FUZZY DOCUMENT REPRESENTATION FOR SEARCH DIVERSIFICATION

SIJIN P
Department of Computer Science and Engineering,
University Visvesvaraya College of Engineering Bangalore University, Bangalore, India
psijin@gmail.com

DR. CHAMPA H. N.
Department of Computer Science and Engineering
University Visvesvaraya College of Engineering, Bangalore University, Bangalore, India
champahn@yahoo.co.in

ABSTRACT

Fuzzy document representation involves transforming the unstructured data into numerical vectors. Such a representation is more useful for text classification and document clustering. The proposed Fuzzy Conceptualization Model (FCM) performs conceptualization and provides a better data representation model on the basis of semantic relatedness and similarity between terms in a word corpus. Word embedding is used to hold the semantically related words in a concept cluster. The concept clusters are inferred and vectored for the given corpus to hold the data in a multidimensional space. FCM determines the fuzzy membership value of a base term by calculating the affinity score between its corresponding word embedding and other word embeddings. A weighing scheme is used to distinguish between exact and approximate matches. The greatest bound for the distribution of base set over the documents gives the best matched documents for a search query. The exact and approximate matches are differentiated by considering the normalized term frequency of a term in the specified concept cluster along with its actual presence. The resultant matrix gives a lower dimensional and discriminated representation of data. Moving with the data points having discriminated and non discriminated nature over an affine vector leads to the clustering of them well with proper anchoring of them with the previous mile stones of each data points. The proposed model is useful for the retrieval of information with short and vague keywords. The experimental analysis of the FCM on synthetic and real data sets shows high accuracy in results.

KEYWORDS: BoW, SVD, FCM.

INTRODUCTION

Document is a collection of objects such as texts, images, tables, charts, graphs etc. Vector representation of documents deals with transforming the unstructured text data into numerical vectors. The document vector representation is achieved by methods like matrix representation, Term-Frequency and Inverted Document Frequency (tf-idf) with Cosine similarity, Bag-of-words (BoW) [8], Fuzzy Bag-of-Words (FBoW) etc. Matrix stores data in the form of a table. It usually contains numerical values and character strings. The most used form of a matrix is its two dimensional representation. BoW maps a document to a fixed length vector. The mapping function in this method is binary. It says only about the presence or absence of base terms in a document by an exact word match. An intuitive illustration of the BoW representation for the documents d1, d2 and d3 with base terms {table, book, on, in} is given here. Document d3 doesn't contain any one of the base terms. The documents are given below.
d1: A bag is on the table
d2: A book is in the draw
d3: pen and desk
The BoW projections of the document d1<1,0,1,0> , d2<0,1,0,1> and d3<0,0,0,0> show, documents d1 and d2 are related, d3 is not related with either d1 or d2 since it doesn't have a membership value for any of the base terms in the given corpus. It is notable that the base term 'bag' is semantically related to another normal term 'book', and the base term 'table' is semantically related to the term 'draw'. But BoW fails to

capture the real semantics between these terms. The BoW method suffers with high dimensionality, Intrinsic extreme sparsity and inability to capture semantic relationship.

The concepts of Fuzzy Bag-of-Words (FBoW) and Fuzzy Bag-of-Word Cluster (FBoW) models [16] modify the above model by considering the number of occurrences of the base terms by the equation

$$z_i=c_i*\sum_{wj\epsilon w} t_i w_j x_j \qquad (1)$$

Where c_i is a controlling parameter, w denotes set of words, t_i is the i^{th} base term, x_j is the number of occurrences of the word w_j, and z_i is the sum of the membership degree.

The FBoW projections for the document d1<1,.01> , and d2<.0001,1> is given by avoiding stop words such as ‘on’, ‘in’, and ‘and’. The result shows the word pairs <bag, book> and <table, draw> show the semantic relationship among themselves.

The proposed FSM model can efficiently list out the more discriminated documents from a word corpus, even for vague and short keywords. It is able to deal with exact and approximate matches by a weight oriented SVD computation. The proposed keyword diversification framework considers the semantic meanings of the terms with respect to a search query. The FCM achieves conceptualization and modeling of data over supervised and unsupervised learning algorithms. The inference clustering allows adjusting the search space based on the concept density.

