© 2015 IJSRSET | Volume 1 | Issue 2 | Print ISSN : 2395-1990 | Online ISSN : 2394-4099
Themed Section: Engineering and Technology

Exploratory Search for Web Results

Ayer Prakash R., Sherine Mary R.
Computer Science and Engineering, Dhanalakshmi College of Engineering, Chennai, Tamilnadu, India

ABSTRACT

Measuring the similarity between documents is an important operation in the text processing field. In this paper, a new similarity measure is proposed. To compute the similarity between two documents with respect to a feature, the proposed measure takes the following three cases into account the feature appears in both documents, the feature appears in only one document, and the feature appears in none of the documents. For the first case, the similarity increases as the difference between the two involved feature values decreases. Furthermore, the contribution of the difference is normally scaled.

Keywords: good retrieval, reduced-size feature space, secure routing, performance analysis, design and validation

I. INTRODUCTION

Clustering is the classification of objects into different groups, or more precisely, the partitioning of a data set into subsets (clusters), so that the data in each subset (ideally) share some common trait - often proximity according to some defined distance measure. Data clustering is a common technique for statistical data analysis, which is used in many fields, including machine learning, data mining, pattern recognition, image analysis and bioinformatics. The computational task of classifying the data set into k clusters is often referred to as k-clustering. Besides the term data clustering (or just clustering), there are a number of terms with similar meanings, including cluster analysis, automatic classification, numerical taxonomy, botryology and typological analysis.

Document clustering aims to group, in an unsupervised way, a given document set into clusters such that documents within each cluster are more similar between each other than those in different clusters. It is an enabling technique for a wide range of information retrieval tasks such as efficient organization, browsing and summarization of large volumes of text documents. Cluster analysis aims to organize a collection of patterns into clusters based on similarity. Clustering has its root in many fields, such as mathematics, computer science, statistics, biology, and economics. In different application domains, a variety of clustering techniques have been developed, depending on the methods used to represent data, the measures of similarity between data objects, and the techniques for grouping data objects into clusters.

II. METHODS AND MATERIAL

Related Work

Requirement Analysis is the first phase in the software development process. The main objective of the phase is to identify the problem and the system to be developed. The later phases are strictly dependent on this phase and hence requirements for the system analyst to be clearer, precise about this phase. Any inconsistency in this phase will lead to lot of problems in the other phases to be followed. Hence there will be several reviews before the final copy of the analysis is made on the system to be developed. After all the analysis is completed the system analyst will submit the details of the system to be developed in the form of a document called requirement specification.

The Requirement analysis task is a process of discovery, refinement, modeling and specifications. The software scope, initially established by a system engineer and refined during software project planning, is refined in detail. Models of required data, information and control

flow and operational behavior are created. Alternative solutions are analyzed and allocated to various software elements.

Both the developer and the customer take an active role in requirement analysis and specification. The customer attempts to reformulate a sometimes-nebulous concept of software function and performance into concrete detail. The developer acts as interrogator, consultant and problem solver. The communication content is very high. Changes for misinterpretation of misinformation abound. Ambiguity is probable.

Requirement analysis is a software engineering task that bridges the gap between the system level software allocation and software design. Requirement analysis enables the system engineer to specify the software function and performance indicate software interface with other system elements and establish constraints that software must meet. It allows the software engineer, often called analyst in this role, to refine the software allocation and build model of the data, functional and behavioral domains and that will be treated by software.

Requirements analysis provides the software designer with models that can be translated into data, architectural, interface and procedural design. Finally, the requirement specification provides the developer and customer with the means to access quality once software builds.

A) Organization

The rest of the paper is organized as follows. In Section we will discuss about our towards the work. In Section 3, we describe the architecture and the various levels. In Section 4, we describe our algorithm used for security. In Section 5, we conclude our process.

