# Developing a Clustering Categorization Approach for Tigrigna Text

Gebrehiwot Assefa Berhe(MSc.)

School of Computing, EiT-M, Mekelle University

Mekelle, Ethiopia.

Abstract—***Tigrigna language is a Semitic language spoken by the Tigray people in Northern Ethiopia and Eritrea which has more than six million speakers worldwide. Even though the amount of the document increase, there are challenging tasks to identify the relevant documents related to a specific topic. So, a mechanism is required for finding, filtering and managing the rapid growth of online information.***

***Hence, this study attempts to design a text categorization system. Here, clustering is used to find natural grouping of the unlabeled Tigrigna text documents. As a result, repeated bisection and direct k-means clustering algorithms are used to obtain documents of natural group of the Tigrigna data set. The repeated bisection clustering algorithm outperforms the direct k-means clustering algorithms.***

***Keywords—categorization, text , clustering,algorithm, data set,k-means, repeated bisection***

## I. BACKGROUND OF THE STUDY

The globalization era provides a growing amount of information and data coming from different sources.. As a result, several online services have been proposed to find and organize valuable information needed by the target users [1]. However, these services are not capable to fully address the users' interest. So, a mechanism is required for finding, filtering and managing the rapid growth of online information. This mechanism is called text categorization [9]. Text categorization (TC) can be defined as the task of determining and assigning topical labels to content [1]. While the text categorization task based on automatically identified categories is called text clustering [2].

Text clustering is used to assign some similar properties of text documents into automatically created groups. It is used to improve the efficiency and effectiveness of text categorization system such as time, space, and quality. The standard text clustering algorithms can be categorized into partitioning and hierarchical clustering algorithms [10]. Partitioning clustering algorithm splits the data points into k partition where each partition represents a cluster. Whereas hierarchical clustering algorithm groups data objects to form a tree shaped structure. It can be bottom up approach which each data points are considered to be a separate cluster and clusters are merged based on a criteria or top down approach where all data points are considered as a single cluster and they are splited into number of clusters based on certain criteria. [6] has compared partitioning and hierarchical methods of text clustering on a broad variety of test data set. It concludes that k-means of the partition clustering algorithm clearly outperforms the hierarchical methods with respect to clustering quality. In addition, a variant of k-means called repeated bisection k-means is introduced and yields even better performance.

Even though more and more organizations are automating all their activities using different text categorization approaches, a number of challenges are remained for text categorization research. Labeling the training document requires a large amount of time and costs since most of the documents in the real word are available in unlabeled format. As a result, unsupervised method has been proposed by[11] that can learn from unlabeled documents.

Tigrigna language is a Semitic language widely spoken in the Tigray region of Ethiopia and Eritrea. Tigrigna is spoken by large immigrant communities around the world, including Sudan, Saudi Arabia, the United States, Germany, Italy, the United

Kingdom, Canada and Sweden. Even though the amount of the document increase, there are challenging tasks to identify the relevant documents related to a specific topic [8]. Hence, the process of searching relevant documents from large collection becomes too expensive as it has to search every document in the entire collection. Sometimes relevant key words are used in irrelevant document and often omitted in relevant documents because the context is not clear to the target audience [4]. In such situations, making the search easier, fast and efficient is quiet difficult for Tigrigna documents.

Since there are a number of documents produced by different subject domains in the form of articles, research results, and reports that are electronically available, there is a need to design text categorizer for efficient search and retrieval. Hence, the present study aims to develop Tigrigna text documents categorization system using clustering approach.

## II. OBJECTIVE

The general objective of this research is to design a text categorization system for Tigrigna text documents using clustering approach. In order to achieve the above stated general objective, the following specific objectives are formulated.

- To conduct a thorough review of literature on the existing text clustering techniques and methods .
- To prepare the appropriate data sets from different documents for training and testing purposes.
- To preprocess the data in order to select discriminating terms of the Tigrigna text documents.
- To cluster the preprocessed Tigrigna text documents in to their natural groups (categories) which can be used for text classifier learning.
- To evaluate the performance of the proposed model using the selected evaluation metrics

## III. METHODOLOGY

Methodology provides an understanding of how a proposed research is conducted in order to obtain information for developing the proposed systems [3]. So the methodology contains tools and techniques that can be used for conducting the study. As a result, the present study uses the following important aspects for constructing a Tigrigna text categorization system.

