Determining Term on Text Document Clustering using Algorithm of Enhanced Confix Stripping Stemming

Titin Winarti
Department of Informatics and Communication
Semarang University

Jati Kerami
Department of Mathematics
University of Indonesia

Sunny Arief
STMIK Jakarta STI&K
Jakarta Indonesia

ABSTRACT
In a term based clustering technique with the vector space model, the issue of high dimensional vector space due to the number of words used always appears. This causes the clustering performance drops because the distance among the points tends to have the same value. The reduction of dimension by decreasing the number of words can be done by stemming. Stemming was used as term selection to reduce the many terms generated on preprocessing. The utilization of algorithm of enhance confix stripping stemmer reduced the terms that must be processed of 199.358 terms resulted from 108 text documents, became 5.476 terms result of the stemming. This reduction would speed up the process and saved the storage media. The evaluation by utilizing clustering was done using confusion matrix. The accuracy of experiment increased.

General Terms
Information Retrieval, Text Mining, K-Means

Keywords
Stemming, Clustering, Confusion Matrix, Enhance Confix Stripping Stemming

1. INTRODUCTION
Stemming is a computational procedure that converts word into its original form (stem) by searching the prefixes, suffixes and removes them based on the rules of a language. The result of the stemming process is called token / term. Stemming has been widely used in the processing of electronic documents [9]. Stemming is used in several fields such as: information retrieval system, question answering (QA), checking of spelling, translator machine, clustering of document, and classification of documents. One of the advantages of using stemming in the development of information retrieval system is: efficiency and index files have already been compressed. Without the process of stemming, the words 'improvement', 'improved' and 'improve' are something different. With the stemming process, each words that has the same root of word can be equated despite not having the words which are exactly the same [6].

The stemming process of each language is different, it is because each language has different rules in the use of the word with affixes. In Indonesian there is complexity in the variation of affixes that becomes the focus point on the establishment of the basic word [7]. Indonesian is a language that has various morphology affixes. Often a basic word or basic form needs to be given the affixes to be able to be used in the talk. These affixes can change the meaning, the types and the functions of a basic word. Which affixes that should be used depend on the purpose of the user in the talk [8].

Stemming algorithm of Indonesian with the best performance (having the least kind of stemming errors) is algorithm of enhanced confix stripping (ECS) stemmer. Algorithm of enhanced confix stripping stemmer, is an improvement algorithm of confix stripping stemmer. Algorithm of enhanced confix stripping (ECS) stemmer can be used to perform stemming on the Indonesian text document [9].

For text documents, clustering has proven to be an effective approach and an interesting research problem. Clustering of text documents plays a vital role in efficient document organization, summarization, topic extraction and information retrieval. Initially used for improving the precision or recall in an information retrieval system [1,2], more recently, clustering has been proposed for use in browsing a collection of documents [3] or in organizing the results returned by a search engine in response to user's query [4] or help users quickly identify and focus on the relevant set of results.

In a term based clustering techniques, the issue of high dimensional space due to the number of words used always appears. This causes the clustering performance drops because the distance among the points tends to have the same value. The reduction of dimension by decreasing the number of words can be done by stemming [12].

Algorithm of stemming used in this study is the ECS algorithm. The result of the ECS algorithm was term, this term would be chosen to be used for the clustering process. The researcher used the K-Means method for clustering. K-means clustering method by utilizing algorithm of ECS stemming on the process of preprocessing would be the solution for clustering of text document[11].

2. TEXT MINING
Text processing functions to convert unstructured textual data into structured data and stored in database [5]. Preprocessing stage consists of several steps, namely: case folding, tokenization, filtering and stemming. The process of case folding eliminates characters other than letters and changes all letters into lowercase. The process of tokenization cuts theinitial data in the form of sentence into words. The data as the results of tokenization process are continued with filtering process. The process of filtering takes the important words from the results of tokenization process. This process step can be done with two techniques, namely stop list (discarding the less important words) and word list (saving the important words). The data of filtering results are then processed by stemming. Stemming stage is the stage of looking for the root of word of each words of filtering results. The articles processed in this study are Indonesian articles. Stemming algorithm for Indonesian used is ECS stemmer [9].