B) Our contribution

The main work is to develop a novel hierarchal algorithm for document clustering which provides maximum efficiency and performance. It is particularly focused in studying and making use of cluster overlapping phenomenon to design cluster merging criteria. Proposing a new way to compute the overlap rate in order to improve time efficiency and "the veracity" is mainly concentrated. Based on the Hierarchical Clustering Method, the usage of Expectation-Maximization (EM) algorithm in the Gaussian Mixture Model to count the parameters and make the two sub-clusters combined when their overlap is the largest is narrated. Experiments in both public data and document clustering data show that this approach can improve the efficiency of clustering and save computing time.

Given a data set satisfying the distribution of a mixture of Gaussians, the degree of overlap between components affects the number of clusters "perceived" by a human operator or detected by a clustering algorithm. In other words, there may be a significant difference between intuitively defined clusters and the true clusters corresponding to the components in the mixture.

Architecture

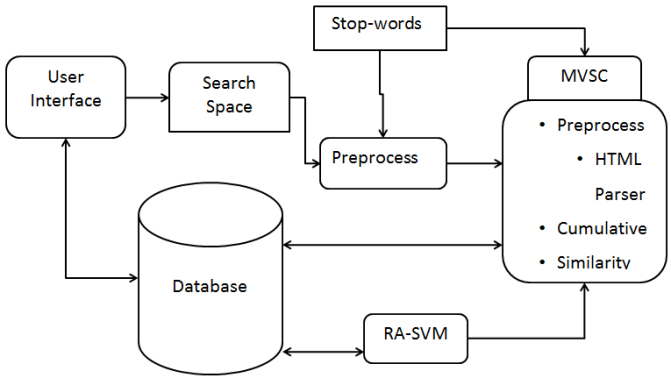

Figure 1: Proposed System Architecture

- Parsing is the first step done when the document enters the process state.
- Parsing is defined as the separation or identification of meta tags in a HTML document.
- Here, the raw HTML file is read and it is parsed through all the nodes in the tree structure.

A) Cumulative Document

- The cumulative document is the sum of all the documents, containing meta-tags from all the documents.
- We find the references (to other pages) in the input base document and read other documents and then find references in them and so on.
- Thus in all the documents their meta-tags are identified, starting from the base document.

B) Document Similarity

- The similarity between two documents is found by the cosine-similarity measure technique.
- The weights in the cosine-similarity are found from the TF-IDF measure between the phrases (meta-tags) of the two documents.
- This is done by computing the term weights involved.
- TF = C / T
- IDF = D / DF.

D → quotient of the total number of documents
DF → number of times each word is found in the entire corpus
C → quotient of no of times a word appears in each document
T → total number of words in the document

- TFIDF = TF * IDF

C) Clustering

- Clustering is a division of data into groups of similar objects.
- Representing the data by fewer clusters necessarily loses certain fine details, but achieves simplification.
- The similar documents are grouped together in a cluster, if their cosine similarity measure is less than a specified threshold [9].

III. RESULTS AND DISCUSSION

Algorithm

The algorithm forms clusters in a bottom-up manner, as follows:

1. Initially, put each article in its own cluster.
2. Among all current clusters, pick the two clusters with the smallest distance.
3. Replace these two clusters with a new cluster, formed by merging the two original ones.
4. Repeat the above two steps until there is only one remaining cluster in the pool.

Thus, the agglomerative clustering algorithm will result in a binary cluster tree with single article clusters as its leaf nodes and a root node containing all the articles.

In the clustering algorithm, we use a distance measure based on log likelihood. For articles *A* and *B*, the distance is defined as

$$\begin{aligned} LL(X) &= \log \prod_{w \in X} p_X(w)^{c_X(w)} \\ &= \sum_{w \in X} c_X(w) \log c_X(w) - N_X \log N_X \end{aligned}$$

$$d(A, B) = LL(A) + LL(B) - LL(A \cup B) \quad (1)$$

The log likelihood $LL(X)$ of an article or cluster X is given by a unigram model:

Here, CX (w) and PX(w) are the count and probability, respectively, of word *w* in cluster *X*, and NX is the total number of words occurring in cluster *X*.