RELAED WORKS

TEXT REPRESENTATION

Text processing involves data clustering, classification and machine translation. Text segmentation is the process of dividing a text into sequence of terms [6]. Text segmentation methods are of two types, statistical based and vocabulary based. POS tagging determines the lexical types of words in a text by measuring lexical probabilities and sequential probabilities of the words in a word corpus. The semantic labeling processes identify the hidden semantic relations among words in a word corpus.

In order to minimize the drawback of BoW , methods like Latent Semantic Analysis (LSA) [10], [9], [4] and Topic modeling [2] have introduced. Topic models further classified into Probabilistic Latent Semantic Analysis (PLSA) and Latent Dirichlet Allocation (LDA). Topic modeling is based on statistical model. It is used for discovering abstract topics from a collection of documents; it is widely used for text modeling, collaborative filtering and text classification. Latent Dirichlet allocation model (LDA) is the first graphical model for topic modeling [2], [1]. LDA focuses on maximizing the separability among known categories. It projects the data onto a new axis in a way to maximize the separation of the two categories. The process of mapping the high dimensional count vectors to a lower dimensional representation is known as LSA. It is proposed to improve the BoW model. LSA transforms the BoW model representation to lower dimensional representation of documents to capture the semantic relationships among them. LSA achieved it by a method called Singular value Decomposition (SVD). The SVD produces a new dimension from the linear combination of all original dimensions. The proposed model adopted the same method with a weighing scheme. In Fuzzy bag of word model a text is represented as the multi set of its words, FCM calls the multi-set as base terms which can be distributed over the documents.

WORD EMBEDDINGS

Word embeddings are words converted into numbers. Word embedding is the mapping of words or phrases of a vocabulary to a vector of real numbers [8], [12]. It offers words with same meaning have similar representation. The works in [13] says the learned word representation can capture the syntactic and semantic regularities in a simple way. The proposed work used the word embedding methods to hold the semantically and syntactically related words along with their varying number of projections together. The learned document representation reduces the complexity in word representation.

SEARCH DIVERSIFICATION WITH CONCEPTUALIZATION

The statement that precedes or follows a specific word in a sentence or passage determines the context of that passage. The surrounded words may be a definition, examples, synonyms or antonyms, substitution for the given term of selection. The information in knowledge base need to be probabilistic in order to model text for inferencing. The proper context of the term in a sentence can identify only by the thorough knowledge about the terms in text segmentation, part of speech tagging, and concept labeling process. This will help to harvest the proper semantic of a word [11]. The lexical semantic relationships among terms can be extracted from a knowledge base, web corpus, probabilistic network, or a probabilistic database [3], [14]. The average number of attributes, features and binary relations issued from a given concept node is known as knowledge density. If semantic coherence is considering, the traditional Longest Cover Method for text segmentation is not suitable. The state-of-the art text segmentation methods will not consider the semantic relationships among various instances and concepts [5], [15]. The proposed method provides a best suit of data representation and clustering via inferred over the knowledge base by measuring and preserving the relative movements of terms over the document vector.

DATA CLUSTERING AND CLASSIFICATION

The state-of-art K-mean algorithm is the most commonly used clustering method. It takes k input parameters and partitions the n objects in to k clusters, by maintaining inter cluster similarity less and intra-cluster similarity more[17]. The fuzzy clustering algorithm classifies the data objects into distinct data lines and can be inferred by a given threshold. In Fuzzy clustering objects are assigned to different clusters. In fuzzy object representation clustering each object is assigned to exactly one cluster [7]. The proposed fuzzy inference algorithm assigned each object to exactly one cluster or else an optimization by allowing the objects to be assigned in different clusters they belong provided with data feed backing ability.

