An Enhanced Data Mining For Text Clustering

Rupali D. Tajanpure
K. K. Wagh Polytechnic, Chandori.
Tal-Niphad, Dist-Nashik

D. B. Kshirsagar
HOD-Computer Engg. Dept.
SRES COE, Kopargaon

ABSTRACT
Text mining is based on the statistical analysis of a term, either word or phrase. Statistical analysis of a term frequency captures the importance of the term within document only. Usually in text mining techniques the basic measures like term frequency of a term (word or phrase) is computed to compute the importance of the term in the document. But with statistical analysis, the original semantics of the term may not carry the exact meaning of the term. To overcome this problem, a new framework has been introduced which relies on concept based model approach. The proposed model can efficiently find significant matching and related concepts between documents according to concept based approaches.

General Terms
Conceptual term frequency, document frequency, term frequency, concept-based similarity, Concept-based analysis algorithm.

Keywords
Concept-based mining model, sentence-based, document-based, corpus-based, concept analysis

INTRODUCTION

Due to the daily rapid growth of the information, there are considerable needs in extracting and discovering valuable knowledge from the vast amount of information found in different data sources today such as World Wide Web. Data mining in general is the field of extracting useful information, and sometimes high-level knowledge, from large sets of raw data. It has been the attention of many researchers to find efficient ways to extract useful information automatically from such information sources.

Text Mining is the process of deriving high quality information from text by discovering patterns and trends through different written resources. Text mining is generally considered more difficult than traditional data mining. This is attributed to the fact that traditional databases have fixed and known structure, while text documents are unstructured, or, as in the case of web documents, semi-structured. Thus, text mining involves a series of steps for data pre-processing and modeling in order to condition the data for structured data mining. Text mining can help in many tasks that otherwise would require large manual effort. Common problems solved by text mining include, but not limited to, searching through documents, organizing documents, comparing documents, extracting key information, and summarizing documents. Methods in information retrieval, machine learning, information theory, and probability are employed to solve those problems.

Natural language processing (NLP) is a field of computer science, artificial intelligence and linguistics concerned with the interactions between computers and human (natural) languages. NLP is a term that links back into the history of Artificial Intelligence (AI), the general study of cognitive function by computational processes, with an emphasison the role of knowledge representations. The need for representations of human knowledge of the world is required in order to understand human language with computers. Text mining attempts to discover new, previously unknown information by applying techniques from natural language processing and data mining. The problem introduced by text mining is that natural language was developed for humans to communicate with one another and to record information. Computers are a long way from understanding natural language.

Clustering can be considered the most important *unsupervised learning* problem; some as every other problem of this kind, it deals with finding a *structure* in a collection of unlabeled data. In other words, clustering could be "the process of organizing objects into groups whose members are similar in some way". A *cluster* is therefore a collection of objects which are "similar" between them and are "dissimilar" to the objects belonging to other clusters.

Each sentence in a document is labeled by a semantic role labeler. This labeler determines the terms that contribute to the semantics of the sentence. Any term that has a semantic role in a sentence is called a concept. These concepts can be words or phrases and are dependent on the semantics of the sentence. Each time a new document is introduced; the designed system scans the new document and extracts matching concepts between the document and all the previously processed documents.

Thematic Roles Background

The study of roles associated with verbs is referred to as a thematic role or case role analysis. Fillmore [2] first suggested that thematic roles are categories which help characterize the verb arguments by providing a shallow semantic language.

Generally, the semantic structure of a sentence can be characterized by a form of verb argument structure. The verb argument structure permits a link between the arguments in the surface structures of the input text and their associated semantic roles.

The important terms used in this paper are given below:

• **Verb Argument structure**: (e.g.: Adam plays the guitar). "plays" is the verb. "Adam" and "the guitar" are the arguments of the verb "plays".
• **Label**: An argument is assigned a label (e.g.: Adam plays the guitar). The first argument "Adam" is preverbal and plays the role of subject and the second argument "the guitar" is post verbal and plays the role of object.
• **Term:** It is either an argument or a verb. It can also be a word or a phrase

• **Concept:** The concept is a labeled term.

In recent times, thematic roles in sentences have been tried to be labeled automatically. The first was proposed by Gildea and Jurafsky [3]. They applied a statistical learning technique to the FrameNet Database. The model presented a discriminative method for determining the probable role of the constituent given the predicator, frame and certain other features.

