Single Pass Algorithm for Text Clustering by Encoding Documents into Tables

Taeho Jo
IT Convergence, KAIST Institute
tjo@kaist.ac.kr, 82-42-869-4297

Abstract

This research proposes a modified version of single pass algorithm specialized for text clustering. Encoding documents into numerical vectors for using the traditional version of single pass algorithm causes the two main problems: huge dimensionality and sparse distribution. Therefore, in order to address the two problems, this research modifies the single pass algorithm into its version where documents are encoded into not numerical vectors but other forms. In the proposed version, documents are mapped into tables and the operation on two tables is defined for using the single pass algorithm. The goal of this research is to improve the performance of single pass algorithm for text clustering by modifying it into the specialized version.

1. Introduction

Text clustering refers to the process of segmenting a particular group of documents into subgroups each of which contains content-based similar documents. A collection or group of documents is given as the input of the task. Several smaller groups of content-based similar documents are generated from the task as its output. Although there are many heuristic approaches to the task, unsupervised learning algorithms have been used as state of the art approaches to it. As an instance of text mining, text clustering is necessary for organizing documents automatically.

The process of encoding documents into numerical vectors for using traditional unsupervised learning algorithms for text clustering causes the two main problems. The first problem is huge dimensionality where documents must be encoded into very large dimensional numerical vectors for preventing information loss. In general, documents must be encoded at least into several hundreds dimensional numerical vectors in previous literatures. This problem causes very expensive cost for processing each numerical vector representing a document in terms of time and system resources. Furthermore, much more training examples are required proportionally to the dimension for avoiding over-fitting.

The second problem is sparse distribution where each numerical vector has zero values dominantly. In other words, more than 90% of its elements are zero values in each numerical vector. This phenomenon degrades the discrimination among numerical vectors. This causes poor performance of text categorization or text clustering. In order to improve performance of both tasks, the two problems should be solved.

Figure 1 illustrates an original document or documents and its or their surrogate given as a table. The table consists of entries of words and their weights indicating their content based importance in the original document. This research adopts the strategy of encoding documents illustrated in figure 1 and applies single pass algorithm under the strategy. A semantic similarity between two documents is computed based on words shared by both tables. The computation will be described in detail in section 4.

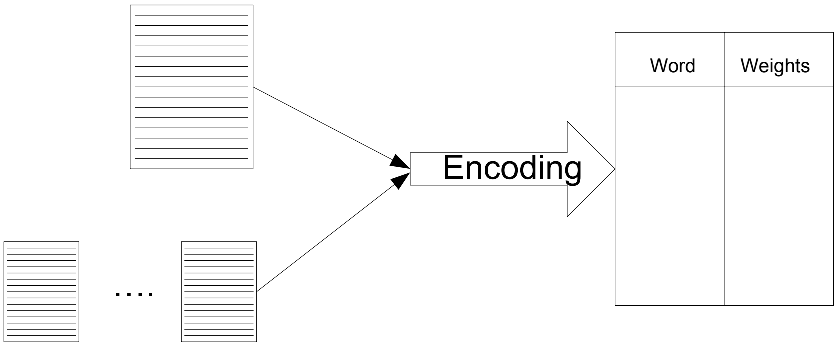

Figure 1. Original Document or Documents and its or their Table as a Surrogate

This research proposes another version of single pass algorithm where documents are encoded into tables. By doing that, it offers three contributions. The first contribution is to avoid the two problems, huge dimensionality and sparse distribution, by encoding documents into another form which is completely different from numerical vectors. The second contribution is to open a way of developing a new class of approaches to text clustering. The third contribution is to make it easy to generate symbolic clustering rules for tracing why a particular document should belong to a cluster, because the table is close to symbolic data rather than numerical data.

This paper consists of six sections, including this section. In section 2, we will explore previous approaches to text clustering and a previous solution to the two problems. In section 3 and 4, the process of encoding documents into tables and the proposed text clustering system are described in detail, respectively. In section 5, the traditional and proposed versions of single pass algorithm are compared with each other in terms of their clustering performance, in order to validate that the proposed version is more desirable. In section 6, the significance of this research and further research will be mentioned as the conclusion of this article.