3. ALGORITHM OF ENHANCED CONFIX STRIPPING STEMMER

Enhanced confix striping stemmer (ECS Stemmer) is the development of confix stripping stemmer (CS stemmer) [9]. Confix Stripping (CS) stemmer is stemming method to Indonesian introduced by Jelita Asian [8]. This stemmer is the development of stemming method for Indonesian introduced by Nazief Adriani [13].

After conducting limited experiments, in this study obtained some failures made by confix stripping stemmer and classified them as follows:

1. No prefix removal rule for words with construction of "mem+p...", for example, "mempromosikan", "memproteksi", and "memprediksi".
2. No prefix removal rule for words with construction of "men+s...", for example, "mensyaratkan", and "mensyukuri".
3. No prefix removal rule for words with construction of "menge+...", for example, "mengerem".
4. No prefix removal rule for words with construction of "penge+...", for example, "pengeboman.
5. No prefix removal rule for words with construction of "peng+k...", for example, "pengkajian".
6. Suffix removal failures – sometimes the last fragment of a word resembles certain suffix. For examples, the words like "pelanggan" and "pelaku" failed to be stemmed, because of the "-an" and "-ku" on the last part of the word should not be removed.

Based on those failures, in this study try to extend confix stripping stemmer, and present our modified confix stripping stemmer that is called enhanced confix stripping stemmer. The improvements deal as follows:

1. Modifying some rules on Table 1 and 2, so that stemming process on words with construction of "mem+p...", "men+s...", "menge+...", "penge+...", and "peng+k..." can be done. These modifications are listed in Table 5.
2. Adding additional stemming step to solve the suffix removal problem. This additional step called catloop algorithm . This step is performed when recoding (step 6, CS stemmer) failed.

In each process of catloop algorithm , the dictionary lookup is performed to check the result upon current word. The processes in catloop algorithm are defined as follows:

1. Restore the word to its prerecoding form and return all the prefixes that have been removed in the last process, so it will create word model like follows:

   [DP+[DP+[DP]]] + Root word

   Next, the prefix removal is attempted. If dictionary lookup succeed, then the process stops. Otherwise, the next step is executed.

2. Return the suffixes that have been removed previously. It means that the return starts from DS ("-i", "-kan", "-an") if exist, then followed by PP("-ku", "-mu", "-nya"), and the last is P ("-lah", "-kah", "-tah", "-pun"). On each returning, step 3) to 5) below is attempted. Special case for DS "-kan", character "k" is restored first and step 3) to 5) is executed. If still failed, then "an" is restored.
3. Prefix removal is performed according to rules defined in Table 1, 2, 3, and 4 (with modifications on Table 5 and 6).
4. Recoding is performed.

If dictionary lookup does not succeed, then return the word to its pre-recoding form and return all the prefixes that have been removed. The next suffix according order in Step 1) is restored and Step 3) to 5) are performed against current word.

Table 1 : Prefix Removal Rules for prefix "me-"

Rule	Construction	Prefix Removal
1	Me{l|r|w|y}V...	Me-{1|r|w|y}V...
2	Mem{b|f|v}...	Mem-{b|f|v}...
3	Mem{rV|V}...	Me-m{rV|V}...| Me-p{rV|V}...
4	MempV...	Mem-pA...dimana A!='e'
5	Mempe	Mem-pe...
6	Men{c|d|j|z}...	Men-{c|d|j|z}...
7	Meng{g|h|q|k}...	Meng-{{g|h|q|k}...
8	MengV...	Meng-V...| meng-kV...|
9	MenV...	Me-nV...| me-tV...
10	MenyV...	Meny-sV...

