Space and Cosine Similarity measures for Text Document Clustering

Venkata Gopala Rao S.
M.Tech Software Engineering,
Vardhaman College of Engineering,
Hyderabad, India.

Bhanu Prasad A.
Associate Professor, Department of IT,
Vardhaman College of Engineering,
Hyderabad, India.

Abstract

Correlation study is at the heart of time-varying multivariate volume data analysis and visualization. Document clustering is related to data clustering concept which is one of data mining tasks and unsupervised classification. Clustering is a useful technique that organizes a large quantity of unordered text documents into a small number of meaningful and coherent clusters, thereby providing a basis for intuitive and informative navigation and browsing mechanisms. It is often applied to the huge data in order to make a partition based on their similarity. Initially, it used for Information Retrieval in order to improve the precision and recall from query. It is very easy to cluster with small data attributes which contains of important items. Furthermore, document clustering is very useful in retrieve information application in order to reduce the consuming time and get high precision and recall. Therefore, we propose to integrate the information retrieval method and document clustering as concept space approach. The method is known as Latent Semantic Index (LSI) approach which used Singular Vector Decomposition (SVD) or Principle Component Analysis (PCA). The aim of this method is to reduce the matrix dimension by finding the pattern in document collection with refers to concurrent of the terms. Each method is implemented to weight of term-document in vector space model (VSM) for document clustering.

Keywords: Similarity measures partitioned clustering, text clustering.

1. Introduction

The abundant texts flowing over the Internet, huge collections of documents in digital libraries and repositories, and digitized personal information such as blog articles and emails are piling up quickly every day. These have brought challenges for the effective and efficient organization of text documents. Clustering in general is an important and useful technique that automatically organizes a collection with a substantial number of data objects into a much smaller number of coherent groups [8, 20]. In the particular scenario of text documents, clustering has proven to be an effective approach for quite some time and an interesting research problem as well. It is becoming even more interesting and demanding with the development of the World Wide Web and the evolution of Web 2.0. For example, results returned by search engines are clustered to help users quickly identify and focus on the relevant set of results. Customer comments are clustered in many online stores, such as Amazon.com, to provide collaborative recommendations. In collaborative bookmarking or tagging, clusters of users that share certain traits are identified by their annotations. Text document clustering groups similar documents that to form a coherent cluster, while documents that are different have separated apart into different clusters. However, the definition of a pair of documents being similar or different is not always clear and normally varies with the actual problem setting. For example, when clustering research papers, two documents are regarded as similar if they share similar thematic topics. When clustering is employed on web sites, we are usually more interested in clustering the component pages according to the type of information that is presented in the page.

For instance, when dealing with universities' web sites, we may want to separate professors' home pages from students' home pages, and pages for courses from pages for research projects. This kind of clustering can benefit further analysis and utilize of the dataset such as information retrieval and information extraction, by grouping similar types of information sources together. Accurate clustering requires a precise definition of the closeness between a pair of objects, in terms of either the pair wised similarity or distance. A variety of similarity or distance measures have been proposed and widely applied, such as cosine similarity and the Jaccard correlation coefficient. Meanwhile, similarity is often conceived in terms of dissimilarity or distance as well [15]. Measures such as Euclidean distance and relative entropy have been applied in clustering to calculate the pair-wise distances. Given the diversity of similarity and distance measures available, their effectiveness in text document clustering is still not clear. Although Strehl et al. compared the effectiveness of a number of measures [17], our experiments extended their work by including more measures and experimental datasets, such as the

averaged Kullback-Leibler divergence, which has shown its effectiveness in clustering text and attracted considerable research interest recently. More specifically, we evaluated five measures with empirical experiments: Euclidean distance, cosine similarity, Jaccard coefficient, Pearson correlation coefficient and averaged Kullback-Leibler divergence. Each of the measures are further discussed. In order to come up with a sound conclusion we have performed an empirical evaluation with seven data sets that each have different characteristics. They contain such things as newspaper articles, newsgroup posts, research papers, and web pages. They all come with a set of categorizing labels, with one category attached to each document. These pre-assigned labels are very useful for cluster validation; we use them to measure the consistency between the resulting clusters and the categories created by human experts.

We use two measures to evaluate the overall quality of clustering solutions purity and entropy, which are commonly used in clustering [23, 22]. However, manually assigned labels are normally not available in clustering, and in these case other measure such as within-cluster distances and between clusters distances [13] can be used for evaluation. These are not used in this paper because all the datasets already have labels.