2. Previous Works

This article concerns the exploration of previous research on text clustering. As mentioned in section 1, there exist various kinds of approaches to text clustering. However, in exploring previous research, we restrict the scope of approaches only to unsupervised learning algorithms. Among unsupervised learning algorithms, based on their popularities, we select only three representative ones: single pass algorithm, Kohonen Networks, and EM algorithm. In this section, we explore previous cases of using one of the three unsupervised learning algorithms.

A simple and popular clustering algorithm is single pass algorithm. When a number of clusters is far less than a number of objects, this algorithm runs in an almost linear complexity to the number of objects. The algorithm has been popularly used for clustering objects especially in industrial worlds, since it is fast enough to implement a real time clustering system. However, note that quality of clustering objects in this algorithm is not as good as that in other clustering algorithms. In 2000, Hatzivassiloglou et al used this algorithm as an approach to text clustering where documents are encoded into numerical vectors together with linguistic features and compared it with complete pair-wise algorithm [Hatzivassiloglou et al 2000].

Kohonen Networks is an unsupervised neural network and was used as a popular approach to text clustering [Kaski et al 1998][Kohonen et al 2000][Bote et al 2002]. WEBSOM was a typical text clustering system where Kohonen Networks was adopted as the approach to text clustering [Kaski et al 1998] [Kohonen, et al. 2000]. In 1998, its initial version was developed by Kaski et al in 1998 [Kaski et al 1998]. Each cluster of documents is identified with a group of relevant words. In the system, not only documents, but also words are clustered using Kohonen Networks.

K means algorithm is also a typical approach to not only text clustering but also any other pattern clustering. It is the simplest version of EM algorithm consisting of E-step and M-step [Mitchell 1997]. In 2000, Vinokourov and Girolami proposed five probabilistic models of hierarchical text clustering as specific versions of the EM algorithm [Vinokourov and Girolami 2000]. In 2003, Banerjee et al proposed two variants of the EM algorithm for soft clustering, where each object is allowed to belong to more than one cluster, and applied them to text clustering and gene expression clustering [Banerjee et al 2003].

When using one of the most three popular approaches, documents should be encoded into numerical vectors. Although a previous literature on text mining mentioned the two problems, it was regarded as natural and unavoidable task to encode documents so. However, this research attempts to find solutions to the two problems without accepting it naturally. The solution proposed in this research is to encode documents into another form. After doing that, this research modifies the single pass algorithm to be able to process the form of data.

3. Document Encoding

This section concerns the process of encoding a document or documents into a table. Figure 2 illustrates the process with three steps. A document or documents is given as input of the process, and a list of words and their frequencies is generated from the process. The three steps illustrated in figure 2 will be explained. After that, the three schemes of weighting words will be also mentioned.

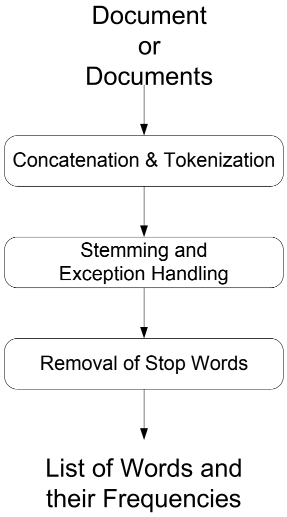

Figure 2. The Process of Mapping Document or Documents into a Table

As illustrated in figure 2, a document or documents may be given as input of this stage. If more than two documents are given as the input, their full texts are concatenated into an integrated full text. The integrated full text becomes the target for the tokenization. The full text is tokenized into tokens by a white space or a punctuation mark. Therefore, the output of this step is a list of tokens.

The next step to the concatenation & tokenization is the stemming & exception handling, as illustrated in figure 2. In this step, each token is converted into its root form. Before doing that, rules of stemming and exception handling are saved into a file. When the program encoding documents is activated, all rules are loaded into memory and the corresponding one of them is applied to each token. The output of this step is a list of tokens converted into their root forms.

The last step of extracting feature candidates from a corpus is to remove stop words as illustrated in figure 2. Here, stop words are defined as words which function only grammatically without their relevance to content of their document; articles (a an, or the), prepositions (in, on, into, or at), pronoun (he, she, I, or me), and conjunctions (and, or, but, and so on) belong to this kind of words. It is necessary to remove this kind of words for more efficient processing. After removing stop words, frequencies of remaining words are counted. Therefore, a list of the remaining words and their frequencies is generated as the final output from the stage illustrated in figure 1.

