual dimensions known as AP-dimensions. If we take into account that the contextual dimension proposed in this study is treated just as an AP-dimension, Wonder 3.0 perfectly adjusts for achieving the integration of such a dimension in a multidimensional model.

However, certain functionalities which have been implemented in Wonder 3.0 to ensure its correct operation need to be mentioned. In order to create an OLAP cube with a contextual dimension, it is necessary to first select the context and the level of the context hierarchy for which the data cube will be created, as the analysis of the data will be implemented by contexts. Once the cube has been created for a specific context, the OLAP can already be implemented, either on the contextual dimension or on the traditional ones.

The contextual dimension can be treated in the same way as the classic dimensions, that is to say, all basic operations can be executed on such a dimension. Due to the structure of the contextual dimension (formed by the context hierarchy and by the domain hierarchy), it is possible to browse the different levels of the context hierarchy and the domain hierarchy. Wonder 3.0 allows the user to select the level of the context hierarchy, as well as the context according to which the data are to be analyzed upon the creation of the OLAP cube. In the same way, once the data cube has been created, Wonder 3.0 enables the user to launch more or less detailed queries taking into account the domain hierarchy for the selected context when the cube was created.

Illustrative example

We now present a real example of how to create and query a contextual cube with data from Twitter. A cube is created with tweets from the Computer Science context and the number of publications from a day of the week which correspond to the search phrases “COMPUTER”, “INTERNET” and “INTERNET COMPUTER”. 4583 tweets are randomly selected. Apart from the tweets we also have the publication date of the tweets so when the tweets are analyzed this publication date is also taken into account.

Before creating the contextual cube the textual dimension has to be built. As we already explained in the “Context Hierarchy Module” and “Domain Hierarchy Module” sections this dimension is automatically created and allows the texts or documents to be
organized into contexts and at the same time a domain hierarchy is created which offers the potential for studying the texts via the topics dealt with in each context.

Fig. 9(a), shows an overview of how to create a cube. As can be seen it is necessary to select the table or the view from where the texts to be analyzed are, as well as the context hierarchy level and the desired context. In this example level 80 and the Computer Science context has been selected which is made up of 1780 tweets. The contexts which these tweets belong to appear in the Tagcloud of this figure.

Fig. 9(b) also shows the Tagcloud which is generated when no contextual dimension is constructed. The aim of this figure is to show the usefulness of our proposal for analyzing unstructured data. In this case it is trickier for the user to carry out tweet analysis from search phrases which have not been organized into contexts.

Fig. 9(c) shows the dimensions which form the new OLAP cube. In this case the cube is formed by two dimensions, (date and context), where the latter is a contextual dimension formed by a hierarchy which contains the values corresponding to level 80. Moreover, this selection has been based on measuring the quantity of tweets.

Once the data cube has been created we carry out a query which highlights the advantages of the contextual dimension.
5. Experiment results and discussion

This section assesses our proposal experimentally to prove the potential of the integration of the contextual dimension into a multidimensional model. Firstly, we will prove that the methodology used to detect the main contexts in textual data automatically is correct. For such a purpose, we have experimented with real data from two social networks with a large amount of documents. We will then show two examples of the use of the contextual dimension integrated in the OLAP Wonder 3.0 tool.

5.1. Data collection

The experimentation is performed using several sets of real data from Twitter and from the social network Dreamcatchers, in English and Spanish respectively. The Twitter data were downloaded from Sentiment140,\(^1\) and are in CSV format and have six fields, among them the text of the tweets and the date. We use these two attributes in this study. On the other hand, Dreamcatchers, developed through a collaborative approach, has the supporting database PostgreSQL 9.4, with a total amount of 61 tables. The collected information is related to the personal data of the user and to the interactions of such a user in his/her profile and with other users. Dreamcatchers contains several fields with textual information, in this study we use the comments of the users as they provide different interactions discussing several topics.

5.2. Context detection evaluation

In order to prove that the proposed methodology for the detection of contexts works regardless of the language and of the number of documents, we have experimented with four datasets from the two aforementioned social networks. Table 1 presents the description of document datasets used in this research, showing the number of final documents (Number of documents) after the syntactic and semantic preprocessing and which will later be used for the detection of the contexts, and the total number of preprocessed documents for each social network (Number of documents preprocessed (Twitter) and Number of documents preprocessed (Dreamcatchers)). As regards Twitter, tweets have been used whereas for Dreamcatchers, we have used the comments. It should be noted, that in both social networks there are a lot of documents discarded, this is because most of the texts found in social networks express sentiments or opinions, and as

\(^1\) http://www.sentiment140.com/.

![Fig. 10. Query results chart.](image-url)

Table 1

<table>
<thead>
<tr>
<th>Experiment</th>
<th>Number of documents</th>
<th>Number of documents preprocessed (Twitter)</th>
<th>Number of documents preprocessed (Dreamcatchers)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Data Set 1</td>
<td>5000</td>
<td>8743</td>
<td>23125</td>
</tr>
<tr>
<td>Data Set 2</td>
<td>10000</td>
<td>17672</td>
<td>50673</td>
</tr>
<tr>
<td>Data Set 3</td>
<td>20000</td>
<td>35240</td>
<td>93911</td>
</tr>
<tr>
<td>Data Set 4</td>
<td>30000</td>
<td>50399</td>
<td>109475</td>
</tr>
</tbody>
</table>
was previously explained, the terms with sentiment orientation are discarded during the semantic preprocessing.

