ITC 3/49

Information Technology and Control
Vol. 49 / No. 3 / 2020
pp. 370-380
DOI 10.5755/j01.itc.49.3.25988

Semantics Based Clustering through Cover-Kmeans with OntoVsm for Information Retrieval

Received 2020/05/04 | Accepted after revision 2020/05/24

http://dx.doi.org/10.5755/j01.itc.49.3.25988

HOW TO CITE: Kumar, R. L., Kannammal, N., Krishnamoorthy, S., Kadry, S., Nam, Y. (2020). Semantics Based Clustering through Cover-Kmeans with OntoVsm for Information Retrieval. *Information Technology and Control*, 49(3), 370-380. https://doi.org/10.5755/j01.itc.49.3.25988

Semantics Based Clustering through Cover-Kmeans with OntoVsm for Information Retrieval

R. Lakshmana Kumar

Hindusthan College of Engineering and Technology, India, e-mail: research.laksha@gmail.com

N. Kannammal

Independant Researcher, e-mail: drkannammal.n@gmail.com

Sujatha Krishnamoorthy

Department of Computer science, College of Science and Technology, Wenzhou-Kean University, Wenzhou, China; e-mail: sujatha@wku.edu.cn

Seifedine Kadry

Department of Mathematics and Computer Science, Faculty of Science, Beirut Arab University, Lebanon; email: s.kadry@bau.edu.lbn

Yunyoung Nam*

Department of Computer Science and Engineering, Soonchunhyang University, Asan 31538, South Korea

***Corresponding author:** ynam@sch.ac.kr

Document clustering plays a significant task in the retrieval of the information, which seeks to divide documents into groups automatically, depending on their content similarity. The cluster consists of related documents within the group (having high intra-cluster similarity) and dissimilar to other group documents (having low inter-cluster similarity). Clustering documents should be considered an unsupervised process that aims to classify documents by identifying underlying structures, i.e. the learning process is unsupervised. Thus there is no need to determine the correct output for an input. Previous clustering methods do not know the semantic associations between words such that the context of documents cannot be correctly interpreted. To address this problem, the advent of semantic ontology information such as WordNet was widely used to enhance text clustering consistency. This paper initially proposes an OntoVSM model to reduce the dimension of the document efficiently. The cover K-means clustering algorithm is proposed for semantic document clustering. The proposed algorithm is a hybrid version of K-means and covers coefficient-based clustering methodology (C3M) that is improved semantically using WordNet ontology. The dimensionality reduction based on semantic knowledge of each term preserves the information without loss. The performance of the proposed work is analyzed through experimental results. This shows that the proposed work gives improved results compared to other standard methods.

KEYWORDS: Semantic Clustering, dimension reduction, WordNet, semantic features.

1. Introduction

The exponential growth of the World Wide Web (WWW) has expanded the number of documents available online. Search engines retrieve many documents while searching for the document using WWW. While most of the documents are important to the topic, others are obsolete, limited-quality documents. Clustering plays an integral part in arranging a large number of documents that have been transferred to clusters from the search engines [14]. Document clustering is one of the methods used by search engines to locate related documents [18]. A large collection of documents can be easily navigated, browsed and organized by organizing similar documents together. It is used in the identification of patterns, machine learning and statistics [11]. It is really helpful for categorization of a collection of documents and the identification of topics [17].

Traditional text clustering algorithms typically rely on the Bag of Words (BOW) method, and an apparent drawback of the BOW is that it lacks the semantic association between terms so that the context of documents cannot be correctly interpreted. While text documents have evolved rapidly, textual details have become a range of glossary wherein they are high-dimensional and often contain semantic information. Therefore, text clustering techniques that can accurately represent the topic of documents and increase clustering accuracy are significantly required, preferably processing data with a limited scale. There are many semantic-based approaches [11], [5] that are being developed.

Vector Space Model (VSM) is used in traditional document clustering methods which identify each text in the collection with a single multidimensional vector, and each portion of this vector represents a particular keyword or concept applicable to the document. VSM text-data representation can easily result in tens, hundreds, or thousands of features. Consequently, any clustering algorithm will suffer from the dimensionality curse. In such sparse and high-dimensional space, any measure of distance that assumes all features are equally significant is likely not to be efficient. This is due to the semantically related words that are not taken into account, which can cause problem [14].

To overcome this problem, this paper has proposed an efficient document representation model called OntoVSM, which has efficiently reduced the dimension of the document features. The proposed work uses traditional VSM with Ontology model for dimensionality reduction. Ontology is a critical, widely used conceptualism for the Semantic Web. The domain of ontology is useful in developing a general glossary which defines the domain of interest. This is essential to unify and exchange domain knowledge and to connect with other domains [18].

Other problems in clustering results are word sense disambiguation, and extracting core semantics from texts. This paper has proposed an efficient document clustering algorithm called cover K-means algorithm to overcome these problems, which combines traditional K-means clustering and cover coefficient-based clustering methodology (C3M), that are improved semantically using WordNet ontology.

The remainder of this paper is organized as follows: Section 2 reviews the related work, which includes dimensional reduction and semantic clustering. Section 3 describes the materials and methodology. The proposed ontology-based dimensionality reduction with cover K-means clustering is explained in Section 4. The performance of experimental results is analyzed in Section 5, and finally, Section 6 concludes the paper.

