International Journal of Information Management Data Insights 2 (2022) 100129

Contents lists available at ScienceDirect

International Journal of Information Management Data Insights

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

Context-aware OLAP for textual data warehouses

Santanu Roy [a,*], Agostino Cortesi [b], Soumya Sen [c]

[a] Future Institute of Engineering and Management, Kolkata, India
[b] DAIS, Ca' Foscari University, Venice, Italy
[c] University of Calcutta, Kolkata, India

ARTICLE INFO

Keywords:
OLAP
Text data warehouse
Information System
Concept hierarchy
Word embedding
Agglomerative hierarchical clustering

ABSTRACT

Decision Support Systems (DSS) that leverage business intelligence are based on numerical data and On-line Analytical Processing (OLAP) is often used to implement it. However, business decisions are increasingly dependent on textual data as well. Existing research work on textual data warehouses has the limitation of capturing contextual relationships when comparing only strongly related documents. This paper proposes an Information System (IS) based context-aware model that uses word embedding in conjunction with agglomerative hierarchical clustering algorithms to dynamically categorize documents in order to form the concept hierarchy. The results of the experimental evaluation provide evidence of the effectiveness of integrating textual data into a data warehouse and improving decision making through various OLAP operations.

1. Introducton

With the incessant growth of textual information in a variety of business systems, it has become more desirable for organizations to analyze both structured data records and unstructured text data simultaneously. In recent times in order to automate the process of organizational data analysis for extractions of business intelligence, the enterprises apply Information System (IS) based work systems (Struijk, Ou, Davison, & Angelopoulos, 2022). IS is a system in which human participants and/or machines perform work (processes and activities) using information, technology, and other resources to produce informational products and/or services for internal or external customers. Performing On-line Analytical Processing (OLAP) operations on data warehouses have been the most widely used technique by the organizations to implement IS enabled decision support systems.

While OLAP tools have been proven very useful for handling structured data, they face challenges in handling text data. Usually, data warehousing technologies and OLAP tools are unable to analyze textual data. Moreover, as OLAP queries of a decision-maker are generally related to a context, contextual information must be taken into account during the exploitation of data warehouses. OLAP systems allow navigation through multiple dimensions from one view to another which can be effectively used to analyze big data. In order to deal with textual data information retrieval (IR) techniques are generally used to evaluate the relevance of data to a query composed of simple keywords expressing needed information. Most often this relevance is based on the terms' frequency in the document. But in a text-OLAP system, the interest is in navigational analysis which may be based on operations corresponding to the analysis of text context at different levels in a data warehousing model.

1.1. The research question

Traditionally data analysis focuses on business data managed by a decision support system with data being mostly stored in data warehouses or structured files. In the era of digitalization and the prolific rise of big data, business analytics must evolve constantly. The volume of unstructured data is more rapidly growing in comparison to the growth of structured data. According to Gartner's magic quadrant of 2019, unstructured data is growing by 30% to 60% year over year. According to the figures from the ITC research firm, the volume of unstructured data is set to grow from 33 zettabytes in 2018 to 175 zettabytes, or 175 billion terabytes by 2025. In many complex fields, such as academics, research communities, company Human Resource activities, medical diagnosis, social media feedbacks, online customer feedback and customer support, decision-makers require helpful indicators and tools to make analyze text data and make business decision. Over the years, data warehouses and OLAP tools have emerged as most useful Information Systems of managing his huge volume of data assist users in the process of business decision making. Data warehouses can be implemented using several data models. Multidimensional database, represented by Multidimensional Data Model (MDM) is often a part of a data warehouse. This model is defined using set of dimensions and facts. The indicators to assess the facts are known as measures. Dimensions are the perspectives or entities based on which an organization wants to perform analytical

* Corresponding author.
E-mail addresses: santanuroy84@gmail.com (S. Roy), cortesi@unive.it (A. Cortesi), iamsoumyasen@gmail.com (S. Sen).

https://doi.org/10.1016/j.jjimei.2022.100129
Received 21 September 2021; Received in revised form 1 October 2022; Accepted 4 October 2022

processing. Each dimension may be associated with a hierarchy known as concept hierarchy. For the navigation and the visualization OLAP uses operations such as roll-up, drill-down, slice, and dice (Sen, Roy, Sarkar, Chaki, & Debnath, 2014).

The traditional OLAP tools are effective when data are numerical but they are not suitable for unstructured data such as text. Because of the fast growth of textual data there is a need for new approaches that take into account the textual content of data in OLAP analysis and it is called text-OLAP. However, this involves not only dealing with the heterogeneity of representations and granularities but also dealing with large volume of data. Large volume of text documents are generated everyday in every organization. Consequently, documents should be integrated into the decision support system. The perfect process of integrating unstructured data on the context of data warehouses is to manage, query and visualize information in a way that is as effective and meaningful with structured data.

In order to capture the notion of text-OLAP it is important to propose OLAP operations to process and analyze textual data and summarize them into an OLAP cube (Cuzzocrea, 2020) for fast and effective decision making. The long-established OLAP operations can't be applied in their conventional form due to the complex nature of unstructured text data (Zhang, Wang, & Feng, 2018).To use the OLAP with textual data, text mining provides the necessary techniques for textual aggregation. Research study reveals there have been few attempts to perform OLAP operations on text data but even the current state-of-the art algorithms on text-OLAP are unable to extract the semantic information from the texts with immaculate precision and accuracy. Embedding the semantics and context in textual data in OLAP analysis (Oukid, Benblidia, Asfari, Bentayeb, & Boussaid, 2015) and aggregating them to enhance the decision-making is a challenge in business intelligence systems. Therefore, it is imperative to modify the traditional data warehousing models and introduce new aggregation techniques (Sen et al., 2014) appropriate for text-OLAP. Most of the existing works employ information retrieval (IR) techniques (Kosmopoulos, Androutsopoulos, & Paliouras, 2015; Lin, Ding, Han, Zhu, & Zhao, 2008; Oukid et al., 2015) to evaluate the Semantic Textual Similarity (STS) between a set of text documents and an OLAP aggregation query containing simple keywords to express the desired information. Often this context analysis is based upon Term Frequency and Inverse Document Frequency (TF-IDF) or Bag-of-Word (BOW) methods (Chakrabarty, Roy, & Roy, 2018; Kim & Gil, 2019; Oukid et al., 2015; Ravat, Teste, Tournier, & Zurfluh, 2008). However these techniques are inadequate to capture the similar contexts across different levels of a dimension table. Thus, the results generated from IR systems suffer from the limitation of extracting contextual information in the development of decision support system (Sarkar & Shankar, 2021) from text data warehouse. Moreover, for a dimension having concept hierarchy (Sen et al., 2014), these feature based Vector Space Models (VSM) are often not suitable to extract the hierarchical relationships among documents due to their frequent near-orthogonality and inability to capture the semantic similarity as a metric of distance between different words having similar meanings or contexts.

This study identifies the possible opportunities for business analysis on text data warehouses by embedding context into the model and subsequently performing OLAP operations. These textual data can attribute to the different decision making process for any organization. In this study the authors propose IS based context-aware work system model that integrate word embedding with agglomerative hierarchical clustering algorithm to perform OLAP operations on textual data warehouses to generate IT enabled corporate reports that may aid in fast and effective business decision making.

1.2. Summary of the proposed methodology

The proposed model presents a novel methodology for the creation of a textual data warehouse with textual dimensions organized by contexts (set of topics) named as contextual dimensions and its implementation in a real OLAP system. This study uses star schema (Sen et al., 2014) to build the conceptual textual data warehousing model.

The proposed methodology processes the text documents and constructs a data cube around a central theme of analysis called fact table F defined by several dimensions Dim_R where $R \in [1, *]$. A set measure(s) of a fact table F is denoted by M, stores values to be aggregated. A Fact F with its dimensions Dim_R and set of measure(s) M, form a star schema model which is formalized as: $\$F$; Dim_1, Dim_2,..., Dim_* ; M_1, M_2,..., M_*).

After arranging the documents according to the star schema, proposed methodology combines word embeddings (De Miranda, Pasti, & de Castro, 2019; Ángel González, Hurtado, & Pla, 2020; Maas et al., 2011; Mikolov, Chen, Corrado, & Dean, 2013a; Mikolov, Sutskever, Chen, Corrado, & Dean, 2013b) (De Miranda et al., 2019) in conjunction with agglomerative hierarchical clustering method to group similar documents by extracting their contextual similarities. At first, the text documents are represented by their word embedding based centroid word vectors and then the hierarchical agglomerative clustering algorithm is applied to word centroid vectors to cluster the documents according to a concept hierarchy existing among the contextual dimensions. This overall approach is dynamic as it is not necessary to declare the number of clusters in the design hierarchy at the beginning of algorithm execution. The dynamically constructed hierarchy of concepts will comprise a sequence of hierarchical mappings ranging from a set of low-level concepts to a broader higher-level concept. The contextual dimension will allow the decision makers to analyze and query on the set of documents after selecting a context that has been automatically extracted during the formation of the concept hierarchy.

1.3. Validation of the proposed methodology

In order to validate our proposed model experimental studies have been carried out on huge sets of publicly available resumes (bio-data) collected from different job portals that can facilitate search using skill set, the domain of specialization, location of a person, and experience as contextual dimensions in the multi-dimensional text data warehousing model. The idea of working on resume dataset has been adopted from the research work carried out in the paper by Oukid et al. (2015). However, the collection of the resumes and the corresponding customization of the resumes into suitable format has been done by the authors of this paper. Therefore the authors work with the resume data set preprocessed and prepared by themselves. As an example, in the resume dataset, the dimension Topic (Dim_T) contains a concept hierarchy on the skill-set specialization domain. An example of concept hierarchy is illustrated in Fig. 1.

1.4. Novelty and the findings of the proposed methodology

The novelty and the findings of the proposed methodology may be listed as follows:

- The proposed methodology uses word embedding algorithm to represent each document by its centroid word vectors. Experiments have been carried out to show the effectiveness of this approach in extracting contextual similarity between documents having very few terms in common. In the result analysis section, it has been shown the proposed word embedding based approach is superior in extracting contextual similarity in comparison to the state-of-the-art VSM models using TF-IDF approaches. The enhanced performance by capturing contextual similarity of the proposed method is measured by the cosine based similarity measure.