Table 2 : Prefix Removal Rules for prefix "pe-"

Rule	Construction	Prefix Removal
1	Pe{w|y}V...	Pe-{w|y}V...
2	PeCerV...	Per-erV... dimana C!={r|w|y|l|m|n}
3	peClerC2...	Pe-ClerC2... dimana C1!={r|w|y|l|m|n}
4	PeCP...	Pe-CP... dimana C!={r|w|y|l|m|n} dan P!='er'
5	PelV...	PelV... kecuali pada kata 'pelajar'
6	Pem{b|f|v}...	Pem-{b|f|v}...
7	Pem{rV|V}...	Pem{rV|V}... | Pe-p{rV|V}...
8	Pen{c|d|j|z}...	Pen-{c|d|j|z}...
9	Peng{g|h|q}...	Peng{g|h|q}...
10	PengV...	Peng-V... | peng-kV... | (pengV-... jika V="e")
11	PengV...	Peng-V... | peng-kV...
12	PenyV...	Peny-sV...
13	PerCAerV...	Per-CAerV... dimana C!='r'
14	PerCAP...	Per-CAP... dimana C!='r' dan P!='er'
15	PerV...	Per-V...| pe-rV...

Table 3 :Prefix Removal Rules for prefix "be-"

Rule	Construction	Prefix Removal
1	BeC_1erC_2	Be-C_1erC_2...dimana C1!={'r'|'l'}
2	Belajar...	Bel-ajar...
3	BerCAerV...	Ber-CaerV...dimana C!='r'
4	BerCAP...	Ber-CAP...dimana C!='r' dan P!='er'
5	BerV...	BerV...|be-rV...

Table 4 :Prefix Removal Rules for prefix "te-"

Rule	Construction	Prefix Removal
1	TeClerC2...	Te-ClerC2...dimana C1!='r'
2	TerCerV	Ter-CerV...dimana C!='r'
3	terClerC2...	Ter-ClerC2... dimana C1!=,,r"

4	TerCP...	Ter-CP...dimana C!='r'dan P!='er'
5	TerV...	Ter-V...|te-rV...

Table 5 : Modified Rules for Table1

Rule	Construction	Prefix Removal
6	Men{c|d|j|z|s}...	Men-{c|d|j|z|s}...
9	MenV...	Meng-V...| meng-kV...| MengV-...if V='e')
4	MempA...	Mem-pA...dimana A!='e'

Table 6 : Modified Rules for Table2

Rule	Construction	Prefix Removal
9	PengC	Peng-C
10	PengV...	Peng-V... | peng-kV... | (pengV-... jika V="e")

In which : C : Consonant

V : Vocal

A : Consonant and Vocal

P : Partikel and Fragmen

4. TERM WEIGHTING

Term weighting aims to determine the weight of each terms. The calculation of term weight requires two things, i.e: Term Frequency (tf) and Inverse Document Frequency (idf). Term Frequency (tf) is the frequency of occurrence of a word (term) in a document. Tf value varies in each documents depending on the occurrence of the word in a document. The great value of tf is equal to the level of term occurrences in the document. The more frequent a term appears on a document, the greater the value of tf on the document and the less frequent a term appears, the smaller the value of tf. Besides Term Frequency, Inverse Document Frequency (idf) is also needed in term weighting[10].