2. Related Work

2.1. Dimension Reduction

Dimension reduction techniques are a significant step in the clustering process, and given the high dimensionality of the data, this is not a simple task. This high dimension of space typically reduces the effectiveness of the clustering mechanism. Many dimension reduction techniques such as PCA, NMF [14] and SVD [8] are proposed. This section explains some dimensionality reduction methods.

Li et al. [9] have introduced a new paradigm of matrix factorization, which concurrently incorporates the goals of clustering and reduction of dimensionality. The grouping is based on the factorization of matrixes, is currently carried out on the embedded subspace, and can offer more efficient and rational solutions.

These dimension reduction does not consider the semantic relationship between words. Only a few approaches have proposed a semantic-based dimension reduction.

Semantic similarity based feature reduction algorithm for graph classification is proposed in [18]. This algorithm uses neural language models to learn vector representations of subtree patterns and then combines related subtree patterns semantically into a new feature. A new ranking is used to pick highly discriminatory features. A new methodology is introduced in [13] based on the semantic structure of the Web data. It incorporates both extraction feature and data visualization and retrieval feature selection strategies, including essential features for efficient text processing. This method reduces the complexity of the dimensions in the feature vector for the efficient retrieval of information. Mendizabal et al. [10] tackles the issue of feature reduction by proposing a new semantic-based proposal which prevents a lack of (lossless) information. Synset characteristics can be classified semantically by using the BabelNet ontological dictionary's taxonomic relationships (mainly hypernyms).

2.2. Semantic Clustering

Stanchev [17] proposes semantic document clustering. It models the WordNet and DBPedia knowledge as a probabilistic graph which can be used to measure the similarities among two words. Cao et al. [19] incorporate named entities as objectives into the clustering of documents, which are the core elements that describe the semantics of documents and, in many cases, are user issues. First, the standard vector space model based on keywords is modified, instead of keywords, with vectors defined over spaces of object names, classes, name-type pairs and identifiers. Furthermore, hierarchical clustering of documents can be done using the similarity measure specified as the vector cosines representing documents.

Fahad and Yafooz [6] suggested a model for the clustering of semantic documents. The pre-processing steps of the document, WordNet semantic knowledge allow us to have the semantic relationship accessible from raw text. By remembering the constraint on the natural language of conventional clustering algorithms, find semantic clustering by logical clustering at COBWEB. The Google Tri-gram Frequency Measure is proposed in [19] to determine the correlation among documents based on the frequencies of words in the comparative documents as well as Google's n-gram corpus as an alternative indicator of semantic similarities. In [17], an innovative model was proposed to stabilize the objects with distinct features for the ships in backpropagation neural methods. The suggested method results in excellent accuracy

3. Materials and Methodology

3.1. Definitions

Semantic_Link_Wt: Let D = {d_1, d_2, d_3,...d_m} be the document collection for which each document is represented as term vector $\overrightarrow{t_k}$ ={t_1,t_2,t_3......t_n} for k=1....m documents and let R={ r_1(synonymy), r_2 (hypernymy), r_3(hyponymy), r_4(meronymy) } be the semantic relations based on which semantic relatedness between two terms are manipulated as weight of the terms. The terms with maximum weight have more important to the documents. Every term in the document gains weight based on the background knowledge it acquires about relationships with other terms. It is defined as

For each document k, the term vector is $\{t_1^k, t_2^k, \ldots, t_n^k\}$ subjected to relation R={$r_{1,}$ $r_{2,}$ $r_{3,}$ r_4} whose link weight vector is $\{lw_{t_1}^k, lw_{t_2}^k, \ldots, lw_{t_n}^k\}$ and is computed as

$$\begin{array}{l}\forall k{:}k=1\ldots m,\\ \quad \forall i{:}i=1\ldots n,\\ \quad\quad lw_{t_i}^{k}=0\\ \quad\quad \forall j{:}j=1\ldots n,\ i\ \ j,\\ lw_{t_i}^{k}+=1\text{, iff}\{\ r\ \ R, \mathrm{Rel}\ r(S(t_i), S(t_j))>0\},\end{array} \tag{1}$$

where $w_{t_i}^{k}$ is link weight, r be any relation of R, $S(t_{i|j})$ is synset of the term, Relr is semantic relatedness measure between synset of the term based on four relations ‘r’.

Low-frequency uncommon terms are static and do not add to the documentation. They can be defined using the tf-idf equation, and the terms whose weight is below the specified threshold are pruned from the remaining term that is regarded as relevant. The frequency is the number of word occurrences and it is described below.

Tf-idf_wt: A document (k D) is a set of document defined as {d1,d2,....dm} have frequency weight denoted for each term as { where n is the number of terms of the k^{th} document. Each ‘w’ is composed using tf (t) and idf(t) that are frequency of a term in corresponding document and frequency of a term in the total collection of N document respectively. It is given as

$$\begin{array}{l} i=1\ldots n\ \ \text{of}\ \ k\,|k\ \ D\\ fw_{t_i}^{k}=tf_{t_{i,k}}\times idf_{t_i}\end{array} \tag{2}$$

on using Tf-Idf method. A term with highest ‘fw’ value is a good discriminator of the document in which it occurs.

