Finding Related Forum Posts Through Twitter Data Using K-Mean Clustering Algorithm

[1]Y.LEKHA SREE, [2]B.PRAJNA

[1]M.TECH SCHOLAR, [2] PROFESSOR
Department of Computer Science & Systems Engineering,
Andhra University College of Engineering (A), Visakhapatnam, India.

ABSTRACT: Forum posts have the specific problem of finding related posts to a post at hand. By considering across the related documents the contents of posts are generally consider as a whole. Here similarity processes are done between two posts with respective segments and should be of same intention. All posts are generally fragmented in the form of group to attain the goal bunches. Now similarities are generally cross view the forums with sections and that will be of same intention. We experimentally illustrate the effectiveness and efficiency of our segmentation method and our overall approach is finding related forum posts. Twitter is one of the biggest platform where users post tweets about everything and anything. In this paper for finding forum posts that are related we generally introduce a novel method. In that method each and every post are considered as set of segments and then they compute similarity contents across every segments with same intention.

Key words:- Forum posts, segmentation, clustering, VSM, term frequency(TF), inverse document frequency(IDF).

1. INTRODUCTION

Forums are generally a online discussion site, where people hold there conversations by posts[1]. It is like a message board and different from chat rooms. A traditional approach for finding related document is that they perform content comparisons across content of posts, the contents are compared by different posts. The relatedness of two posts can then be based on a comparison across segments that serve the same goal. Every posts are generally considered as segments. Segments are generally said as parts (or) sections. In this the relatedness between two posts should be based on similarities with respective to segments. The segmentation methods play important role by developing work with monitoring the no of text features ,it identify by parts of post. While this process is performing significant jumps are occurred because of that segmentation. Now segmentation of all posts are generally clustered in the form of intention cluster so that the similarities are calculated across segmentation with same intention. Here, Clustering plays an important role.

Generally existing forums range from domains like twitter. Twitter is a social networking service on which users post and interact with messages known as "tweets". The relatedness between the forums are compared based on segmentation process.. Work are done this direction has been in the form of questions in Q&A archives but not for richer- content posts .The compression can be performed by information retrieval method TF/IDF or BM25 variants or language- model based methods or using topics generated by topic modeling techniques like LDA paraphrasing techniques or even auxiliary external services with the latter been used especially for documents with short and poor content.

This paper is organized as follows. In section II, we review the related research study in the fields of data mining. In section III, presented the methodology and description of various processing's carried out on the dataset. The section IV consists of existing system .The section V consists of proposed system . Section VI consists of discussion of results and section VII,VIII the conclusion and references.

2. RELATED STUDY

Dimitra Papadimitriou et al. [1] proposed finding Related Forum Posts through Content Similarity over Intention based Segmentation.
J. Jeon et al.[2] proposed Finding semantically similar questions based on their answers. T. C. Zhou et al.[3] proposed syntactic structure of
questions posted in such forums in order to match questions. H. Misra et al .[4] proposed The first is topical segmentation where adjacent pairs of text blocks are compared for overall similarity based on terms or topics. S. Robertson et al.[5] proposed a method on elaborating forum posts of multiple segments using content similarity. J. Berant et al.[6] proposed matching technique when the comparison of the text of the segments.

3. METHODOLOGY

The following are the methodology used in this paper involves dataset, text preprocessing ,vector space model and k-mean clustering**.**

- **Data Set:**

The data set used in this are html pages it can be any web pages particularly in this paper twitter conversations taken from web .This consists of bag of words .These words undergo the preprocessing process(which is discussed in the next section).The dataset consists of set of documents .The frequency ,relative frequency are calculated for the words which are produced at the disambiguate process(discussed in the procedure section)

- **Text pre-processing:**

  - ✓ Filtering: Punctuation marks and special characters are removed from the plain text document.
  - ✓ Tokenization: Sentences are split into individual words or tokens.
  - ✓ Stop word removal: The words (e.g. "and", "the" etc.) which do not convey meaning as a dimension in the vector space are removed

  .

