Cluster-based Hierarchical Topic Trees for Topic Detection

Man Xuan [1], Han-joon Kim [1], Jae Young Chang [2]

[1.] School of Electrical and Computer Engineering, University of Seoul, Seoul, 130-743, Korea
[2.] Department of Computer Engineering, Hansung University, Seoul, 136-792, Korea
mmxuan23@gmail.com, khj@uos.ac.kr, jychang@hansung.ac.kr

Abstract: Extracting topic keywords from on-line text documents is highly significant in text mining applications. In our work, extracted keywords are represented as a hierarchical topic tree. For this, we basically use incremental clustering technique for incoming online documents. Moreover, we define a cluster-based measure similar to the *tf-idf* measure and a probabilistic inequality to determine subsumption relationships among keywords. In this paper, with Google news data, we empirically analyze our proposed method in terms of the threshold value of incremental clustering algorithm, the range of keyword extraction measure and the amount of text data and prove its superiority.
[Xuan M, Kim HJ, Chang JY. **Cluster-based Hierarchical Topic Trees: An Empirical Analysis**. *Life Sci J* 2014;11(7):706-710] (ISSN:1097-8135). http://www.lifesciencesite.com. 102

Keywords: Text mining; topic keywords; topic trees; clustering

1. Introduction

A topic tree is a sort of hierarchical structure with topic words. The root of a topic tree is a general topic word, the internal nodes and leaves of a topic tree correspond to more specific topic words. Topic tree can be generated by computing so called 'subsumption relationships' among topic words; that is, it is represented as the parent-child relationship in a hierarchical tree structure (Lawrie and Croft, 2001) (Kim and Lee, 2008). As related work, Sanderson and Croft (1999) proposed a probabilistic algorithm to determine the subsumption relationship by evaluating co-occurrences of two words. However, in our empirical study, this conventional algorithm has been evaluated to be more or less ineffective.

As a new strategy, we intend to use clustering techniques to build topic trees for incoming online documents. Describing clusters by topic trees have two advantages. Firstly, users can figure out the main content of a cluster only by observing the hierarchical relationships of the cluster's topic trees, and know more detail by reading the internal nodes and leaves of the cluster's topic trees. Secondly, topic tree structure can settle the polysemy problem that a word has more than two meanings. Through extensive experiments, we have found that more descriptive topic words in describing a cluster are more likely to be the parent. Hence, we determine the subsumption relationship by combining cluster description measure and co-occurrence of two topic words. To achieve this, we define a cluster-based measure similar to the conventional *tf-idf* weighting measure (Manning et al., 2008). It reflects how important a particular term (or word)[1] is within the document in a given document corpus. The *tf-idf* value increases proportionally to the term frequency (*tf*) in the document, but decreases proportionally to the document frequency, the number of documents that the term occurs in the corpus; that is, *idf* is a direct measure of the informativeness of the term in the whole corpus. Based on the idea of *tf-idf* weighting scheme, we try to identity words important to each cluster since we build topic trees from each cluster. The proposed cluster description measure that utilizes clustering results has been found to be very effective for building topic trees. In this paper, with recent *Google* news data, we empirically analyze our proposed method in terms of the threshold value of incremental clustering algorithm, the range of the keyword extraction measure and the size of document corpus, while evaluating real topic trees generated.

2. Generating Topic Trees

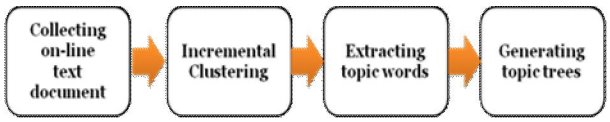

Figure 1. Process of Topic Tree Generation

Figure 1 is the process of generating the topic trees with clustering results. It consists of three steps:

① Perform clustering the on-line text documents with the incremental clustering algorithm

[1] The definition of 'term' depends on the application; that is, terms can be single words, keywords, or longer chunk phrases. Throughout the paper, the 'term' and 'word' are used interchangeably according to the context.