Link_frequency_wt: Let Wlink_tfidf be the total score for each term of document d obtained on summing link weight and frequency weight.

$$W_{\mathrm{link\_tfidf}}=\alpha\ lw_{t_i}^{k}+(1\text{-}\alpha)\ fw_{t_i}^{k}, \tag{3}$$

where α is a weight parameter to adjust the contribution of both link and frequency weight.

The following are the concepts suggested for the reduction of term vectors by double level dimensionality.

The term pruning is determined by both semantic correlation and frequency value information. The frequency-based decision cannot be taken alone since more semantically relevant words can often be omitted due to minimal tfidf, which results in knowledge loss. Reduction of dimensionality with maximal pruning of terms is not promoted as it leads to information loss which affects the accuracy of clusters.

The value of $W_{\mathrm{link\_tfidf}}$ falls in any one of the following cases:

Case 1: terms which have maximum mutual semantic count and maximum tfidf weight are significant to document.

Case 2: terms which have maximum semantic count and minimum tfidf weight can contribute semantic information to document similarity.

Case 3: terms which have a minimum semantic count (i.e. term which is not correlated with many terms) and maximum tfidf weight (i.e. frequency in a particular document). More occurring terms are important to document.

Case 4: terms which have minimum semantic link count and minimum tfidf weight are not important.

The terms with $W_{\mathrm{link\_tfidf}}$ > threshold (σ) form term vectors for a document. The threshold is fixed to satisfy the first three cases.

4. Proposed Clustering with Dimensionality Reduction

This section explains the proposed OntoVSM model with document clustering. According to the feature selection method, the number of terms is reduced in the proposed work. Clustering is a data mining strategy that uses specific features to bind related members together.

A data matrix D with I= {1,.., n} individuals with k features F= {1,.., k} is given as input for clustering process. Each individual or object (i) is represented as a vector of k features or dimensions. Any entry xik can hold numerical or categorical value. The k features are multidimensional that describe the object. When the object to be clustered is web documents, k features are taken as terms occurring in the document and each xik is frequency or number of occurrence of that term in the corresponding document in data matrix D. This depiction of the set of documents in a vector space is called the vector space model (VSM) and is the basic input for information retrieval, clustering and classification of documents.

Term Weight: Traditional weight for the term is of local and global type. Where the term takes 1 for the inclusion of term and 0 for the absence of term, the local types are discrete. It is the Term Frequency (TF) type where the term only takes the number of term occurrence in the document. The global weight approach represents the term frequency in the target document as well as in the whole collection called Inverse Document Frequency (IDF).

The proposed work is evaluated against TF-IDF global term weight method.

The tf-idf weighting scheme assigns to term t a weight in document d given by

Tf-Idf = Local weight. Global Weight

$$\text{tf-idft,d} = \text{tft,d} \times \text{idft}, \tag{4}$$

where $\mathrm{tf}_{t,d}$ is the term frequency of a term in a candidate document, and df_t is the entire documents that have the term t. In order to scale the term frequency with the growing collection inverse document frequency, Idf_t is found with N collection as

$$Idf_t = \log \frac{N}{df_t} \cdot \tag{5}$$

Based on the Bag-of-Word concept, which states that the importance of a document to a query is determined using the frequencies of terms, the term-document matrix which is in VSM could be used for knowledge recovery. Also used for retrieval is the document vector of binary values (0 or 1), but it showed low results relative to the frequency vector.

Feature Selection and Feature Extraction: Two methods used for dimension reduction are the selection of features and extraction of features. Selection of features extracts a subset which removes unnecessary features and preserves the originality of information. Extraction of the feature is a mapping of a high-dimensional vector to a small space. Both approaches help reduce the search space. In this work, unsupervised method of selection of features is modified for dimension reduction. In the case of the tf-idf method, which preserves the originality of the input, the terms are pruned depending on certain parameters such as frequency.

The VSM mentioned above is called Bag-of-Words in which each term is separate from each other and is not regarded as a link between them. In using this model for the retrieval, the estimation of similarity is based only on the number of terms that exist. Thus it defines two vectors as identical though they are not that identical. To take the hidden semantic into account, the data matrix is packed with semiconducting features that will boost the interpretability of results as IR follows the Bag-of-Concept model. The context information is embedded using information sources such as Ontology, which are defined as explicit tabilizedzation specifications. The concepts are placed in a taxonomical way associated with each other using taxonomical relationships. Using Ontology, the semantic relation between the terms is extracted to create a strong profile for the documents.

Semantic similarity and relatedness: Semantic similarity and semantic relatedness are two steps to map relevant related terms as documents discriminator. Semantic similarities show identical terms like car and motorcycle both are vehicle sharing generally. In contrast, Semantic relatedness identifies terms that are one-to-one related even though they are not the same type as bread and jam, or paper and pencil.

WordNet ontology is used to manipulate context information when consolidating documentation for the work. The term is hierarchically related to another based on relationships such as identities, synonyms, antonyms, hyper-hyponyms and meronyms, among others. Two approaches combine semantic information between terms: concept mapping and embedded procedures (inputs are linked to ontology concepts).