Notice that this definition is equivalent to the weighted information loss after merging two articles:

$$d'(A, B) = (N_A + N_B)H(A \cup B) - (N_A H(A) + N_B H(B)) \quad (2)$$

Where

$$H(X) = -\sum_{w \in X} P_X(w) \log P_X(w) \quad .$$

To avoid expensive log likelihood recomputation after each cluster merging step, we define the distance between two clusters with multiple articles as the maximum pair wise distance of the articles from the two clusters:

$$d(C_1, C_2) = \max_{A \in C_1, B \in C_2} d(A, B) \quad (3)$$

Where C1 and C2 are two clusters, and A, B are articles from C1 and C2, respectively.

Once a cluster tree is created, we must decide where to slice the tree to obtain disjoint partitions for building cluster-specific LMs. This is equivalent to choosing the total number of clusters. There is a tradeoff involved in this choice. Clusters close to the leaves can maintain more specifics of the word distributions. However, clusters close to the root of the tree yield LMs with more reliable estimates, because of the larger amount of data.

We roughly optimized the number of clusters by evaluating the perplexity of the Hub4 development test set. We created sets of 1, 5, 10, 15, and 20 article

clusters, by slicing the cluster tree at different points. A back off trigram model was built for each cluster, and interpolated with a trigram model derived from all articles for smoothing, to compensate for the different amounts of training data per cluster. Then, the set of LMs that maximizes the log likelihood of the Hub4 development data was selected. Given a cluster model set LM={LMi}, the test set log likelihood was obtained as an approximation to the mixture-of-clusters model

$$\begin{aligned} P(w \mid LM) &= \sum_i P(LM_i) * P(w \mid LM_i) \\ &\approx P(LM_{i^*}) * P(w \mid LM_{i^*}) \\ &\propto P(w \mid LM_{i^*}) \end{aligned}$$

Where

$$i^* = \arg\max_i P(LM_i \mid A) \quad ,$$

and P(LMi) and P(LMi|^) are the prior and posterior cluster probabilities, respectively.

In training, *A* is the reference transcript for one story from the Hub4 development data. During testing, *A* is the 1-best hypothesis for the story, as determined using the standard LM.

Note that P(w|LM) depends on the smoothing weights used to compute P(w|LMi), which in turn determine which cluster a story is assigned to, which in turn determines the best smoothing weights. Therefore, we jointly optimize smoothing and cluster assignment in an iterative procedure. First, the posterior probabilities of the smoothed cluster LMs given reference transcripts for a story were calculated. Then, stories with the highest posterior probability of a *same* cluster LM were merged. The interpolation weight for the cluster LM and the general LM was tuned by maximizing the likelihood of the segments in the story cluster corresponding to the cluster LM. These steps were iterated until all cluster assignments became stable and the interpolation weights converged.

Psuedocode for clustering algorithm:

```
1: procedure INITIALIZATION
2:   Select k seeds s_1, ..., s_k randomly
3:   cluster[d_i] ← p = arg max_r {s_r^t d_i}, ∀i = 1, ..., n
4:   D_r ← Σ_{d_i ∈ S_r} d_i, n_r ← |S_r|, ∀r = 1, ..., k
5: end procedure
6: procedure REFINEMENT
7:   repeat
8:     {v[1 : n]} ← random permutation of {1, ..., n}
9:     for j ← 1 : n do
10:      i ← v[j]
11:      p ← cluster[d_i]
12:      ΔI_p ← I(n_p − 1, D_p − d_i) − I(n_p, D_p)
13:      q ← arg max_{r, r≠p} {I(n_r+1, D_r+d_i) − I(n_r, D_r)}
14:      ΔI_q ← I(n_q + 1, D_q + d_i) − I(n_q, D_q)
15:      if ΔI_p + ΔI_q > 0 then
16:        Move d_i to cluster q: cluster[d_i] ← q
17:        Update D_p, n_p, D_q, n_q
18:      end if
19:    end for
20:  until No move for all n documents
21: end procedure
```

