

Topic Oriented Semi-supervised Document Clustering

Jiangtao Qiu Changjie Tang
School of Computer, Sichuan University
Chengdu, 610065, P.R. China
{qiujiangtao, tangchangjie}@cs.scu.edu.cn

ABSTRACT

In our study on developing a text mining prototype system, it is needed to group documents according to author's need. However, Traditional documents clustering are usually considered an unsupervised learning. It cannot effectively group documents under user's need. To solve this problem, we propose a new documents clustering approach. The main contributions include: (1) Describes user's need by using multiple-attributes topic; (2) Proposes a topic-semantic annotation algorithm; (3) Proposes an optimizing hierarchical clustering algorithm to find out the best clustering solution on clustering tree by using criterion function. Experiments have validated feasibility and effectiveness of the new approach.

Categories and Subject Descriptors:

H.2.8 [Database Application]: Data Mining

General Terms: Algorithm, Design

Keywords

Document clustering, Ontology, Semantic Annotation.

1. INTRODUCTION

Our research focuses on developing a text mining prototype system whose tasks includes mining association events and generating hypotheses on documents collection. When implementing the prototype system, we face a problem how group the documents by user's need from the collected documents.

Document clustering is a critical component of research in text mining. Traditional document clustering includes following steps: (a) extracting feature vector of document; (b) clustering document by parameters such as similarity threshold, the number of clusters. Traditional clustering, however, is the unsupervised learning. It cannot effectively group documents under need of user.

Example 1: Collect documents about people named 'Yao Ming' from Internet. The corpus involved several peoples named 'Yao Ming'. Now it is needed to group documents according to same 'Yao Ming' being in same cluster. Traditional clustering

Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided that copies are not made or distributed for profit or commercial advantage and that copies bear this notice and the full citation on the first page. To copy otherwise, or republish, to post on servers or to redistribute to lists, requires prior specific permission and/or a fee.

Proceedings of SIGMOD2007 Ph.D. Workshop on Innovative Database Research 2007(IDAR2007), June 10, 2007, Beijing, China.

approaches only group documents by similarity of feature vector of documents. It is often difficult to meet user's need by using traditional approaches.

In recent years, Researchers have introduced prior knowledge to document clustering process, which is called semi-supervised document clustering. Topic-driven document clustering approach is proposed in [8]. It first defines multiple topics before clustering. Each topic can be represented as a vector, and then documents in the collection will be clustered by their similarity with topics. Some semi-supervised clustering approaches[4,6] assign class label to parts of documents, *i.e.* some constraints are put, and then clustering process will consult these constraints.

The above semi-supervised document clustering approaches can do nothing on grouping documents according to user's need.

Zhao[9] proposed a method to group documents by topic. It represent a topic by using a series of topic elements, which include *who, where, when, object* etc. Based on topic elements, an index may be built for each document. By their indices, documents with same topic will be grouped to same cluster.

Obviously, topic in [9] is not generated from user's need. Zhao's method cannot solve our problem.

To solve the problem, we propose a new documents clustering approach to group documents according to user's need, which is called topic oriented documents clustering. The main steps include: (1) design a multiple-attributes topic structure to represent user's need. (2) make topic-semantic annotation for each document, and then compute topic-semantic similarity between documents. (3) build dissimilarity matrix. (4) group documents based on optimizing hierarchical clustering algorithm.

Another motivation for topic-semantic annotation in this paper is to reduce dimensionality of feature vector. Dimensionality reduction of feature vectors is a hotspot of research in text mining. The dimensionality of document vectors may reach thousands even tens thousands. It results in huge time cost on documents clustering. Because only words that are able to being mapped to attributes of topic will be extracted from a document, our approach also may reduce dimensionality effectively.

The remaining of this paper is organized as follows. Section 2 gives a brief introduction to related works. Section 3 presents our works on topic-semantic annotation and computing topic-semantic similarity of documents. Section 4 proposes an optimizing document hierarchical clustering algorithm. Section 5 gives a thorough study on clustering performance in comparison with unsupervised clustering. Section 6 summarizes our study.

2. RELATED WORKS

Document clustering is a traditional subject in data mining and machine learning. In recent studies, many new technologies are introduced into documents clustering. In [5], a hierarchical document clustering approach based on frequent item-sets is proposed. In [7], author use particle swarm algorithm to generate fast and high-quality clustering.