Inverse Document Frequency (idf) is the frequency of term occurrence in the whole documents. Idf value associates with the distribution of terms in the various documents. Idf value is inversely equal to the number of documents that contain the term. Term that rarely appears in the whole documents has greater idf value than the term that frequently appears. If each documents in the collection contains the term, the idf value of the term is zero (0). This suggests that any term that appears on the document in

The collection is not useful to distinguish documents based on specific topics. The illustration of tf-idf algorithm is shown in Figure 1/Fig.1.:

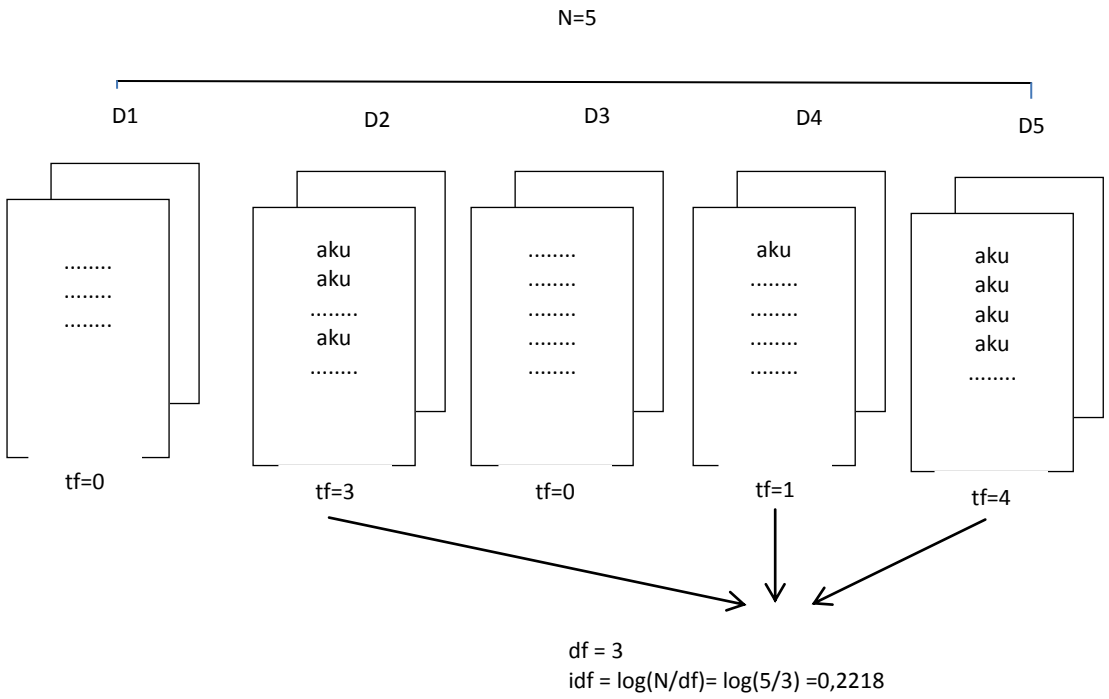

Fig. 1 llustration of tf-idf algorithm

In which
D1, ..., D5= documents

tf = the number of terms searched on each documents

N = total documents

df = the number of documents that contains the term searched.

The equation in calculating tf-idf value is [10]:

$$W_{i,}=tf_{ij}\text{x } idf_j=tf_{i,j}\times\log(\frac{N}{dfj}) \quad (1)$$

In which:

$W_{i,j}$ = the term weight to j on the document to-i

tf_{ij} = the number of j term occurrence into i

N = the number of documents in a whole

df_j = the number of documents that contains j term document

Weight calculation of certain terms in a document by using tf X idf shows that the best description of the document is term that most frequently appears in the document and less frequently appears in other documents [11].

5. CLUSTERING

Clustering is the process of grouping or classifying objects based on information obtained from the data describing the relationship among objects in principle to maximize the similarities among members of the same class and to minimize the similarities among the class / cluster [10].

For our analysis have chosen K-means algorithm to cluster documents. K-Means is a method of data analysis or data mining method that performs the modeling process without supervision (unsupervised) and is one of the methods that performs data grouping with partition system. This is an iterative Partitional clustering process that aims to minimize the least squares error criterion [6]. As mentioned previously, Partitional clustering algorithms have been recognized to be better suited for handling large document datasets than Hierarchical ones, due to their relatively low computational requirements [7].