- The agglomerative hierarchical clustering algorithm categorizes the text documents (resumes) according to a concept hierarchy. The proposed method shows highly improved performance in dynamically forming the concept hierarchy based on contextual dimensions in comparison to the state-of-the-art methods. The novelty of the proposed

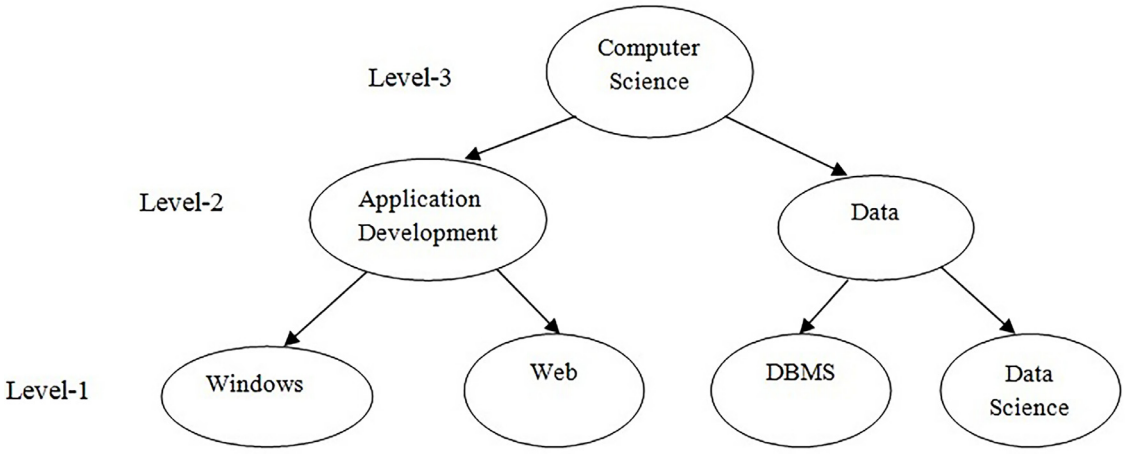

Fig. 1. Example of concept hierarchy of the dimension-Topic (Dim_T).

method of the formation of the concept hierarchy is understood by the following two facts: a) Number of clusters (concepts) is not needed to be declared before the execution of the algorithm. According to the proposed methodology, different linkage criteria of the agglomerative algorithm are applied and accordingly Silhouette Score(s) is measured by varying the number of clusters. The number of clusters and the corresponding linkage criterion producing the highest Silhouette score are chosen dynamically by the proposed algorithm. b) During the formation of the concept hierarchy new concepts are automatically added to the concept hierarchy based on their contextual similarity. This is a major improvement over the state-of-the-art algorithms which try to capture contextual similarity based on Vector Space Model (VSM). Existing works require the leaf concepts in the concept hierarchy to be supplied (as input) statically as a collection of a few terms or words belonging to the different concepts (topics). Thus the existing approaches are heavily limited in dynamically adding new concepts represented by new words or terms during the processing of the documents. c) After the formation of the concept hierarchy a case study (related to the H.R. Manager's activity in correspondence to a job advertisement and subsequent job applications by candidates) is carried out to perform OLAP aggregation operations on the set of documents. The proposed model in this research work can extract useful information to support business intelligence. d) The state-of-the-art algorithms represent the text documents by TF-IDF based vectors. With the increase in the number of documents, these sparse TF-IDF based vectors can be very high dimensional. In contrast, the proposed model represents each document by less dimensional dense centroid word vectors. Therefore, processing the less dimensional word vectors for contextual similarity computation with a posed OLAP query is much faster than the processing time required to scan through the high dimensional TF-IDF vectors. Hence, during experimental evaluation proposed model shows considerable improvement in speeding up the execution time of OLAP operations.

This paper can be seen as a further contribution in the recent discussions of the IJIM Data Insights on issues related to the application of data analytics techniques for decision support systems. In particular, it is worth mentioning that our contribution on analytical processing of textual data can be combined in an effective way with the recent advancements on Natural Language Processing (NLP) and Big Data Analytics (Atkinson & Escudero, 2022; Georgiadou, Angelopoulos, & Drake, 2020), as well as on security management of textual contents (Fujii, Sakaji, Masuyama, & Sasaki, 2022; Wadud et al., 2022), business intelligence (Unhelkar et al., 2022) and complex decision-making problems (Razavisousan & Joshi, 2022).

This paper aims to provide a model to perform OLAP operations in textual data warehouses. The proposed model can capture the contextual similarity between the documents and thus categorizes the documents according to a dynamically formed concept hierarchy existing among contextual dimensions.

1.5. Organisation of the paper

The rest of the paper is organized as follows: In Section 2, we discuss the related works in the domain of text-OLAP. The limitations of the existing works is discussed in Section 3. Section 4 presents the foundation concepts associated with the proposed methodology. The proposed methodology is described in Section 5. Materials and methods needed for the experimental evaluation are discussed in Section 6. Experimental results and the performance analysis of the proposed methodology is discussed in Section 7. Discussion on the effectiveness work is presented in Section 8. Section 9 concludes.

2. Related work

In this digital era, data warehouses are extensively used in the industry for organizing and analyzing large amounts of data. A survey work presented in Bouakkaz, Ouinten, Loudcher, & Strekalova (2017) broadly classifies the text-OLAP and aggregation techniques 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 the data structure. Approaches that are not based on the data structure are further classified into four subcategories, approaches based on linguistic knowledge, approaches based on external knowledge, approaches based on graphs and approaches based on statistical information. Details concerning these approaches are developed next.

2.1. Approaches based on data structure and data models

The X-OLAP: (XML-OLAP) proposed by Park, Han, & Song (2005) is based on the text mining approach. XML-OLAP is based on the text mining technique that aggregates the text content of XML documents. This approach to analyzing XML documents stored in a data warehouse is represented by a multidimensional model.

The DocCube: DocCube was introduced by Mothe, Chrisment, Dousset, & Alaux (2003). 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 like the content of a fact table that contains links.

Topic Cube: Zhang, Zhai, & 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 discovers the content of the text documents.

Text Cube: In order to introduce the semantic aspect in the textual aggregation Lin et al. (2008) proposed an approach for data cube called Text Cube. The main idea is to give the user the possibility to make semantic navigation in the data dimension. To achieve that, two OLAP operations such as the pull-up and push-down.

The R-Cube: Perez, Aramburu, Berlanga, & Pedersen (2007) 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.

The Cube Index: Azabou, Khrouf, Feki, Soulé-Dupuy, & Vallès (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. 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.

2.2.1. Approaches based on linguistic knowledge

The approaches based on linguistic knowledge consider a corpus as a set of the vocabulary mentioned in the documents but the results are sometimes ambiguous. To overcome this obstacle, techniques based on lexical knowledge and syntactic knowledge previews have been introduced. Kohomban & Lee (2007) 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 scientific concepts, rather than adverbs, verbs, or adjectives.

2.2.2. Approaches based on external knowledge

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, Song, Teste, & Trojahn (2020) 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 et al. (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. The same concept propagation technique has been used in the research work discussed in Chakrabarty et al. (2018). The work (Chakrabarty et al., 2018) uses a context-aware Fuzzy Classification based technique to capture the semantic ontology from the text documents and classifies them to aggregate in terms of their relevant concepts.

2.2.3. Approaches based on graph

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 keyword-ranking approaches have been tried. The first proposed is an approach called TextRank (Mihalcea & Tarau, 2004) where the edges represent the co-occurrence relations between the keywords. Two successive research works by Bouakkaz, Loudcher, & Ouinten (2016) focus on textual aggregation techniques. In their earlier work Bouakkaz et al. (2016) proposed a method that performs aggregation of keywords of documents based on the construction of a graph using the affinities between keywords. Term Frequency (TF) based keyword extraction technique has been used in this work. The following work (Bouakkaz et al., 2017) tries to capture semantic aggregation of the keywords by applying k-means algorithm using Google Similarity Distance Measure.

2.2.4. Approaches based on statistical methods

The approaches based on statistical methods use the occurrence frequencies of terms and the correlation between terms. Landauer, Foltz, & 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. Ravat et al. (2008) proposed a second aggregation function called TOP-Keywords to aggregate keywords. They computed the frequencies of terms using the TF-IDF function, and then they selected the first k most frequent terms.

The papers discussed in this Section offer quite a large choice of methodologies applicable to a variety of datasets to perform OLAP operations on text data. Most of the techniques conglomerate text mining approaches with OLAP aggregation operations.

3. Limits of the state-of-the-art methodologies

It is identified from the literature survey of Section 2 that the existing research works on text-OLAP suffer from the following limitations. 1 summarizes a few of the works which deal with context-aware textual data warehouses.

1. Works described in Azabou et al. (2015), Bouakkaz et al. (2016), Chakrabarty et al. (2018), Manuel Pérez-Martínez, Berlanga-Llavori, Aramburu-Cabo, & Pedersen (2008), Oukid et al. (2015), Ravat et al. (2008) try to focus on capturing contextual information during text-OLAP analysis. However, in all of these schemes, the documents are represented using either of the models between the BOW model, TF calculation, or using TF-IDF feature vectors. These techniques are often not suitable to grasp the semantic relationships between contexts during the comparison of related parts of different documents. In BOW representations each word of the vocabulary is represented as a 'one-hot' vector with as many components (features) as the size of the vocabulary, and only one non-zero component (corresponding to the particular word). Thus the resulting vector is a high dimensional sparse vector (mostly zero components). Standard feature selection algorithms can be used to reduce the dimension. However, if the concept hierarchy is formed in bio-medical text document datasets for OLAP analysis then the number of concepts (class) may extend up to the order of a few thousands (Kosmopoulos et al., 2015). In these kinds of scenarios even with the least number of features per class (after application of the feature selection algorithm), the total number of features representing each document may contain a significantly huge number of features in the Vector Space Model (VSM). OLAP query processing in these high dimensional feature vectors can be very slow.
2. Regarding text-OLAP, very little number of works have addressed the concept hierarchy (Sen et al., 2014) existing in a certain contextual domain. Studies suggested in Chakrabarty et al. (2018) and Oukid et al. (2015) highlight the contextual dimensions having concept hierarchy. Both of the approaches use the relevance/concept propagation technique to calculate contextual term weights of the documents across the different levels of the concept hierarchy. However, both of these assume a static structure of the concept hierarchy with a few arbitrary terms related to a concept. This method is highly inefficient as any concept (other than the statically mentioned concepts at the beginning) discovered with the increasing size of the dataset does not get categorized into a proper domain of specialization topic (class). Study

Table 1
Comparison of the works on context-aware text-OLAP.