Although there are other schemes of weighting words, we will mention only three schemes as representative ones. For first, we can assign frequencies themselves to words as their weights. For second, we may assign normalized frequencies generated from dividing their frequency by the maximum frequency. For third, we can weights words using equation by equation (1),

$$weight_i(w_k) = tf_i(w_k)(\log_2 D - \log_2 df(w_k) + 1) \quad \mathbf{(1)}$$

where $weight_i(w_k)$ indicates a weight of the word, w_k, which indicates its content based importance in the document, i, $tf_i(w_k)$ indicates the frequency of the word, w_k in the document, i, $df(w_k)$ is the number of documents including the word, w_k, and D is the total number of documents in a given corpus. Among the three schemes, we adopt the third for weighting words in this research.

4. Proposed Text Clustering System

This section concerns the proposed version of single pass algorithm and the text clustering system which adopts the proposed version. Figure 3 illustrates the modules involved in implementing the proposed text clustering system. The first module is document encoder given as the interface of the system and encodes documents into tables. The second module is similarity measurer and computes a semantic similarity between two documents. The third module is document arranger and arranges documents into their content based corresponding clusters or creates a new cluster.

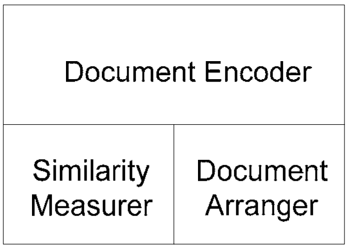

Figure 3. The Modules inolved in Implementing the Proposed Text Clustering System

Figure 4 illustrates the initialization of the single pass algorithm as its first step applicable to the first document. The initialization refers to the process of creating the first cluster and containing the first document in the cluster. The first document is given as the input of the step. The first document contained in the cluster becomes its prototype which represents it[1]. Therefore, from the initialization, a cluster with a document is generated as the output as illustrated in figure 4.

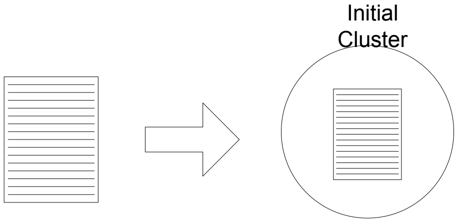

Figure 4. The Initialization of the Single Pass Algorithm

Figure 5 illustrates the process of generating a normalized value as a similarity between two documents. The role of document encoder was already mentioned above. The process of encoding documents into tables was already described in detail in section 3. The module, similarity measurer, computes a similarity between two tables based on words shared by both tables. The process illustrated in figure 5 generates a normalized real value as the output.

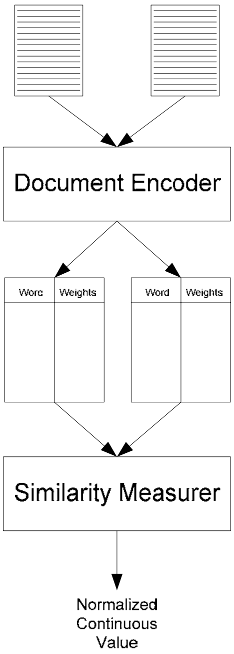

Figure 5. The Process of Generating a Similarity between Two Documents

[1] In other literatures on single pass algorithm, average over similarities of a document with contained ones in a cluster is used as a similarity between the document and the cluster. However, in this research, a similarity between a document and the first document in the cluster is used as the similarity between the document and the cluster for fast clustering.

Figure 6 illustrates the process of generating an output table from two input tables for computing a semantic similarity between two tables. Let the two tables be ‘Table 1’ and ‘Table 2’. By getting words shared by Table 1 and Table 2, the output table, Table 3, is built, and each word in Table 3 has its two weights: one from Table 1 and the other from Table 2. From the three table, we can define four sums of weights as follows.