Concept Based Mining Model

A concept-based mining model analyzes terms on the sentence, document, and corpus levels. This mining model can effectively discriminate between non important terms with respect to sentence semantics and terms which hold the concepts that represent the sentence meaning. The mining model consists of sentence-based concept analysis, document-based concept analysis, corpus-based concept-analysis, and concept-based similarity measure. The term which contributes to the sentence semantics is analyzed on the sentence, document, and corpus levels rather than the traditional analysis of the document only. This model can efficiently find significant matching concepts between documents, according to the semantics of their sentences. The similarity between documents is calculated based on a new concept-based similarity measure.

The model captures the semantic structure of each term within a sentence and document rather than the frequency of the term within a document only. Each sentence is labeled by a semantic role labeler that determines the terms which contribute to the sentence semantics associated with their semantic roles in a sentence. Each term that has a semantic role in the sentence, is called a concept. Concepts can be either words or phrases and are totally dependent on the semantic structure of the sentence. When a new document is introduced to the system, the proposed mining model can detect a concept match from this document to all the previously processed documents in the data set by scanning the new document and extracting the matching concepts.

The concept-based mining model consists of sentence-based concept analysis, document-based concept analysis, corpus-based concept-analysis, and concept-based similarity measure, as depicted in following figure 1

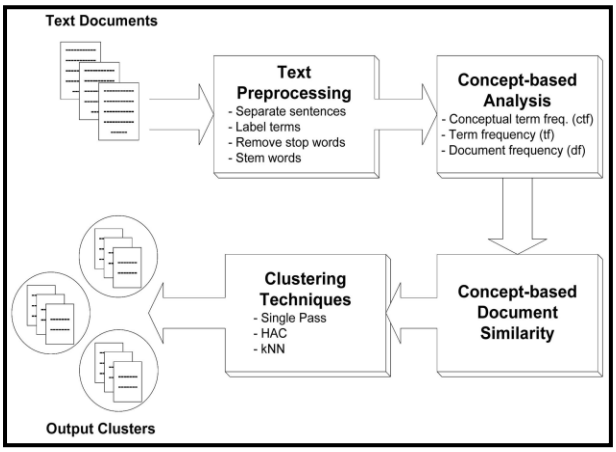

Figure 1. The Concept-based mining model

A raw text document is the input to the mining model. Each document has well-defined sentence boundaries. Each sentence in the document is labeled automatically based on the PropBank notations. After running the semantic role labeler, each sentence in the document might have one or more labeled verb argument structures. The number of generated labeled verb argument structures is entirely dependent on the amount of information in the sentence.

The sentence that has many labeled verb argument structures includes many verbs associated with their arguments.

The labeled verb argument structures, the output of the role labeling task, are captured and analyzed by the concept-based mining model on sentence, document, and corpus levels. In this model, both the verb and the argument are considered as terms. One term can be an argument to more than one verb in the same sentence. This means that this term can have more than one semantic role in the same sentence.

In such cases, this term plays important semantic roles that contribute to the meaning of the sentence. In the concept-based mining model, a labeled terms either word or phrase is considered as concept. The objective behind the concept-based analysis task is to achieve an accurate analysis of concepts on the sentence, document, and corpus levels rather than a single-term analysis on the document only.

The concepts can be identified by using natural language processing on the text document. That is by giving the structure to each sentence. This structure is called verb argument structure. See the example for verb argument structure of a sentence.

Example:
Sentence: He *hits* a ball.
Verb: hits
Arg0: He
Arg1: a ball
These labels are according to the Prop bank notations [1]. A single word may have different senses. Using this semantic role, we can get the content in which the word is being used in that sentence.

Mining Model Step Wise Process

- Preprocessing of Text
- Identify the Concepts
- Calculating Conceptual Term Frequency at
  - Sentence Level
  - Document Level
  - Corpus Level
- Cluster the Documents

Preprocessing of Text:
In this module each document is read from the corpus. In each document, the sentences are separated. As the raw text data is unstructured data, we have to give a proper structure to each sentence. So each sentence is given a verb argument structure.

These arguments are labeled as ARG0, ARG1, ARG2 etc. basing on the number of verbs for which the term is argument. Another important technique in text mining is reducing the dimensionality of the text. That is we have to remove some unnecessary words. This can be done using standard stop lists. Each word is checked against the standard stop word list. If it is a stop word, then it is treated as insignificant word and it is removed from the process.