In order to get a good understanding of text categorization and the Tigrigna language relevant published text documents are reviewed. Different books, journal articles, news, previous related research works and electronic publication on the web have been consulted in order to design an effective Tigrigna text categorization system.

The data sets are collected from different books, articles, journals etc. These data sets collected are in word document format. The format of data sets is converted in to text for preprocessing. Then the data was preprocessed through two phases. In the first phase, python programming language is used to remove extraneous characters from the collection. Python is used because the researcher is familiar with the programming language and it is an effective tool for text processing. In the second phase, the preprocessing is done using python and Perl programming. Like python, Perl is also used based on its familiarity with the researcher. The researcher uses python programming to remove the irrelevant term of Tigrigna language from the text and the Perl programming to create a document term matrix that is used for the clustering purpose.

The main tasks performed for developing Tigrigna text document categorization are clustering and classification using gCluto , which is the most freely available tools of text categorization. gCluto is selected for clustering Tigrigna text documents because it has an intuitive graphical user interface, and interactive visualizations of the clustering solutions [6].

## IV. PREPROCESSING TIGRIGNA DOCUMENTS

In order to categorize the text documents by applying clustering and classifier algorithms, documents preprocessing is required to make ready the data set for training and testing. In this study, the Tigrigna text documents are preprocessed before using them for categorization purposes. Text documents are tokenized in to word-level data set that can be analyzed by a Machine Learning algorithm.

In tokenization stage, the Tigrigna texts are partitioned into discrete units. These units contain a list of words in the text. According to [7], a word in Tigrigna language is a combination of two or more than two letter of Tigrigna word that has a meaning. However, it is not sufficient to split the Tigrigna text into tokens (words) for tokenization purposes because there are punctuation marks and digits embedded with each of the word. Hence, the punctuation marks and digits are tokenized from the Tigrigna corpus since they do not carry any information to describe the content.

After tokenizing, a bag-of-words are identified. However, there are terms of the same root written in different forms due to their grammatical use. In Tigrigna language one term has suffix, prefix and infix. These extra characters added to form word variants are stemmed from the document corpus using the stemming algorithm developed by [5]. Though added extra characters are stemmed to come with the root of a word, there are also challenges because of affixes that change the structure of the term. For instance, words "□□□□□", "□□□□", "□□□□" and "□□□□" are the affixes of the word "□□□" (succede). The word "□□□□□" has both prefix ("□") and suffix ("□") terms and it is stemmed in to "□□□". The second word "□□□□" has only prefix terms "□" and stemmed in to "□□□". Both "□□□□" and "□□□□" have "□" prefix terms and stemmed in to "□□□" and "□□□" terms. "□□□", "□□□", "□□□" and "□□□" require another morphological analysis of the terms in order to stem them in to their same root term "□□□". Such problems need a better stemming algorithm that considers the various morphological structure of the Tigrigna language.

The terms that do not discriminate one document from other documents which are called stop words are identified after stemming words in to their root. Stop words are words that frequently appear in nearly all the documents. In the current study, stop word list of common Tigrigna words are prepared and used for comparison. As a result, the term appears in the stop word lists are removed from the Tigrigna documents using python programming language.

Even though the words are stemmed to their root and stop word are removed from the documents, the Tigrigna text documents are in high dimensionality. In the current research, rarely appearing terms make the document to have a high dimensions. So, in this research words that appear in three or fewer than that are considered as rare words and hence removed them from terms that are used for document representation. Finally, term-document matrix of the relevant terms and their tf×idf with reduced dimension from 2623 size feature to 1200 size feature is created for all the Tigrigna documents which is used for text clustering and classification.

## V. CLUSTERING

Clustering becomes crucial in text categorization when document category is not known. This study also uses clustering as a complementary step to text classification. Since Tigrigna documents used in this study are not labeled. In order to make the data ready for clustering task, the researcher constructs the document term matrix. The rows represent the list of documents and the columns represent the list of terms. The value in the ith row and jth column represents the tf×idf of the term in a given documents.