- Sum_Weight 1: The sum of weights of words contained in Table 1
- Sum_Weight 2: The sum of weights of words contained in Table 2
- Sum_Weight 3: The sum of weights from table 1 of words contained in Table 3
- Sum_Weight 4: The sum of weights from table 2 of words contained in Table 3

Therefore, the similarity between Table 1 and Table 2 is computed using equation (4).

$$similarity = \frac{Sum\_Weight3 + Sum\_Weight4}{Sum\_Weight1 + Sum\_Weight2} \quad \textbf{(2)}$$

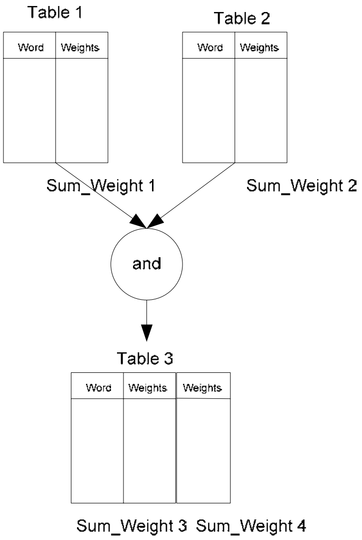

Figure 6. The Process of computing a Similarity between Two Tables

Figure 7 illustrates the process of arranging documents or creating one more cluster. The threshold of similarity is given as the parameter of the single pass algorithm. For each successive document, its similarities with prototypes of clusters are computed using equation (4). If its maximum similarity is less than the threshold, one more cluster is created and it is contained in the cluster. Otherwise, the document is arranged into the cluster corresponding to the maximum similarity.

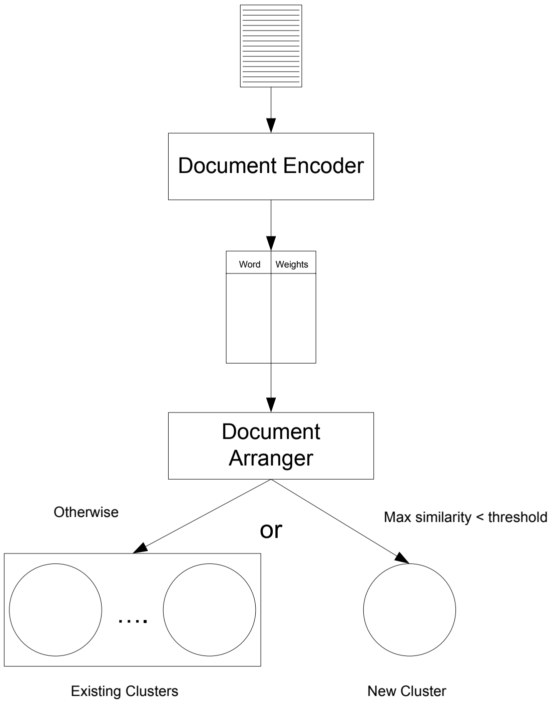

Figure 7. The Process of arranging Documents or creating one more Cluster

Table 1 summarized the difference and shared between the traditional and proposed versions of single pass algorithm. Basically, in the both versions, single pass algorithm consists of the two steps: initialization and arrangement. In the traditional version, documents are encoded into numerical vectors, while in the proposed one, they are encoded into tables. In the traditional version, a similarity between two documents is computed based on Euclidean distance or inner product between their corresponding numerical vectors, while in the proposed version, it is computed based on words shared by their two corresponding tables. Single pass algorithm is a very fast clustering algorithm, but its reliability is very poor since prototypes of clusters are fixed while clustering objects.

Table 1. The Different and Shared Points between Traditional and Proposed Version

	Traditional Version	Modified Version
Clustering Process	Initialization and Arrangement	
Encoding Documents	Numerical Vectors	Tables
Semantic Similarity	Inner Product Cosine Similarity Euclidean Distance	Equation (2)

5. Empirical Results

This article concerns two sets of experiments for comparing the two versions of single pass algorithm with each other in text clustering, and it consists of four sections. The first section describes the proposed measure for evaluating results of text clustering, and it was proposed by Jo and Lee in 2007 [Jo and Lee 2007]. The second section presents results of comparing the two versions on the test bed, NewsPage.com. The third section does those of doing that on one more test bed, 20NewsGroups. The last section visualizes the comparisons of the two versions as pie charts for the discussion on the results.

5.1. Evaluation Measure

In this section, we describe the measure which is called clustering index, for evaluating results of clustering objects. The proposed evaluation measure is targeted only for exclusive clustering, what is called hard clustering, where each object is allowed to belong to only one cluster. Therefore, we must use exclusively labeled objects as test bed for using the proposed evaluation measure. Two factors, intra-cluster similarity and inter-cluster similarity, are involved in the proposed measure. The former should be maximized and the latter should be minimized for the desirable clustering; both factors are given as normalized values between zero and one.

