Concept Driven Approach Using Semantic Lingo for Web Document Clustering

Mrs. Poonam Milind Thakre.

Department of Computer Engineering, Universal College of Engineering, Vasai.

---***---

Abstract -.Web document clustering is an efficient method to make the search results easier to scan. It aims to cluster search results into meaningful groups and provide a navigator to easily access the satisfying results for users. Traditional clustering algorithms do not consider relationships among the words so that can not accurately represent the meaning of documents. To overcome this problem semantic information from ontology such as WordNet has been used to improve the Quality of Web search clustering. Our key goal is to improve our system by overcome various problems such as Synonym and polysemy, high dimensionality and assigning appropriate description for generated cluster.

***Key Words*:** Clustering, wordNet, Semantic, ontology, Concept.

1. INTRODUCTION

We present a novel algorithm 'Semantic Lingo' for clustering Web search result based on largest lexical databases. First, a document corpus is pre-processed into term frequency files, in which each document is represented as a list of its term frequencies. In addition, common phrases are extracted using a suffix array algorithm Second the inverted document frequency of each term is calculated and each term weight is computed by multiplying the term frequency and inverted document frequency.[2] Inverted term-document files are generated for each term and the term-document matrix is constructed based on term weights using VSM (vector space model). Third, with the extracted common phrases, we conduct key concept induction using LSA techniques. Fourth, with the list of key concepts, we utilize WordNet to inspect their synonyms and hyponyms. The documents are allocated based on each key concept and its synonyms and hyponyms Using WordNet, hyponyms of each concept are detected and used to construct a corpus-related ontology.[3] Last documents are linked to the ontology through the key concepts.

2.1 Web Document Clustering Applications.

Web Document clustering is unsupervised learning and is applied in many fields of business and science. Initially, document clustering was studied for improving the precision or recall in information retrieval systems. Document clustering has also been used to automatically generate hierarchical clusters of documents. Following are few applications of document clustering.

1. Finding Similar Documents: To find similar documents matching with the search result document. Clustering is able to discover documents that are conceptually alike compared to search-based approaches which discover documents sharing many of the same words.
2. Organizing Large Document Collections: To organize large number of uncategorized documents in taxonomy identical to the one human would create for easy retrieval.
3. Search Optimization: Clustering helps a lot in improving the quality and efficiency of search engines as the user query can be first compared to the clusters instead of comparing it directly to the documents. Clustering is used in organizing the results returned by a search engine in response to a user's query. Following this principle of cluster-based browsing by automatically organizing search results into meaningful categories are Teoma, vivisimo clustering engine, MetaCrawler, WebCrawler.

2.2 Architecture of Web Document Clustering

First, a document corpus is preprocessed into term frequency files, in which each weight is computed by multiplying the term frequency and inverted document frequency. Inverted term-document files are generated for each term and the term-document matrix is constructed based on term weights. Third, with the extracted common phrases, we conduct key concept induction using LSA techniques. [6] Fourth, with the list of key concepts, we utilize WordNet to inspect their synonyms and hyponyms. The documents are allocated based on each key concept and its synonyms and hyponyms. Fifth, using WordNet, [2] hyponyms of each concept are detected and used to construct a corpus-related ontology. Sixth, documents are linked to the ontology through the key concepts.

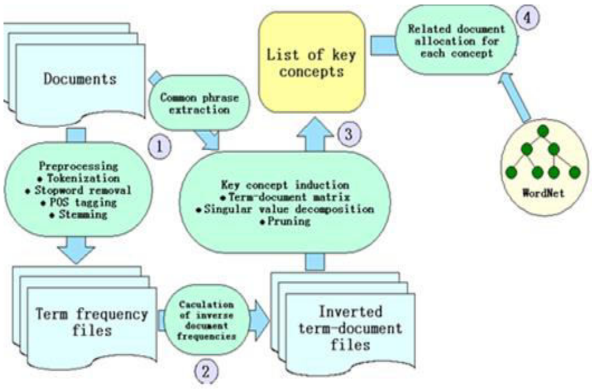


Figure 1 Architecture of Web Document Clustering

1. Load Documents.

In this module Documents are loaded from database which is the initial stage

2. Preprocessing

In this module, Input documents are preprocessed to remove stop words and stemming is applied.It consist stop word removal and stemming .