Identify the Concepts:
After completion of first step, we are remained with the labeled terms which are significant for matching that is to find the similarity. So each labeled term is treated as a concept. There are three types of concept analysis which are as follows:

1>Sentence based Concept analysis

2>Document based Concept analysis
3>Corpus based Concept analysis

1>Sentence based Concept analysis: ***Atsentence level (Conceptual term frequency ctf)*** **-** To analyze each concept at the sentence level, a new concept-based frequency measure, called the conceptual term frequency *(ctf)* is proposed.

2>Document based Concept analysis: ***At document level (term frequency tf)***- To analyze each concept at the document level, the concept- based term frequency *tf*, the number of occurrences of a concept (word or phrase) *c* in the originaldocument is calculated. The *tf* is a local measure on the document level.

3**>Corpus based Concept analysis:-*** ***At corpus level (document frequency df)***-To extract concepts that can discriminate between documents, the concept-based document frequency *df*, the number of documents containing concept *c*, is calculated. The *df* is a global measure on the corpus level.

Calculating Conceptual Term Frequency:-

a) Calculating *ctf* of Concept c in Sentence s: The *ctf* is the number of occurrences of concept *c* in verb argument structures of sentence *s*. The concept *c*, which frequently appears in different verb argument structures of the same sentence *s*, has the principal role of contributing to the meaning of *s*. In this case, the *ctf*is a local measure on the sentence level.

b) Calculating *ctf* of Concept *c* in Document *d*: A concept *c* can have many *ctf* values in different sentences in the same document *d*. Thus, the *ctf* value of concept *c* in document d is calculated by:

$$ctf = \sum_{n=1}^{sn} ctfn \qquad (1)$$

Where *sn* is the total number of sentences that contain concept *c* in document *d*. Taking the average of the *ctf* values of concept *c* in its sentences of document *d* measures the overall importance of concept *c* to the meaning of its sentences in document *d*. A concept, which has *ctf*values in most of the sentences in a document, has a major contribution to the meaning of its sentences that leads to discover the topic of the document. Thus, calculating the average of the *ctf* values measures the overall importance of each concept to the semantics of a document through the sentences. To illustrate the calculation of *ctf* in a document, consider a concept *c* which appears twice in document *d* in the first and the second sentences. The concept *c* appears five times in the verb argument structures of the first sentence *s1*, and three times in the verb argument structures of the second sentence *s2*. In this case, the *ctf*value of concept *c* is equal to (5+3)/2 = 4

Cluster Documents: Clustering is the task of assigning a set of objects into groups (called **clusters**) so that the objects in the same cluster are more similar (in some sense or another) to each other than to those in other clusters. Clustering is a main task of explorative data mining, and a common technique for statisticaldata analysis used in many fields, including machine learning, pattern recognition, image analysis, information retrieval, and bioinformatics. To cluster the documents we need a clustering algorithm. There are many clustering techniques/ algorithms [8] exist which are as follows:

Clustering techniques / Algorithms:-

1. Hierarchical Agglomerative Clustering (HAC)
2. k-Nearest Neighbor Clustering.

These clustering techniques were chosen because they require pair-wise document similarity information only. Other classes of clustering techniques that rely on having information about the original document vectors (such as k-means) were not considered for comparison since such comparison would not be accurate due to differences in the input to each technique. Each clustering algorithm accepts as its input a document similarity matrix without having to rely on the original feature vectors.

1 Hierarchical Agglomerative Clustering (HAC) isnon-incremental clusteringmethods that mainly rely on having the whole document set ready before applying the algorithm. This is typical in offline processing scenarios.It is a straightforward greedy algorithm that produces a hierarchical grouping of the data. It starts with all instances each in its own cluster, and then repeatedly merges the two clusters that are most similar at each iteration. Complexity of HAC is O (n^2), which could get infeasible for very large document sets.

2 K-Nearest Neighbor Clustering is incremental clustering algorithm. For each new document, the algorithm computes its similarity to every other document, and chooses the top k documents. The new document is assigned to the cluster where the majority of the top k documents are assigned.