② Rank the candidate topic words in each cluster by the proposed *ctf-cdf-icf* measure (which will be described in detail in this section), and choose top-*k* topic words
③ Discover the parent topic words for each topic word and then build topic trees

Steps 1&2 (*Extracting topic keywords*): As for clustering technique, the incremental clustering is suitable for topic trees since topic keyword extraction needs to be performed for textual document streams (Walls et al., 1999). The incremental clustering is a dynamic clustering technique that can continuously decompose newly incoming documents while inserting them into the current clusters. The algorithm should be more robust to noise and outliers (Kaufman and Rousseeuw, 1987). In this regard, we have previously proposed 'incremental *k*-medoid clustering' algorithm (Xuan and Kim, 2013), and use the algorithm in this paper.

As stated before, we need to isolate significant words from a set of clusters rather than from a set of documents. Thus, based on the idea of *tf-idf* weighting scheme, we define the following three measures: *cluster term frequency*, *cluster document frequency*, and *inverse cluster frequency*. These measures are combined by multiplication.

Firstly, the words that can describe a cluster effectively tend to appear more time in the cluster. Thus we calculate the *cluster term frequency* (*ctf*) as a measure to reflect a term's describing power to a cluster. We first define the number of occurrences of the term t in the cluster c as 'term frequency' in this paper. Furthermore, in computing the final *ctf* value, we need to normalize the term frequency to prevent a bias towards length of documents. Accordingly, the *ctf* of term t in cluster c is calculated as follows:

$$ctf(t,c) = \frac{tf(t,c)}{\sum_{i=1}^{n} tf(t_i,c)} \qquad (1)$$

where $tf(t,c)$ is the frequency of the term t in the cluster c. $\sum_{i=1}^{n} tf(t_i,c)$ is the sum of term frequencies of all the terms occurring in the cluster c. And, n is the total number of terms in the cluster c.

Secondly, we have seen that the words that can describe a cluster well tend to often appear in most documents in the cluster. For this reason, we calculate the *cluster document frequency* (*cdf*) as a measure to embody a term's describing power to a cluster. We define the number of documents which have the given term t in the cluster c as 'document frequency' in this paper. By normalizing the measure, we get the *cdf* value; the *cdf* of term t in cluster c is defined as follows:

$$cdf(t,c) = \frac{df(t,c)}{D} \qquad (2)$$

where $df(t, c)$ is the frequency of documents where the term t occurs in the cluster c. D denotes the total number of documents in the cluster c.

Finally, we consider another observation that the words that occur in all clusters cannot well describe any cluster. Therefore, we define the *inverse cluster frequency* (*icf*) as a measure to incarnate a term's describing power. We first define the number of clusters in which the given term occurs as 'cluster frequency'. For the term t, its cluster frequency and its describing power have an inverse relationship with each other. Consequently, the *icf* measure of term t is defined as follows:

$$icf(t) = \log_2 \frac{C+1}{cf(t)+1} \qquad (3)$$

where $cf(t)$ is the frequency of clusters where the term t occurs, and C denotes the total number of clusters in the current document corpus.

Up to now, we describe three measures *ctf*, *cdf*, and *icf* to reflect a term's description power to a cluster. Finally, to express the description power of a given term, we combine the above three measures by multiplication, which is denoted as *ctf-cdf-icf*. Furthermore, since the *ctf-cdf-icf* measure varies in value depending on the cluster, the measure is necessary to normalize; as a result, the following normalized *ctf-cdf-icf* (shortly, *nCCI*) ranges from 1 to (1+d) as shown in Equation 4.

$$nCCI(t) = \frac{t - t_{\min}}{t_{\max} - t_{\min}} \cdot d + 1 \qquad (4)$$

where t_{max} (or t_{min}) is the largest (or smallest, resp.) value of *nCCI* in a set of candidate words selected from a cluster, and d is a positive number, which is maximally set to be 0.8. With this measure, we can identify a set of topic keywords from each cluster.