METHODOLOGY

BASELINE APPROACH

Let A is an MXN term-document matrix of real numbers or complex numbers which represents a collection of documents. Each column of A corresponds to a document and each row corresponds to words related to the base terms. The factorization of A is done by Singular Value Decomposition method. This is achieved by solving the equation

$$A=S\sum U^T \qquad (2)$$

Consider another two matrices $B=A^TA$ and $C= AA^T$, it is possible to perform SVD on matrix A by using the generated matrices B and C. S is the matrix of the Eigen vectors of B. ∑ is the diagonal matrix of the singular values obtained as square roots of the Eigen values of B. U is the matrix of the Eigen vectors of C. Hence the SVD of A is

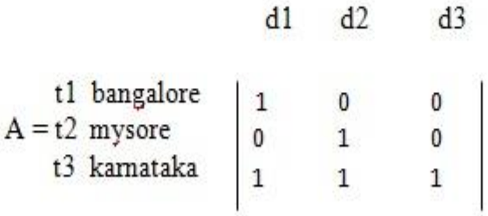
$$A_k= S_k\sum_k U_k^T \qquad (3)$$

where k is the number of singular values. In Least Semantic Indexing (LSI) ∑ values will all zeros except the first k entries along its diagonal.

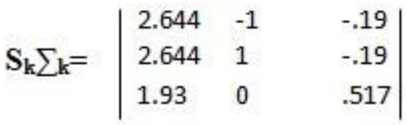
The following documents, have used to show the usage of SVD on a data corpus.

d1: bangalore, Karnataka
d2: mysore, Karnataka
d3: Karnataka
q: bangalore, mysore

The matrix A is given as



$$A = \begin{array}{l|ccc|} & d1 & d2 & d3 \\ \text{t1 bangalore} & 1 & 0 & 0 \\ \text{t2 mysore} & 0 & 1 & 0 \\ \text{t3 karnataka} & 1 & 1 & 1 \end{array}$$



$$S_k\sum_k = \begin{vmatrix} 2.644 & -1 & -.19 \\ 2.644 & 1 & -.19 \\ 1.93 & 0 & .517 \end{vmatrix}$$

Fig 1: The document-term matrix

Fig 2: The Term matrix

$$\sum_k U_k{}^T = \begin{vmatrix} .71 & .71 & 1.93 \\ -1 & 1 & 0 \\ -.71 & -.71 & .52 \end{vmatrix}$$

Fig 3: The Document matrix

Here the search query is bangalore, mysore . The query can be represented in matrix form as the average of matrices representing the query terms such as bangalore and mysore.

The document-term matrix is given in Fig 1. The cosine values of terms and documents are calculated from the values as illustrated in Fig.2 and Fig.3. The cosine distance of each documents with the given query has calculated as cos(d1,q)=0.8240, cos(d2,q)=0.7030, cos(d3,q)=0.6783.

The cosine values should be in the range of -1 to 1, the three documents with the given query have a cosine distance in the specified range of 0 to 1(-1 to 0 shows the range of dissimilarity), so they are related. In other words for a search query “bangalore, mysore”, the documents d1(bangalore, karnataka), d2(mysore, karnataka) and d3(karnataka) are produced.

FUZZY CONCEPTUALIZATION MODEL

The Fuzzy Conceptualization Model performs conceptualization and measure of co-occurrence count of a typed term by considering the semantic relationship between the base terms set and the concept clusters they belongs. Word embeddings is used to hold the semantically related words in a concept cluster. FCM determines the fuzzy membership value of a base term by calculating the affine score between its corresponding word embedding and other word embeddings. Seperate weighing score is given to the exact and approximate matches, since majority of the search queries are short keywords with a distinct search term or a fuzzy set of it. For example the fuzzy set like (Donald Trump, the president). An SVD computation on the base term, document matrix gives a lower dimensional and weighted representation of data. The proposed document representation frame work is given in Fig. 4.