Some researchers use ontology to improve document clustering performance. Algorithm COSA (Concept Selection and Aggregation) [3] first extracts feature vector of documents, and then maps words in feature vector to a concept in concept tree of ontology. The support of each concept is counted. Those concepts with large support will be split into sub concepts and concepts with small support count will be aggregated to parent concept. This method dramatically reduces vectors dimensionality. WordNet contains semantic relationships in synset. In [2], author exploits WordNet’s hypernym-hyponym relationship to obtain fewer but more general concepts and thus further improve SOM’s documents clustering ability.

3. TOPIC-SEMANTIC ANNOTATION

In order to implement topic oriented documents clustering, we need address three issues: (1) How to represent user’s need? (2) How to represent relationship between the need and documents? (3) How to evaluate similarity of documents by the need?

In this study, we propose a multiple-attributes topic structure to represent the user’s need, and then represent relationship between topic and documents by annotating topic-semantic for documents. Topic-semantic similarity of documents is just similarity of documents by the need.

3.1 Building Core Ontology

Ontology describes relationships between words and concepts. We exploit ontology to generate topic-semantic annotation for documents. In this study, the definition of core ontology in [3] is quoted.

Definition 1 (Core Ontology): A core ontology is a sign system $O := (L, F, C^*, H, Root)$, which consist of

- A lexicon: The lexicon L contains a set of natural language terms.
- A set of concepts C^* .
- The reference function F with $F: 2^L \rightarrow 2^{C^*}$. F links sets of terms $\{L_i\} \subset L$ to the set of concepts they refer to. In general, one term may refer to several concepts and one concept may be referred to by several terms. The inverse of F is F^{-1} .
- A heterarchy H : Concepts are taxonomically related by the directed, acyclic, transitive, reflexive relation H ($H \subset C^* \times C^*$), it is also called as concept tree.
- A top concept $Root \in C^*$. For all $C \in C^*$ it holds: $H(C, Root)$.

In this study, HowNet[10] is exploited to build core ontology. HowNet is a Chinese-English lexical Knowledgebase implemented by professor Dong Zhen-Dong. It describes relationship among concepts and attributes. Sixteen relationships are defined in HowNet such as hyponym, synonym, antonym. HowNet include multiple files. We use modified lexicon file in HowNet as dictionary that is used in splitting word in Chinese

documents, and then use modified sememe file as heterarchy of core ontology.

Example 2: Exploring a word in HowNet.

Cancer N disease

Description of word ‘cancer’ is that it is a noun and its concepts is *disease*

Observation 1: Many words have a preference for a specific semantic category. In other words, many words have background. For instance, background of word *football match* is sport; background of *security holder* is finance. Background of words may well reflect content of document.

According to Observation 1, we add background description for words in HowNet. Then add a sub-tree of background concepts into concept tree of core ontology.

Example 3: adding background description for word ‘cancer’.

Cancer N disease, background | medical

Figure 1 illustrates a sub-tree of concepts. It includes eighteen background such as *military*, *sport*, *business*, *industry*, *entertainment*. Some backgrounds have hypernymy relationship. For instance, *Industry* and *Theory* is sub-background of *Science*.

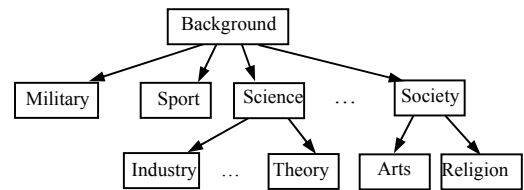


Figure 1: Background Concept Tree

3.2 Topic-semantic Annotation for Document

Definition 2 (Topic and Attributes of Topic): Topic is a user’s focus that is represented by a word. Let C is concepts set in core ontology. Attributes of topic consist of a collection of concepts $\{p_1, \dots, p_n\} \subset C$; attributes can well describe the topic.

By Definition 2, we pay attention to attributes that may well describe topic when choosing attributes for topic.

Example 4: Corpus involved peoples named ‘Yao Ming’ are available. It is needed to cluster documents according to ‘Yao Ming’. Therefore, we set ‘Yao Ming’ as topic. There are three considerations (motivations) on choosing attributes for topic ‘Yao Ming’. (1) Different peoples have different background. Words may represent background. For instance, when words *coach*, *stadium* emerge in a document, it can be inferred that the peoples involved in this document is related to ‘sport’. (2) The places where peoples have grown up and lived may well discriminate different peoples. (3) Some people names, institution and organization names that do not occur in dictionary are called named entity. Named entities may be used to describe semantic of topic. Therefore, at last, *background*, *place* and *named entity* are chose as attributes of topic ‘Yao Ming’.