The input file formats in gCLuto are sparse matrix format where the first line contains information about the size of the matrix, and the remaining N lines contain information for each row. The information for each row contains the position number of the term and its tf×idf in the given documents. As shown in figure 1, the first line of the input matrix file contains three numbers describe the number of rows in the matrix (n) (i.e. the number of documents in the given data set), the number of columns in the matrix (m) (i.e. the number of unique terms), and the total number of non-zero entries in the n×m matrix respectively. Next lines show the position number of the term in the document and their tf×idf value in each documents rounded to two digits.

```
1462  1200  83242
1 1.52 2 1.63 3 0.87 4 2.09 5 1.60 6 0.42 7 1.84 8 9.21 9 2.34 10 1.41 11 2.12 12 1.84 13
1.99 14 4.40 15 1.59 16 0.64 17 2.20 18 1.50 19 1.08 20 0.79 21 2.43 22 1.12 23 1.37 24
1.52 25 1.63 26 6.94 27 0.91 28 1.67 29 0.54 30 2.04 31 1.35 32 1.20 33 1.91 34 1.47 35
0.88 36 1.82 37 1.25
3 0.87 38 1.27 39 1.62 40 2.25 41 3.61 42 0.63 43 1.22 44 2.43 45 6.81 46 2.97
47 1.67 3 0.87 6 0.42 48 1.59 49 1.10 50 1.82 51 1.23 52 1.31 16 0.64 53 1.91 54 1.86 55
0.28 23 1.37 56 1.30 26 2.31 57 1.77 37 1.25
58 1.80 59 0.71 60 1.14 61 2.12 62 1.52 63 0.94 64 1.09 65 2.77 26 3.47 29 0.54 30 2.04
31 1.35 66 2.05 67 2.27 57 1.77
68 1.50 69 0.74 3 1.75 6 0.42 70 1.37 51 1.23 71 1.50 16 0.64 72 1.89 53 1.91 73 1.51 74
0.90 75 1.34 26 1.16 28 0.83 31 1.35 76 1.78
.  .  .
```

Figure 1: Input file format for gCluto clustering tools

The gCluto clustering tool supports a number of clustering algorithms, including repeated bisection, and direct k-means which are used in the present study. The results for each of the clustering algorithms are presented in the experiment. The clustering algorithms cluster the Tigrigna text documents in to seven, eight, nine, ten, eleven and twelve subject categories. The performance of each subject category is measured. The eight subject categories result high purity and low entropy. So the researcher uses the eight subject categories for this study. The performance of the given algorithm is inversely proportional with entropy and directly proportional with purity. The clustering results of the clustering algorithms are also analyzed using mountain visualization[12].

The results for repeated bisection clustering algorithms are presented in table 1. This result contains statistics about the discovered clusters.

| Cluster | Size | ISim | ISdev | ESim | ESdev | Entropy | Purity |
|---|---|---|---|---|---|---|---|
| 0 | 86 | 0.535 | 0.156 | 0.015 | 0.003 | 0.053 | 0.977 |
| 1 | 111 | 0.107 | 0.041 | 0.011 | 0.003 | 0.412 | 0.703 |
| 2 | 118 | 0.076 | 0.028 | 0.013 | 0.004 | 0.431 | 0.763 |
| 3 | 167 | 0.050 | 0.019 | 0.009 | 0.004 | 0.713 | 0.311 |
| 4 | 196 | 0.050 | 0.017 | 0.011 | 0.004 | 0.441 | 0.755 |
| 5 | 243 | 0.049 | 0.019 | 0.011 | 0.004 | 0.595 | 0.638 |
| 6 | 327 | 0.043 | 0.017 | 0.014 | 0.006 | 0.761 | 0.398 |
| 7 | 214 | 0.032 | 0.011 | 0.012 | 0.004 | 0.819 | 0.322 |

Table 1: Experimental results using repeated bisection clustering algorithm

The clustering statistics shows that cluster “7” and cluster “3” are less accurate than the other cluster because they have high entropy and less purity as shown in table 1 while cluster “0” has highest accuracy than the other clusters because it has low entropy (i.e. 0.053) and high purity (i.e. 0.977). As a result, the repeated bisection clustering algorithm gives a better performance of the cluster “0”, followed by cluster “2”, “4”, “1”, and ”5” while cluster ”6”,”7”,and “3” have least performance in repeated bisection clustering algorithm as elaborated in table 1. Results are also visualized using mountain visualization in figure 2 below.