Works	Data Format	Approach used	Formation of Concept Hierarchy	Nature of Concept hierarchy
(Ravat et al., 2008)	XML	TF-IDF	No	NA
(Azabou et al., 2015)	Text	TF-IDF	No	NA
(Lin et al., 2008)	Text	Cube	No	NA
(Bouakkaz et al., 2016)	Text	Graph/TF-IDF	No	NA
(Bouakkaz et al., 2017)	Text	Graph/TF-IDF	No	NA
(Oukid et al., 2015)	Text	TF-IDF	Yes	Static
(Chakrabarty et al., 2018)	Text	TF-IDF	Yes	Static
Proposed Work	Text	Word Vector	Yes	Dynamic

suggested in Bouakkaz et al. (2017) tries to aggregate the keywords by using Google Similarity Distance Measure. However, this study also suffers from the problem of static declaration of the number of clusters as it uses k-Means algorithm to find the similarity between keywords.

4. Preliminaries and theoretical foundations

Our proposal (discussed in Section 5) is based on a list of theoretical results already introduced in the literature. In this section we list them systemically, giving credit to who introduced them and we indicate how they constitute a determining element in our solution.

An overview of the word embedding technique and associated Word2Vec algorithm is presented in Section 4.1. The use of word embedding based centroid vector has also been highlighted in Section 4.1. The proposed methodology uses hierarchical agglomerative clustering algorithm applied over the centroid vectors to categorize the documents according to the concept hierarchy. An extensive discussion is made in Section 4.2 on agglomerative hierarchical clustering algorithm with different linkage criteria. The utility of dendrograms in the proposed methodology is also explained.

4.1. Word embedding

In recent few years, word embeddings have generated a lot of interest in the text analysis (Ángel González et al., 2020) research domain ever since two very simple log-linear models (Mikolov et al., 2013a; Mikolov et al., 2013b) were proposed that outperformed all previous of NLP models. Word2Vec has become the most reliable technique to be used as the basis of all NLP models. Of course, there have been proposals on improvement using Deep Learning Recursive Neural Networks (RNN) based on Long-Short Term Memory (LSTM) (Alcamo, Cuzzocrea, Bosco, Pilato, & Schicchi, 2020) nodes and also the very recent BERT algorithm (Devlin, Chang, Lee, & Toutanova, 2019), but in the last five years word embedding has proven to be a strong baseline. Both the Deep Neural-Net LSTM model and word embedding based models are scalable to very large corpus sizes and produce accurate results. However, the word embedding model is very simple in architecture. Word embedding based algorithms also have the advantage of drastically reduced time complexity. Therefore recent works (Krishna & Sharada, 2019; Periñán-Pascual, 2021) on capturing semantic context in the text are still employing the word embedding technique as one of the state-of-the-arts in the text mining task.

We briefly explain the working principle of the Skip-gram model. Let there be a corpus, a sequence of words w_1, w_2,..., w_T. The window is defined by parameter c, where c words at the right and left of the target are taken. For Skip-gram, each context is predicted independently given the target. The objective function to be maximized is defined as :

$$\frac{1}{T}\sum_{t=1}^{T}\sum_{-c\le j\le c, j\neq 0}\log p(w_{t+j}\mid w_t) \tag{1}$$

Skip-gram models the probability of word w_{t+j} being observed within word w_t's context window as a probability. The probability $p(w_{t+j}\mid w_t)$ is defined as a softmax, where $u_w$is a target embedding vector for w and v_w is a context embedding vector. The u_w embeddings are the ones that are kept, v_w is a side product. The following definition is used for Skip-gram:

$$p(W_c\mid W_t)=\frac{\exp v_{wc}^{T}u_{wt}}{\sum_{w=1}^{W}\exp v_w^{T}u_{wt}} \tag{2}$$

Finally, using these equations word embedding vectors of the text documents are generated.

The similarity between any two embedding vectors represented as $\vec{w}_i$ and $\vec{w}_j$ respectively, is measured by the Cosine Similarity distance (Oukid et al., 2015) value and is calculated as:

$$CSM(\vec{w}_i,\vec{w}_j)=\frac{\vec{w}_i.\vec{w}_j}{||\vec{w}_i||.||\vec{w}_j||} \tag{3}$$

4.1.1. Document centroids

Having computed the dense vectors of all the vocabulary words, the simplest method to obtain a dense vector (of the same dimensionality) for a text document $d=\langle w_1, w_2, ..., w_n\rangle$ of n consecutive word occurrences is to simply compute the centroid $\vec{d}$ of the dense vectors $\vec{w}_i$ of the word occurrences:

$$\vec{\mathbf{d}}=\frac{1}{n}\sum_{i=1}^{n}\vec{w}_i=\frac{\sum_{j=1}^{|v|}\vec{w}_j.TF(w_j,d)}{TF(w_j,d)} \tag{4}$$

However, as suggested by Kosmopoulos et al. (2015), we hereby compute the document centroid vectors using Eq. (5) by taking its IDF scores of tokens/words into consideration. As shown in Kosmopoulos et al. (2015), this modification results in improved document categorization performance.

$$\vec{\mathbf{d}}=\frac{\sum_{j=1}^{|v|}\vec{w}_j.TF(w_j,d).IDF(w_j)}{\sum_{j=1}^{|v|}TF(w_j,d).IDF(w_j)} \tag{5}$$

Here $|v|$ is the vocabulary size, w_j is the j-th vocabulary word, $\vec{w}_j$ represents its embedding, TF(w_j,d) is the term frequency of w_j in d and IDF(w_j) represents the inverse document frequency of w_j.

4.2. Agglomerative hierarchical clustering algorithm

The agglomerative clustering technique performs a hierarchical clustering using a bottom up approach to form the concept hierarchy existing in the contextual dimension. The distance between two clusters has been computed based on the length of the straight line drawn from one cluster to another. We have represented the documents using word vectors, therefore, as discussed in Mikolov et al. (2013a, 2013b) if they are mapped into the Euclidean Space, it may be observed that the similar pair of words tend to exhibit similar displacement vectors. Such that the straight line distance between 'DBMS' and 'MS-Access' will be equal to the straight line distance between 'Data Science' and 'Python'. Keeping this property of the word vectors in mind, in the proposed methodology we use the Euclidean Distance as the distance metric to compute the distance between two clusters. After selecting a distance metric, it

Table 2
Parameters of the Lance-Williams update formula for different agglomeration methods with the definition of dissimilarity measure.

Method	α_i	α_j	β	γ	Dissimilarity Measure
Single Linkage	0.5	0.5	0	0.5	d_{ij}
Complete Linkage	0.5	0.5	0	0.5	d_{ij}
Average Linkage	$\frac{N_i}{N_i+N_j}$	$\frac{N_j}{N_i+N_j}$	0	0	d_{ij}
Ward	$\frac{N_i+N_m}{N_i+N_j+N_m}$	$\frac{N_i+N_j}{N_i+N_j+N_m}$	$\frac{-N_m}{N_i+N_j+N_m}$	0	d_{ij}^2

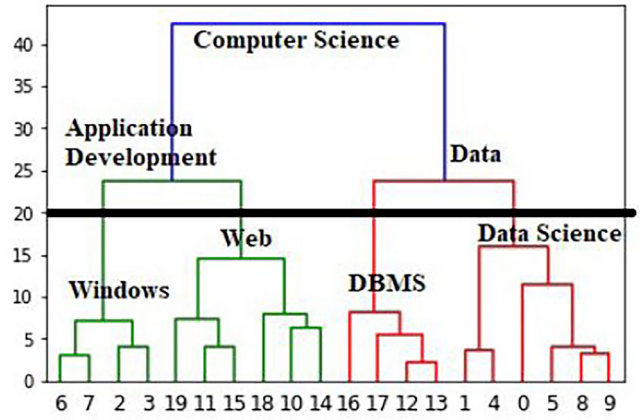

Fig. 2. Dendrogram using average linkage agglomerative clustering algorithm.

is necessary to determine from where the distance is computed (linkage criterion). For example, it can be computed between the two most similar parts of a cluster (single-linkage), the two least similar bits of a cluster (complete-linkage), the center of the clusters (mean or average-linkage), or some other criterion. Many linkage criteria have been developed. All hierarchical methods described in this work can be easily implemented through the widely used Lance-Williams dissimilarity update formula (Theodoridis & Koutroumbas, 2009). Lance-Williams updated formula allows us to calculate this distance straightforwardly, according to the following equation:

$$d_{km} = \alpha_i.d_{im} + \alpha_j.d_{jm} + \beta.d_{ij} + \gamma.|d_{im} - d_{jm}| \quad (6)$$

The coefficients α_i, α_j, β and γ define the agglomerative criterion. The two different types of dissimilarity measures are also described in Table 2. The dissimilarity measure can either be the Euclidean distance or its squared value. N_i, N_j and N_m are the number of documents in clusters i, j and m respectively.

4.2.1. Use of dendrogram in the formation of concept hierarchy

During the formation of the concept hierarchy the dendrogram (Ángel González et al., 2020) structure has been referred to get a visual representation for working out the number of concepts (clusters) of the formed hierarchy. It is created as an output of hierarchical clustering algorithm and displayed graphically as a tree diagram. The use of a dendrogram is to determine the number of clusters that best fits the data in terms of compactness and closeness. The different parts of a dendrogram are demonstrated in Fig. 2. This dendrogram is achieved by experimenting with 20 random documents (CVs) selected from our dataset. The horizontal axis indicates the number of documents and the vertical axis corresponds to the dissimilarity measure between the documents.

In the proposed methodology we have used the Silhouette Coefficient(s) (Shahapure & Nicholas, 2020) to more accurately select the optimal number that best fits the documents. The cut-off method has also been used in dendrograms to visually represent the concepts in the hierarchy. The Silhouette Coefficient (s) is defined for each sample and is composed of two scores: a- mean distance between a sample and all other points in the same class and b- mean distance between a sample and all other points in the next nearest cluster.

The Silhouette Coefficient s for a single sample is then given as:

$$s = \frac{b-a}{max(a,b)} \quad (7)$$

The best value is 1 and the worst value is -1. For agglomerative hierarchical clustering, the Silhouette Coefficient can be computed for several cuts (k=2,3,..,N-1). The user selects the k with the maximum Silhouette Coefficient value.

5. Proposed methodology

In this Section, we introduce and discuss a novel context-aware model that uses word embedding in conjunction with agglomerative hierarchical clustering algorithms to dynamically categorize documents in order to form the concept hierarchy.

The proposed methodology can be broadly divided into 4 steps: (i) The raw text files are first represented as document centroid vectors (see Section 5.1), (ii) The agglomerative hierarchical clustering algorithm is applied for the formation of the concept hierarchy (see Section 5.2). Two novel algorithms to perform the first two tasks are proposed in this paper. (iii) Clusters are labelled according to the relevant concepts (see Section 5.3). Finally, (iv) OLAP aggregation operations are performed according to the business requirement (see Section 7.4). A schematic diagram representation of the methodology is provided in Fig. 3.