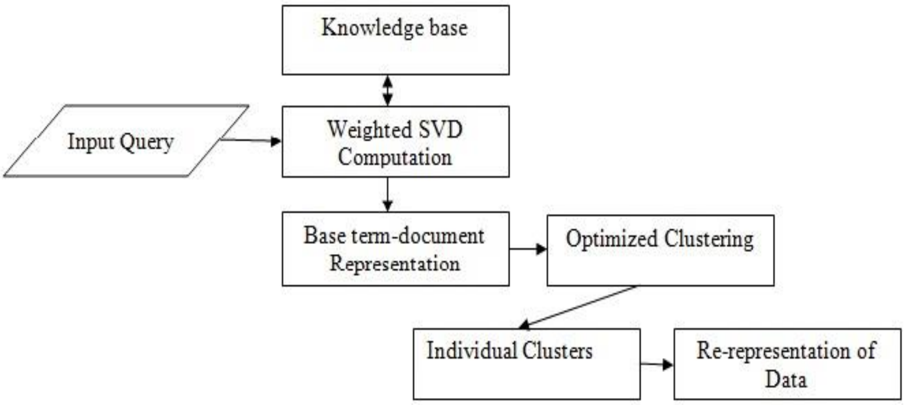


Fig 4: Document Representation Framework

The baseline approach gives unique weight to a concept cluster, even though it produces relevant results, the result set size may be large. The FCM considers the normalized term frequency of a term from the concept cluster or clusters it belongs.

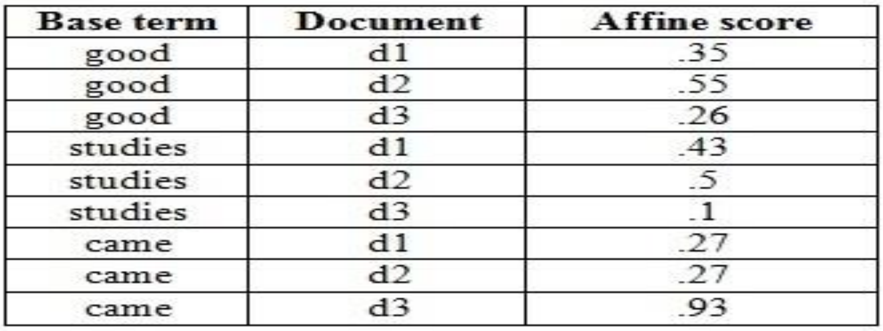
The following documents, have used to show the usage of weighed SVD on a data corpus.

d1: raju is good in studies
d2: juli came by train, she is good
d3: victor came for conference

The base term set is given as {good, studies, came}

The actual presence of a term in a document has given a value of the absolute sum of the normalized term frequency of that term (ntf) plus one(actual presence is represented by 1). The partial presence is represented by the normalized term frequency of that term. The affinity score obtained for the above example is given in Table I.

Table I. Affinity Score Table



Base term	Document	Affine score
good	d1	.35
good	d2	.55
good	d3	.26
studies	d1	.43
studies	d2	.5
studies	d3	.1
came	d1	.27
came	d2	.27
came	d3	.93

Each concept cluster contains the instances and attributes related to the given terms, or it can be projected to the contextual branches of the conceptual networks to fetch out the results. The Fig. 5 shows the distribution of concept clusters over the entities of a knowledge base.

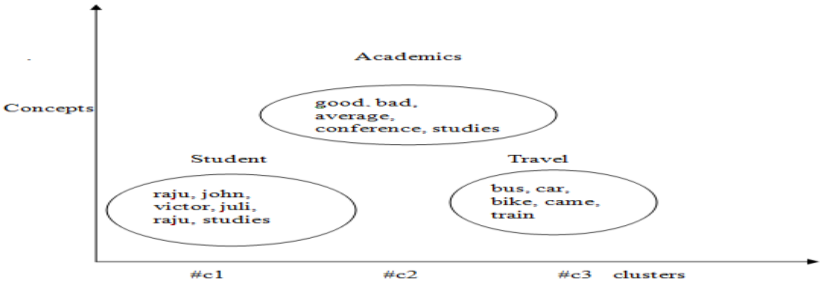


Fig 5: Concept distribution over a knowledge base

By using a weighted SVD Computation, the base terms-document relations are identified. The knowledge of how the base terms are distributed over the documents will give the intuition that which concepts are highly related. The cluster quality is not guaranteed here.

The optimized mean value algorithm is used for data clustering. The three concept clusters obtained for good, studies, and came plotted to a common axis in Figure 6.