Until results of clustering documents are evaluated, their exclusive target labels are not opened. When we start to evaluate the results of clustering objects, the target labels are opened and similarities of objects are computed based on their target labels. Note that a similarity between two objects for clustering them is conceptually different from that for evaluating the results. A similarity between two documents denoted by d_i and d_j is given as a binary value one zero and one as expressed in equation (3),

$$sim(d_i,d_j)=\begin{cases}1 & \text{if } d_i,d_j \in c_t \\ 0 & \text{otherwise}\end{cases} \quad \textbf{(3).}$$

If two documents have their identical labels, their similarity becomes one, and otherwise, their similarity becomes zero.

The intra-cluster similarity indicates how much entities are similar with each other within a particular cluster; it means the cohesion of a particular cluster. For the desirable clustering, intra-cluster similarity should be maximized closely to one. A cluster c_k includes a series of documents and is denoted as a set of documents by $c_k=\{d_{k1},d_{k2},...,d_{k|c_k|}\}$. The intra-cluster similarity of the cluster, c_k, σ_k is computed using the equation (4),

$$\sigma_k = \frac{2}{|c_k|(|c_k|-1)}\sum_{i>j} sim(d_{ki},d_{kj})\ \textbf{(4),}$$

which indicates the average similarity of all pairs of different documents included in the cluster, c_k . If a series of clusters as the result of text clustering is denoted by $C = \{c_1, c_2, ..., c_{|C|}\}$, the average intra-cluster similarity, $\bar{\sigma}$ is computed using the equation (5),

$$\bar{\sigma} = \frac{1}{|C|}\sum_{k=1}^{|C|} \sigma_k\ \textbf{(5),}$$

by averaging the intra-cluster similarities of the given clusters.

The second factor involved in evaluating text clustering is the inter-cluster similarity. For a desirable clustering, this factor should be minimized closely to zero for reinforcing the discrimination among clusters. The inter-cluster similarity between two clusters, c_k and c_l, δ_{kl}, is computed using the equation (6),

$$\delta_{kl} = \frac{1}{|c_k||c_l|}\sum_{i=1}^{|c_k|}\sum_{j=1}^{|c_l|} sim(d_{ki},d_{lj})\ \textbf{(6),}$$

which indicates the average similarity of all possible pairs of two documents belonging to their different clusters. If the total number of clusters is denoted by $|C|$, the total number of pairs of clusters becomes $\frac{|C|(|C|-1)}{2}$. The average inter-cluster similarity $\bar{\delta}$ is computed using the equation (7),

$$\bar{\delta} = \frac{2}{|C|(|C|-1)}\sum_{k>l} \delta_{kl}\ \textbf{(7),}$$

by averaging all possible pairs of different clusters.

After calculating the two factors using equation (5) and (7), the clustering index is obtained by combining the two factors with each other, as expressed in equation(8),

$$CI = \frac{\bar{\sigma}^2}{\bar{\sigma}+\bar{\delta}}\ \textbf{(8).}$$

A value of the clustering index is given as a normalized continuous value between zero and one. If the intra-cluster similarity is close to one and the inter-cluster similarity is close to zero, the clustering index becomes close to one, according to equation (8). The value of clustering index which is close to one indicates the desirable results of clustering objects. The measure described in this section will be used for evaluating results of clustering documents.

5.2. NewsPage.com

This section concerns the set of experiments for comparing the two versions of single pass algorithm on the first test bed. The test bed used in this set of experiments is Newspage.com which is a collection of electronic news articles. The two versions of single pass algorithm are compared with each other; one is the version where documents are encoded into numerical vectors, and the other is the version where documents are encoded into tables. We use the measure described in the previous section for evaluating the performance of text clustering. The goal of this set of experiments is to observe the results of comparing the two versions on the test bed named as NewsPage.com.

Table 2 illustrates the number of news articles in each category in the first test bed, NewsPage.com. There are totally 1,200 news articles which are exclusively labeled with one of five categories: 'business', 'health', 'law', 'internet', and 'sports'. The source of this test bed is from the web site, www.newspage.com; the test bed is named after the URL address. We made the test bed as text files by copying and pasting full texts of news articles. In this test bed, each news article is given as an ASCII text file.