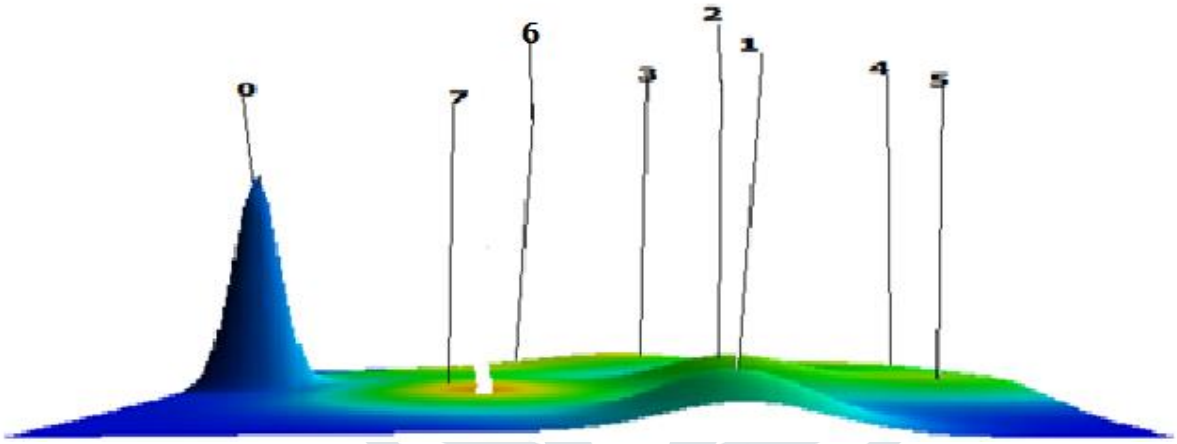


Figure 2: Repeated bisection clustering algorithm results visualization

As shown in figure 2, each peak represents a single cluster in the clustering. The height of each peak on the plane of clustering result is proportional to the internal similarity of the corresponding cluster. As shown in table 1, the internal similarity ”ISim” elaborates that cluster “0” with 0.535 “ISim” has the highest similarity but cluster “7” with 0.032 “ISim” has the least internal similarity than other clusters. As a result, the height of peak in figure 2 also depicts accordingly that cluster “0”, cluster “1” has the highest internal similarity than other clusters whereas cluster “7” has the least internal similarity. This indicates that the documents clustered within the cluster “0” are more similar than the documents clustered within cluster “7”.

The direct (basic) k-means clustering algorithm results are also presented as shown in table 2 below.

| Cluster | Size | ISim | ISdev | ESim | ESdev | Entropy | Purity |
|---|---|---|---|---|---|---|---|
| 0 | 91 | 0.486 | 0.173 | 0.014 | 0.003 | 0.051 | 0.978 |
| 1 | 111 | 0.091 | 0.039 | 0.011 | 0.004 | 0.353 | 0.793 |
| 2 | 114 | 0.076 | 0.022 | 0.011 | 0.005 | 0.649 | 0.544 |
| 3 | 197 | 0.056 | 0.022 | 0.014 | 0.005 | 0.597 | 0.442 |
| 4 | 152 | 0.053 | 0.033 | 0.012 | 0.004 | 0.804 | 0.303 |
| 5 | 211 | 0.041 | 0.014 | 0.011 | 0.004 | 0.824 | 0.365 |
| 6 | 282 | 0.040 | 0.013 | 0.011 | 0.004 | 0.489 | 0.727 |
| 7 | 304 | 0.041 | 0.017 | 0.014 | 0.007 | 0.802 | 0.329 |

Table 1: Experimental results using direct k-means clustering algorithm

The direct k-means clustering algorithm statistics in table 2 shows that the cluster "0" is clustered more accurately than other cluster categories because it has low entropy (i.e. 0.051) and high purity (i.e. 0.978). As shown in the above table 2, cluster "4" is clustered less accurately than other clusters because it has low purity (i.e. 0.303). As a result, the direct k-means clustering algorithm gives a better performance of the cluster "0" followed by cluster "1", "6", "2" and "3" where cluster "4", "5", and "7" have lower performances than other categories.

Like the repeated bisection clustering algorithm, direct k-means clustering algorithm results are also visualized using the mountain visualization as shown in figure 3 below. As a result, each cluster is represented in the figure by certain peak of the figure. As indicated in table 2, the cluster "0" documents has more internal similarity than other cluster documents because the cluster "0" has high internal similarity shown in the column "ISim" which is 0.486 where the cluster "5", "6" and "7" have low internal similarity. Their internal similarities are 0.041, 0.040 and 0.041 respectively. As a result, the peak for cluster "0" is high and for cluster "5", "6" and "7" are very low.