Mapping or changing terms into their correct ontology definition is concept mapping, but missing the relationship does not increase precision. The aspect of polysemous and synonyms, where a term can be mapped to more than one concept, would increase when mapped. The embedded approach involves the task of mapping the association between two terms, using specifics of taxonomy and integrated into the algorithm.

The increase in clustering accuracy is obtained on the best results of semantic similarity or semantic relatedness between two terms that will have reliable clusters and strongly bonded cluster members. In this study, semantic relatedness with identity, synonym, hypernym, hyponym and meronym relationships using WordNet is used for the reduction of semantic-based dimensionality before clustering. The proposed work is compared with the existing TF-IDF frequency-dependent method and the Latent Semantic Analysis feature selection process for dimensionality reduction in the latent semantic space.

The proposed work uses WordNet ontology to calculate the semantic relationship between terms depending on which reduction of the aspect is accomplished. Two clustering algorithms K-means and Cover Coefficient clustering technique (C3M) are subject to the reduced matrix of the vectors. The measurement is similar to a conventional TF-IDF term weight approach and commonly used LSA algorithm.

OntoVSM: The document-term matrix with reduced size improves clustering technique efficiency. For many applications, clustering is performed as an offline method for static collection of documents. For several implementations, the clustered classes are used. This paper deals with the dimension reduction using ontology for vector space, and the reduced vector is called OntoVSM. The relevant criteria are that any technique of reducing dimensionality is possible even if the originality of the document details is retained even after reduction. The feature selection technique incorporates a set of features and outputs a subset that satisfies the reduction limit. The terms of the document are pre-processed to remove non-informative words through stop word elimination and stemming process. Each noun is featuring document vectors. The appearance or absence of features is represented as either 1 or 0.

The significance of a term is determined based on the lexical relation it has with other terms. A term that is linked to many other terms is considered very relevant for documentation. When the relation existing between terms are semantic relations, i.e. synonyms, hypernyms, hyponyms, meronyms etc., of WordNet, they are said to be semantically related. Weightage to the term is differentiated according to the type of relationship in existing work, but the proposed work considers all relationships similarly since only the link is counted.

Let D be the collection of documents $\{d_{1,}d_{2,}d_{3,........,}d_m\}$ and let T be the collection of terms in each document $T=(t_1,t_2,t_3.......t_n\}$. Related terms are preserved for each document to eliminate irrelevant terms. The pruning of the terms is calculated for each term by relationship-based weight calculation. The weight calculation, along with the lexical relationship between the terms, is the occurrence of terms in the documents. Since different terms are not necessary to document, the occurrence of the term is given priority, and the term that appears in the highest document cannot be a strong discriminator of a specific document.

All semantic relatedness and frequency determine the appropriate terms for the documents, thus pruning irrelevant terms and the dimension. The relationship-based semantic relatedness is assessed using ontology from WordNet. Only four types of identity relation, synonymy, hyper-hyponymy and meronymy are included in this study.

Parts-of-speech plays an essential role in dimensionality reduction and cluster performance. Many studies have shown that it is a noun form that has proved to be successful. Every noun may be polysemous and synonymous, and may also increase the dimensionality or information loss. In this strategy, the process adopted combines all the factors. Any term is described by a synset when mapped to WordNet ontology (collection of similar terms with the same meaning), and the number of synsets depends on the number of meanings the term carries. The sense is selected by the representation word# pos#sense no., where the word is the term, pos is parts of speech (verb, noun, adjective, etc.) and sense no. selected in this work is 1.

Algorithm1 explains the proposed OntoVSM dimensionality reduction.

Algorithm-1. OntoVSM

```
Input: Input Dnxm, ⟦lw⟧_(t_i)^k=0, T, λ =1,  σ = 4, α = 0.7
Output: RDnxm
1. Set link weight lw=0 of each term
2. Compare a term ti with other terms tj on condition ti≠tj
3. if Rel r (ti, tj) = TRUE
4.        ⟦lw⟧_(t_i)^k □(+=) 1
5. Else
6.        Go to S2
7. End If
8. For each ⟦lw⟧_(t_i)^k > λ
9.     Compute ⟦fw⟧_(t_i)^k = ⟦tf⟧_(t_(®,k) ) × ⟦idf⟧_(t_i)
10. End For
11. For each term
12.     Wlink_tfidf = α ⟦lw⟧_(t_i)^k  + (1-α) ⟦fw⟧_(t_i)^k
13. End For
14. For each term if Wlink_tfidf ≥ σ
15.        Add to term set T of each document k ϵ D
16. End For
```

The resultant document vectors of document collection are the input for the clustering process. The input is subjected to two clustering algorithm K-means and Cover Coefficient and studied that hybrid of two algorithms solves each other's limitation and yields good results on computing time. The steps of the hybrid Cover-K-means algorithm and its base Cover-Coefficient algorithm are given next section.