Table 2. NewsPage.com

Category Name	#Document
Business	400
Health	200
Law	100
Internet	300
Sports	200
Total	1200

Table 3 illustrates the five subgroups of news articles of this test bed for evaluating approaches to text clustering. Each subgroup consists of 500 news articles (100 news articles per category). A file name of each ASCII text file consists of its category name and a sequential number. For example, if a particular news article belongs to the category, 'health' and its sequential number five, its ASCII file name is assigned to the news article as 'health005'. As shown in table 3, five subgroups are exclusive with each other.

Table 3. Five Sub-collections of NewsPage.com

Category Name	Subgroup 1	Subgroup 2	Subgroup 3	Subgroup 4	Subgroup 5
Business	100 1 ~ 100	100 51 ~ 150	100 101 ~ 200	100 151 ~ 250	100 201 ~ 300
Health	100 1 ~ 100	100 26 ~ 125	100 51 ~ 150	100 76 ~ 175	100 101 ~ 200
Law	100 1 ~ 100	100 1 ~ 100	100 1 ~ 100	100 1 ~ 100	100 1 ~ 100
Internet	100 1 ~ 100	100 51 ~ 150	100 101 ~ 200	100 151 ~ 250	100 201 ~ 300
Sports	100 1 ~ 100	100 26 ~ 125	100 51 ~ 150	100 76 ~ 175	100 101 ~ 200
Total	500	500	500	500	500

Differently from k means algorithm and Kohonen Networks, the similarity threshold is given as the parameter of the single pass algorithm, instead of the number of clusters. In the clustering algorithm, the number and the size of clusters are determined automatically, depending on the similarity threshold. The parameter is given as a continuous normalized value between zero and one, and if it is close to zero, the small number of large clusters is resulted in. If it is close to one, the large number of small clusters is resulted in. Since the test bed has a small number of target categories, the parameter is set as 10^{-6} close to zero.

Table 4 illustrates the three groups by input size, and within each group the two versions of single pass algorithm are compared with each other. In the first group, documents are encoded into 100 dimensional numerical vectors in the traditional version, and they are encoded into 10 entries tables in the proposed version. In the second group, they are encoded into 250 dimensional numerical vectors in the traditional version, while they are encoded into 25 entries tables in the proposed version. In the third group, they are encoded into 500 dimensional numerical vectors and 50 entries tables in the traditional and proposed version, respectively. Note that we set the dimensions of numerical vectors based on previous dimensions in the previous literatures.

Table 4. Input Sizes for Comparison of the two Versions of Single Pass Algorithm

Groups of Input Sizes	Traditional	Proposed
Small Input Sizes	100 dimensional numerical vectors	10 entries table
Medium Input Sizes	250 dimensional numerical vectors	25 entries table
Large Input Sizes	500 dimensional numerical vectors	50 entries table

Figure 8 illustrates the results of comparing the two versions of the single pass algorithm on this test bed. The x-axis of figure 1 contains the three groups of bars by input size as shown in table 3. Within each group, the black bar indicates the performance of the previous version, while the white bar does that of the proposed version. The y-axis indicates the logarithmic clustering index computed by equation (9),

$$\frac{1}{-\log_{10} CI} \quad (9)$$

where the base of the logarithm is ten. The reason of rescaling the clustering index logarithmically is that the performance difference between the two versions in the original scale is too big to display with a bar-graph.

The difference between the two versions is outstanding by the only logarithmic scale. In the original scale, the proposed version is better 1000 times than the traditional version. In the logarithmic scale, the proposed version is better almost three times than the traditional version. In the traditional version, its clustering performances are not influenced by the dimension of numerical vectors; they are almost identical as illustrated in figure 1. In the proposed version, its performance is highest when documents are encoded into 50 entries tables.

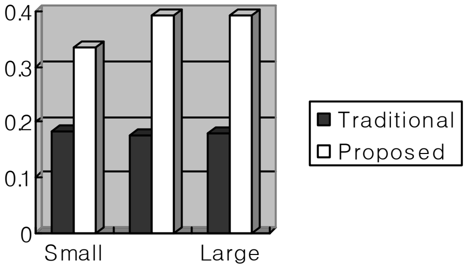

Figure 8. The Results of Comparing Two Versions on Subgroups of NewsPage.com

5.3. 20NewsGroup

This subsection concerns another set of experiments for comparing the two versions of single pass algorithm one more time. The test bed used in this set of experiments is another collection of electronic news articles called 20NewsGroups. In this set of experiments, documents are encoded according to the rules illustrated in table 3. The parameter of both versions of single pass algorithm is set identically to that in the previous set of experiments. The goal of this set of experiments is to observe the comparison of two versions one more time on another test bed.