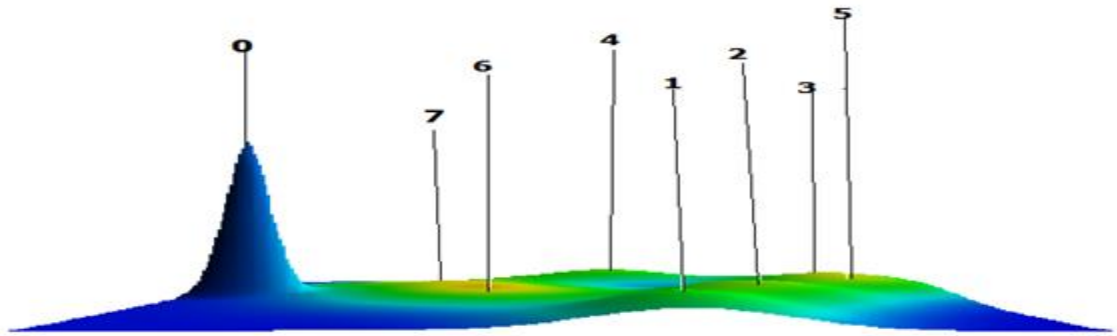


Figure 3: Direct k-means clustering algorithm results visualization

## VI. DISCUSSION

Both repeated bisection and direct k-means clustering have high performance in cluster "0" and low performance in cluster "7". The cluster "0" has deqeq ("□□□"), naaxte ("□□□□"), gdi ("□□"), and kala (□□□) as descriptive and discriminating terms as shown in table 3. So cluster "0" documents illustrate about small enterprise, trade and company because these terms are the descriptive and discriminating terms of the cluster "0" as shown in table 3. These are similar topic features. Hence, the cluster "0" instances are clustered more accurately than other clusters as shown in table 3.

| Cluster | | Feature | % | Feature | % | Feature | % | Feature | % |
|---|---|---|---|---|---|---|---|---|---|
| 0 | Descriptive | Deqeq (□□□) | 31.1% | naaxte (□□□□) | 22.3% | gdi (□□) | 11.6% | kala (□□□) | 3.2% |
| | Descriminating | Deqeq (□□□) | 18.2% | naaxte (□□□□) | 12.6% | gdi (□□) | 5.8% | kala (□□□) | 1.7% |
| 1 | Descriptive | Hmam (□□□) | 12.0% | Easo (□□) | 9.8% | aiecyvi (□□□□□) | 9.4% | TEna (□□□) | 7.5% |
| | Descriminating | Hmam (□□□) | 7.4% | Easo (□□) | 6.3% | aiecyvi (□□□□□) | 6.1% | TEna (□□□) | 4.7% |
| 2 | Descriptive | tmhr (□□□□) | 31.9% | tmharo (□□□□) | 9.4% | memhra (□□□□) | 5.2% | yuniversi (□□□□□) | 4.6% |
| | Descriminating | tmhr (□□□□) | 22.1% | tmharo (□□□□) | 6.3% | memhra (□□□□) | 3.6% | yuniversi (□□□□□) | 3.3% |
| 3 | Descriptive | Hadega (□□□) | 17.0% | meTQaE (□□□□□) | 5.4% | aaxeberti (□□□□□) | 4.7% | suda (□□) | 3.6% |
| | Descriminating | Hadega (□□□) | 11.5% | meTQaE (□□□□□) | 3.7% | aaxeberti (□□□□□) | 3.2% | suda (□□) | 2.5% |
| 4 | Descriptive | mesno (□□□) | 8.6% | hiektar (□□□□) | 7.6% | kuntal (□□□□) | 6.8% | hier(□□) | 3.5% |
| | Descriminating | mesno (□□□) | 6.1% | hiektar (□□□□) | 5.6% | kuntal (□□□□) | 5.0% | deqeq (□□□) | 2.9% |
| 5 | Descriptive | priezidan (□□□□□) | 4.9% | parlama (□□□□) | 4.4% | mereSa (□□□) | 4.2% | poletika (□□□□) | 3.8% |
| | Descriminating | Priezidan (□□□□□) | 3.4% | parlama (□□□□) | 3.3% | mereSa (□□□) | 3.2% | poletika (□□□□) | 2.9% |
| 6 | Descriptive | frdi (□□□) | 11.7% | ftHi (□□□) | 6.2% | Hgi (□□) | 3.6% | TrEa (□□□) | 3.6% |
| | Descriminating | frdi (□□□) | 9.7% | ftHi (□□□) | 5.2% | Hgi (□□) | 3.1% | TrEa (□□□) | 3.0% |
| 7 | Descriptive | Sde(□□) | 13.6% | Hbura (□□□) | 9.1% | Aiertra (□□□□) | 9.0% | suda (□□) | 6.6% |
| | Descriminating | sde(□□) | 10.5% | aiertra (□□□□) | 7.1% | Hbura (□□□) | 6.4% | suda (□□) | 5.1% |