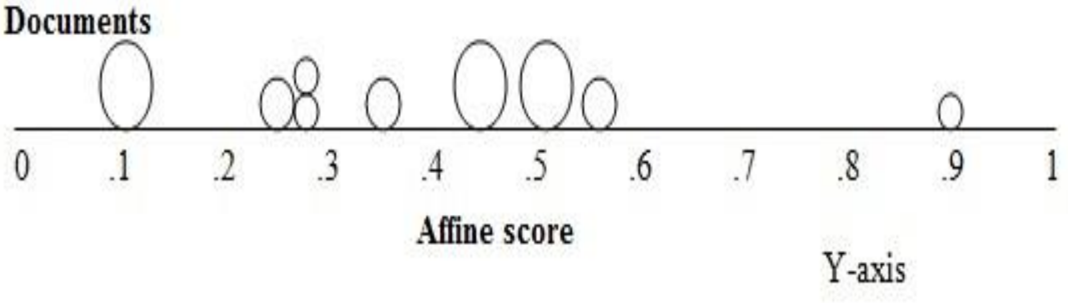


Fig 6: Document representation over concepts

The small circle represents the documents for the term "came". The bigger circle represents the documents for the term "good" and the biggest circle represents the documents for the term "studies". For each and every concept clusters here, the clusters belong to good, came and studies, tried to bring the document's affinity scores to the respective mean values of the selected clusters. In such an act they over took some of the documents belong to other clusters. For example the concept handle "good" (good is the representative of the concept clusters it belongs or simply a term) , the mean obtained for the documents belong to this concept is 0.39, The three matched documents are approached to this dynamic virtual cluster center, at that time the document d3 has to jump over the documents d1 and d2 belong to the concept term "came". So the d3 of "good" jumped by 0.02 to over comes the 0.27 value of d1 and d2 of "came". For d1 of "good" no need to over take any value to approach mean, but for d2 of "good" has to over come d1 and d2 of "studies" by -.13 and -.06 respectively. This method produced highly separable clusters by compromising quality (since some jumps are long jumps) refer Figure. 7.

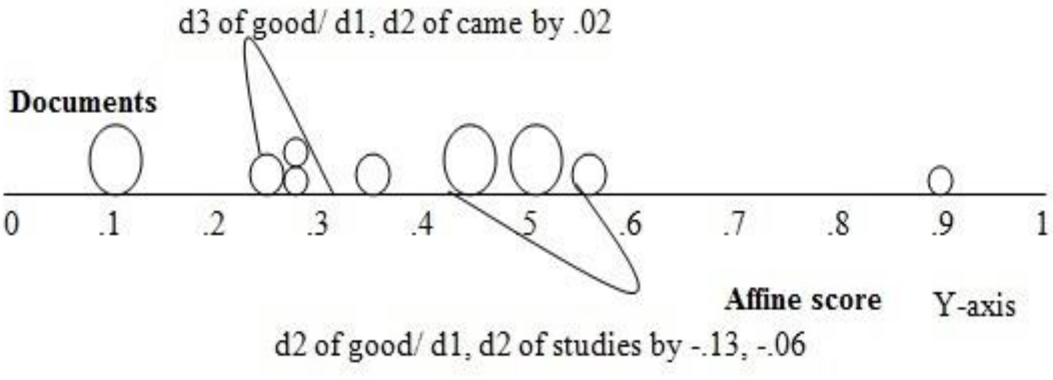

d3 of good/ d1, d2 of came by .02
Documents
0
.1
.2
.3
.4
.5
.6
.7
.8
.9
1
Affine score
Y-axis
d2 of good/ d1, d2 of studies by -.13, -.06

Fig 7: Document jumping process

The document jump for term "good" is shown in Figure. 8, the old value is given in brackets. Similar jump is possible for all the other terms(i.e came, studies) in the given corpus. The clustered data can be represented with the modified values in separate vectors and can memorize all the previous mile stone values and the values used for jumps in order to attain clustering as depicted in Algorithm 1. This approach will help to set a very clear bound to each and every clusters for an accurate inferencing over the knowledge base by reducing the dimensionality problem.