The second test bed is 20NewsGroup which is a large collection of news articles. The collection consists of 20 categories and 20,000 news articles. The test bed was obtained by downloading the collection from the web site, kdd.ics.uci.edu as an integrated compressed file. Each news article is exclusively labeled with one of the twenty categories. This fact is the reason for adopting the collection as the test bed for evaluating the performance of text clustering, instead of the most standard test bed, Reuter21578.

Ten subgroups are built from the test bed for evaluating the approaches to text clustering. The twenty predefined categories are grouped into two groups of ten categories. Each subgroup consists of ten categories and 500 news articles; each category in the subgroup contains 50 news articles. Among ten subgroups of documents, five subgroups span over ten categories of twenty categories. The other five subgroups span over the other ten categories.

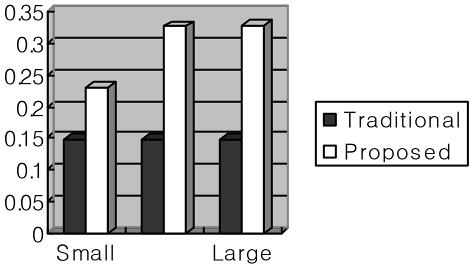

Figure 9. The Results of Comparing Two Versions on Subgroups of 20NewsGroups

Figure 9 illustrates the results of comparing the two versions on this test bed. Although results of this set of experiments are different from those of the previous set in terms of the detail clustering indices, they are similar as each other, in terms of their trends. Like the results in the previous set of experiments, the proposed version performs better several hundreds times than the traditional version in the original scale of clustering index. Only in logarithmic scale, the proposed version is outstandingly better than the traditional version, as illustrated in figure 2. In this set of experiments, we also judge that the proposed version of single pass algorithm is recommendable.

5.4. Discussion

This section concerns the discussion on the comparisons of the two versions of single pass algorithm. The comparisons are visualized as pie charts as illustrated in figure 3, 4, and 5. In each pie chart, the black part indicates the portion of the traditional version in terms of the logarithmic clustering index computed by equation (8). The white part indicates the portion of the proposed version. Figure 3 and 4 visualizes the comparisons of the two versions on NewsPage.com and 20NewsGroups, respectively, while figure 5 visualizes the entire comparison of the two versions.

Figure 3 visualizes the comparison of the two versions of single pass algorithm on the test bed, NewsPage.com. The logarithmic clustering index averaged over the three groups is 0.1798 in the traditional version. In the proposed version, it is 0.3737. The ratio of the proposed version to the traditional version is 67:33 only in the logarithmic scale. Based on figure 3, it is judged that the proposed version is clearly better than the traditional version.

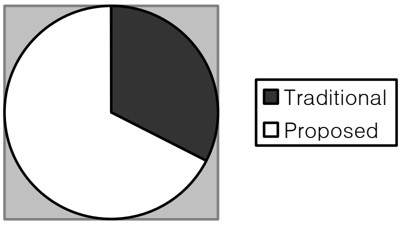

Figure 10. Visualization of Comparison of Two Versions on NewsPage.com

Figure 4 visualizes the comparison of the two versions of single pass algorithm on the test bed, 20NewsGroups. The portion of traditional version indicates 0.1481 by the averaged logarithmic clustering index. That of proposed version indicates 0.2944. The ratio of the proposed version to the traditional version becomes 66.34; the ratio is similar as that in the previous test bed. The comparison of the two versions on this test bed characterizes almost identically to that on the previous test bed.