Cover – K-means and basic Cover Coefficient Clustering Methodology (C3M)

The steps in hybrid algorithm Cover-K-means is given as

Step 1: For initial clustering, construct C matrix (document-document) whose entry is c_{ij} ($1 \leq ®, j \leq m$) for the input D matrix with $\{d_{1,} d_2, \ldots . d_m\}$ as rows and $\{t_{1,} t_{2,\ldots\ldots} t_n\}$ be the discriminator terms for each document. The entry is computed as

$$c_{ij} = \alpha_i \times \sum\nolimits_{k=1}^{n} d_{ik} \times \beta_k \times d_{jk}\,,\ 1 \leq \mathrm{i}, \mathrm{j} \leq \mathrm{m}, \tag{6}$$

where $\alpha_{®}$ and β_k are the reciprocals of the ith row sum and the kth column sum and is given as

$$\alpha_{\mathrm{i}} = \left[\sum\nolimits_{j=1}^{n} d_{ij}\right]^{-1}, 1 \leq \mathrm{i} \leq \mathrm{m}, \text{and} \beta_{\mathrm{k}} = \left[\sum\nolimits_{j=1}^{m} d_{jk}\right]^{-1}, 1 \leq \mathrm{k} \leq \mathrm{n}. \tag{7}$$

Every c_{ij} entry indicates the probability of selecting any term of the document (d_i) from the document (d_j), which is information about how long d_i covers d_j. As the document vector consists of semantically related terms, the value gives semantic coverage capacity of each document.

Step 2: Seed document is selected based on the seed power of each document calculated as

$$P_i = \delta_i \times \psi_i \times \sum_{j=1}^{n} d_{ij}, \tag{8}$$

where δi = c_{ii}: Decoupling coefficient of d_i.mm

$\psi_i = 1 - \delta_i$: Coupling coefficient of d_i.

Step 3: The number of clusters is found by $n_{c\,=}\ \delta \times m$

Step 4: Assign the non-seed document to the nearest seed that covers the non-seed document using the highest value of cij.

Step 5: The non-seed document that is not covered by any seed is assigned to the ragbag cluster.

Step 6: The seed as centroid and the formed cluster is given as input to K-means to cluster the incoming documents following regular steps of K-means.

The tabilized centroids found initially with static inputs increases the effectiveness and accuracy of cluster formation and increase the speed of the query-doc matching in many applications. The C matrix and seed document calculation are adopted from C^{3}M algorithm.

Cover Coefficient Clustering Methodology (C^{3}M)

After the dimension reduction, the documents are clustered using cover coefficient clustering. C^{3}M is a single pass partitioning type of algorithm. Basic steps of C^{3}M algorithm is given below:

Step 1: Find Number of cluster Nc.

Step 2: Find Seed power for each document.

Step 3: Identify first Nc seed document on sorting.

Step 4: Each non seed document is assigned to seed which covers its maximum.

Step 5: Uncovered documents are moved to ragbag cluster.

The document collections $\{d_1, d_{2\ldots\ldots} d_m\}$is considered as D matrix and the index terms $T=\{t_1, t_{2\ldots} \ldots . t_n\}$ is given. The C matrix is a document-by-document matrix whose entries c_{ij} ($1 \leq i, j \leq m$) indicate the probability of selecting any term of the document (d_i) from document (d_j), where d_i and d_j are the members of D matrix.

Using D matrix, construct S probability matrix. Multiplying S matrix with the transpose of $S'(S^{'T})$ forms the m-by-m C matrix. By multiplying S and $S^{'T}$, the C matrix is constructed.

The entries in C matrix c_{ij} is computed using

$$c_{ij} = \alpha_i \times \sum_{k\ 1}^{n} d_{ik} \times \beta_k \times d_{jk}\,,\ 1 \leq i,\ j \leq m, \tag{9}$$

where $\alpha_{®}$ and β_k are the reciprocals of the ith row sum and the kth column sum.

$$\alpha_i = \left[\sum_{j=1}^{n} d_{ij}\right]^{-1}, \quad 1 \leq i \leq m,$$
$$\beta_k = \left[\sum_{j=1}^{m} d_{jk}\right]^{-1}, \quad 1 \leq k \leq n. \tag{9}$$

The C matrix (c_{ii}) diagonal values are referred to as a decoupling coefficient and are denoted with the symbol δi. This calculation indicates exactly how irrelevant the document is to the other documents. The coefficient of coupling is determined by using i^{th} row off-diagonal entries sum, and it is denoted with the symbol ψ_i.This coefficient shows the extent of coupling of di with the other documents of the database.

To select seed documents, cluster seed power for all documents is computed, by using the Equation (8).

The threshold value is calculated as:

$$Tr = \sum_{d_i \in document} P_i / Current\ No.of\ documents. \tag{11}$$

If the Seed Power $P_i \geq$ Tr form a new cluster with cluster id *i*. Otherwise, compute the semantic similarity between the current documents with other documents in all clusters. The document is assigned to the highest semantic score cluster.

5. Experimental Results

This section presents the experimental evaluation to assess the quality of clustering algorithms. Three real-time data sets are used for cluster evaluation. The minimum configuration required is Intel Dual Core Processor with 2 GB RAM. The whole experiment was carried in Java so JDK 1.8 was used.

Data Sets

BBC Dataset: The BBC data set consists of 2225 BBC news website documents relating to the reports from the period 2004-2005 in five topical fields. The dataset is divided into five groups of life, such as business, entertainment, politics, sport and technology.

R8 Dataset: R8 dataset is a Reuter's subset of the collection (21578). The data Reuters-21578 contains eight most commonly used classes. The labels of the **class** are acq, crude, earn grain, interest, money, ship and trade.