Concept-Based Analysis Algorithm

1. d_{doci} is a new Document
2. *L* is an empty List (L is a matched concept list)
3. s_{doci} is a new sentence in d_{doci}
4. Build concepts list C_{doci} from s_{doci}
5. for each concept $ci \in C_{doci}$ do
6. computectf_i of *ci* in d_{doci}
7. computetf_i of *ci* in d_{doci}
8. computedf_i of *ci* in d_{doci}
9.d_k is seen document, where k ={0, 1, . . . ; doci-1}
10. s_k is a sentence in d_k
11. Build concepts list C_k from s_k
12. for each concept $c_j \in C_k$ do
13. if ($c_i == c_j$) then
14. updatedf_i of c_i
15. compute$ctfweight = avg(ctf_{i,}ctf_j)$
16. add new concept matches to *L*
17. end if
18. end for
19. end for
20. output the matched concepts list *L*

The procedure begins with processing a new document (at line 1) which has well defined sentence boundaries. Each sentence is semantically labeled according to Prop Bank notations. The lengths of the matched concepts and their verb argument structures are stored for the concept-based similarity calculations.

Each concept (in the for loop, at line 5) in the verb argument structures, which represents the semantic structures of the sentence, is processed sequentially. Each concept in the current document is matched with the other concepts in the previously processed documents. To match the concepts in previous documents is accomplished by keeping a concept list L, which holds the entry for each of the previous documents that shares a concept with the current document. After the document is processed, L contains all the matching concepts between the current document and any previous document that shares at least one concept with the new document. Finally, L is output as the list of documents with the matching concepts and the necessary information about them. The

concept-based analysis algorithm is capable of matching each concept in a new document *(d) with* all the previously processed documents in *O(m)*time, where m is the number of concepts in d.

Concept based similarity measure

This similarity measure is a function of the following factors:
1. the number of matching concepts, *m*, in the verb argument structures in each document *d*,
2. the total number of sentences, *sn*, that contain matching concept c_i in each document *d*,
3. the total number of the labeled verb argument structures, *v*, in each sentence *s*,
4. thectf_i of each concept c_i in *s* for each document *d*, where i=1,2,….*m*,
5. thetf_i of each concept c_i in each document *d*, where i=1,2,….*m*
6. thedf_i of each concept c_i, where i=1,2,….*m*
7. the length, *l*, of each concept in the verb argument structure in each document *d*,
8. the length, *Lv*, of each verb argument structure which contains a matched concept, and
9. the total number of documents, *N*, in the corpus.

$$Sim_c(d1,d2)=\sum_{i=1}^{m} max\left(\frac{li1}{Lvi1},\frac{li2}{Lvi2}\right) X weight_{i1} X weight_{i2} \quad (2)$$

$$weight_i = (tfweight_i + ctfweight_i) * log\left(\frac{N}{dfi}\right) \quad (3)$$

The concept-based weight of concept *i* in document *d* is calculated by (3). In (3), the $tfweight_i$ value presents theweight of concept *i* in document *d* at the document level.

In (3), the $ctfweight_i$value presents the weight of the concept *i* in document *d* at the sentence level based on the contribution of concept *i* to the semantics of the sentences in *d*.

In (3), the *log* (N/df_i) value rewards the weight of the concept *i* on the corpus level, when concept *i* appears in a small number of documents.

The sum between the two values of $tfweight_i$ and$ctfweight_i$in (3) presents an accurate measure of the contribution of each concept to the meaning of the sentences

The multiplication between *log* (N/df_i) value and ($tfweight_i$ + $ctfweight_i$) value in (3) finds the concepts that can efficiently discriminate among documents of the entire corpus.

Equation (2) assigns a higher score, as the matching concept length approaches the length of its verb argument structure, because this concept tends to hold more conceptual information related to the meaning of its sentence.

In (4), the tf_{ij} value is normalized by the length of the document vector of the term frequency tf_{ij} in document d, where j=1,2 . . . , *cn*, and

$$tfweight_i = \frac{tfij}{\sqrt{\sum_{j=1}^{cn}(tfij)^2}} \quad (4)$$

cn is the total number of the concepts which has a term frequency value in document d.

In (5), the ctf_{ij} value is normalized by the length of the document vector of the conceptual term frequency ctf_{ij}in document *d*, where j=1, 2 . . . ,*cn*

$$ctfweight_i = \frac{ctfij}{\sqrt{\sum_{j=1}^{cn}(ctfij)^2}} \quad (5)$$

where *cn* is the total number of concepts which has a conceptual term frequency value in document *d*.