5.1. Computation of document centroid vectors

The initial preprocessing of raw text involves the removal of stop words present in the documents to prepare a corpus of meaningful terms. Upon the large corpus of documents (resumes), Skip-gram algorithm is executed to form the word embedding vectors. Subsequently document centroids are computed using Eq. (5) presented in Section 4.1.1. As a result the text documents are represented as a N-dimensional [N=50,100,200] dense word vectors. The steps for the formation of document centroid vectors from the raw text documents are formalized in the form of a novel algorithm and presented as Algorithm 1.

5.2. Categorization of documents using hierarchical agglomerative clustering

After representing the text documents as N-dimensional document centroid vectors (output of Algorithm 1), we apply the agglomerative hierarchical clustering algorithm to the centroid vectors to categorize them into a set of clusters. Each cluster represents a concept of a dimension having concept hierarchy. We have used the state-of-the-art standard agglomerative hierarchical clustering algorithm (Theodoridis & Koutroumbas, 2009) keeping the linkage criterion generic. Lance-Williams dissimilarity update formula (Eq. (6)) has been used as the generic dissimilarity measure. However, the final choice of the linkage criterion and selection of the number of clusters has been decided based on the Silhouette Coefficient (s) (Eq. (7)) score being obtained during the experiment. The linkage method producing the highest Silhouette score value is considered to be the suitable agglomeration linkage criterion for a particular dataset. The methodology of the formation of the concept hierarchy is presented as Algorithm 2.

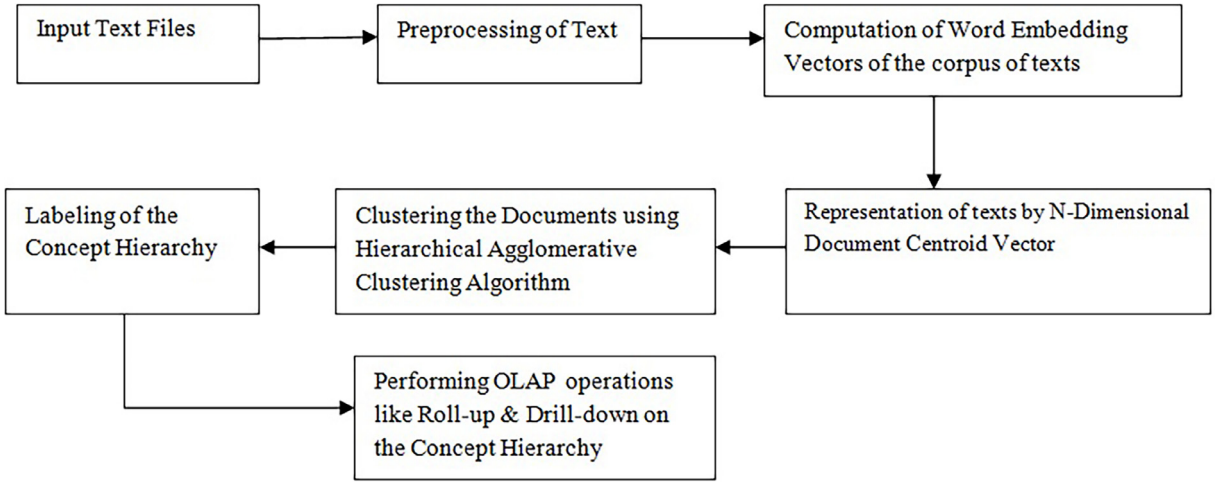

Fig. 3. Schematic diagram of the proposed methodology.

Algorithm 1 Construction of the word embedding based document centroid vectors.

Input: $\{d_i : d_i \in D\}$ - Documents (in raw text format) in the corpus D,
N - Number of documents,
c - Window size of the Word2Vec model,
D' - Dimensionality of the embedding
$|v|$ - Vocabulary size
Result: $\{d_i : d_i \in D\}$ - Centroid vectors of each document
Preprocessing: Each document d_i is preprocessed by performing stop word removal and tokenization. Each preprocessed document d_i is represented as a sequence of meaningful words. $d_i = \langle w_1, w_2, ..., w_j \rangle$
Repeat $\forall d_i : d_i \in D$
Repeat $\forall w_j : w_j \in d_i$
- Compute Term frequency $TF(w_j, d_i)$ of the j^{th}vocabulary word w_j in document d_i by counting the number of occurrences of w_j in d_i.
- Compute the Inverse Document Frequency $IDF(w_j)$ of the j^{th} vocabulary word in document d_i as: $log\frac{N}{N(w_j)}$, $N(w_j)$ is the number of documents containing the word w_j.
- Compute the word embedding vector w_j of the word w_j belonging to d_i by executing the Skip-gram algorithm with window size c.
- Compute the document centroid vector d_i of document d_i by combining the results of Term Frequency, Inverse document Frequency and word embedding vector of each word w_j belonging to document d_i by using Eq. (5) as:

$$\boldsymbol{d_i} = \frac{\sum_{j=1}^{|v|} \vec{w}_j . TF(w_j, d_i) . IDF(w_j)}{\sum_{j=1}^{|v|} TF(w_j, d_i) . IDF(w_j)}$$

End{Repeat}
End{Repeat}
Return$\{\vec{d}_i : d_i \in D\}$

5.3. Labeling of the concept hierarchy

After applying the agglomerative hierarchical clustering algorithm the documents are clustered in a hierarchy of concepts. Labeling is the task of selecting descriptive and human-readable labels/names for the clusters that summarize the concept or topic of the clusters. Labels distinguish the clusters from each other. This is the only step that requires human intervention.

In the proposed methodology we pick the twenty top most frequently occurring terms belonging to a particular cluster concept. After that, we consulted domain experts and referred to query logs on related topics from the FAQs in different job portals. Thereafter we were suggested with one or more concept names as labels for that particular cluster but finally, we chose the one which has the highest average cosine similarity value with the twenty top most frequently occurring terms. The predecessor concepts in the hierarchy are marked by the clades of the produced dendrogram. Combining the descendant clusters, the connecting clades are also labeled with appropriate concept names. Finally, the root of the hierarchy is labeled with the most generalized concept that covers all the descendant concepts.

Algorithm 2 Dynamic formation of concept hierarchy.

Input: $\{\vec{d}_i : d_i \in D\}$ - Document centroid vectors (Output of Algorithm-1),
N - Number of documents,
Linkage criterion (selected based on maximum Silhouette score s),
$d_{(c_i,c_j)}$ - Distance metric (using Eq. (6)) between any two clusters c_i and c_j, $\forall c_i, c_j \in C$
α_i, α_j, β and γ - Values of coefficients defining linkage criterion
Result: Assignment of documents in the concept hierarchy
Repeat $i = 1$ to N
- $c_i = \{\vec{d}_i\}$
End{Repeat}
$C = \{c_1, c_2, ..., c_N\}$
Repeat $C.size > 1$
- $\{c_{min1}, c_{min2}\} =$ minimum $d_{(c_i,c_j)}$, $\forall c_i, c_j \in C$
- Remove c_{min1}, c_{min2} from C
- Add $\{c_{min1}, c_{min2}\}$ to C
End{Repeat}

6. Materials and methods for experimental evaluation

We accomplished the experiments to illustrate the effectiveness of the proposed methodology by using the following components.

6.1. Benchmark datasets creation

The selection of the dataset is crucial to explain the functionalities of the proposed method. Here resume dataset is chosen for the following reasons: (i) Resumes are generally structured in nature and hence different contextual dimensions can be well defined. In a text-OLAP environment users or decision makers usually pose queries based on the notion of context. For example the candidature of a candidate can be shortlisted by referring to several contextual dimensions together, like- expertise in skill-set,years of experience,location of work,qualification,etc. Thus contextual information during the exploitation of data warehouses

Table 3
Dataset description.

Dataset	Characteristics	No. of Samples	No. of Features	Value range	Missing values	Nature	No. of classes
Dataset-I	Multivariate	850	200	0.0 - 1.00	No	Dense	5
Dataset-II	Multivariate	80	50	0.0 - 1.00	No	Dense	4
Dataset-III	Multivariate	80	200	0.0 - 1.00	No	Sparse	4

must be taken into account. With the structured format of the resume dataset, it is easier to segregate the different partitions of a resume based on contextual factors. The contextual factors can be represented as contextual dimensions. For example, a resume can be easily formatted and segmented into a collection of contextual dimensions like, skill-set, location, etc. (ii) Some of these contextual dimensions such as skill-set and location maintain concept hierarchy, therefore the OLAP operations such as roll-up, drill-down can be performed. OLAP analysis on the contextual concept hierarchy allows a decision-maker to navigate through the OLAP cube and to observe the data along several analysis axes organized in different hierarchical levels. For instance, the decision-maker can observe the competencies in 'Computer Science' for the year 2022 in India and then, by a drill down operation he observes those for the city Kolkata, etc. Further drill down can also be done on skill-set by observing 'Computer Science' in a more detailed view of specializations in 'DBMS', 'Web Technology', 'Machine Learning' etc.

As a workbench, we have considered the resume dataset (Chakrabarty et al., 2018; Oukid et al., 2015) for experimental purposes. The dataset was obtained from different resume portals from where 850 resumes of candidates belonging to various job specializations were collected from different publicly accessible job portals available online at websites like: www.freeresumesites.com, www.resumeworld.com, www.eresumex.com, www.freshersworld.com, www.linkedin.com, etc. Thereafter, the raw text documents have been preprocessed to prepare the datasets with the desired format.

Experiments have been carried out on three benchmark datasets, referred as: Dataset-I,[1] Dataset-II[2] and Dataset-III.[3] Table 3 presents a description of the dataset.

- Preprocessing:

After the collection of resumes, we manually extracted and formatted the documents according to the dimensions in our proposed star schema for text-OLAP analysis. In the proposed OLAP model, we have experimented with three dimensions: Skill-set specialization Topic (Dim_T), Location of candidate (Dim_L), and Experience (Dim_E). After extraction of the relevant sections from the resumes, data cleaning techniques such as Text Tokenization, and Stop-words removal are performed to create the final text corpus. Job profile resumes are notably different text documents with a higher density of specialized terminology. Hence, we haven't performed stemming techniques on our dataset. For example, terms like 'Data-Mining', 'Deep-Learning', 'Encapsulation' remain unaltered in our methodology, However, as a result, terms like 'Program' and 'Programming' turned out to be very similar in the Word2Vec Model. Similarly, the text is also not converted in lower case as certain terms like 'R', 'RDBMS', 'MVC', 'SVM' are always written in upper case.