NewsGroups (NG) Dataset: It is a series of around 20,000 newsgroup articles, partitioned (nearly) equally across 20 different **newsgroups**. The compilation of 20 newsgroups has become a common data set for research on machine learning techniques in text applications, for instance, text classification and clustering.

Table 1 shows the data set configuration used for the experiments.

Table 1
Data set Configuration

Dataset	No of class	No of docs	No of terms	Avg Terms/ Doc
BBC	5	100	16542	165.42
R8	8	100	4023	40.23
NG20	20	100	9078	90.78

The proposed framework is evaluated for quality assurance using standard relevant metrics such as precision (P), recall ®, and F-measure (FM) in the field of information retrieval.

Performance Analysis

This section analyses the performance of the proposed work. After the preprocessing and dimension reduction, the numbers of terms are reduced. Table 2 shows the details of the terms after dimensionality reduction.

Table 2
Dimensionality reduction results

Data set	Total terms after the process	No of Key Terms On reduction		No of terms Avg Terms/Doc	
		Using Freq	OntoVSM	Using Freq	OntoVSM
BBC	5760	1050	952	81.77%	83.47%
R8	1682	223	210	8.74%	87.51%
NG 20	3493	608	569	82.59%	83.71%

Form Table 2, the proposed OntoVSM decreases the terms by more than 80%. The values also show that Into VSM works better than the conventional approach of the frequency of terms. More reduction of terms leads to poor extraction of semantic information.

Table 3 shows the summary of evaluation metrics for BBC data set.

Table 3

Result Summary of BBC Data set

Algorithm	P	R	FM	Acc	CT (sec)
A1	0.56	1.0	0.72	0.72	2.217
A2	0.49	1.0	0.66	0.70	2.13
A3	0.91	0.45	0.60	0.75	1.815
A4	0.81	0.80	0.80	0.89	3.5
A5	0.84	0.82	0.83	0.96	3.7
A6	0.65	1.0	0.79	0.81	1.484

where A1 = K-means- TFIDF, A2 = K-means-OntoVSM, A3 = K-means-LSA, A4 = C3 M-TFIDF, A5= C3 M-OntoVSM, and A6 = Cover-K-means-OntoVSM, P=precision, R= recall, FM=fmeaure, Acc= Accuracy and CT= computing time for the throughout results.

Figure 1 shows the comparison of evaluation metrics for BBC data set. From that result, K-means LSA gives high precision compared to other methods. The recall of proposed Cover-K-means is much better with the recall of 1 than K-means LSA and Cover OntoVSM. The F-Measure achieved by Cover-K-means is much better than K-means-LSA.

Figure 1

Comparison of evaluation metric – BBC data set

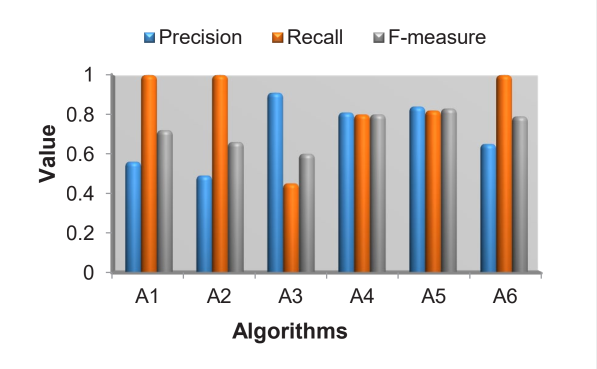


Table 4 shows a summary of results for the R8 Data set.

Table 4

Result summary of R8 data set

Algorithm	P	R	FM	Acc	CT (sec)
A1	0.61	0.59	0.60	0.75	0.793
A2	0.48	1.0	0.65	0.79	0.572
A3	0.97	0.31	0.47	0.77	0.32
A4	0.85	0.78	0.81	0.89	1.3
A5	0.87	0.81	0.84	0.97	1.45
A6	0.78	0.72	0.75	0.83	0.287

Figure 2 shows the graphical representation of precision, recall and f-measures. The precision of K-means-LSA is better than Cover-K-mean. The algorithm C3M- OntoVSM shows a nearer precision performance to K-means-LSA. The F-Measure of Cover-K-means using OntoVSM dimension reduction is much better than K-means LSA by more than 50%.

Figure 2

Evaluation metrics for the R8 data set

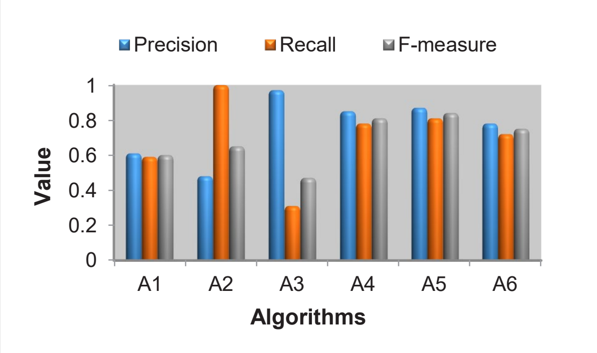


Table 5 and Figure 3 shows the summary of precision, recall, F-measure of various algorithms for newsgroup dataset.