The standard K-means algorithm works as follows. Given a setOf data objects D and a pre-specified number of clusters k, k dataobjects 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 earestor most similar centroid. Next, new centroids are recomputed 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 givendata 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 [10]. Clustering divides data into groups that have the same characteristics objects.This method seeks to minimize the variation among the existing data in a cluster and to maximize variation among existing data in other clusters [12]. Here are the steps of K-Means algorithm:

1. Determining a lot of k-clusters to be formed.
2. Generating random value for the center of initial cluster (centroid) as many as k-cluster.
3. Calculating the distance of each input data to each centroids by using Eucledian Distance formula until the closest distance of each data with centroid is found. Here is the Eucledian Distance equation:

$$d(x_i,\mu_i) = \sqrt{(xi - \square i)^2} \qquad (2)$$

In which:

d (x_i, μ_i) = the distance among x clusters with the center of μ cluster on words to i,

x_i = the words weight to i on the cluster that distance is tried to be found,

μ_i = the words weight to i on the center of cluster.

4. Classifying each data based on its closeness to the centroid (the smallest distance).
5. Updating the value of centroid. The new centroid value is obtained from the related mean cluster by using formula:

$$Ck = n_k + \sum d_i \qquad (3)$$

In which:

n_k = the amount of data in the cluster

d_i = the number of distance values included in each cluster

6. Performing iteration of step 2 through 5 until there is not changing of members of each cluster.
7. If step 6 has been met, then the mean value of the cluster center (μj) on the last iteration will be used as parameter to determine the classification of data.

For clustering quality evaluation are using confusion matrix as a measure of quality of the clusters A confusion matrix (Kohavi and Provost, 1998) contains information about actual and predicted classifications done by a classification system. Performance of such systems is commonly evaluated using the data in the matrix.

6. RESULTS AND DISCUSSION

The dataset used were 108 students' articles from various faculties in University of Semarang. The dataset would be clustered 5. The initial phase was the pre-processing comprising of case folding, tokenization, filtering and stemming. Of the 108 articles, the number of term before being stemmed as many as 199.358 terms and after being stemmed as many as 2.624 terms were found. There was a decrease of 98.68% term that would be used for clustering process before stemming and after stemming. Fig.2 shows the distribution of the dataset:

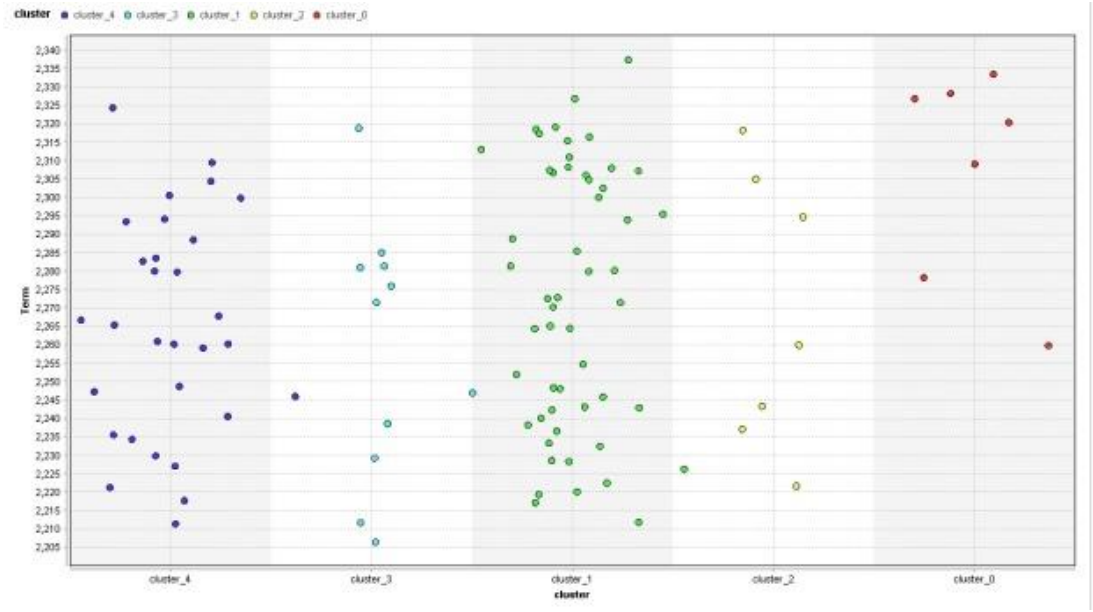

Fig.2 Distribution of Dataset

After the election of term had been done, weighting of matrix was made and then clustering process with K-means was done by using Rapid Miner Studio learning machine with k=5. Following are the results of the clustering process.

Cluster Model:

Cluster 0: 7 items of data

Cluster 1: 53 items of data

Cluster 2: 7 items of data

Cluster 3: 11 items of data

Cluster 4: 30 items of data

The total amount of data = 108 items of data.

The result of the clustering process can be seen in the following figure (Fig.3):

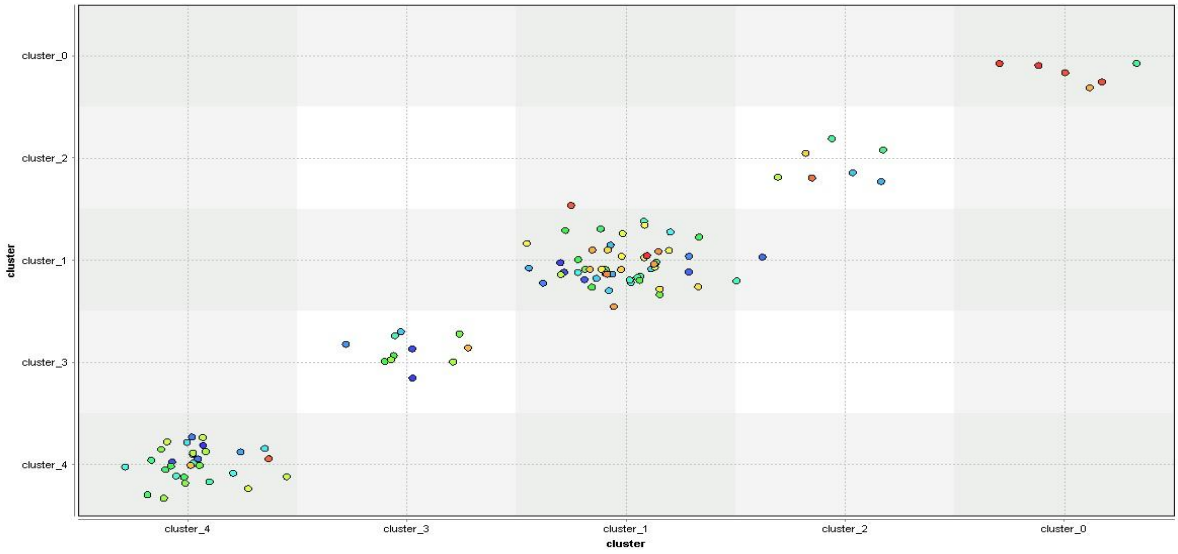

Fig. 3. The results of clustering

The following table shows the confusion matrix as the result of clustering evaluation with stemming. In this research to evaluate the results of clustering using confusion matrix with rapid miner studio tools. Table 7 and 8 shows the confusion matrix as the result of clustering evaluation without stemming.

Table 7 : The Result of Evaluation without Stemming

Akurasi	60,71%
Precision	85,82%
Recall	54,61%
Sensitivity	56,43%
Specificity	43,57%
G-Mean	49,59%
F-Measure	68,09%

Table 8 : The Result of Evaluation with Stemming

Akurasi	72,00%
Precision	81,52%
Recall	67,96%
Sensitivity	72,04%
Specificity	27,96%
G-Mean	44,88%
F-Measure	76,49%

Base on the experiment, there is improvement if use stemming for clustering.