2. Related Work

2.1 Similarity Measures

Before clustering, a similarity/distance measure must be determined. The measure reflects the degree of closeness or separation of the target objects and should correspond to the characteristics that are believed to distinguish the clusters embedded in the data. In many cases, these characteristics are dependent on the data or the problem context at hand, and there is no measure that is universally best for all kinds of clustering problems. Moreover, choosing an appropriate similarity measure is also crucial for cluster analysis, especially for a particular type of clustering algorithms. For example, the density-based clustering algorithms, such as DBScan [4], rely heavily on the similarity computation. Density-based clustering finds clusters as dense areas in the data set, and the density of a given point is in turn estimated as the closeness of the corresponding data object to its neighbouring objects. Recalling that closeness is quantified as the distance/similarity value, we can see that large number of distance/similarity computations are required for finding dense areas and estimate cluster assignment of new data objects. Therefore, understanding the effectiveness of different measures is of great importance in helping to choose the best one. In general, similarity/distance measures map the distance or similarity between the symbolic description of two objects into a single numeric value, which depends on two factors the properties of the two objects and the measure itself. In order to make the results of this study comparable to previous research, we include all the measures that were tested in [17] and add another one the averaged Kullback-Leibler divergence. These five measures are discussed below. Different measure not only results in different final partitions, but also imposes different requirements for the same clustering algorithm.

2.2 Metric

Not every distance measure is a metric. To qualify as a metric, a measure d must satisfy the following four conditions. Let x and y be any two objects in a set and d(x, y) be the distance between x and y.

1. The distance between any two points must be nonnegative, that is, d(x, y) _ 0.
2. The distance between two objects must be zero if and only if the two objects are identical, that is, d(x, y) = 0 if and only if x = y.
3. Distance must be symmetric, that is, distance from x to y is the same as the distance from y to x, ie. d(x, y) = d(y, x).
4. The measure must satisfy the triangle inequality, which is d(x, z) _ d(x, y) + d(y, z).

2.3 Cosine Similarity

When documents are represented as term vectors, the similarity of two documents corresponds to the correlation between the vectors. This is quantified as the cosine of the angle between vectors, that is, the so-called cosine similarity. Cosine similarity is one of the most popular similarity measure applied to text documents, such as in numerous information retrieval applications [21] and clustering too [9]. Given two documents ta and tb , their cosine similarity is

$$SIM_C(\vec{t_a}, \vec{t_b}) = \frac{\vec{t_a} \cdot \vec{t_b}}{|\vec{t_a}| \times |\vec{t_b}|}$$

Where t_a and t_b are m-dimensional vectors over the term set T = {t1, . . . , tm}. Each dimension represents a term with its weight in the document, which is non-negative. As a result, the cosine similarity is non-negative and bounded between [0, 1]. An important property of the cosine similarity is its independence of document length. For example, combining two identical copies of a document d to get a new pseudo document d0, the cosine similarity between d and d0 is 1, which means that these two documents are regarded

to be identical. Meanwhile, given another document l, d and d0 will have the same similarity value to l, that is, sim (t_d ,t_l) =sim(t_{d0} ,t_l). In other words, documents with the same composition but different totals will be treated identically. Strictly speaking, this does not satisfy the second condition of a metric, because after all the combination of two copies is a different object from the original document. However, in practice, when the term vectors are normalized to a unit length such as 1, and in this case the representation of d and d0 is the same.

2.4 Clustering algorithm

For all subsequent experiments, the standard K-means algorithm is chosen as the clustering algorithm. This is an iterative partitioned clustering process that aims to minimize the least squares error criterion [15]. As mentioned previously, partitioned clustering algorithms have been recognized to be better suited for handling large document datasets than hierarchical ones, due to their relatively low computational requirements [16, 9, 3]. The standard K-means algorithm works as follows. Given a set of data objects D and a pre-specified number of clusters k, k data objects are randomly selected to initialize k clusters, each one being the centroid of a cluster. The remaining objects are then assigned to the cluster represented by the nearest or most similar centroid. Next, new centroids are re-computed for each cluster and in turn all documents are re-assigned based on the new centroids. This step iterates until a converged and fixed solution is reached, where all data objects remain in the same cluster after an update of centroids. The generated clustering solutions are locally optimal for the given data set and the initial seeds. Different choices of initial seed sets can result in very different final partitions. Methods for finding good starting points have been proposed [1]. However, we will use the basic K-means algorithm because optimizing the clustering is not the focus of this paper. The K-means algorithm works with distance measures which basically aims to minimize the within-cluster distances. Therefore, similarity measures do not directly fit into the algorithm, because smaller values indicate dissimilarity. The Euclidean distance and the averaged KL divergence are distance measures, while the cosine similarity, Jaccard coefficient and Pearson coefficient are similarity measures. We apply a simple transformation to convert the similarity measure to distance values. Because both cosine similarity and Jaccard coefficient are bounded in [0, 1] and monotonic, we take D = 1 − SIM as the corresponding distance value. For Pearson coefficient, which ranges from −1 to +1, we take D = 1 − SIM when SIM _ 0 and D = |SIM| when SIM < 0.