The precision of K-means and C3M is much better than the proposed Cover-K-means. K-means achieves 24.4% better precision than Cover-K-means. The proposed Cover-K-means with dimension reduction achieves maximum recall value of 1 than K-means and Cover-K means-Onto. Cover-K-means achieves much better F-measure value by 69.5% than K-means LSA.

Table 5

Result summary of NG20 data set

Algorithm	P	R	FM	Acc	CT (sec)
A1	0.61	1.0	0.76	0.73	1.249
A2	0.63	1.0	0.77	0.77	1.202
A3	0.94	0.33	0.49	0.69	0.953
A4	0.83	0.82	0.82	0.88	1.5
A5	0.84	0.88	0.86	0.98	1.64
A6	0.71	1.0	0.83	0.81	0.864

Figure 3

Evaluation metrics for news group data set

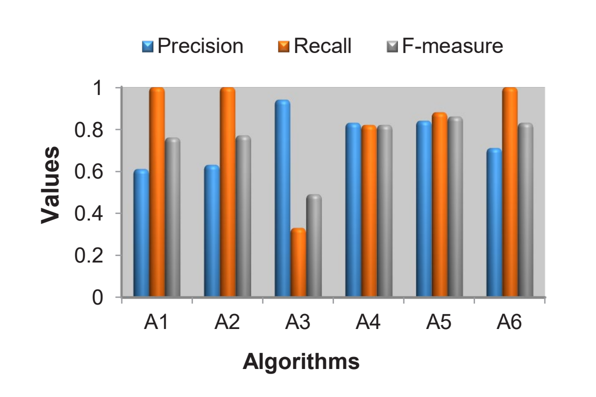
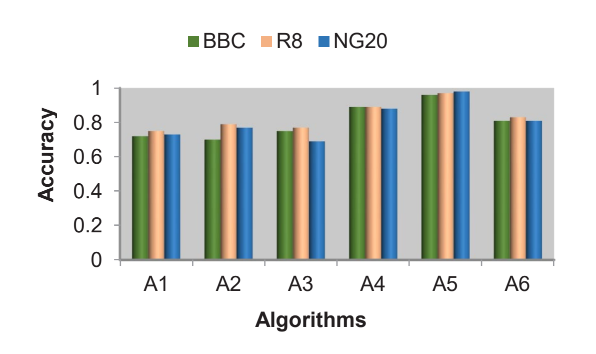


Figure 4 shows the accuracy comparison of various methods for the BBC, R8 and NG20 data set.

Figure 4

Accuracy comparison of three data set

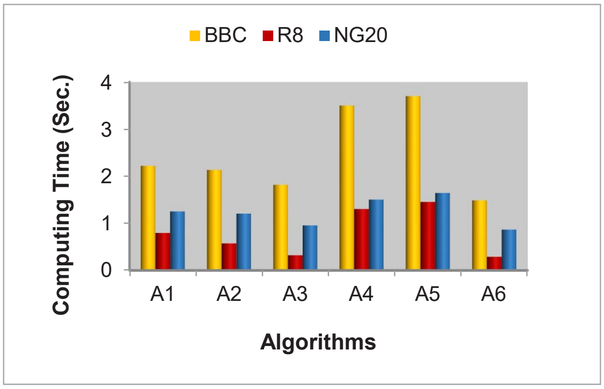


The Cover-K-means is better than K-means-LSA for BBC and NG20 dataset

Figure 5 shows the comparison of computing time for the BBC, R8 and NG20 data set. For R8 dataset Cover-K-means with Onto dimension reduction achieves a little improvement of 0.03 secondslesser computing time than K-means LSA. In contrast, two methods of C3M shows very high computing time. For NG20 dataset, Cover-K-means OntoVSM achieves less computing time with the difference of 0.89 sec when compared with K-means LSA.

Figure 5

Computing time for three data set



6. Conclusion

This paper proposes a semantic clustering using dimensionality reduction. The vector space model is modified using ontology called OntoVSM. It efficiently reduces the terms compared to the frequency method. The traditional K-means and cover coefficient clustering is improved semantically with WordNet ontology. The experimental results show that both frequency and OntoVSM method reduces the terms of more than 80%. It also indicates that clustering with OntoVSM performance has improved results compared to other methods.

Acknowledgement

This research was supported by Korea Institute for Advancement of Technology (KIAT) grant funded by the Korea Government (MOTIE) (P0012724, The Competency Development Program for Industry Specialist) and the Soonchunhyang University Research Fund.