2.1Stop words Elimination

Stop words are words which are filtered out prior to, or after, processing of natural language data. A stop word is a commonly used word in our daily life, that a search engine has been programmed to ignore, both when searching and when retrieving them as a result of a search query. The major work is to identify the mostly weighted words are called as keywords for the documents that reduce the dimensions of the matrix. Stop word elimination is done based on ASCII values of each letter without considering the case (either lower case or upper case) and sum the each letter corresponding ASCII value for every word and generate the number. Assign number to corresponding word, and keep them in sorted order.

2.2 Stemming

Stemming is the process for reducing modified words to their stem; base or root forms generally a written word form. Stem is a part of a word like ing, er, etc., The term is used with slightly different meanings. An algorithm for removing derivations endings and inflectional in order to reduce word forms to a common stem. In this stemming algorithm the suffixes and prefixes were eliminated according to the conditions by which the stemming procedure was applied.[4]

A stemmer for English, for example, should identify the string "cats" (and possibly "catlike", "catty" etc.) as based on the root "cat", and "stemmer", "stemming", "stemmed" as based on "stem". A stemming algorithm reduces the words "fishing", "fished", and "fisher" to the root word, "fish". On the other hand, "argue", "argued", "argues", "arguing", and "argus" reduce to the stem "argu" (illustrating the case where the stem is not itself a word or root) but "argument" and "arguments" reduce to the stem "argument".

3. Phrases Extraction

To extract the common phrases from the document corpus, we use the suffix array algorithm. We define some rules. First, the phrases appear in the document corpus at least a specified number of times. Second, the phrases do not cross sentence boundaries. Third, if a phrase is contained in a longer phrase, the latter is regarded as more complete than the former. Fourth, if a phrase begins with a stop word or ends with a stopword, it is regarded as meaningless and discarded. Note that if a phrase contains a stopword in the middle of it, we do not discard it because a meaningful phrase can contain a stopword.

4. Feature Selection

The inverted document frequency of each term is calculated and each term weight is computed by multiplying the term frequency and inverted document frequency. Inverted term-document files are generated for each term and the term-document matrix is constructed based on term weights. To construct the term-document matrix, the tfidf (term frequency-inverted document frequency) is applied to calculate the weights of terms. In the vector space model document d is represented as a feature vector d = (tft1, ..., tfti), where tft returns the absolute frequency of term t ∈ T in document d ∈ D, where D is the document corpus and T = {t1, t2, ..., ti} is the set of all different terms occurring in D. Inverted document frequency of each term is calculated as follows [2]

$$Aij = tfij.\log(\frac{N}{dif})$$

Where tfij is the term frequency, dfi denotes the number of documents in which term i appears, and N represents the total number of documents in the collection. Each term weight is computed by multiplying the term frequency and inverted document frequency. To calculate the weight wt of term t: as follows:

$Wt = tft \times (\log_2 n - \log_2 dft + 1)$ (1)

Where dft is the document frequency of term t that how many documents in which term t appears. With weight wt of term t, the inverted file of term t is constructed. Inverted term-document files are generated for each term and the term-document matrix is constructed based on term weights. [2]

we use the vector space model (VSM) and singular value decomposition (SVD), the latter being the fundamental mathematical construct underlying the latent semantic analysis (LSA) technique.LSA aims to represent the input collection using abstract terms found in the documents rather than the literal terms appearing in them, by approximating the original term-document matrix using a limited number of orthogonal factors.

5 Clusters Lables using Word Net

with the extracted common phrases, we conduct key concept induction using LSA techniques. with the list of key concepts, we utilize WordNet to inspect their synonyms and hyponyms. The documents are allocated based on each key concept and its synonyms and hyponyms.

6. Document Allocation.

Documents are allocated to each key concept based on the cosine similarity between each document and the set including the key concept and its synonyms and hyponyms. For each concept, if the cosine similarity between a document and the concept exceeds a predefined threshold, the document is allocated to the corresponding group represented by the concept.[5]

EXPERIMENTAL RESULTS

This section presents the results of the methodologies to assign documents to clusters which are more relevant to each other.
Figure 2 shows Precision, Recall and Purity of cluster using FI measures. It shows cluster labels generated by Semantic Lingo are more relevant and efficient then Lingo.

1. Precision

First, let us assume that from a set D of documents, in response to the user's query the set A of documents was returned. Further, let R denote the set of all documents in D that are relevant to the query. Finally, let RA be the intersection of R and A.

Definition — Precision is the fraction of the retrieved documents which is relevant.

Precision = |RA|/|A|.

2. Recall

Definition — Recall is the fraction of the relevant documents which has been retrieved.