6.2. Software and libraries

To validate our model, we have developed our prototype application written in Python-3.6 with Application Software Spyder (64-bit) and IDE Anaconda-3. For text preprocessing, developing the word-embedding and centroids of documents by importing the Word2Vec model and executing agglomerative clustering algorithm, the following libraries have been used- nltk, numpy, scikit-learn, scipy, tensorflow, gensim etc. We test skip-gram neural architecture by varying the embedding sizes. To find the best parameters configuration, we run a grid search using this setting: embedding size [50,100,200] and finally settle with 200, topic and similarity thresholds respectively in [0,0.5] and [0.5,1] with a step of 0.01. Window size is taken as 5. The centroid of the documents is stored in a numpy array of shape: Number of documents × Embedding size. Later we store the values (centroid vectors) of the numpy array in a.csv file. The hierarchical agglomerative clustering is executed on the.csv file.

7. Experimental results

In this section some experimental results are discussed.

7.1. Performance of document centroid vector in capturing semantic text similarity

In this section, the effectiveness of the document centroid vector in capturing semantic text similarity is discussed. The Word2Vec Skip-gram algorithm is applied over the large corpus of text documents to form the word vectors. Each text document is then represented by its document centroid vector. Word2Vec allows learning complex semantic relationships using simple vectorial operators (Mikolov et al., 2013a). For example, it may be written as: $vec(DBMS) - vec(Access) + vec(Python) = vec(DataScience)$.

In Fig. 4, we can see the Word2Vec t-SNE (Van der Maaten & Hinton, 2008) visualization of our implementation, using the resume dataset and a window size of 2. The t-SNE algorithm reduces the dimensionality of the vectors and plots the high dimensional word vectors into 2-dimensions. For a better visibility, we have partially shown the right hand side figure with the zoomed portion of the diagram having a vocabulary size $|v|$ of 700 words. On the right hand side figure, it can be seen that words with semantic similarity are in close proximity with each other. The diagram has been grouped into three clusters: marked as Black, Red, and Green. The words enclosed in the black coloured group are: Association, Data Science, R, Python, Classification and Data Warehouse. They signify that they belong to the concept-Data Science and Machine Learning. The blue marked cluster has the highest number of contextually related words, like: C, C++, VB, VC++, PHP, HTML, ASP.NET, MVC, Java, Hibernate, Struts, API, JSON, Tomcat etc. Based on the context, it can be said that these highlight the specialization of Application development Programming. The third cluster marked in red contains words related to job specialization in Database handling with keywords like: DBMS, RDBMS, SQL, PL/SQL, Access, Stored Procedure and Oracle etc. This observation proves how Word2Vec can group the semantically similar words into close distanced vector space vicinity.

7.2. Evaluation of proposed centroid based method over TF-IDF models in capturing semantic similarity

In order to illustrate the effectiveness of the method let's consider two small pieces of paragraphs extracted from two separate resumes with specialization in Application development. **Text-1**: *"Experience in*

[1] https://drive.google.com/file/d/1t96dYfcMTnAkx7c_4iMLlHlJ8baHng60/view?usp=sharing.
[2] https://drive.google.com/file/d/1lRmuGiY0OpI6yoYQ09X60sZMLgROTuG9/view?usp=sharing.
[3] https://drive.google.com/file/d/1R_Uaf2cFmpWsgYSkyfQ8yv7Ovne38576/view?usp=sharing.

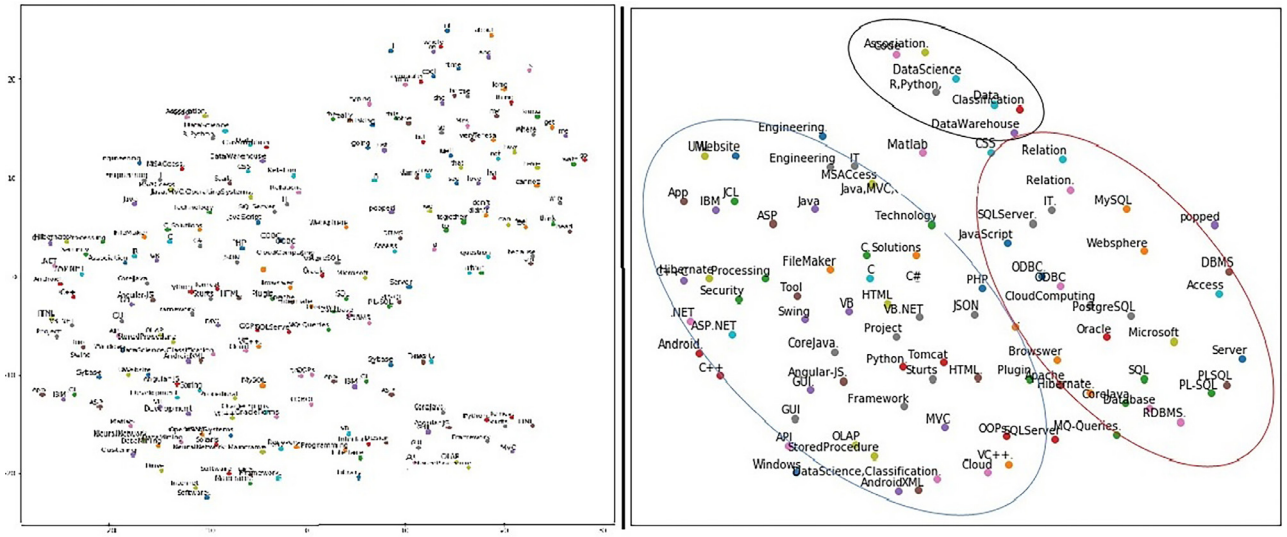

Fig. 4. Left: Word relationships using dimensionality reduction by t-SNE algorithm Right:Zooming in of the rectangle in the left figure.

Table 4
Semantic similarity between word vectors and document centroid.

Experience	ASP.NET	C#	coding	Centroid embedding
Work (0.872)	.NET (0.913)	ASP.NET (0.986)	code (0.985)	Working
Working (0.868)	C# (0.901)	.NET (0.978)	Programming (0.864)	C#
years (0.712)	VB.NET (0.834)	VB.NET(0.902)	programming(0.845)	.NET
Professional(0.644)	framework(0.77)	programming(0.837)	language(0.733)	programming
Industry (0.605)	MVC (0.654)	application (0.745)	skill (0.662)	-

Table 5
Cosine similarity between the word vectors of Tex-1 and Text-2.

Word Vector of Text-1	Word Vector of Text-2	Cosine Similarity between Text-1 and Text-2
Experience	Working	0.868
ASP.NET	Java	0.563
C#	Hibernate	0.474
coding	framework	0.558
Average Similarity		0.616

ASP.NET in C# coding" and **Text-2:** *"Working in Java Hibernate framework"*. After cleaning (excluding stop words) the modified texts look like: **Text-1:** *"Experience ASP.NET C# coding"* and **Text-2:** *"Working Java Hibernate framework"*.

Since Text-1 and Text-2 have no words in common, thus the document similarity between these is zero according to BOW or TF-IDF models. Experimental results depicted in Table 4 prove the superiority of the proposed centroid based method. The table header corresponds to the words of Text-1 and the last column contains the most similar word to the centroid embedding computed using Eq. (5).

In Table 5, we show the pairwise cosine similarity between Text-1 and Text-2 calculated using the Skip-gram word vector model. We find the average pairwise word similarity between Text-1 and Text-2 is 0.616 which is a major improvement over the TF-IDF model where the similarity value turned out to be zero between the two pieces of texts.

7.3. Evaluation of the agglomerative hierarchical clustering algorithm

In this section, we present the experimental result of deciding the number of clusters in the hierarchy by applying different linkage criteria, such as Single, Complete, Average, and Ward. The total number of clusters (k) in the hierarchy is chosen based on the maximum Silhouette score (s).

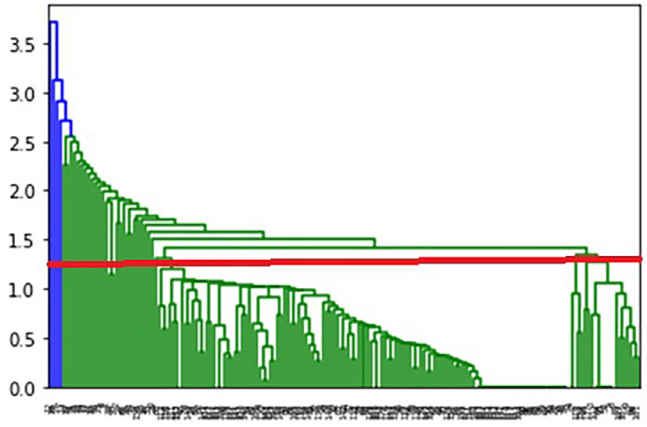

Fig. 5. Dendrogram for average linkage criterion agglomerative clustering algorithm applied on 375 documents.

When the corpus is made up of 375 text documents (resumes), it can be seen in Table 6 that the maximum Silhouette score ($s = 0.419$) is achieved with the Average Linkage method for the number of clusters (k)=4. Table 6 also confirms that with 850 documents maximum Silhouette score ($s = 0.467$) is obtained by using Ward linkage criterion for the number of clusters=5. Hence the number of clusters (k) is chosen as 5 for 850 documents.

The result of Table 6 can be visualized in Fig. 5 which represents the dendrogram produced by applying the agglomerative algorithm on 375 text documents. If the cut-off line is set at value 1.25 (marked in red) then we get 2 distinct sets of separated clusters. The large numbered left hand side clusters are similar to each other and represent the

Table 6
Linkage criteria wise Silhouette scores with varying number of documents and clusters.