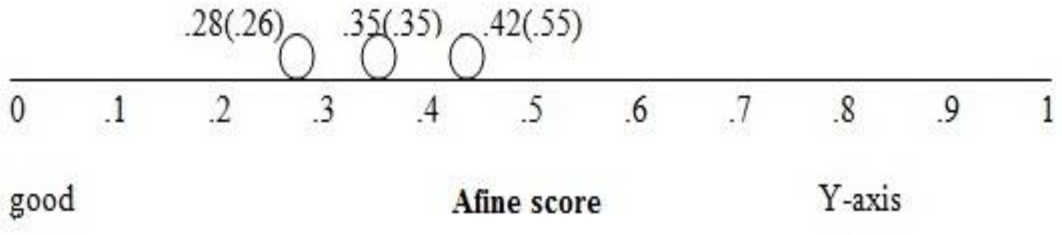


Fig 8: Document jumping process for the term "good”

After the first iteration of the algorithm, it is noted that, the documents are clustered for the base terms "came" and "studies", but for "good", it still remains for further iteration. The second iteration of the algorithm clustered the documents on the context of "good" in the given corpus , it is shown in Figure 9.

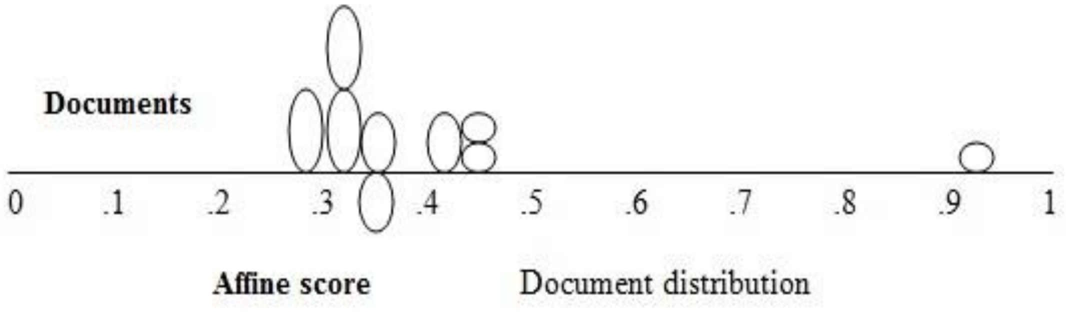


Fig 9: Inference Clustering

ALGORITHM

Algorithm 1: DataClusteringAlgorithm

```
Input: A query q with n key words and a data set contains terms, concepts and
Instances and the number of iterations k.
Output: <= n numbers of data clusters with mile stone values.
while(no.of iterations<k)  do
    for all concepts $c_i$ selected do
        if $c_{ival}<c_{imean}$ do
            for all $c_{i+1}$, i=1 to n do
                $milestone_{left}=max(compare(c_{ivall},c_{i+1val}))$
                $update(c_{ivall}=milestone_{left})$
            end
        end
        if $c_{ival}>c_{imean}$ do
            for all $c_{i+1}$, i=1 to n do
                $milestone_{right}=min(compare(c_{ivalr},c_{i+1val}))$
                $update(c_{ivalr}=milestone_{right})$
            end
        end
        $c_{inew}=concat(c_{ivall},c_{ivalr})$
        if $milestone_{left}$>cimean and $milestone_{right}$<cimean
            remove(ci);
        increment k
end
```

EXPERIMENTAL SETUP

The paper emphasizes more on efficient document representation process for machine learning applications. The experimental analysis shows some significant signs, to achieve relevancy without penalizing with computation time and data discrimination problems. FCM with a proper clustering is a better option. Document clustering is the process in which, group the documents which show some common behavioral patterns, have common attributes, shares some common instances, obey some association rules, under a common concept. Data discrimination and dimensional reduction are the main problems commonly arise in such situations. FCM uses logical jumps which can remember the previous states. The experimental analysis with synthetic dataset classroom and the real world data set Iris, FCM produces high quality clusters and an adjustable inference frame work based on concept density.

DESCRIPTION OF DATA SET

The task of document representation is to assign a numerical value to the selected objects, and let them distribute over the concept clusters.

Class room: The class room dataset is created for the experimental analysis of the various modules of BoW, FBoW and FCM algorithms which are used for studies and comparisons. This dataset contains concepts such as Academics, Travel etc, and their various attributes. The data set is still accumulating.