Recall = |RA|/|R|.

Ideally, both of the measures would be equal to 1.0. In practice, however, a trade-off between

Them must be sought – in most situations an algorithm will attain higher precision at the

Cost of worse recall and vice versa.

3. Cluster label quality

g – The total number of created groups

u – The total number of groups judged as useful

Cluster label quality is the fraction of all created groups that the user judged as

Useful: q = u/g

The values of q can range from 0.0 (no useful groups created) to 1.0 (all discovered Groups are useful).

4. Purity of cluster

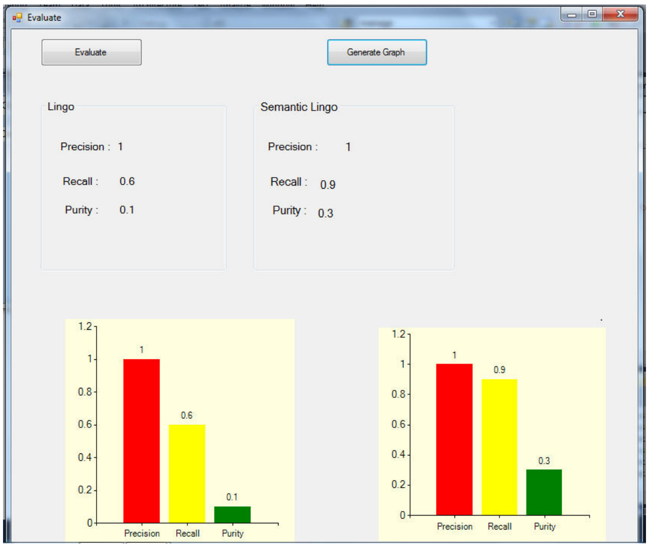
Purity, which is a function of the relative size of the largest class in the resulting clusters. This is the number of documents of the largest class in a cluster divided by the cluster size. The overall purity of the clustering solution is obtained by taking a weighted sum of the individual cluster purities

Purity of a cluster = the number of occurrences of the most frequent class / the size of the cluster (this should be high)

Table 1shows, evaluation criteria for both Lingo and Semantic Lingo in terms of Pricision , Recall , cluster label quality and Purity.

Table 1 Evaluation Criteria for Lingo and Semantic Lingo

Algorithm	Precision	Recall	Cluster label quality	Purity of cluster
Lingo	1.00	0.6	1.00	0.1
Semantic Lingo	1.00	0.9	1.00	0.4



Lingo

Precision : 1

Recall : 0.6

Purity : 0.1

Semantic Lingo

Precision : 1

Recall : 0.9

Purity : 0.3

Figure2. shows Final graph generated for Lingo and Semantic Lingo.

3. CONCLUSIONS

We will try to solve problems in text clustering such as polysemous and synonyms words, high dimensionality and properly assign documents to clusters. It gives implicit and explicit relationship between documents. We proposed a novel method, Semantic Lingo which identifies key concepts and assign documents to the clusters. Based on the vector space model, LSA techniques are used to identify the meaningful key concepts. . The higher number of matrix transformation leads to demanding memory requirements. So we design semantic Lingo for specific application like web search result clustering.

ACKNOWLEDGEMENT

Motivation and guidance are the keys towards success. I would like to extend my thanks to all the sources of motivation.

REFERENCES

1. Carrot2. http://project.carrot2.org/.

2. B. R. Prakash and M. Hanumanthappa, "Web Snippet Clustering and Labeling using Lingo Algorithm", International Journal of Advanced Research in Computer Science, vol. 3, no. 2, pp. 262-265, 2012

3. S. Bechhofer, M. Hauswirth, J. Hoffmann, and M. Koubarakis, An information visualization tool for personalized exploratory document collection analysis editors, *ESWC*, volume 5021 of *Lecture Notes in Computer Science*, pages 139–15

4. Maitri P. Naik, Harshadkumar B. Prajapati, Vipul K. Dabhi,"A survey on semantic Approach", *Electrical Computer and Communication Technologies (ICECCT) 2015 IEEE International Conference on*, pp. 1-10, 2015.3.Springer, 2008

5. Manoj Kumar Sarma, 2019 IEEE International Conference on Intelligent Techniques in Control, Optimization and Signal Processing (INCOS)

6. V.M.Navaneethakumar, r.C.Chandrasekar (2012), “A Consistent Web Documents Based Text Clustering Using Concept Based Mining Model", International Journal of Computer Science Issues, Vol. 9