Number of Documents	No. of Cluster	Silhouette Score (s)			
		Single Linkage	Complete Linkage	Average Linkage	Ward Linkage
375	2	0.283	0.346	0.290	0.349
	3	0.324	0.357	0.325	0.378
	4	0.362	0.403	0.419	0.405
850	3	0.392	0.364	0.391	0.367
	4	0.380	0.407	0.380	0.424
	5	0.407	0.466	0.406	0.467

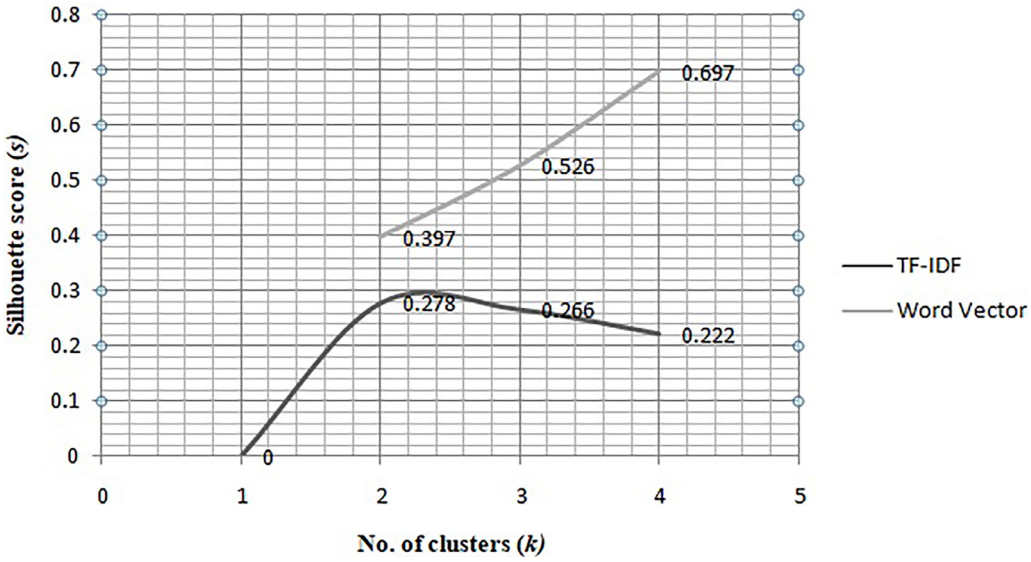

Fig. 6. Comparison of Silhouette scores (s) between proposed model and TF-IDF vector models.

concept 'Application Development'. On the other hand, the right hand side clusters represent the concept 'Data'. The formed concept hierarchy has already been shown in Fig. 1. If we drill-down even further in the concept hierarchy then it can be seen that 'Level-1' of Fig. 1 consists of 4 concepts. Therefore, the documents should be ideally categorized in 4 clusters. However, with bare eyes it is quite difficult to set a cut-off value in the dendrogram of Fig. 5 to select 4 clusters. In such scenarios, the determination of the optimal number of clusters is made by referring to the Silhouette score (s).

7.3.1. Performance comparison of centroid vectors over the TF-IDF models in the formation of clusters

A comparative study (in terms of Silhouette scores (s) between the centroid word vectors over the TF-IDF based models is depicted in Fig. 6. The agglomerative hierarchical clustering algorithm is executed on both Dataset-II and Dataset-III containing word vector and TF-IDF vector respectively. Fig. 6 confirms for all three values of the number of clusters (2,3 and 4), the Silhouette scores achieved from the centroid word vector are better than the ones achieved from the TF-IDF vector.

7.4. Case study on OLAP operations

The final concept hierarchy formed using Dataset-I (875 documents), is represented in Fig. 7. With the inclusion of new documents, the cluster (concept) 'Data Science' of Fig. 1 (375 documents) has been further divided into two separate clusters 'Machine Learning' and 'Data Warehouse'. Therefore the total number of clusters/concepts is 5 (Number of leaf nodes) according to Fig. 7. In this section we focus on a case study where an H.R. Manager wants to study the resumes that are submitted in response to a recruitment advertisement. Traditional methods of resume screening by the H.R. managers are not a standardized process. Earlier it has been done manually. Now-a-days, they have a computerised tool where they put the requirement by the organization by specifying the parameters as per the requirement of the different departments of the organization. The computerized tools are developed as per the requirement of the organization and parameters specific. These tools work by just matching the terms of the given parameters. Therefore if the terms are different but belong to the same category or domain it fails to identify the required skill level. These tools lack the intelligence as no context awareness are incorporated.

In the proposed model, since resumes sharing similar context (domain) are hierarchically clustered, thus searching the documents from the highest level of generalized requirement to the most specific requirement is much more efficient. Various OLAP operations (like-roll-up, drill-down) are performed which will extract knowledge and aid the manager in making decisions.

In order to include the other two dimensions along with the dimension Topic (D_T), each document is tagged with the working Location of the candidate (Dim_L) and job Experience (Dim_E). After clustering the documents according to the concept, we have represented each document according to its dimension values to carry out further OLAP operations. For example, a random sample under the concept 'Web Application Development' (member of cluster 'Web') may be expressed as $\langle id-678, \}Kolkata', 8\rangle$. This means a resume of a candidate with document id-678 has been categorized with specialization in web application development having 8 years of industry experience and is currently located in Kolkata.

If the H.R. manager of a company wants to analyze the resumes for candidates working in Kolkata then he would like to perform the slice operation on the dimension Location (Dim_L) using the selection criteria Location= 'Kolkata'. On the other hand, a dice operation by using se-

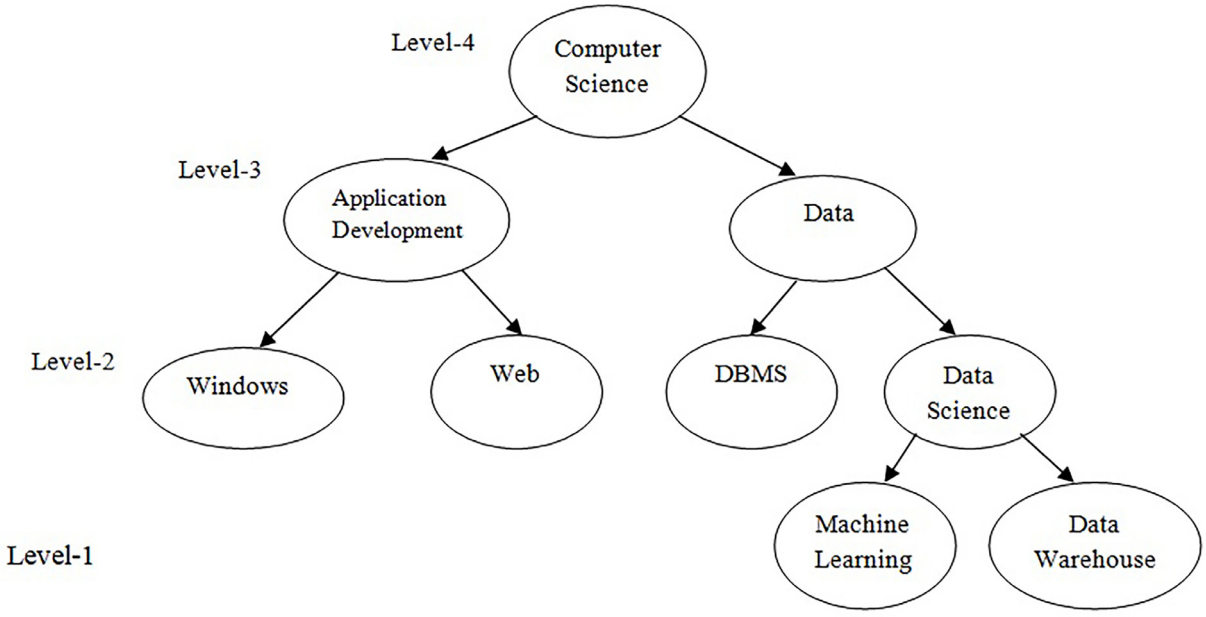

Fig. 7. Concept hierarchy formed with 850 Documents.

Table 7
Drill-down operation on the concept hierarchy on Dimension- Topic (D_T).

Level-4	No. of Files	Level-3	No. of Files	Level-2	No. of Files	Level-1	No. of Files
Computer Science	850	Application Development	583	Web	462	-	-
				Windows	121	-	-
		Data	267	DBMS	94	-	-
				Data Science	173	Machine Learning	132
						Data Warehouse	41

Table 8
Dice operation on the resume dataset.

Selection Criteria	Count (No.of Resumes)	(%) of total number of documents
Topic='Data Science' & Experience='2 years'	16	9.25
Topic='Data Science' & Experience='5 years'	67	38.72
Topic='Web Application' & Experience='2 years'	248	42.54
Topic='Web Application' & Experience='5 years'	184	31.56

lection conditions on the dimensions Topic and Experience is shown in Table 8.

Results investigated during the case study prove the importance of performing OLAP operations on organizational text data repositories for business oriented improvised decision making purpose.

7.5. Comparison of OLAP query execution time between proposed model and TF-IDF models

At level-2 (with the maximum number of clusters), we have performed the various OLAP operations such as: roll-up, drill-down, slice, and dice on documents represented using both 200-dimensional centroid word vector and also as TF-IDF vector format. Fig. 8 shows the query execution time on the centroid word vector is much faster than the higher dimensional TF-IDF word vectors.

The existing methods which address text-OLAP, use the TF-IDF vectors to represent the text documents in the vector space model. With a large dataset having a huge vocabulary set, these TF-IDF vectors can be very sparse and very high dimensional. However, the proposed model uses much lesser dimensional dense word vectors for document representation. Processing the less dimensional word vectors for contextual similarity computation with a posed OLAP query is much faster than the

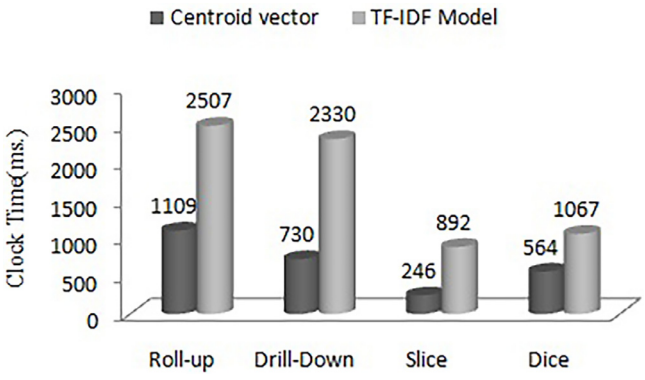

Fig. 8. Comparison of OLAP operation time in wall clock (milliseconds) between proposed centroid vector based model and TF-IDF based models.

processing time required to scan through the high dimensional TF-IDF vectors. Hence, during experimental evaluation proposed model shows considerable improvement in speeding up the execution time of OLAP operations. Results in Fig. 8 confirm that the proposed centroid vector

model significantly reduces the OLAP query processing time compared to the TF-IDF based models for all four OLAP operations. Roll-up operation is almost 2.3 times faster in the proposed model compared to the existing models. Dice operation is performed even faster with a speed-up of almost 3.2 times. The improvements in the query execution time of the proposed model are visible for the slice(3.6 times faster) and dice (1.9 times faster) operations.