References

1. Allab, K., Labiod,L., Nadif,M. A Semi-Nmf-Pca Unified Framework for Data Clustering. IEEE Transaction Knowledge Data Engineering, 2017, 29(1), 2-16. https://doi.org/10.1109/TKDE.2016.2606098
2. Balasubramaniam, K. Hybrid Fuzzy-Ontology Design Using FCA Based Clustering for Information Retrieval in Semantic Web. Procedia Computer Science, 2015, 50, 135-142. https://doi.org/10.1016/j.procs.2015.04.075
3. Bouras, C., Tsogkas, V. A Clustering Technique for News Articles Using WordNet. Knowledge-Based Systems, 2012, 36, 115-128. https://doi.org/10.1016/j.knosys.2012.06.015
4. Capizzi, G., Lo Sciuto, G., Woźniak, M., Damaševicius, R. A Clustering Based System for Automated Oil Spill Detection by Satellite Remote Sensing. In: Rutkowski L., Korytkowski M., Scherer R., Tadeusiewicz, R., Zadeh, L., Zurada, J. (Eds.) Artificial Intelligence and Soft Computing. ICAISC 2016. Lecture Notes in Computer Science, 9693. Springer, Cham, 2016, 613-623. https://doi.org/10.1007/978-3-319-39384-1_54
5. Dang, Q., Zhang, J., Lu Y., Zhang, K. WordNet -Based Suffix Tree Clustering Algorithm. International Conference On Information Science and Computer Applications, 2013. https://doi.org/10.2991/isca-13.2013.12
6. Fahad, S., Yafooz, S. Design and Develop Semantic Textual Document Clustering Model. Journal of Computer Science and Information Technology, 2017, 5(2), 26-39. https://doi.org/10.15640/jcsit.v5n2a4
7. Jensi Dr, R., Wiselin Jiji, G. A Survey On Optimization Approaches to Text Document Clustering. International Journal of Computer Science and Application (IJCSA), 2013, 3(6), 31-44. https://doi.org/10.5121/ijcsa.2013.3604
8. Kuang, Da., Choo, J., Park, H. Nonnegative Matrix Factorization for Interactive Topic Modeling and Document Clustering. In: Celebi, M. (Ed.) Partitional Clustering Algorithms. Springer, Cham, 2015, 215-243. https://doi.org/10.1007/978-3-319-09259-1_7
9. Li, R., Zhang, L., Du, B. A Robust Dimensionality Reduction and Matrix Factorization Framework for Data Clustering. Pattern Recognition Letters, 2019, 128, 440-446. https://doi.org/10.1016/j.patrec.2019.10.006
10. Mendizabal, I. V., Basto-Fernandes, V., Ezpeleta, E., Méndez, J. R., Zurutuza, U. SDRS: A New Lossless Dimensionality Reduction for Text Corpora. Information Processing & Management, 2020, 57(4). 1-13. https://doi.org/10.1016/j.ipm.2020.102249
11. Onan, A., Bulut, H., Korukoglu, S. An Improved Ant Algorithm with LDA Based Representation for Text Document Clustering. Journal of Information and Science, 2017, 43(2), 275-292. https://doi.org/10.1177/0165551516638784
12. Pamba, R. V., Sherly, E., Mohan, K. Self-Adaptive Frequent Pattern Growth-Based Dynamic Fuzzy Particle Swarm Optimization for Web Document Clustering. In: Verma, N., Ghosh, A. (Eds.) Computational Intelligence: Theories, Applications and Future Directions - Volume II. Advances in Intelligent Systems and Computing, 799. Springer, Singapore, 2019, 15-25. https://doi.org/10.1007/978-981-13-1135-2_2
13. Saravana Kumar, C. S., Santhosh, R. Effective Information Retrieval and Feature Minimization Technique for Semantic Web Data. Computers & Electrical Engineering, 2020, 81, 1-14. https://doi.org/10.1016/j.compeleceng.2019.106518
14. Shah, N., Mahajan, S. Semantic-Based Document Clustering: A Detailed Review. International Journal of Computer Applications, 2012, 52(5), 42-52. https://doi.org/10.5120/8202-1598
15. Sharma, I., Jain, A., Sharma, H. Stream and Online Clustering for Text Documents. In: Kamal, R., Henshaw, M., Nair, P. (Eds.) International Conference on Advanced Computing Networking and Informatics. Advances in Intelligent Systems and Computing, 870. Springer, Singapore, 2019, 469-475. https://doi.org/10.1007/978-981-13-2673-8_49
16. Soares, V. H. A., Campello, R. J. G. B., Nourashrafeddin, S., Milios, E., Naldi, M. C. Combining Semantic and Term Frequency Similarities for Text Clustering. Knowledge and Information Systems, 2019, 61, 1485-1516. https://doi.org/10.1007/s10115-018-1278-7
17. Stanchev, L. Semantic Document Clustering Using Information from WordNet and DBPedia. IEEE 12th International Conference on Semantic Computing (ICSC), Laguna Hills, USA, 2018, 100-107. https://doi.org/10.1109/ICSC.2018.00023
18. Sun, Z., Huo, H., Huan, J., Vitter, J. S. Feature Reduction Based on Semantic Similarity for Graph Classification. Neurocomputing, 2020, 397, 114-126. https://doi.org/10.1016/j.neucom.2020.02.047
19. Tru, H. C., Khanh, C. L, Ngo, V. M. Exploring Combinations of Ontological Features and Keywords for Text Retrieval. In: Ho, T. B., Zhou Z. H. (Eds.) PRICAI 2008: Trends in Artificial Intelligence. PRICAI 2008. Lecture Notes in Computer Science, 5351. Springer, Berlin, Heidelberg, 2008, 603-613. https://doi.org/10.1007/978-3-540-89197-0_55

This article is an Open Access article distributed under the terms and conditions of the Creative Commons Attribution 4.0 (CC BY 4.0) License (http://creativecommons.org/licenses/by/4.0/).