Definition 3 (Distance of Words): Let doc , $para$, sen be collections of words derived from a document, paragraph and sentence respectively. The distance of two words w_1, w_2 is defined as following function:

$$dis(w_1, w_2) = \begin{cases} 1, & \text{if } w_1, w_2 \in sen \\ 2, & \text{elseif } w_1, w_2 \in para \\ 3, & \text{elseif } w_1, w_2 \in doc \\ \infty, & \text{else} \end{cases}$$

Note that mutual information $I(t_i, t_j) = \log(p(t_i, t_j) / p(t_i) * p(t_j))$ may be used to compute correlation scale of two words. However, mutual information only explores occurring frequency of two words. In fact, correlation scale of two words is also related to their distance.

Definition 4 (Correlation Scale of Words): Let t_i and t_j be two words. Correlation scale of t_i and t_j , $\xi(t_i, t_j)$, is equal to $I(t_i, t_j) / dis(t_i, t_j)$. $\xi(t_i, t_j) = I(t_i, t_j) / dis(t_i, t_j)$.

Lemma 1: correlation scale of two words is in direct proportion to their occurring frequency in same documents and is in inverse proportion to their distance.

Proof. Let t_i and t_j be two words. By Definition 4, following increasing of distance of two words, their correlation scale will decrease. Thus correlation scale of two words is in inverse proportion to their distance. By formula of mutual information $I(t_i, t_j) = \log(p(t_i, t_j) / (p(t_i) * p(t_j)))$, $p(t_i, t_j)$ will increase when occurring frequency of two words increase, thus mutual information of two words $I(t_i, t_j)$ will also increase. By Definition 4, correlation scale of two words is in direct proportion to occurring frequency of them in same document.

Definition 5 (Semantical Correlation of Words): Let t_i and t_j be two words. If t_i may describe semantic of t_j and vice versa, we call t_i and t_j is semantical correlation.

As Mei point out in [5], “the semantic of a word can be inferred from its context, and words sharing similar contexts tends to be semantically similar”. We use this as an observation in this study.

Observation 2: The semantic of a word can be inferred from its context to a certain extent of accuracy.

Observation 3: In the view of linguistics, one document may include several paragraphs and the document may involve multiple themes. However, most of documents meet paragraph-theme criterion, i.e. one paragraph only involves one theme.

We can derive lemma 2 on condition that conclusions of Observation 2 and Observation 3 are correct.

Lemma 2: Let t_i be a word, T be topic, $d = dis(t_i, T)$ be the distance of t_i and T . If themes of paragraphs in one document are unknown, t_i and T is semantical correlation only when $d \leq 2$.

Proof. I : By Lemma 1, the correlation scale of word t_i and topic T will increase while the distance of t_i and T shortens. II : By Observation 2, one topic may derive its semantics from its contexts. III: Thus words that have shorter distance with topic are much more suitable to annotate semantic for topic. IV: By Observation 3, only words occurring in same paragraph with topic share same background with topic when themes of paragraphs are unknown. V : By III, IV, only words whose distance with topic is not greater than 2 are suitable to annotate topic semantic. i.e. if themes of paragraphs are unknown, words and topic are semantically correlation only when $d \leq 2$.

By Lemma 2, words that do not occur with topic in same paragraphs cannot be used to make semantic annotation for topic if themes of paragraphs are unknown.

Semantic annotation for topic needs to extract words that both are semantically correlation with topic and may be mapped to one of attributes. By Lemma 1, correlation scale of two words is in direct proportion to their co-occurring frequency. Thus we also extract co-occurring frequency of words with topic.

Definition 6 (Topic-semantic Annotation): Let s be a document, T be topic, $\{p_1, \dots, p_n\}$ be attributes collection of T . \forall word $t_i \in s$, if t_i may be mapped to one attribute p_j and t_i are semantical correlation with T , 2-tuple $\langle t_i, p_j \rangle$ will be added into attribute vector of p_j , P_j . After exploring all words in s , vectors collection $\{P_1, \dots, P_n\}$ may be derived. We call this process topic-semantic annotation.