7. CONCLUSION

Research of documents clustering by utilizing algorithm of ECS Stemmer in determining term for clustering had described the linkage among documents. The clustering results showed the existence of similar document, which represented the similarity among documents. The evaluation was conducted by using confusion matrix, the evaluation results of clustering process after being stemmed was that accuracy value was 72%, Precision was 81,52%, Recall was 67,96%, F-Measure was 76,49%.

For further development in clustering,the determination of the center point of initial cluster and the determination of k magnitude need to be done, so that clustering results are proper.

8. ACKNOWLEDGMENTS

We thank to Semarang University and Gunadarma University for allowing to use their laboratory, equipment and instruments, and experimental area.

9. REFERENCES

[1] Sharma, D.: Improved stemming approach used for text processing in information retrieval system. Master of Engineering in Computer Science & Engineering, Thapar University, Patiala (2012)

[2] Moral, C., Antonio, A., Imbert, R., Rmirez J.: A survey of stemming algorithms in information retrieval. Inf. Res.: Int Electron. J. **19**(1) (2014)

[3] Maurya, V., Pandey, P., Maurya, L.S.: Effective information retrieval system. Int. J. Emerg. Technol. Adv. Eng. **3**(4), 787–792 (2013)

[4] Singhal, A.: Modern information retrieval: a brief overview. IEEE Data Eng. Bull. **24**(4), 35–43 (2011)

[5] J.B. Lovins, 1968, "Development of a stemming algorithm, "Mechanical Translation and Computer Linguistic., vol.11, No.1/2, pp. 22-31.

[6] N. Sandhya, Y. Sri Lalitha, V.Sowmya, Dr. K. Anuradhaand Dr. A. Govardhan, 2011, Analysis of Stemming Algorithm for Text Clustering, International Journal of Computer Science Issues, ISSN 1694 - 0814

[7] Arifin, A. Z. and A. N. Setiono. 2002. Classification of Event News Documents in Indonesian Language using Single Pass Clustering Algorithm. Proc. of the Seminar on Intelligent Technology and its Application.

[8] Asian, 2005, Stemming Indonesian, In Proc. Twenty-Eighth Australasian Computer Science Conference (ACSC 2005), Newcastle, Australia. CRPIT, 38. Estivill-Castro, V., Ed. ACS. 307-314.

[9] Arifin, A.Z., I.P.A.K. Mahendra dan H.T. Ciptaningtyas. 2009 "Enhanced Confix Stripping Stemmer and Ants Algorithm for Classifying News Document in Indonesian Language", Proceeding of International Conference on Information &Communication Technology and Systems (ICTS).

[10] Arai, K., Barakbah, A. R.. 2007. Hierarchical K-Means: an algorithm for centroids initialization for K-Means, the Faculty of Science and Engineering, Saga University, Vol. 36, No.1

[11] Alfina, T., Santosa, B. and Barakbah, A.R. 2010. Analisa Perbandingan Metode Hierarchical clustering, K-Means dan Gabungan Keduanya dalam Cluster Data (Studi kasus: Problem Kerja Praktek Jurusan Teknik Industri ITS). Jurnal Teknik ITS Vol. 1, (Sept, 2012) ISSN: 2301-9271. Surabaya

[12] Liu T., S. Liu, Z. Chen and Wei-Ying Ma. "An Evaluation on Feature Selection for Text Clustering". Proceedings of the 12th International Conference (ICML 2003), Washington, DC, USA. PP 488-495. 2003.

[13] Adriani, M., Asian, J., Nazief, B., Tahaghoghi, S.M.M., Williams, H.E. 2007. Stemming Indonesian: A Confix-Stripping Approach. Transaction on Asian Langeage Information Processing. Vol. 6, No. 4, Artikel 13. Association for Computing Machinery: New York