Singular Value Decomposition (SVD)

The singular value decomposition is a method whi8ch can fi8nd the patterns in the matrix and identify which words and documents are similar to each other. It creates the new matrices from term (t)x document (d) matrix A that are matrices U, ∑ and V such that $A=USV^T$ which can be illustrated. The SVD matrix shows where U has orthogonal, unit-length column ($U^TU=I$ and it is called right singular vectors, unit-length column ($V^TV=I$) and ∑ is diagonal matrix (k x k) of singular values, where k is the rank of A($\leq$min(t,d). generally, $A=U\sum V^T$ matrix must all be the full rank the amount of dimension reduction, k need to choice correctly in order to represent the real structure in the data.

Principal Component Analysis PCA

Principal component analysis is a method to find k "principal axes" which are othonkormal coordinate systems that can capture most of the variance in data. Basically, PCA is formed from singular Vector Decomposition (SVD) on the covariance matrix which used Eigen vector or value of covariance matrix.

3. Performance Evaluation

In order to know the performance for quality of clustering, there are two measurements which are F-measure and entropy [17]. This basic idea is from information retrieval concept. In this technique, each cluster is considered as if it were the result of query and each class as if it were the desired set of documents for the query. Furthermore, the formulation of F-measure involves Precision and recall for each cluster j and class i are as follows:

$$Recall(i,j) = \frac{n_{ij}}{n_i} ; Precision(i,j) = \frac{n_{ij}}{n_j}$$

Where n_{ij} is the number of documents with class label *i* in cluster *j*, n_i is the number of documents with class label *I* and n_j is the number of documents in cluster *j*. Thus, the F-measure cluster *j* and class *i* is obtained as this below equation:

$$F(i,j) = \frac{(2*Recall(i,j)*Precision(i,j))}{Recall(i,j)+ Precision(i,j)}$$

The higher f-measure is the higher accuracy of cluster, includes precision and recall. Another measurement which related to the internal quality of clustering is entropy measurement (and it can be formulated:

$$E_j = -\sum_i P(i,j).\log P(i,j)$$

Where, *P (i, j)* is probability that a document has class label *I* and is assigned to cluster *j*. Thus, the total entropy of clusters is obtained by summing the entropies of each cluster weighted by the size of each cluster:

$$E = \sum_j \frac{n_j}{n} E_j$$

Where, *nj* is size of cluster *j* and n is total document number in the corpus. The lower value of entropy, the higher quality of cluster internally.

3.1 Level-of-Detail Correlation Exploration

The level-of-detail exploration of correlation clusters with the hierarchical quality threshold algorithm. Samples that are not in the current level being explored can be either hidden or de-emphasized respectively. Parallel coordinates show the correlation relation quantitatively. In our case, the number of axes in a level equals the number of samples. The thickness of each axis is in proportion to the number of samples it contains in the next level, which provides hint for user interaction. The user can simply click on an axis to see the detail or double click to return. For each level in the parallel coordinates, we sort the axes by their similarity so that sample correlation patterns can be better perceived. The samples along the path from the root to the current level are highlighted in white and green in the volume and parallel coordinates views, respectively. By linking the parallel coordinates view with the volume view, we enable the user to explore the hierarchical clustering results in a controllable and coordinated fashion.

4. Conclusion

The document clustering can be applied using concept space and cosine similarity. In this paper found that except for the Euclidean distance measure, the other measures have comparable effectiveness for the partitioned text document clustering task. Pearson correlation coefficient and the averaged measures are slightly better in that their resulting clustering solutions are more balanced and have a closer match with the manually created category structure. Meanwhile, the Jaccard and Pearson coefficient measures find more coherent clusters. Despite of the above differences, these measures' overall performance is similar. Considering the type of cluster analysis involved in this study, which is partitioned and require a similarity or distance measure, we can see that there are three components that affect the final results—representation of the objects, distance or similarity measures, and the clustering algorithm itself.