7.6. Computational complexity

The proposed model first converts the corpus of text documents into word embedding based N-dimensional documents centroid vectors. After that, the hierarchical agglomerative clustering algorithm is applied to the centroid vectors to form the concept hierarchy. Once the concept hierarchy is performed, documents are aggregated and retrieved according to different dimensions using OLAP queries on the concepts (clusters). The computational complexity of the Word2Vec Skip-gram algorithm is: $Q = O(C \times (D + D \times log_2(V)))$ (Mikolov et al., 2013a), C is the window or context size, D is the dimensionality of the embeddings and V is the vocabulary size. In addition, the computational complexity of the hierarchical clustering algorithm is $R = O(N^2)$, where N is the number of documents. Therefore, in total the computational complexity in forming the concept hierarchy is:

$O(Q + R) = O(C \times (D + D \times log_2(V))) + N^2)$.

This analysis shows the computational cost of the proposed method is an expensive one. However, in this paper we have prioritized extracting semantic information from the text documents to form the concepts (clusters) with better accuracy. Since this entire analysis is performed in off-line mode, thus for the sake of contextual accuracy this compromise can be made at the cost of heavy computational complexity. Accurate knowledge is critical in making managerial decisions and empowering organizations with business intelligence. Having said that, once the documents are clustered according to the concept hierarchy then the concept wise real time (on-line mode) document retrieval time is improved. Then the searching time complexity will be $O(log_2 k)$, where k is the number of clusters.

8. Discussion

The synthesis and analysis of the resume datasets considered for the simulation of our proposed model underline the importance of maintaining textual data warehouses. Findings from the analysis of the result implicate the benefits of leveraging managerial decision making by performing OLAP operations on the textual data warehouses. Our experimental model serves as a decision support making tool for the HR managers to analyze and manage the huge volume of the contextually variant sets of resumes that are received in response to a job offer. By the implementation of the proposed model and making a case study on the resume datasets, the following analytics driven benefits were established:

8.1. Key findings

In particular, the results of our experiments are contrasting with that of existing TF-IDF based models dealing with textual data warehouses. During the empirical evaluation, the findings of this research work turn out to be promising in performing OLAP aggregation operations on textual data warehouses. The findings more specifically reveal the following facts: i) Text documents represented as dense word embedding based centroid vectors perform significantly better than the sparse TF-IDF vectors in capturing the contextual similarities between documents ii) With fewer number of documents TF-IDF models are better than word vectors but as the number of documents increases the performance of the word centroid vectors drastically improves in capturing contextual similarities iii) Agglomerative hierarchical clustering algorithm forms the concept hierarchy dynamically. Based on the Silhouette Coefficient value the linkage criterion of the clustering algorithm is chosen. Experiments with varying Silhouette Coefficient values ensure the better formation of the clusters in terms intra cluster similarity and inter cluster separability. iii) Automatic extraction of the underlying contexts in the concept hierarchy allows decision makers to query upon the set of documents without knowing the contexts in advance. iv)Execution time of text OLAP aggregation operations on the concept hierarchy using centroid vectors is considerably faster than that of TF-IDF based word vectors.

8.2. Implications for theory

With the emergence of digitalization, the generation of unstructured data is more than ever evident. In the business sector, virtually all organizations are operating with the huge voluminous text data generated every day. The rapid advancement of data storage hardware and the growth of the generation of digital unstructured data on the web has solved the problem of availability of data for every organization. However, it has become harder than ever for organizations to keep up with this data. In the competitive market scenario, it has become a mandate for every organization to efficiently manage the data and get access to the right data at the right time for effective business decision making purposes. Management of organizations is finding it very hard to have a grasp on the hundreds of emails, reports, scholarly articles, medical diagnosis reports, customer feedback, product documentation, forum discussion, transaction management report, customer profiling, resume submission for job recruitment, and other plethora of applications. Related study reveals that organizations have already opted for comprehensive IT enabled data retrieval tool sets for quick access to the data needed for managerial analysis leading to business intelligence. However, these IT enabled tools often suffer from the accuracy of the retrieved data. With the emergence of AI and NLP and high volume of distributed computing, organizations are increasingly looking for alternative Information System enabled work systems. Literature survey (Carvalho, 2000) on Information System reveals there are four categories of Information Systems, such as: IS1, IS2, IS3 and, IS4, all with different purpose of handling information to communicate with different stakeholders of an organization. According to Alter (2008) the four types of objects 'all deal with information; they all are somewhat related to organizations or the work carried out in organizations; and they all are related to information technology, either because they can benefit from its use or because they are made with computers or computer-based devices.'

In this paper, the authors have proposed a model which is a solution for managing a huge repository of textual documents and goes beyond the retrieval of documents by mere keyword-matching with the user query. The proposed model can dive into the contextual semantics of the documents and performs hierarchical clustering of text documents according to their contextual similarity. This work is intended to be one of the small first steps in the direction of making an Information System which will ensure effective reporting of data between the management and operation modules of an organization. According to the definition of Information Systems, this proposed framework can be conceptualized as an IS2 work system. The sharing of information from operational subsystems allows the management of an organization to analyze and discover hierarchical relationships existing among different actors. In this paper, we presented a framework for integrating text into multi dimensional data model capable of OLAP text analysis. The proposed conceptual star schema is surrounded by contextual dimensions. In the proposed measure each document is represented by word embedding based centroid vectors of weighted concepts. Next, the agglomerative hierarchical clustering algorithm categorizes the documents into a concept hierarchy based on their contextual similarities. Documents arranged in a concept hierarchy allow users to query the corpus of documents from different levels of granularity. The H.R. manager of a company is highly benefited from this system in terms of discovering new business facts through fast OLAP querying on hierarchically organized resumes based on the sim-

ilarity of the domain of expertise of different candidates. The existing works on text-OLAP represent data using high dimensional sparse TF-IDF vectors. Processing of OLAP queries on these high dimensional vectors results in more execution time than the processing of the same set of OLAP queries on much smaller dimensional word vectors. The dense nature of the word vectors also increases the accuracy of the hierarchical clustering. Increase in the accuracy of clusters means more number of relevant documents are categorized in a group representing a particular context. The notion of extracting hierarchical contextual similarity between documents through OLAP queries is highly beneficial for the H.R. managers of a company. As already shown, in the proposed model, two separate resumes with specialization in 'ASP.NET C#' and 'Java Hibernate framework' are considered to be under the same genre or context 'Object Oriented Programming' despite sharing very few or no keywords between them. Therefore, clustering accuracy increases. Since OLAP queries are executed on these clusters, thus accuracy of clustering in turn results in the enhancement of accuracy during the retrieval of relevant documents in correspondence to an OLAP query.

To summarize the impact of this research work, the proposed Information System based context-aware work system helps in taking organizational managerial decisions through fast and accurate OLAP query processing.

8.3. Implications for practice

The implications for practice have been broadly classified into three sections, such as A) Descriptive analytics, B) Discovery analytics and C) Predictive analytics.

A) Descriptive analytics

Descriptive analytics deals with reporting and visualizations. In the different documents based on the underlying contextual information, concept hierarchy may be inferred. For instance in the given example Fig. 1 provides a descriptive visualization of the notion of concept hierarchy with the associated theory on the contextual dimensions that are extracted from the resumes based on the skill-set or domain of specialization of the candidates. The experimentally formed context-aware concept hierarchy depicted in Fig. 7 confirms our claim.

B) Discovery analytics

Discovery analytics can forecast early signals through text summarization OLAP operations. Results investigated from the case study depicted in Table 8, epitomize the importance of early knowledge discovery for developing a decision support system. The case study contributes to the knowledge of knowing the candidature of different candidates with respect to their area of specialization and job experience. This may serve as an aid to figure out the demand and supply of skilled human resource according to the latest industry needs.

In the case of our dataset the following observations can be witnessed from the analysis carried upon Table 8. It can be seen that there is a huge difference in the percentage of the total number of resumes between the candidates with 2 years of experience (9.25%) and candidates with 5 years of experience (38.72%) for the job specialization in Data Science. In contrast, this difference is not that drastic for the job profiles in 'Web Application Development' for the 2 years (42.54%) and 5 experience (31.56%) holders respectively. The inference that can be taken out from this data is a) opportunity to work on the projects in Data science domain for the entry level candidates is still relatively less. However, candidates are enhancing their skills in Data Science Projects when they are deployed into the domain after gaining some years of experience and b) there is a uniform distribution of entry-level and moderately experienced candidates in the domain of web application development.

The proposed model claims to fasten the OLAP query processing time by reducing the number of features of the text documents. Experimental results show drastic improvements in speeding up aggregation operations like- roll-up, drill-down, slice, and dice. The supporting results are given in Tables 7, 8 and Fig. 8.

C) Predictive analytics:

The proposed model uses word embedding based technique combined with agglomerative hierarchical clustering algorithm to predict the class of the documents during the formation of the concept hierarchy. Application of word embedding technique on the resume dataset ensures the contextually similar text documents are categorized with higher accuracy in terms of cosine similarity. Results reported in Tables 4 and 5 confirm the superiority of the proposed method over TF-IDF based models. The context-aware ensemble clustering method is dynamic as it does not require the number of clusters to be defined at the beginning. This methodology further ensures the better formation of clusters in terms of intra-cluster cohesion and inter-cluster separation. The Silhouette scores (s) shown in Fig. 6 confirm our claim.

In general, this research work serves as a potential tool for the development of Information Retrieval (IR) systems at a time when several existing organizations and start-ups are investing in developing text data analytics models with the motive of deriving actionable insights from the large volume of text data. The practice of text summarization technique and OLAP operations provide an excellent facility to the management of the organizations in keeping-up and processing high volume of text data which are hard to manage manually. The simulated model on the resume dataset enables the managers to drastically reduce the longer recruitment processing time by automating the resume short-listing time as per the requirement of skill-set specialization, job location, and preferred experience.

8.4. Limitations

There remains room for improvement in the proposed model. This model demands the documents be formalized in a certain format so that inherent contextual dimensions can be extracted by the algorithm. The experiments performed in this paper can be further performed on a bigger sized data set. Alternate methodologies can be further investigated which may result in lesser computational complexity. Query optimization strategies have not been applied in this work, and hence remain as a scope for further work.

8.5. Future work

Further work can be carried out to compare the performance of the proposed model with deep learning based algorithms using RNN or BERT. This algorithm can be also explored in other domains like scientific journal, bio-medical engineering etc. Satisfactory performance of the proposed algorithm with a higher sized dataset and higher numbers of features is another challenge. Another research challenge is the formation of the lattice of cuboids over textual data warehouse to enable a holistic decision support system.