Figs. 11–14 show the results obtained after calculating the Silhouette Coefficient once the Complete Link hierarchical clustering algorithm has been applied for the amounts of 5000, 10000, 20000, and 30000 documents for each social network. As mentioned previously, the Silhouette Coefficient (Rousseeuw, 1987) is used as measure to evaluate the performance of the hierarchical clustering algorithms since this measure allows us to determine the number of clusters for which the hierarchical clustering algorithms work better.

For each dataset, cuts were made for seven numbers of clusters (17, 25, 40, 60, 80, 100, and 120). In all cases, the values of the social network Dreamcatchers improve those of Twitter and stabilize at around 60 clusters. In this way, the validity of the methodology to determine the contexts is proven, regardless of the number of documents. Moreover, we have to point out the fact that it is fully automatic and independent from the language thanks to the use of lexical resources such as MCR 3.0 and the Stanford POS and Stanford NER tools.

5.3. Usage example

We have selected two practical examples to show the advantages offered by the integration of the Contextual Dimension in a multidimensional model for the analysis of textual data (particularly, in social networks) together with structured data. For the first example, Data Set 4 from Twitter has been selected. Once the Contextual Dimension for these texts has been created, the COMPUTER_SCIENCE contexts have been selected for an amount of 80 clusters.

Fig. 15 shows a first general query, using as a context computer science and retrieving the number of documents (in this case, tweets) according to the days of the week. In total, this context has 1680 tweets distributed on four weekdays (Friday, Saturday, Sunday, and Tuesday). As mentioned in Section 3.3, for each context of the context hierarchy, a domain hierarchy which enables more and less detailed queries from the main topics present in this context is created.

Fig. 16 shows the result of the query executed in the context COMPUTER_SCIENCE with several search sentences. In this case, the values represent the number of tweets which contain all the terms present in the search sentence for days of the week. On the other hand, it is possible to see how the roll-up and drill-down operations of the multidimensional model work on the domain hierarchy. Due to this, more and less specific search sentences such as “INTERNET” and “INTERNET PHONE TWITTER” respectively have been selected. After analyzing the number of tweets of these sentences for each day of the week, we can see that in all cases, the number is higher for the phrase “INTERNET”.

Just as for the first example, Data Set 4 of Dreamcatchers has been selected, and in this case, context ANATOMY and 100 clusters. Fig. 17 shows the general query with the number of documents (in this case comments) by city in the ANATOMY context (910 comments).

Fig. 18 shows the result of the query launched in the ANATOMY context with several search sentences. In this case, the values represent the number of comments which contain all terms present in the search sentence by the city of the user posting the comment. We have selected the cities “Camagüey, Ciego de Ávila, Ciudad Habana, La Habana, Pinar del Río and Santiago de Cuba”. Besides,
more and less specific search sentences such as “BARBA”, “BARBA PELO”, “BOCA”, “BOCA NARIZ”, “CORAZÓN” and “CORAZÓN PECHO” have been selected to be able to check the roll-up and drill-down operations of the multidimensional model.

5.4. Performance evaluation

In this section we have included the response time for the four queries carried out in Section 5.3 to give an example of the performance of the method. Below we list the characteristics of the computer we used to carry out the test.

- Operating System Type: 64 bits.
- CPU: 8 x Intel Core i7-4700MQ 2.4GHz.
- RAM Memory: 7.7GB.

Table 2 shows the execution results in milliseconds for each of the queries carried out in the previous section. Although the number of documents we have used for this evaluation is not very high, it should be totally clear that the system is capable of tackling large amounts of data as are found in the real world.

Moreover, it should be taken into account that up to a maximum of 30,000 documents have been used in experiments, nevertheless one of the advantages of our proposal is that precisely by having the documents previously clustered in contexts the quantity of texts to analyze is considerably reduced.

<table>
<thead>
<tr>
<th>Query</th>
<th>Number of documents</th>
<th>Execution time (milliseconds)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Query 1</td>
<td>1680</td>
<td>1325</td>
</tr>
<tr>
<td>Query 2</td>
<td>1680</td>
<td>290</td>
</tr>
<tr>
<td>Query 3</td>
<td>910</td>
<td>255</td>
</tr>
<tr>
<td>Query 4</td>
<td>910</td>
<td>539</td>
</tr>
</tbody>
</table>
6. Conclusions

This research has the aim of analyzing textual data once they have been previously grouped into contexts. For such purpose, we have built a new dimension called Contextual Dimension. Such a dimension consists of two components: the contexts detected in the texts (context hierarchy) and the main topics treated in each context of the context hierarchy. The new dimension is obtained through data mining techniques (hierarchical clustering algorithms) and its creation is fully automatic and independent from the language of the texts. Then, this dimension is integrated into a multidimensional model, allowing the analysis of textual data by contexts and topics in the same way and together with the conventional dimensions. This integration allows business analysts to take decisions knowing the full extent of the information in a record. In order to enable integration, the multidimensional model needs to be extended (storage and query extensions).

The experimentation took place with real data from social networks and the Wonder 3.0 system was used as the OLAP server for the integration of the Contextual Dimension. The results proved the feasibility of the methodology for the automatic creation of the Contextual Dimension, as well as the query possibilities for the multidimensional analysis of the textual data integrated with conventional dimensions.

This research is just a starting point, opening up a broad range of possibilities for the future. Firstly, once the texts have been organized into contexts it would be very interesting to extract the entities present in them and to incorporate them as a new dimension into the multidimensional model. Finally, the presented mul-
tidimensional model would be extended to support the sentiment analysis by the extracted entities for a given context.

Acknowledgment

The research reported in this paper was partially supported by the Andalusian Government (Junta de Andalucía) under projects P11-TIC-7460 and P10-TIC-6109.

References