5. References

[1] D. Arthur and S. Vassilvitskii. K-means++ the advantages of careful seeding. In Symposium on Discrete Algorithms, 2007.
[2] M. Craven, D. DiPasquo, A. McCallum, T. Mitchell, K. Nigam, and S. Slattery. Learning to extract symbolic knowledge from the World Wide Web. In AAAI-98, 1998.
[3] D. R. Cutting, J. O. Pedersen, D. R. Karger, and J. W. Tukey. Scatter/gather: A cluster-based approach to browsing large document collections. In Proceedings of the ACM SIGIR, 1992.
[4] M. Ester, H.-P. Kriegel, J. Sander, and X. Xu. A density-based algorithm for discovering clusters in large spatial databases with noise. In Proceedings of 2nd International Conference on KDD, 1996.
[5] N. Friburger and D. Maurel. Textual similarity based on proper names. In Proceedings of Workshop on Mathematical Formal Methods in Information Retrieval at 25th ACM SIGIR Conference, 2002.
[6] E. Han, D. Boley, M. Gini, R. Gross, K. Hastings, G. Karypis, V. Kumar, B. Mobasher, and J. Moore. Webace: A web agent for document categorization and exploration. In Proceedings of the 2nd International Conference on Autonomous Agents. 1998.
[7] A. Hotho, S. Staab, and G. Stumme. Wordnet improves text document clustering. In Proceedings of the SIGIR Semantic Web Workshop, Toronto,, 2003.
[8] A. K. Jain, M. N. Murty, and P. J. Flynn. Data clustering: a review. ACM Computing Surveys (CSUR), 31(3):264–323, 1999.
[9] B. Larsen and C. Aone. Fast and effective text mining using linear-time document clustering. In Proceedings of the Fifth ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 1999.
[10] D. D. Lewis. Reuters-21578 text categorization test collection distribution 1.0. http://www.research.att.com/ lewis, 1999.
[11] J. Lin. Divergence measures based on the Shannon entropy. IEEE Transaction on Information Theory, 37(1):145–151, 1991.
[12] D. Milne, O. Medelyan, and I. H. Witten. Mining domain-specific thesauri from wikipedia: A case study. In Proc. of the International Conference on Web Intelligence (IEEE/WIC/ACM WI'2006), 2006.
[13] J. M. Neuhaus and J. D. Kalbfleisch. Between- and within-cluster covariate effects in the analysis of clustered data. Biometrics, 54(2):638–645, Jun. 1998.
[14] M. F. Porter. An algorithm for suffix stripping. Program, 14(3):130–137, 1980.
[15] G. Salton. Automatic Text Processing. Addison-Wesley, New York, 1989.

[16] M. Steinbach, G. Karypis, and V. Kumar. A comparison of document clustering techniques. In KDD Workshop on Text Mining, 2000.
[17] A. Strehl, J. Ghosh, and R. Mooney. Impact of similarity measures on web-page clustering. In AAAI-2000: Workshop on Artificial Intelligence for Web Search, July 2000.
[18] N. Z. Tishby, F. Pereira, and W. Bialek. The information bottleneck method. In Proceedings of the 37th Allerton Conference on Communication, Control and Computing, 1999.
[19] E. Voorhees and D. Harman. Overview of the fifth text retrieval conference (trec-5). In Proc. of the Fifth Text REtrieval Conference (TREC-5), 1998.
[20] P. Willett. Recent trends in hierarchic document clustering: a critical review. Information Processing and Management: an International Journal, 24(5):577–597, 1988.
[21]. Pang-Ning Tan, M.S., Vipin Kumar, *Introduction to Data Mining*. Pearson International ed. 2006: Pearson Education, Inc.
[22]. M.A. Hearst, a.J.O.P. *Re-examining the cluster hypothesis*. 1996: In Proceeding of SIGIR '96.
[23]. Jardine, N.a.v.R., C.J., *The Use of Hierarchical Clustering in Information Retrieval*. Information Storage and Retrieval. Vol. 7. 1971.
[24]. Steinbach M., K.G., Kumar V., *A Comparison of Document Clustering Techniques*. 2000, University of Mineasota.
[25]. Saveresi, S.M., D.L. Boley, S.Bittanti and G. Gazzaniga, *Cluster Selection in Divisive Clustering Algorithms.* 2002.
[26]. Larose, D.T., *An Introduction to Data Mining*. Discovering Knowledge in Data. 2005: Willi & Sons, Inc.
[27]. El-Sonbaty, Y.a.I., M.A., *Fuzzy Clustering for Symbol Data*. IEEE Transactions on Fuzzy Systems, 1998.
[28]. Rodrigues, M.E.S.M.a.S., L. *A Scalable Hierarchical Fuzzy Clustering Algorithm for Text Mining*. in *The 5th International Conference on Recent Advances in Soft Computing*. 2004.
[29]. Aberer, K., *EPFL-SSC*, L.d.s.d.i. repartis, Editor. 2003.
[30]. S. Deerwester, e.a., *Indexing by latent semantic analysis.* Journal of American Society for Information Science and Technology, 1990. **41**: p. 391-407. 11. Smith, L., *A Tutorial*