Step 3 (*Building topic trees*): Sanderson and Croft (1999) have proposed a way to determine the parent-child 'subsumption' relationship by calculating the co-occurrence probability of two topic words. Their idea is that for two topical terms t_i, t_j if

$$Pr(t_i|t_j) \geq 0.8 \text{ and } Pr(t_i|t_j) > Pr(t_j|t_i) \qquad (5)$$

then t_i is said to subsume t_j. Here, $Pr(t_i|t_j)$ is the probability that t_i occurs in the document set in which t_j occurs, and 0.8 was determined empirically. In other words, t_i is a general topic word relatively compared to t_j, and thus in a topic tree, t_i can be t_j's parent and t_j can be t_i's child.

However, in our experiments including their idea, we have found that topic words which are more powerful to describe a cluster are more likely to be the

parent (or more general). Therefore, we have concluded that another parameter is required to determine more effective subsumption relationships in constructing hierarchical topic trees. Our proposed idea is that both cluster description measure and the degree of co-occurrence of two topic words can be used to determine the subsumption relationship between them. Accordingly, to discover subsumption relations among words extracted from clusters, we consider the following probabilistic inequality. For two topic words t_i and t_j, if

$$Pr(t_i|t_j) \geq 0.7 \text{ and } Pr(t_i|t_j) \cdot nCCI(t_i) > Pr(t_j|t_i) \cdot nCCI(t_j) \quad (6)$$

then t_i is said to subsume t_j. Here, $nCCI(t_i)$ is the measure for describing words extracted from clusters, which can express cluster description power. In Equation 6, $Pr(t_i|t_j) \cdot nCCI(t_i)$ is the probability of t_i subsumes t_j , not depending only upon $Pr(t_i|t_j)$. In addition, we have modified the first condition of Sanderson and Croft (1999) as $Pr(t_i|t_j) \geq 0.7$, not 0.8. This is because the conventional lower bound 0.8 is too strict in the current on-line documents, and actually our new lower bound 0.7 has showed the best result in our experiments. Another difference is that in Sanderson and Croft's idea, a child may have more than one parent, but we choose only one parent for each topic word based on the probability $Pr(t_i|t_j) \cdot nCCI(t_i)$.

3. Results

3.1 Experimental Setup

To evaluate our proposed method for building topic trees, we have prepared the datasets collected from the *Google* news U.S. edition (*http://news.google.com*). It contains lots of documents about the categories such as *world, business, elections, technology, entertainment, sports, science,* and *health* from August 12, 2012 through October 29, 2012. As an initial set of clusters, the first cluster contains 10 documents about *smart phone*, the second cluster contains 10 documents about *Olympics*, and the third cluster contains 10 documents about *Oscar*.

Because topic words should be noun, we have utilized two natural language processing tools to extract noun phrases: *Illinois Part of Speech Tagger* (http:// cogcomp.cs.illinois.edu/page/software_view/POS) and *Illinois Chunker* (http://cogcomp.cs.illinois.edu/page/software_view/Chunker). In addition, to evaluate the accuracy for each pair of subsumption relations within the topic trees generated, we calculate the following subsumption accuracy (Brachman, 1983):

$$\text{Subsumption Accuracy} = \frac{\text{The correct number of subsumption relations}}{\text{The total number of subsumption relations}} \quad (7)$$

3.2 Results and Discussion

Figure 2 is part of topic trees generated by proposed method. With the topic trees, we can understand the main content contained in the current documents, and moreover it is possible to effectively grasp the change of documents while observing the change of topic trees over time.

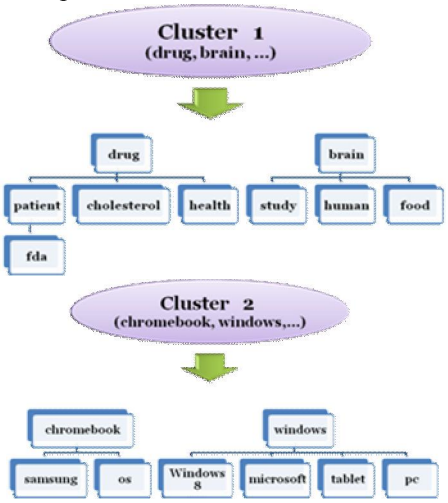

Figure 2. Examples of topic trees

During the clustering process, we choose the clusters that have more than N documents, in which the positive N influences the size of clusters being clustered. Figure 3 shows the accuracies and the number of clusters for different N values. When $N = 2$, the accuracy is abnormally high; this is because the total documents are clustered into too many clusters, and most of topic words corresponds to root nodes without any child topic words. In contrast, with the increase of N, topic trees generated are more reasonable and the accuracy becomes increased. As seen in the figure, when N is 5, we achieve the best accuracy.