  - ✓ Stemming: Words are reduced to their base form. For example, the words "process", "processing", are reduced to the stem "process" using Porter's algorithm.
  - ✓ Pruning: Very low frequency words are removed. A small set of Keywords extracted from the dataset known as Feature vector. These Keywords are required to create vector space model. We have used frequency based method to extract the feature vector.

- **SEGMENTATION**

Segmentation is a key data mining technique.[4]The key to segmentation is to decide how to split the database up. Segment is a group of consumers that react in a similar way to a particular approach. So the key to segmentation is to decide how to split the database up.

- **Vectors Space model:**

Another name of VSM is Term Frequency Inverse D3ocument Frequency model i.e. TF-IDF model. It is the standard retrieval technique used in text mining using the feature vector each document is represented as an n-dimensional vector. The value of each element in the vector reflects the priority of the corresponding feature in the document. The similarity between documents can be measured by calculating the distance between document vectors. If the Documents contain the same keywords they are similar. the term frequency tf(i,j) is calculated(frequency of term i in the document j). The term frequency is normalized with respect to the maximal frequency of all terms occurring in a document.

freq (i,j)
If (/.j) Max (f(x,j): w€j)
.x= Any Term with maximum frequency.

The Document frequency (dfi) of a term is the number of documents in which term I occurs. If D is size of documents (i.e. total number of documents) in a database then, Inverse Document Frequency is given by,
idf (i, j) = log (01 dfi) (3.2)
Weight(Wi) of term is calculated as, Wi = tfi * log (O/dfi) (3.3)
Weight of a term is normalized with respect to the maximum weight of all terms present in a document. Cosine similarity used as similarity measure for vector space retrieval. If two vectors matches completely their similarity should be equal to 1. If two vectors have no keywords in common, the similarity should be equal to O. If some part of a vector matches then the similarity should be between 0 and 1.as:Co sine similarity between two vectors is calculated.

- **K-Mean Method**

If a vector of documents (Dl, D2 . . · Dn) is given ,K-means clustering Algorithm will partition the n documents into K clusters (K :::; n) such that cosine distance between them is minimum.

1. Randomly select initial centroid that divides the documents into k clusters.
2. Compute Cosine distance of each document from the centroid of each of the clusters. Assign that document to the cluster with the closest centroid.