9. Conclusion

This article proposed a novel methodology to construct a textual data warehouse and perform OLAP operations on the textual data after categorizing the text documents in a concept hierarchy by capturing the contextual similarity between the text documents. The proposed model outperforms existing models in capturing contextual similarity between texts. The proposed model also proves its supremacy with respect to the ability to add new clusters during the formation of the concept hierarchy. Despite being computationally expensive during the initial word vector formation time, the proposed model is able to retrieve documents in logarithmic time once the concept hierarchy is formed. Experimental results show the efficiency of the proposed model in faster processing of OLAP queries in comparison to the existing techniques. In the end, the findings from a case study render the necessity of OLAP tools over textual documents of an organization to explore and analyze the business facts.

Acknowledgement

Work partially supported by iNEST (Interconnected NordEst Innovation Ecosystem), funded by PNRR (Mission 4.2, Investment 1.5), NextGeneration EU (Project ID: ECS 00000043).

References

Alcamo, T., Cuzzocrea, A., Bosco, G. L., Pilato, G., & Schicchi, D. (2020). Analysis and comparison of deep learning networks for supporting sentiment mining in text corpora. In *Proceedings of the 22nd international conference on information integration and web-based applications & services*. In *iiWAS '20* (pp. 91–96). New York, NY, USA: Association for Computing Machinery. 10.1145/3428757.3429144.

Alter, S. (2008). Defining information systems as work systems: Implications for the is field. *European Journal of Information Systems, 17*(5), 448–469.

Ángel González, J., Hurtado, L.-F., & Pla, F. (2020). Transformer based contextualization of pre-trained word embeddings for irony detection in twitter. *Information Processing & Management, 57*(4), 102262. 10.1016/j.ipm.2020.102262.

Atkinson, J., & Escudero, A. (2022). Evolutionary natural-language coreference resolution for sentiment analysis. *International Journal of Information Management Data Insights, 2*(2), 100115. 10.1016/j.jjimei.2022.100115.

Azabou, M., Khrouf, K., Feki, J., Soulé-Dupuy, C., & Vallès, N. (2015). Diamond multidimensional model and aggregation operators for document olap. In *2015 ieee 9th international conference on research challenges in information science (rcis)* (pp. 363–373). IEEE.

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.

Bouakkaz, M., Ouinten, Y., Loudcher, S., & Strekalova, Y. (2017). Textual aggregation approaches in olap context: A survey. *International Journal of Information Management, 37*(6), 684–692.

Carvalho, J. A. (2000). Information system? which one do you mean? In *Information system concepts: An integrated discipline emerging* (pp. 259–277). Springer.

Chakrabarty, A., Roy, S., & Roy, S. (2018). A context-aware fuzzy classification technique for olap text analysis. In *Recent findings in intelligent computing techniques* (pp. 73–85). Springer.

Cuzzocrea, A. (2020). Sppolap: Computing privacy-preserving olap data cubes effectively and efficiently algorithms, complexity analysis and experimental evaluation. *Procedia Computer Science, 176*, 3831–3842.

De Miranda, G. R., Pasti, R., & de Castro, L. N. (2019). Detecting topics in documents by clustering word vectors. In *International symposium on distributed computing and artificial intelligence* (pp. 235–243). Springer.

Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019). BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 conference of the north American chapter of the association for computational linguistics: Human language technologies, volume 1 (long and short papers)* (pp. 4171–4186). Minneapolis, Minnesota: Association for Computational Linguistics. 10.18653/v1/N19-1423.

Fujii, M., Sakaji, H., Masuyama, S., & Sasaki, H. (2022). Extraction and classification of risk-related sentences from securities reports. *International Journal of Information Management Data Insights, 2*(2), 100096. 10.1016/j.jjimei.2022.100096.

Georgiadou, E., Angelopoulos, S., & Drake, H. (2020). Big data analytics and international negotiations: Sentiment analysis of brexit negotiating outcomes. *International Journal of Information Management, 51*, 102048. 10.1016/j.ijinfomgt.2019.102048.

Kim, S.-W., & Gil, J.-M. (2019). Research paper classification systems based on tf-idf and lda schemes. *Human-centric Computing and Information Sciences, 9*(1), 1–21.

Kohomban, U., & Lee, W. (2007). *Optimizing classifier performance in word sense disambiguation by redefining sense classes* (pp. 1635–1640).

Kosmopoulos, A., Androutsopoulos, I., & Paliouras, G. (2015). Biomedical semantic indexing using dense word vectors in bioasq. *J BioMed Semant Suppl BioMedl Inf Retr, 3410*, 959136040–1510456246.

Krishna, P. P., & Sharada, A. (2019). Word embeddings-skip gram model. In *International conference on intelligent computing and communication technologies* (pp. 133–139). Springer.

Landauer, T. K., Foltz, P. W., & Laham, D. (1998). An introduction to latent semantic analysis. *Discourse processes, 25*(2-3), 259–284.

Lin, C. X., Ding, B., Han, J., Zhu, F., & Zhao, B. (2008). Text cube: Computing ir measures for multidimensional text database analysis. In *2008 eighth ieee international conference on data mining* (pp. 905–910). IEEE.

Maas, A., Daly, R. E., Pham, P. T., Huang, D., Ng, A. Y., & Potts, C. (2011). Learning word vectors for sentiment analysis. In *Proceedings of the 49th annual meeting of the association for computational linguistics: Human language technologies* (pp. 142–150).

Van der Maaten, L., & Hinton, G. (2008). Visualizing data using t-sne. *Journal of machine learning research, 9*(11).

Manuel Pérez-Martínez, J., Berlanga-Llavori, R., Aramburu-Cabo, M. J., & Pedersen, T. B. (2008). Contextualizing data warehouses with documents. *Decis. Support Syst., 45*(1), 77–94. 10.1016/j.dss.2006.12.005.

Mihalcea, R., & Tarau, P. (2004). TextRank: Bringing order into text. In *Proceedings of the 2004 conference on empirical methods in natural language processing* (pp. 404–411). Barcelona, Spain: Association for Computational Linguistics. https://aclanthology.org/W04-3252.

Mikolov, T., Chen, K., Corrado, G., & Dean, J. (2013a). Efficient estimation of word representations in vector space. *arXiv preprint arXiv:1301.3781*.

Mikolov, T., Sutskever, I., Chen, K., Corrado, G. S., & Dean, J. (2013b). Distributed representations of words and phrases and their compositionality. In *Advances in neural information processing systems* (pp. 3111–3119).

Mothe, J., Chrisment, C., Dousset, B., & Alaux, J. (2003). Doccube: Multi-dimensional visualisation and exploration of large document sets. *Journal of the American Society for Information Science and Technology, JASIST, Special, 54*, 650659.

Oukid, L., Benblidia, N., Asfari, O., Bentayeb, F., & Boussaid, O. (2015). Contextualized Text OLAP Based on Information Retrieval. *International Journal of Data Warehousing and Mining (JDWM), 11*(2), 1–21. 10.4018/ijdwm.2015040101.

Park, B.-K., Han, H., & Song, I.-Y. (2005). Xml-olap: A multidimensional analysis framework for xml warehouses. *Dawak*.

Perez, J., Aramburu, M., Berlanga, R., & Pedersen, T. (2007). R-cubes: Olap cubes contextualized with documents. In *Proceedings of the 2007 ieee 23rd international conference on data engineering*. IEEE Press. Null; Conference date: 15-04-2007 Through 20-04-2007.

Periñán-Pascual, C. (2021). Measuring associational thinking through word embeddings. *Artificial Intelligence Review*, 1–38.

Ravat, F., Song, J., Teste, O., & Trojahn, C. (2020). Efficient querying of multidimensional rdf data with aggregates: Comparing nosql, rdf and relational data stores. *International Journal of Information Management, 54*, 102089.

Ravat, F., Teste, O., Tournier, R., & Zurfluh, G. (2008). Top_keyword: An aggregation function for textual document olap. In *International conference on data warehousing and knowledge discovery* (pp. 55–64). Springer.

Razavisousan, R., & Joshi, K. P. (2022). Building textual fuzzy interpretive structural modeling to analyze factors of student mobility based on user generated content. *International Journal of Information Management Data Insights, 2*(2), 100093. 10.1016/j.jjimei.2022.100093.

Sarkar, B. D., & Shankar, R. (2021). Understanding the barriers of port logistics for effective operation in the industry 4.0 era: Data-driven decision making. *International Journal of Information Management Data Insights, 1*(2), 100031. 10.1016/j.jjimei.2021.100031.

Sen, S., Roy, S., Sarkar, A., Chaki, N., & Debnath, N. C. (2014). Dynamic discovery of query path on the lattice of cuboids using hierarchical data granularity and storage hierarchy. *Journal of Computational Science, 5*(4), 675–683. 10.1016/j.jocs.2014.02.006.

Shahapure, K. R., & Nicholas, C. (2020). Cluster quality analysis using silhouette score. In *2020 ieee 7th international conference on data science and advanced analytics (dsaa)* (pp. 747–748). IEEE.

Struijk, M., Ou, C., Davison, R., & Angelopoulos, S. (2022). Putting the is back into is research. *Information Systems Journal, 32*(3), 1–4. http://dro.dur.ac.uk/33883/.

Theodoridis, S., & Koutroumbas, K. (2009). clustering algorithms ii: Hierarchical algorithms. *Pattern Recognition (Fourth Edition): Academic Press*.

Unhelkar, B., Joshi, S., Sharma, M., Prakash, S., Mani, A. K., & Prasad, M. (2022). Enhancing supply chain performance using rfid technology and decision support systems in the industry 4.0–a systematic literature review. *International Journal of Information Management Data Insights, 2*(2), 100084. 10.1016/j.jjimei.2022.100084.

Wadud, M. A. H., Kabir, M. M, Mridha, M. F., Ali, M. A., Hamid, M. A., & Monowar, M. M. (2022). How can we manage offensive text in social media-a text classification approach using LSTM-BOOST. *International Journal of Information Management Data Insights, 2*(2), 100095. 10.1016/j.jjimei.2022.100095.

Zhang, D., Zhai, C., & Han, J. (2009). Topic cube: Topic modeling for olap on multidimensional text databases. In *Proceedings of the 2009 siam international conference on data mining* (pp. 1124–1135). SIAM.

Zhang, Z., Wang, H., & Feng, X. (2018). Olap on multidimensional text databases: Topic network cube and its applications. *Filomat, 32*(5), 1973–1982.