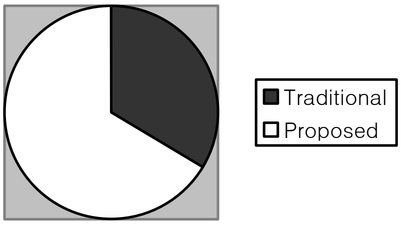

Figure 11. Visualization of Comparison of Two Versions on 20NewsGroups

Figure 5 visualizes the entire comparison of the two versions. Entirely over the two test beds, the portion of traditional version indicates 0.1639 by the generally averaged clustering index in its logarithmical scale, while the proposed version indicates 0.3341. The ratio of the proposed version to the traditional version is 67:33, overall. In the original scale of the clustering index, the portion of the traditional version is too tiny to be compared with that of the proposed version. Only in the logarithmic scale, the ratio clearly leads to the judgment that the proposed version is more desirable for text clustering.

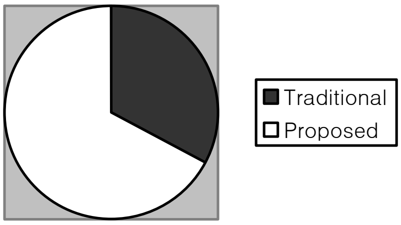

Figure 12. Visualization of General Comparison of Two Versions

6. Conclusions

The significance of this research is to specialize the single pass algorithm to be more suitable for text clustering, solving the two problems completely. We used a more suitable measure for evaluating approaches to text clustering, rather than F1 measure. In section 5, the proposed version worked better than the traditional version through the two sets of experiments. The reason of the better performance of the proposed version is that the two main problems were addressed by encoding documents into another form different from numerical vectors. From the empirical validation in section 5, we may conclude that the proposed version is more desirable than the traditional version.

There may be many ways of computing weights of words. In this research, we computed weights of words using equation (1), because of the popularity in the information retrieval. Note that the weights do not reflect exactly the relevancy of words to a given category or a content of a document. We need to develop several state of the art schemes for computing weights. In further research, we will compute weights of words using by combining multiple schemes with each other.

If we could develop various schemes for computing weights of words, we may define multiple tables to a document or corpus. There are two ways for treating multiple tables. The first way is to integrate multiple tables corresponding to a document or a corpus into a table. The second way is to treat the multiple tables as a committee. In further research, we will evolve the proposed approach by encoding a document or corpus into multiple tables.

Note that there is another clustering algorithm, k means algorithm, as well as single pass algorithm. Like the single pass algorithm, we can modify the k means algorithm so. The difference of the k means algorithm from the single pass algorithm is that a number of clusters is given as the parameter instead of the similarity threshold and prototypes of clusters change continually during clustering objects. In order to modify the k means algorithm, we must define one more operation where a table representing a group of tables should be defined. By building a table consisting of words spanning over tables, we can do that.

In this research, documents were encoded into tables with their fixed size. Note that the optimal size of tables depends on their corresponding document. We must optimize the size of each table for satisfying the two factors; reliability and efficiency. In other words, too large tables cause poor efficiency and too small one cause poor reliability. In further research, we will develop a scheme for sizing tables differently.

Literatures

Ambroise, C. and Govaert, G., “Convergence of an EM-type algorithm for spatial clustering”, Pattern Recognition Letters, Vol 19, No 10, 1998, pp919-927.

Banerjee, A., Dhillon, I., Ghosh, J., and Sra S., “Generative model-based clustering of directional data”, The Proceedings of the 9th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, 2003, pp19-28.

Bote, G., Vincent, P., Felix, M. A., and Solana, V. H., “Document Organization using Kohonen’s Algorithm”, Information Processing and Management, Vol 38, No 1, 2002, pp79-89.

Hatzivassiloglou, V., Gravano, L., and Maganti, A., “An Investigation of Linguistic Features and Clustering Algorithms for Topical Document Clustering”, The Proceedings of 23rd SIGIR, 2000, pp224-231.

Kaski, S., Honkela, T., Lagus, K., and Kohonen, T., “WEBSOM-Self Organizing Maps of Document Collections”, Neurocomputing, Vol 21, 1998, pp101-117.

Kohonen T., Kaski, S., Lagus, K., Salojarvi, J., Paatero, V., and Saarela, A., “Self Organization of a Massive Document Collection”, IEEE Transaction on Neural Networks, Vol 11, No 3, 2000, pp574-585.

Mitchell, T. M., Machine Learning, McGraw-Hill, 1997.

Vinokourov, A. and Girolami, M., “A Probabilistic Hierarchical Clustering Method for Organizing Collections of Text Documents”, The Proceedings of 15th International Conference on Pattern Recognition, 2000, pp182-185.