Mathematical Framework

The formulation of the concept-based mining model is explained as follows:

- A concept *c* is a string of words, *c* =‘‘w_{i1}, w_{i2}, . . .w_{in} ’’where *n* is the total number of words in concept *c*.
- A sentence *s* is a string of concepts, *s* = ‘‘c_{i1},c_{i2} . . . c_{im}’’ where *m* is the total number of concepts generated from the verb argument structures in sentence *s*, Thus, $\boldsymbol{c_i \epsilon s}$ if c_i is a substring of *s*.
- A document *d* is a string of words, *d* = ‘‘w_{i1}, w_{i2}, . . .w_{it} ’’ where *t* is the total number of words in document *d*.
- The function *freq*= (str_{sub},str_{total}) is the number of times that substring str_{sub} appears in string str_{total}.
- The concept-based term frequency of document *d* is *tf = freq(c_i ,d)*
- The conceptual term frequency of sentence *S* is *ctf_s=freq(c_i ,s)*
- The conceptual term frequency *ctf* of document *d* is calculated by (1).
- The concept-based weighting of a concept is $weight_i$ = ($tfweight_i$ + $ctfweight_i$) * *log* (N/df_i)
- The concept-based similarity between documents d1 and d2 using concepts is

$Sim_c(d1,d2)= \sum max(l_{i1}/Lv_{i1}, l_{i2}/Lv_{i2})X\ weight_{i1}\ X weight_{i2}$ as calculated in (2).

CONCLUSIONS

This work bridges the gap between natural language processing and text mining disciplines. A new concept based mining model composed of four components, is proposed to improve the text clustering quality. By exploiting the semantic structure of the sentences in documents, a better text clustering result is achieved. The first component is the sentence-based concept analysis which analyzes the semantic structure of each sentence to capture the sentence concepts using the proposed conceptual term frequency *ctf* measure. Then, the second component, document-based concept analysis, analyzes each concept at the document level using the concept-based term frequency *tf*. The third component analyzes concepts on the corpus level using the document frequency *df* global measure. The fourth component is the concept-based similarity measure which allows measuring the importance of each concept with respect to the semantics of the sentence, the topic of the document, and the discrimination among documents in a corpus. By combining the factors affecting the weights of concepts on the sentence, document, and corpus levels, a concept-based similarity measure that is capable of the accurate calculation of pair wise documents is devised. This allows performing concept matching and concept-based similarity calculations among documents in a very robust and accurate way. The quality of text clustering achieved by this model significantly surpasses the traditional single term- based approaches.

There are a number of possibilities for extending this paper. One direction is to link this work to Web document Clustering. Another direction is to apply the same model to text classification. The intention is to investigate the usage of

such model on other corpora and its effect on classification, compared to that of traditional methods.

REFERENCES

[1] P. Kingsbury and M. Palmer **"*Propbank: the next level of treebank*"**. In Proceedings of Treebanks and Lexical Theories, 2003.

[2] C. Fillmore, **"*The Case for Case Universals in Linguistic Theory"***, Holt, Rinehart and Winston, 1968.

[3] D. Gildea and D. Jurafsky, **"*Automatic Labeling of Semantic Roles*"**, Computational Linguistics, vol. 28, no. 3, pp. 245-288, 2002.

[4] S. Pradhan, W. Ward, K. Hacioglu, J. Martin, and D. Jurafsky**, "*Shallow Semantic Parsing Using Support Vector Machines*"**, Proc. Human Language Technology/North Am. Assoc. for ComputationalLinguistics (HLT/NAACL), 2004.

[5] S. Pradhan, K. Hacioglu, W. Ward, J.H. Martin, and D. Jurafsky, **"*Semantic Role Parsing: Adding Semantic Structure to Unstructured Text*"**, Proc. Third IEEE Int'l Conf. Data Mining(ICDM), pp. 629-632, 2003.

[6] S. Pradhan, K. Hacioglu, V. Krugler, W. Ward, J.H. Martin,and D. Jurafsky, **"*Support Vector Learning for Semantic Argument Classification*"**, Machine Learning, vol. 60, nos. 1-3,pp. 11-39, 2005.

[7] S. Shehata, F. Karray, and M. Kamel, **"*Enhancing Text Clustering Using Concept-Based Mining Model*"**, Proc. Sixth IEEE Int'l Conf.Data Mining (ICDM), 2006.

[8] A.K. Jain and R.C. Dubes, **"*Algorithms for Clustering Data*"**, PrenticeHall, 1988.

[9] M. Steinbach, G. Karypis, and V. Kumar, **"*A Comparison of Document Clustering Techniques*"**,Proc. Knowledge Discovery and Data Mining (KDD) Workshop Text Mining, Aug. 2000.