Iris: Iris data set is a collection of three set of flowers named Iris-Setosa, Iris-Virginica, Iris-Versicolor, with their attributes such as sepal length, sepal width, petal length, petal width. The base term queries can be distributed over various documents, the supervised version of FCM concentrates to distribute the relevant documents obtained for the query on three distinct data lines and to measure the most related and relevant documents on each concepts.

EXPERIMENTAL ANALYSIS

Relevancy Measures: The experimental measures include accuracy, semantic preservation rate (SPR), and computation time. The classification accuracy can be defined as the sum of the correctly identified data and incorrectly rejected data to the sum of incorrectly identified data and correctly rejected data. The sensitivity of a classification is represented by true positive rate of the data classified. It is a direct representation of relevancy measure. F-Score is the harmonic mean of precision and recall. The positive predictive value is known as precision. Sensitivity is called recall.

The ratio of relevant results obtained with respect to the result set is known as semantic preservation rate. The computation time includes, the sum of the time required for data fetch, tts representation and classification along with clustering.

Implementation: The FCM algorithm implementation shows the data convergence on a specific data line by remembering the previous results.

EXPERIMENTAL RESULTS

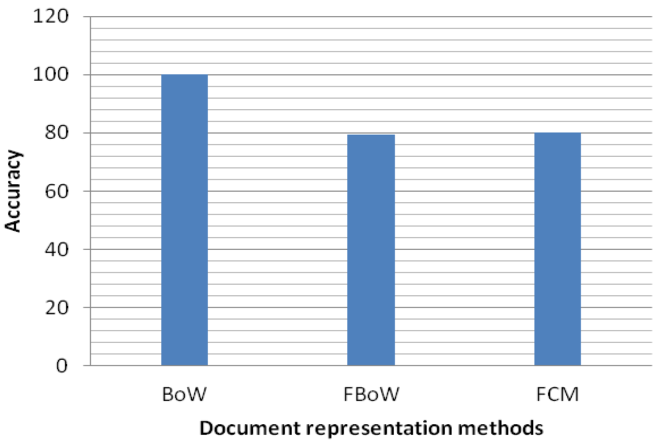


Fig 10: Accuracy Comparison

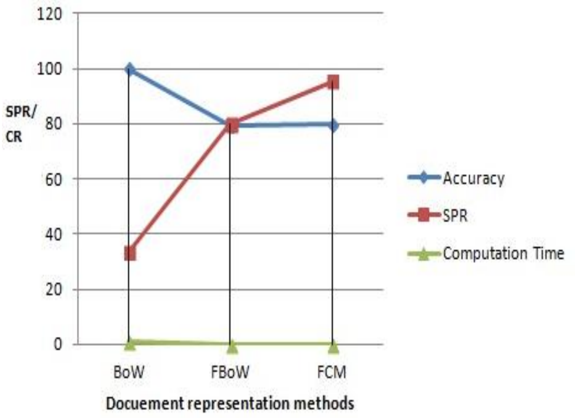


Fig 11: SPR/CR Comparison

Fig 10 compares the accuracy measures of FCM with the methods like BoW and FBoW. FCM shows high accuracy without penalizing SPR rate. Fig 10 shows the SPR/Computation Rate(CR) comparisons of the FCM with other methods, and shows its high rate of relevancy level without penalizing accuracy for a large margin.

CONCLUSION

Data mining is the process of performing logical inference over the data sets to identify the related information such as common concepts, attributes, instances and some semantic patterns. The proposed algorithm produces highly relevant results without compromising much with accuracy and computation time. FCM uses the existing document representation methods for document representation, classification and clustering. The relevancy level has increased by a weighted SVD computation without losing data accuracy much. The actual match and relevant matches are adjusted by fuzzy conceptualization of the documents to the selected concepts. The documents distribute over the concepts, and clustered to a unique data line. FCM algorithm can memorize the previous results in such a way that it can measure, how the documents are related to different concepts in terms of cosine distance and concept density. Thus FCM reduced the hard mapping issues of BoW methods and data discrimination problems of other FBoW methods and allows the data points to converge by the given thresholds to produce more high quality clusters.