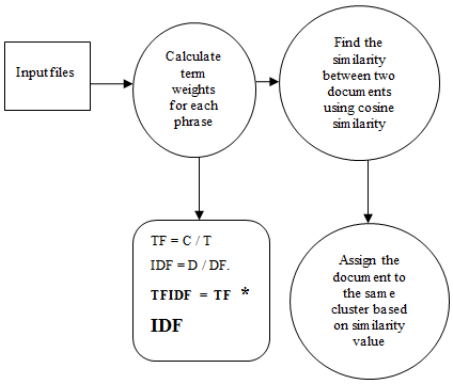

Figure 2: Data Flow Diagram

IV. CONCLUSION

Given a data set, the ideal scenario would be to have a given set of criteria to choose a proper clustering algorithm to apply. Choosing a clustering algorithm, however, can be a difficult task. Even ending just the most relevant approaches for a given data set is hard. Most of the algorithms generally assume some implicit structure in the data set. One of the most important elements is the nature of the data and the nature of the desired cluster. Another issue to keep in mind is the kind of input and tools that the algorithm requires. This report has a proposal of a new hierarchical clustering algorithm based on the overlap rate for cluster merging. The experience in general data sets and a document set indicates that the new method can decrease the time cost, reduce the space complexity and improve the accuracy of clustering. Specially, in the document clustering, the

newly proposed algorithm measuring result show great advantages. The hierarchical document clustering algorithm provides a natural way of distinguishing clusters and implementing the basic requirement of clustering as high within-cluster similarity and between-cluster dissimilarity.

V. FUTURE ENHANCEMENT

In the proposed model, selecting different dimensional space and frequency levels leads to different accuracy rate in the clustering results. How to extract the features reasonably will be investigated in the future work.

There are a number of future research directions to extend and improve this work. One direction that this work might continue on is to improve on the accuracy of similarity calculation between documents by employing different similarity calculation strategies. Although the current scheme proved more accurate than traditional methods, there are still rooms for improvement.

VI. REFERENCES

[1] Cole, A. J. & Wishart, D. (1970). An improved algorithm for the Jardine-Sibson method of generating overlapping clusters. The Computer Journal 13(2):156-163.

[2] D'andrade,R. 1978, "U-Statistic Hierarchical Clustering" Psychometrika, 4:58-67.

[3] Johnson,S.C. 1967, "Hierarchical Clustering Schemes" Psychometrika, 2:241-254.

[4] Shengrui Wang and Haojun Sun. Measuring overlap-Rate for Cluster Merging in a Hierarchical Approach to Color Image Segmentation. International Journal of Fuzzy Systems,Vol.6,No.3,September 2004.

[5] Jeff A. Bilmes. A Gentle Tutorial of the EM Algorithm and its Application to Parameter Estimation for Gaussian Mixture and Hidden Markov Models. ICSI TR-97-021, U.C. Berkeley, 1998.

[6] E.M. Voorhees. Implementing agglomerative hierarchical clustering algorithms for use in document retrieval. Information Processing and Management, 22(6):465–476, 1986.

[7] Sun Da-fei,Chen Guo-li,Liu Wen-ju. The discussion of maximum likehood parameter estimation based on EM algorithm. Journal of HeNan University. 2002,32(4):35~41

[8] Khaled M. Hammouda, Mohamed S. Kamel , efficient phrase-based document indexing for web document clustering , IEEE transactions on knowledge and data engineering, October 2004

[9] Haojun sun, zhihuiliu, lingjunkong, A Document Clustering Method Based On Hierarchical Algorithm With Model Clustering, 22nd international conference on advanced information networking and applications,

[10] Shi zhong, joydeepghosh, Generative Model-Based Document Clustering: A Comparative Study, The University Of Texas.