Table 2: Descriptive & Discriminating features of clustering algorithm

On the other hand, cluster "7" has sde (□□), Hbura (□□□), suda (□□), and aiertra (□□□□) as descriptive and discriminating terms of this cluster as shown in table 3. So the cluster "7" documents elaborate about refugee, united nations, Sudan and Eritrea because these terms are the descriptive and discriminating terms of the cluster "7" as shown in table 3. The documents in cluster "7" discuss the political disorder of Eritrea and Sudan which leads their citizen to migration to other country. Besides, they also elaborate the help from united nation to the refugee of Eritrea and Sudan. As a result, these documents in cluster "7" discuss different topics: the political disorder of the two countries, the migration of their citizen and social aid from United Nations to migrants. Due to this the clustering algorithm faces difficulty to find common word between these concepts. This decreases the performance of the clustering algorithm.

Based on table 1 and 2 results, the total purity and entropy of the clustering algorithm is computed, as depicted in table 4.

| **Clustering algorithm** | **Purity** | **Entropy** |
|---|---|---|
| Repeated bisection | 0.56 | 0.611 |
| Direct k-means | 0.516 | 0.624 |

Table 4: Comparison of direct k-means and repeated bisection clustering algorithms

As shown in table 4, repeated bisection clustering algorithm has higher purity and lower entropy than the direct k-means clustering algorithm. Hence, the repeated bisection clustering algorithm is has better performance.

## VII. REFERENCES

[1] Addis, A. , Study and Development of Novel Techniques for Hierarchical Text Categorization. University of Cagliari, Italy, 2010.

[2] Baker, D. and Kachites, A. ,Distributional Clustering of Words for Text Classification: ACM SIGIR, pp.96-102, 1998.

[3] Dawson, C. ,Practical Research Methods. New Delhi: UBS Publishers , 2002.

[4] Feng, Y. , Multi-Label Text Categorization Using K-Nearest Neighbor Approach with M-Similarity. Proceedings of the 12th International Conference on String Processing and Information Retrieval. Buenos Aires, Argentina: Springer, pp.3-12, 2005.

[5] Girma Berhe , A Stemming Algorithm Development for Tigrigna Language Text Documents. MSc Thesis. Addis Ababa University, Addis Ababa, Ethiopia, 2001.

[6] Karypis, G., Steinbach, M. and Kumar, V., A Comparison of Document Clustering Techniques. New York, USA: ACM Press/Addison-Wesley Publishing Co, 2004.

[7] Kassa Gebrehiwot, A Tigrigna Language Dictionary. Addis Ababa, Ethiopia: EMAY Printers, 2003.

[8] Leslau, W., Documents Tigrigna. Paris: Libraire CKlincksieck, 1998.

[9] Maron, M. and Kuhns, J. , Probabilistic Indexing and Information Retrieval. London: ACM, 23, pp.30-35, 1960.

[10] McCallum, A., Nigam, K., Thrun, S. and Mitchell, T., Text Classification from Labeled and Unlabeled Documents Using EM. Boston: Kluwer Academic Publishers, 39(2), pp.103–134 , 2000.

[11] Rafael, A., Li, X. and Lee, J., Managing Content with Automatic Document Classification. Journal of Digital Information, pp.23-40, 2004.

[12] Zhao, Y., Comparison of Agglomerative and Partitioning Document Clustering Algorithms. Washington DC: ACM Press, 2002.