In this study, theme of paragraph does not be involved. Therefore, we quote conclusion of Lemma 2 in Algorithm 1.

Algorithm 1 (ODSA): Ontology based Document topic Semantic Annotation

Input: document d , Topic t , attributes collection of t A , ontology

Output: Array of Vectors vec

Method:

```

1  $S = findPara(d, t)$ ; //find out paragraphs in which topic emerges
2  $L = split(S)$ ; //splitting words and deriving words collection
3  $preprocess(L)$ ;
4 for each word  $w$  in  $L$ 
5   for each attribute  $attr$  in  $A$ 
6     if ( $find(w, conceptTree, attr)$ )
7        $insert(vec[attr], w)$ ;
8   end
9 end
10  $sort(vec)$ ;
11 return  $vec$ ;

```

Algorithm 1 only extracts words from paragraphs in which topic emerges. In step 5~8 of Algorithm 1, function $find$ maps words to concept tree of ontology. If the word can be mapped into one of attributes, insert the word into vectors. In order to reduce dimensionality of vectors and better reflect background of topic, instead of words, we add background of words into vector.

In Algorithm 1, outer loop (step 4~9) processes all words extracted. Let number of words be m , then number of looping is m . Inner loop (step 5~8) find out mapping relationship between words and each attribute in attributes collection. Let the number of attributes be n . Then the number of looping is n . Thus the time complexity is $O(m \times n)$.

Example 5: The following is a paragraph selected from website www.china.com.

Houston Rockets center Yao Ming grabs a rebound in front of Detroit Pistons forward Rasheed Wallace and Rockets forward Shane Battier during the first half of their NBA game in Auburn Hills, Michigan.

Figure 2: A paragraph about YaoMing

It can be easily observed from this paragraph that words *rebound*, *forward*, *center*, and *game* may be mapped to *sport* background. The four words occur four times. In this paragraph, there are three place names *Detroit*, *Huston* and *Michigan*, they occur one times respectively. Named entity is *Rasheed Wallace* and *Shane Battier* that occurs two times. Topic-semantic annotation of the paragraph is listed as following when setting *Yao Ming* as topic.

Topic: Yao Ming

Attributes: p_1 =background, p_2 =place, p_3 =named entity

Feature vectors:

P_1 ={<sport, 4>

P_2 ={<Huston, 1>, <Michigan, 1>, < Detroit ,1>}

P_3 ={< Rasheed Wallace, 1>, < Shane Battier, 1>, < Auburn Hills, 1>}

3.3 Dissimilarity Matrix

On building dissimilarity matrix of corpus, dissimilarity function is needed. Dissimilarity of two documents is their topic-semantic dissimilarity in topic oriented document clustering. We propose a dissimilarity criterion function (Equation 2).

$$Sim(v_1, v_2) = \frac{\sum_{k=1}^n W_{1k} \times W_{2k}}{\sqrt{\sum_{k=1}^n W_{1k}^2 \times \sum_{k=1}^n W_{2k}^2}} \quad (1)$$

$$DisSim(d_1, d_2) = 1 - \frac{\sum_{i=1}^n f(l_j, Sim(v_{1i}, v_{2i}))}{n} \quad (2)$$

Equation 2 is based on traditional similarity criterion function of vector (Equation 1). Equation 2 first pairwise computes similarity of attribute vectors in two documents. Then computes sum of all similarities and its average value. Dissimilarity of two documents is one minus the average value. In Equation 2, v_{1i} is feature vector of attribute p_i of document d_1 ; v_{2i} is feature vector of attribute p_i of document d_2 . $Sim(v_{1i}, v_{2i})$ is similarity of vector v_{1i} and v_{2i} . Function f is strengthen-weaken function. Both input and output zone of the function are $[0,1]$. The function enlarges or shortens input value according to curves in Figure 3.

l_j in Equation 2 are one of curves in Figure 3. Motivation of using

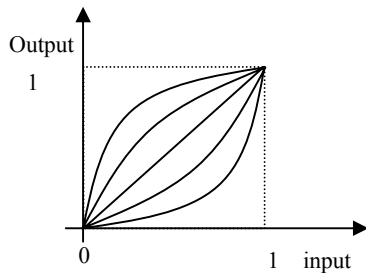


Figure 3: Strengthen-weaken Function

strengthen-weaken function is that user may enlarge or shorten similarity value on a feature vector. For instance, in our works, little similarity on named entity reflects large similarity on topic, i.e. named entity has large distinguishing ability. Therefore, we use strengthen-weaken function to enlarge similarity value on feature vector of named entity.