Repeat step 2 until there is no change in newly formed clusters

4. EXISTING SYSTEM

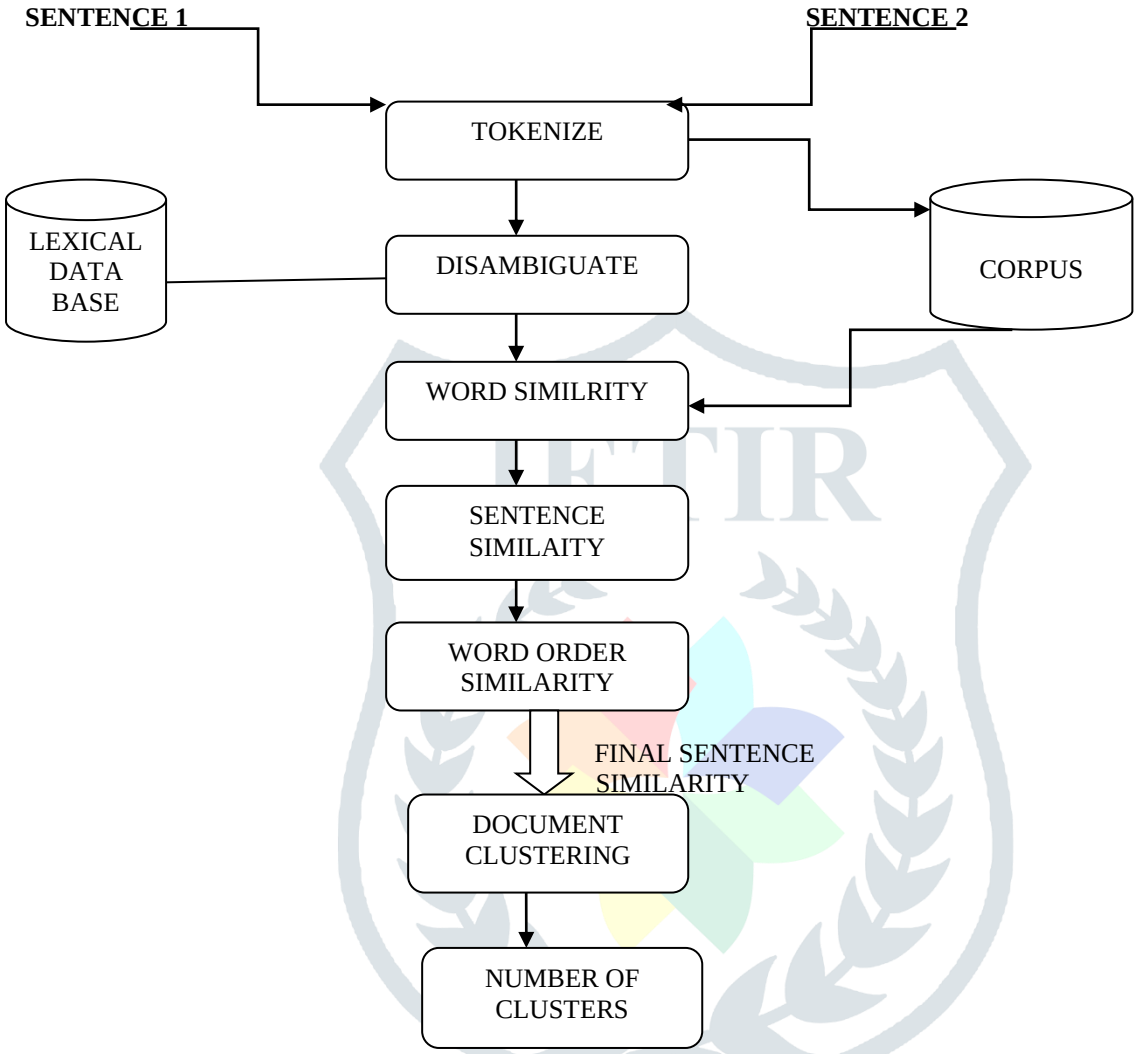
Text document clustering had been implemented in various ways so far. Initially the text documents are pre processed which includes Tokenization, removal of stop words and word stemming. We have many algorithms for performing those operations.[3]Then feature selection is performed to proceed for document clustering. We can go for any of the feature selection methods like tf and idf calculation, tf-df calculation or tf2 calculation. All these methods are used to remain only those words that are frequently repeated in the document. After feature selection is performed, clustering algorithm is applied on the documents to obtain the clusters

5. PROPOSED SYSTEM:

In this paper the semantic similarity is calculated between sentences is a long dealt problem in the area of natural language processing[2].The semantic analysis field has a crucial role to play in the research related to the text analytics. The semantic similarity differs as the domain of operation differs. In this paper, we present a methodology which deals with this issue by incorporating semantic similarity .To calculate the semantic similarity between words and sentence using the document clustering algorithms play an important role in helping users to speedily navigate, sum up and organize the information improved document clustering algorithm is given which generates number of clusters for any text documents and calculate similarity measures to place similar documents in proper clusters

Procedure:

In the procedure the sentences are para phrased by tokenizing them[6] i.e. the bag of words in the twitter data are broken into words. These tokenized words also involves preprocessing are send to Corpus database which is a collection of sentences which grammatically and linguistically correct.The tokens ambiguity is removed in disambiguate process and sent to the lexical database where the lexical analysis is done to the words. Now using NLP(Natural language processing) the words are grouped into articles, auxiliary verbs, common verb, conjunction, interjection , preposition and pronoun.



The word and sentence similarity are found by cosine similarity method. The process followed here follows clustering of the words according to their intention .The documents are clustered using K-mean method.

Algorithm :

Input: Dataset set D = {d 1, d2 . .. dn}
Output: Set of Cluster Numbers C along with document numbers m associated.