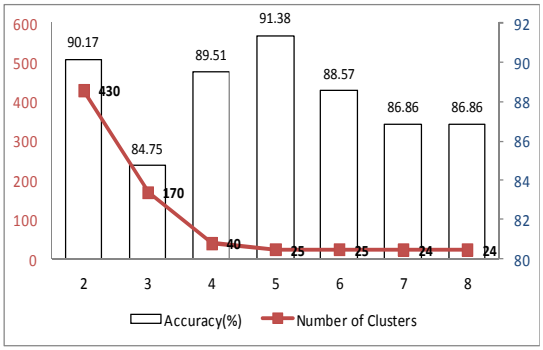

Figure 3. Changes of accuracies from varying the threshold value of incremental clustering

As stated in Section 2, *ctf-cdf-icf* values are mapped into 1 to (1+*d*) for *n*CCI. Figure 4 shows the change of accuracies from varying *d* value. When *d* = 0, it corresponds to the base line method done by Sanderson and Croft (1999). In our work, we set the maximum value to be 0.8, and we have achieved the best performance when *d* is between 0.3 and 0.5.

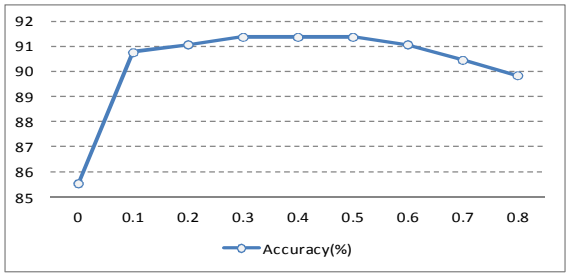

Figure 4. Changes of accuracies from varying the *d* value of *n*CCI

Figure 5 shows the change of accuracy as the amount of documents increases. Both of the base line method and the proposed method shows a good performance at the initial time. This is because the initial clusters are built manually, which have generated reasonable subsumption relations. It implies that if we improve the quality of clusters generated, the accuracy of subsumption relations will be enhanced. As expected, we have seen that the accuracy is getting stable with the increase of documents and the proposed method always shows a better performance than base line one. Moreover, the proposed method can generate more reasonable topic trees.

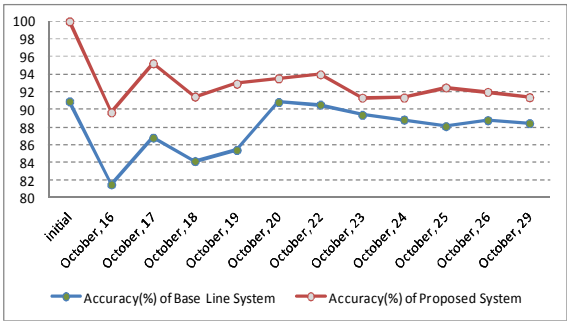

Figure 5. Changes of accuracies with the increase of documents

Figures 6 and 7 are part of topic trees generated by the Sanderson and Croft (base line) method and the proposed one, respectively. As shown in these figures, topic trees generated by the proposed method are more precise and distinct than base line. In Figure 6, we have seen that the trees have several incorrect subsumption relations such as relation between the topic words 'actress' and 'venice'. Moreover, the topic words 'chromebook', 'samsung', 'os', 'academy aware', and 'climate' have no relationship with any other ones. In contrast, Figure 7 shows more reasonable topic trees so that one can catch the main content of the incoming documents while browsing the relations between topic words. The words 'samsung' and 'os' are subsumed by the word 'chromebook'. The word 'film' subsumes its related words such as 'oscar', 'actor', 'actrees', 'phoenix', and 'venice', and moreover the 'oscar' subsumes the word 'academy award'. The word 'climate' has the hierarchical relation with the words 'temperature', 'global warming'. Like this, we have found that most of topic words have hierarchical relations with reasonable topic words through the proposed method. The empirical results suggest that the topic detection with topic trees works fairly well using the technique we proposed.