4. TOPIC ORIENTED DOCUMENTS CLUSTERING

Current clustering algorithms often need user to set some parameters such as the number of clusters, radius or density threshold. If users lack experience to choice parameters, it is difficult to produce good clustering solution. To solve the problem, we proposed a new clustering algorithm without parameter setting. It first build clustering tree by using hierarchical clustering algorithm. Then recommend best clustering solution on clustering tree to users by using a criterion function.

Based on inner criterion function in [8], we propose a new clustering evaluating function *DistanceSum*. The motivation is that singly evaluating intra-cluster similarity or inter-cluster dissimilarity cannot reflect quality of a clustering solution. We believe that all samples being in one cluster and each sample being in one cluster are two the worst clustering solution. Criterion function *DistanceSum* evaluates clustering solution by exploring both intra-cluster similarity and inter-cluster dissimilarity.

$$DistanceSum = \left(\sum_{i=1}^n \frac{\sum_{p,q \in C_i} dis(p,q)}{|C_i|} + \sum_{C_j, C_k \in C} dis(C_j, C_k) / |C| \right) \quad (3)$$

Let $C=\{C_1, \dots, C_n\}$ be a clustering solution. After clustering, documents are partition into clusters $\{C_1, \dots, C_n\}$. $|C_i|$ is number of samples in cluster C_i ; $|C|$ is number of cluster in corpus C .

$\sum_{p,q \in C_i} dis(p,q)$ is called intra-cluster distance sum of NO.1 cluster. $\sum_{C_j, C_k \in C} dis(C_j, C_k)$ is sum of distance inter-cluster.

Inter-cluster distance of two clusters is the distance of the furthest two samples in two clusters.

Criterion function *DistanceSum* shows that value of *DistanceSum* will reach the greatest at two situations where all samples are in one cluster and each sample is in one cluster respectively. After merging two clusters, intra-cluster distance sum of new cluster will increase, but inter-cluster distance sum of clustering solution will decrease. If the merging is reasonable, *DistanceSum* of solution will decrease, or increase in inverse.

The less *DistanceSum* is, the more reasonable documents partition is. Therefore, we choose the solution with the least *DistanceSum* as final solution. Without parameter setting, we can get the best clustering solution by using Algorithm 2.

Algorithm 2(OHC): Optimizing Hierarchical Clustering

Input: Dissimilarity Matrix *matrix*

Output: Best Solution *P*

Method:

```
1  $C = \{C_1, \dots, C_m\}$ ;  $\min = +\infty$ ;  $P = \emptyset$ ; //initiation
2 while ( $|C| \geq 1$ ) {
3    $\text{sum} = \text{DistanceSum}(C)$ ;
4   if ( $\text{sum} < \min$ ) {
5      $\min = \text{sum}$ ;
6      $P = C$ ;
7   }
8   select  $D_{st}$  where  $(s, t) = \text{argmin}_{i, j} d_{ij}$  in  $C$ ;
9   merge  $C_s$  and  $C_t$  into a new cluster  $C_u$ ;
10  return  $P$ ;
```

Algorithm 2 is an agglomerative hierarchical clustering algorithm. In step 1, each sample is one cluster. Each iteration in the Algorithm 2 will merge two clusters with minimum dissimilarity to derive a new cluster. After last iteration, all samples are merged into one cluster. And then clustering tree is built.

Note that there are m samples at first and each loop will decrease one cluster until all samples are merged into one cluster, thus the algorithm loops m times and its time complexity is $O(m)$.

5. EXPERIMENTS

To the best our knowledge, topic oriented document clustering has not been well addressed in the existing works. Experiments, in this study, will compare Algorithm 1 ODSA to the unsupervised clustering approach that extract feature vector of document and compute TFIDF weight of words (TFIDF method hereafter) on time performance and dimensionality of vector.

5.1 Dataset and Environment

We collected Chinese web pages from Internet as test dataset. These web pages are involved three peoples named ‘Li Ming’. There are 25 pieces of web pages about ‘Li Ming’ in Dalian Shide football club, 25 pieces of web pages about ‘Li Ming’ in Shanghai Zhongyuan football club and 10 pieces of web pages about ‘Li Ming’ in Discuz Company.