1. U= {DI I i € N}
2. Distribute the documents into groups us mg Dividing and Conquer Merge sort strategy.
   Apply Divide Strategy on Input Corpus.
   Apply Divide Strategy till documents are equally placed in groups.
   Go to step 3.
   Conquer the Clusters obtained .
3. Now apply K- means algorithm on every partition iteratively till we get the same clusters.
4. Calculate the similarity of the documents using cosine similarity measure.
   Similarity of the document in step 4 is calculated as,
   for S=Di I i (N Where,D- Documents,N- Number of documents)
   for i=1 to n Cosine Similarity Matrix

$$CS\ ixj = \begin{bmatrix} 1 & D(1,2) & D(1,3) & \ldots\ldots & D(1,n) \\ D(2,1) & 1 & D(2,3) & \ldots\ldots & D(2,n) \\ .. & .. & .. & \ldots\ldots & D(3.n) \\ .. & .. & .. & \ldots\ldots & D(4,n) \end{bmatrix}$$

D- Documents, N- Number of documents ,for i=1 to n Cosine Similarity Matrix

6. RESULTS AND DISCUSSION:

DOCS	D1	D2	D3	D4	D5	D6	D7
D1	1.0000000000000002	0.512138586014856	0.33949613138654217	0.36675964658696875	0.28629263316429343	0.49636118254354733	0.5750571147918879
D2	0.512138586014856	0.749560133945585 3	0.7495601339455853	0.7413427615665005	0.7350384280697501	0.7951397053695745	0.5094959106 89468
D3	0.33949613138654217	0.741342761566 5005	0.8450488747837589	0.8450488747837589	0.9156329667884879	0.7978301283535065	0.5295943733017672
D4	0.28629263316429343	0.735038428069 7501	0.9156329667884879	0.9999999999999999	0.8342359764064262	0.7730071957102241	0.5129811763 26248
D5	0.49636118254354733	0.795139705369 5745	0.7978301283535065	0.8342359764064262	0.7708985955699947	0.7708985955699947	0.5952079322124026
D6	0.5750571147918879	0.795139715369 5745	0.6543836944600566	0.7730071957102241	0.6361727621408504	0.6363950818118621	0.6717138324995514
D7	0.4740127395961923	0.645972932633325	0.5479610079166131	0.6483910078470022	0.5295943733017672	0.7913279354652619	0.6378336317838798

The above table consists of document similarity values of the 7 twitter documents ,these similarity are used for intention segmentation.

7. CONCLUSION

This paper presented an approach to calculate the semantic similarity between two words, sentences or paragraphs. The algorithm initially disambiguates both the sentences and tags them in their parts of speeches. The disambiguation approach ensures the right meaning of the word for comparison. The similarity between words is calculated based on a previously established edge-based approach. The information content from a corpus can be used to influence the similarity in particular domain. Semantic vectors containing similarities between words are formed for sentences and further used for sentence similarity calculation. Word order vectors are also formed to calculate the impact of the syntactic structure of the sentences. Since word order affects less on the overall similarity than that of semantic similarity, word order similarity is weighted to a smaller extent.Specifically ,with comparision the mean precision is increased by 15% in this approach.

8. REFERENCES

[1] Dimitra Papadimitriou ,Georgia Koutrika, Yannis Velegrakis and John Mylopoulos Finding Related Forum Posts through Content Similarity over Intention-based Segmentation .

[2] J. Jeon, W. B. Croft, and J. H. Lee, “Finding semantically similar questions based on their answers,” in Proceedings of the 28th ACM SIGIR Conference, ser. SIGIR ’05. New York, NY, USA: ACM, 2005, pp. 617–618**.**

[3] T. C. Zhou, C.-Y. Lin, I. King, M. R. Lyu, Y.-I. Song, and Y. Cao, “Learning to suggest questions in online forums.”in AAAI, 2011

[4] H. Misra, F. Yvon, J. M. Jose, and O. Cappe, “Text segmentation via topic modeling: an analytical study,” in CIKM, 2009, pp. 1553–1556.

[5] S. Robertson, S.Walker, and M. Hancock-Beaulieu, “Okapi at TREC-7: Automatic adhoc, filtering, VLC and interactive track,” TREC ’98, pp. 199–210, 1998.

[6] J. Berant and P. Liang, “Semantic parsing via paraphrasing.” in ACL (1), 2014, pp. 1415–1425.