ACKNOWLEDGMENT

We would like to thank Dr. K R Venugopal Vice Chancellor, Bangalore University, Dr. H.N Ramesh, Dr. Dilip Kumar, Dr. Deepa Shenoy, Dr. Arunalataha J S, Dr. Thriveni J , Dr. Pushpa C N, Dr. Kiran K, Dr. Dharmendra Chauhan, Dr. Venkatesh M, Dr. Tanuja R, Dr. Manjula S H, Dr. Vimala H S, Dr. Lata B T, Dr. Samyama Gunjal for their guidelines and valuable suggestions for the research works to proceed. Thankfully remembers all UVCE staffs, and friends for their constant support to the research.

REFERENCES

1) S. Balakrishnama and A. Ganapathiraju, “Linear Discriminant Analysis-a Brief Tutorial," Institute for Signal and information Processing, vol. 18, pp. 1-8, 1998.
2) D. M. Blei, A. Y. Ng, and M. I. Jordan, “Latent dirichlet allocation," Journal of machine Learning research, vol. 3, no. 10 pp. 993-1022, 2003.
3) J.T. Chien and C.H. Chueh, “Topic-based Hierarchical Segmentation," IEEE Transactions on Audio, Speech, and Language Processing, vol. 20, no. 1, pp. 55-66, 2012.
4) S. T. Dumais, “Latent semantic analysis," Annual review of information science and technology, vol. 38, no. 1, pp. 188-230, 2004.
5) W. Hua, Z. Wang, H. Wang, K. Zheng, and X. Zhou, “Short Text Understanding Through Lexical-Semantic Analysis." ICDE, pp.495- 506, 2015
6) W. Hua, Z. Wang, H. Wang, K. Zheng, and X. Zhou, “Understand Short Texts by Harvesting and Analyzing Semantic Knowledge,"IEEE Transactions on Knowledge and Data Engineering, vol. 29, pp. 499 512, 2017.
7) H.P. Kriegel and M. Pfeifle, “Density-based Clustering of Uncertain Data," Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining, pp. 672-677, 2005.
8) M. Kusner, Y. Sun, N. Kolkin, and K. Weinberger, “From Word Embeddings to Document Distances," pp. 957-966, 2015.
9) T. K. Landauer and S. Dumais, “Latent semantic analysis," Scholarpedia, vol. 3, no. 11, p. 4356, 2008.
10) C. Li, A. Sun, J. Weng, and Q. He, “Tweet Segmentation and Its Application to Named Entity Recognition," IEEE Transactions on Knowledge and Data Engineering, vol. 27, no. 2, pp. 558-570, 2015.
11) X. Liu, Y. Song, S. Liu, and H. Wang, “Automatic Taxonomy Construction from Keywords," pp. 1433-1441, 2012.
12) T. Mikolov, K. Chen, G. Corrado, and J. Dean, “Efficient Estimation of Word Representations in Vector Space," Cornell University Library, 2013.
13) T. Mikolov, W.-t. Yih, and G. Zweig, “Linguistic Regularities in Continuous Space Word Representations, Proceedings of the 2013 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, pp. 746-751, 2013.
14) D. Papadimitriou, G. Koutrika, Y. Velegrakis, and J. Mylopoulos, “Finding related forum posts through content similarity over intention-based segmentation," IEEE Transactions on Knowledge and Data Engineering, vol. 29, no. 9, pp. 1860-1873, 2017.
15) Y. Song, H. Wang, Z. Wang, H. Li, and W. Chen, “Short Text Conceptualization using a Probabilistic Knowledgebase," Proceedings of the Twenty-Second international joint conference on Artificial Intelligence, vol. 3, pp. 2330-2336, 2011.
16) R. Zhao and K. Mao, “Fuzzy Bag-of-Words Model for Document Representation," IEEE Transactions on Fuzzy Systems, 2017.
17) W. Zhao, H. Ma, and Q. He, “Parallel k-means Clustering based on Mapreduce," IEEE International Conference on Cloud Computing, pp. 674-679, 2009.