Dataset *dataset1* consists of web pages about two athlete ‘Li Ming’. *dataset2* consists of web pages about ‘Li Ming’s in Shanghai Zhongyuan football club and in Discuz Company.

In the experiments, topic is people ‘Li Ming’; attributes include {background, place, named entity}. Experiments are preformed on an INTEL C3 1.0G PC with 256M memory, running Windows XP OS. All algorithms are implemented in Java.

5.2 Criterion Function

We propose two measures (equation 4 and 5) to compute precision and recall of one clustering solution.

$$\text{Precision} = \frac{1}{k} \sum_{i=1}^k \max(\text{Prec}(C_i, C'_j) \mid j = 1, \dots, n) \quad (4)$$

$$\text{Recall} = \frac{1}{k} \sum_{i=1}^k \max(\text{Reca}(C_i, C'_j) \mid j = 1, \dots, n) \quad (5)$$

Prec and *Reca* in Equation 4, 5 are two popular measures in evaluating classification:

$$\text{Precision } \text{Prec} = \text{TP} / (\text{TP} + \text{FP}) \quad \text{Recall } \text{Reca} = \text{TP} / (\text{TP} + \text{FN})$$

In the experiment, class label of each sample in dataset is available. $C' = \{C'_1, \dots, C'_n\}$ is class labels collection and there are n class labels in the collection. One clustering solution $C = \{C_1, \dots, C_k\}$ partition samples into k clusters. In Equation 4, precision of each cluster $C_i \in C$ will be computed according to each class label $C'_j \in C'$, $\text{Prec}(C_i, C'_j)$, i.e. regard samples whose class label is C'_j as positive samples. And then choose the greatest value from n results as precision of one cluster. In the last step, average value of precision of all clusters in C just is precision of the solution. Recall of clustering solution (Equation 5) takes similar steps as the above.

We use F-Measure $F = 2 * \text{Precision} * \text{Recall} / (\text{Precision} + \text{Recall})$ as criterion function for clustering solution.

5.3 Experiment 1

The purposes of the experiment are to (1) compare time performance of ODSA and TFIDF approach on building dissimilarity; (2) compare dimensionality of feature vector extracted by using two approaches. In order to reduce dimensionality of feature vectors in TFIDF method, terms in feature vector whose weight are less than threshold will be deleted. We set five weight thresholds {1, 2, 3, 4, 5}.

Figure 4 shows that ODSA only spend 16 seconds to build dissimilarity matrix on *dataset1* while TFIDF method spend 70 seconds in the best situation where threshold is 5. On *dataset2*, ODSA only spend 2 second while TFIDF spent 18 seconds at the best situation. It can be conclude that ODSA have better time performance than TFIDF method. This is because that (1) ODSA only extracts words belonging to one of attributes from parts of paragraph while TFIDF extract all words in the whole document; (2) TFIDF method will compute weight of each word by exploring frequency of the word in all documents.

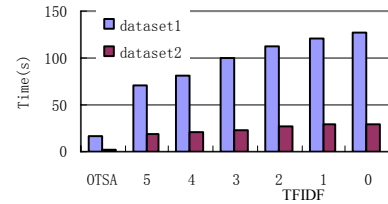


Figure 4: Time Performance

Figure 5 illustrates average dimensionality of vectors extracted from documents by using ODSA and TFIDF methods. In ODSA, dimensionality of one sample is dimensionality sum of all feature vectors in the sample. From Figure 5, it can be observed that dimensionality on ODSA is less than TFIDF approach. ODSA

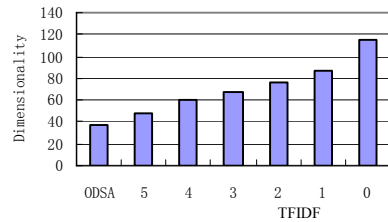


Figure 5: Dimensionality Analysis

may well reduce dimensionality.

5.4 Experiment 2

In experiment 2, we build dissimilarity matrix by using ODSA and TFIDF approach (at five thresholds), and then use Algorithm 2 OHC to produce clustering solutions on two dissimilarity matrixes. We compare clustering solution by using F-Measure.

Experiment 2 use two datasets *dataset1* and *dataset2*. Results are shown in Table1 and Table 2.