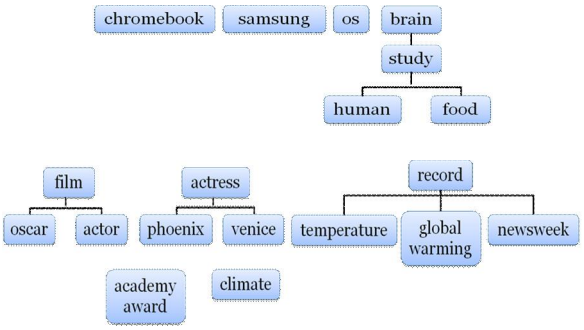

Figure 6. Topic trees generated by the baseline method

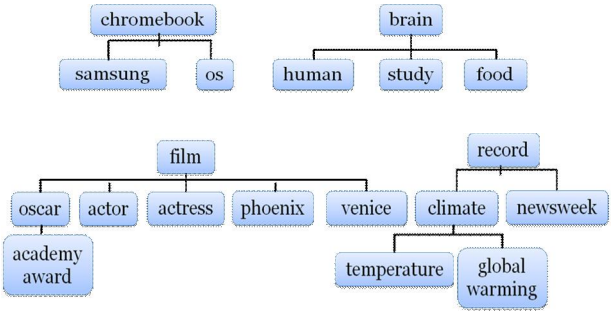

Figure 7. Topic trees generated by the proposed method

4 Conclusions

This paper presented a novel cluster-based technique of discovering hierarchical topic trees for incoming on-line documents. As a new strategy, we used clustering techniques to build topic trees, and proposed a probabilistic approach to discovering subsumption relations between topic words. Unlike the conventional method, we consider the measure of

description power for clusters to achieve more accurate subsumption relations. In short, the proposed topic trees have been built by combining clustering results and probabilistic subsumption relationships. Our empirical study using *Google* news shows that the proposed method can build more reasonable topic trees than the conventional one. In the future, we will develop hierarchical topic detection system using MapReduce under the Hadoop computing environment.

Acknowledgements:

This work was supported by Mid-career Researcher Program through the National Research Foundation of Korea (NRF) grant funded by the Korea government (MISP) (grant number: NRF-2013R1A2A2A01017030), and was financially supported by Hansung University for Jae Young Chang.

Corresponding Author:
Dr. Han-joon Kim
School of Electrical and Computer Engineering
University of Seoul, Korea
E-mail: khj@uos.ac.kr

References

1. Lawrie D, Croft W. Finding topic words for hierarchical summarization. Proceedings of the 24th annual international ACM SIGIR conference on Research and development in information retrieval 2001:349-57.
2. Kim HJ, Lee SG. Building topic hierarchy based on fuzzy relations. Neurocomputing 2003; 51:481-6.
3. Sanderson M, Croft B. Deriving concept hierarchies from text. Proceedings of the 22nd annual international ACM SIGIR conference on Research and development in information retrieval 1999:206 -13.
4. Manning CD, Raghavan P, Schutze H. Introduction to Information Retrieval. Cambridge University Press, 2008:117-20.
5. Walls F, Jin H, Schwartz R. Topic Detection in Broadcast News. Proceedings of the DARPA Broadcast News Workshop 1999:193-8.
6. Kaufman L, Rousseeuw PJ. Clustering by means of Medoids. Statistical Data Analysis Based on the L1–Norm and Related Methods 1987:405-16.
7. Xuan M, Kim HJ. Medoid-based Incremental Clustering for Large Data Streams with MapReduce. Proceedings of the 3rd International Conference on Engineering and Applied Science (ICEAS) 2013:1-9.
8. Brachman R. What IS-A is and isn't. An Analysis of Taxonomic Links in Semantic Networks. IEEE Computer 1983; 16(10):30-6.

3/10/2014