Table 1: Comparison of clustering solution on *dataset1*

ID	Number of cluster	F (%)
ODSA	5	56
TFIDF(1)	7	40.7
TFIDF(2)	7	38.9
TFIDF(3)	7	37
TFIDF(4)	7	33.7
TFIDF(5)	7	33

Table 2: Comparison of clustering solution on *dataset2*

ID	Number of cluster	F (%)
ODSA	3	80
TFIDF(1)	6	43.5
TFIDF(2)	6	50
TFIDF(3)	6	47
TFIDF(4)	6	47
TFIDF(5)	6	43.9

Table 1 and Table 2 show that clustering solution based on ODSA has greater advantage than TFIDF approach on F-Measure. Because two ‘Li Ming’ in *dataset1* have same background, football, and two ‘Li Ming’ in *dataset2* have different background, we can see better clustering solution in Table 2 than Table 1.

5.5 Experiment 3

Experiment 3 evaluates function *DistanceSum* on *dataset1*. *DistanceSum* value of each clustering solution during agglomerative hierarchical clustering is computed, and then the values are illustrated in Figure 6.

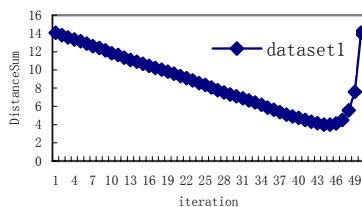


Figure 6: Evaluation of clustering solutions

Figure 6 shows that *DistanceSum* arrive the greatest value at two situations where all samples are in one cluster (NO.50 iteration)

and each sample is in one cluster (NO.1 iteration). *DistanceSum* value will decrease after each merging of clusters, i.e. quality of clustering is improved. At NO.46 iteration, quality of clustering reaches the best and there are 5 clusters at this time.

6. CONCLUSION

Traditional document clustering are unsupervised learning approaches. Traditional approaches often fail to obtain good clustering solution when users want to group documents according to their need. Focusing on this problem, we propose a new clustering approach. First, we describe need of user with a multiple-attributes topic. Then proposed a Topic-semantic annotation algorithm based on ontology and a dissimilarity criterion function. We also proposed an optimizing hierarchical clustering algorithm that may provide the best clustering solution to user by using a criterion function without parameter setting. Experiments show that new approach is feasible and effective.

7. REFERENCES

- [1] Benjamin C.M. Fung, Ke Wang and Martin Ester. Hierarchical Document Clustering Using Frequent Itemsets. Proceedings of the SIAM International Conference on Data Mining,2003.
- [2] Chihli Huang, Stefan Wermter and Peter Smith. Hybrid Neural Document Clustering Using Guided Self-Organization and WordNet. Intelligent Systems, IEEE,3/2004
- [3] Hotho A, Maedche A, Staab S. Ontology-Based document clustering. In Proc. of the Workshop “Text Learning: Beyond Supervision” at IJCAI 2001. Seattle, WA, USA, August 6.
- [4] K.Wagstaff, C. Cardie, S.Rogers and S. Schroedl. Constrained k-means clustering with background knowledge. In Proc. Of 18th International Conference on Machine Learning (ICML-2001), page 577-584, 2001.
- [5] Qiaozhu Mei, Dong Xin, Hong Cheng, Jiawei Han, ChengXiang Zhai. Generating Semantic Annotation for Frequent Pattern with Context Analysis. KDD 2006, Philadelphia,USA.
- [6] Sugato Basu, Mikhail Bilenko and Raymond J.Monney. A probabilistic framework for semi-supervised clustering. In Proc. Of the 10th Int’l Conference on Knowledge Discovery and Data Mining, 2004.
- [7] Xiaohui Cui, Thomas E. Potok and Paul Palathingal. Document Clustering Using Particle Swarm Optimization. Swarm Intelligence Symposium, 2005. SIS 2005. Proceedings 2005 IEEE.
- [8] Ying Zhao, George Karypis. Topic-driven Clustering for Document Dataset. Proc. SIAM Data Mining Conference, 2005
- [9] Zhao Shi-Qi, Liu Ting, Li Sheng. A Topical Document Clustering Method[J]. Journal of Chinese Information Processing. Vol. 21 , No. 2 Mar. , 2007.
- [10] Zhen-dong Dong, Qiang Dong. HowNet.http://www.keenage.com