International Journal of Image Processing Techniques – IJIPT
Volume 1 : Issue 1
Publication Date : 09 January 2014

Text Categorization of Indonesian Language Text Documents using Bracewell Algorithm

Aini Rachmania[1] , Jafreezal Jaafar[2],
Department of Computer & Information Sciences
Universiti Teknologi PETRONAS
Tronoh, MALAYSIA
[1]aini.rachmania@gmail.com, [2]jafreez@petronas.com.my

***Abstract*— With the Internet fast progressing, the number of news documents increases significantly and the demand for easy navigation around the news becomes pivotal. The online news domain needs a classifier that is able to classify news articles accurately at a low cost computation. Apart from that, because it is always updated, the classifier also needs to be able to detect new topic. This paper presents the topic identification and category classification method which enable categorization of news articles and identification of topic as new news arise. During the training phase, keywords are extracted from each document. Then, a document is classified to a predefined category. Lastly, the topic of the document is identified. Testing was conducted on Indonesian news corpus. The result shows that the classifier was able to classified Indonesian text documents with a satisfactory accuracy level as high as 93.84%.**

***Keywords*—text categorization, indonesian text documents, information retrieval, news domain, classsification, Bracewell Algorithm**

I. INTRODUCTION

Being the world's largest computer network, the internet has more than 100 million and growing number of computers communicating with one another over a set of standards and protocols [1]. By the last half of 1996, the Internet has covered more than 60 million of documents distributed among 12 million hosts and 600.000 servers, and these numbers are continuously increasing every day [2]. With the growing of the internet, the online news websites also took a rapid growth. This makes the demand for an easy and quick news become very important to users. Especially the needs to read relevant news about a topic or event the user knows. The news websites then created a technique to make an easy navigation among news for the user. A technique associated with such tasks is automatic text classification, that is, assigning documents to one or more pre-defined categories and topic [3].

The news domain presents a unique challenge in terms of categorization that other domain rarely presents. Apart from the fact that a news article can be classified into more than one category, news that belong to the same category may differ greatly from each other [4]. This is due to the characteristic of news domain where it is updated on a daily or hourly basis. Another requirement that occurs due to the particular characteristic is an effective and low-cost computation while maintaining high performance. This is because as an event develops, the more frequent the news is updated, the larger the space required to compute [5].

This paper proposes a technique to overcome the high computational issue often encountered during classification without decreasing the accuracy of the performance. Moreover, this paper also tries to answer the need to classify news into topics and indentifies whether it needs a new topic. The classifier implements Bracewell's algorithm [4] which uses the keywords from the training data to be compared with the keywords of the testing data to find the similarity. The technique has shown significant improvement on the accuracy and reduced computational cost when applied on Indonesia corpus.

II. RELATED WORKS

Text classification is a technique which automaticly assign a set of text documents to predefined categories according to the content [6]. Two of the major difficulties it presents are dealing with the high dimensionality and the accuracy [2]. A number of methods have been discussed in the literature for text classification. Naive Bayes [7], Decision Trees [8], K-Nearest Neighbors [9] and Single Pass Clustering [10] are among the most common methods used. Text classification can be divided into two main variants: text clustering and text categorization. Text clustering are related to finding group latent structure in a collection of document, while text categorization (often called text classification) can be defined as structurization of document repository into a structurized predefined group [11].

As it progressed, a lot of algorithms have been used to classify the Indonesian documents corpora. One of the earliest works is news events classification using the single pass clustering algorithm [10]. This algorithm uses the standard cosine similarity to calculate the likelihood between documents, and classifies them using the single pass clustering by calculating the similarity between each document to the cluster's representative. Although the concept is quite similar to Bracewell's proposed algorithm, the precision and recall results are not high (below 80%). Apart from that, the topic

classification method is not capable of detecting new topics as the news domain requires.

The trend in text classification algorithms continued as researchers kept developing. In 2008, Arifin developed a classifier for Indonesian News Documents by using Ants Based Clustering Algorithm [12]. While the F-measure value can reach up to 0.86 points, the algorithm takes a costly computation. This is because the ants based algorithm uses the ants analogy to find the shortest path between documents. The more ants used in the algorithm, the better the result. However, the more ants used, the longer it takes for the trial phase to be completed [12]. Another work done in 2008 was also by Arifin et al [13] using the Suffix Tree Clustering to classify online news documents. While the precision level can reach up to 80%, the algorithm needs a lengthy computation during the suffix tree construction process. This is because the algorithm running time depends to the number of documents collection and the number of words each document that are about to be classified possesses.

A rather more conventional algorithm using the Naive Bayes Classifier was used in 2009 [11]. It proposed two major phases: the learning phase and the classification phase. In the learning phase, the documents are processed to the predefined categories. During the classification phase, the method generates the category for each news articles. Even though the F-measure result can reach up to 92.26%, the algorithm shares similar issue with the suffix tree clustering, which is the dependency towards the number of words a document has. The algorithm can't work with a document that has more than 1000 words [11] .

III. METHOD

The method for the classifier implemented in this paper adopted the Bracewell algorithm. However, before a document is classified, it underwent several process in order to remove the noises from the documents and choose only the highly contributive terms to represent the data.

A. Preprocessing

Preprocessing is a control process for the words list dimension. The process is generally divided into four steps:

1) **Filtering**: this step eliminates all non-alphabetic characters such as numbers, symbols and punctuations. These characters are considered as delimiters.

2) **Case folding**: this step unifies the words by converting the letters in the words to the same case, either upper or lower.

3) **Stopwords removal**: this step removes words that are considered to be contributing small role in determining the content of the article. These words include preposition, conjunctions, pronouns and articles.

4) **Stemming**: this step transforms the words to their root forms by eliminating the words' suffix, infix and prefix. In this research, the stemming algorithm used is the enhanced confix stripping stemmer [12].

B. Keywords Selection

After the stemming process is done, text representation process is required to transform the previously digital text document into a more efficient and comprehensive model. This process is required to further analyse the article and select the kewords that best represent the document. Among the common approaches for text representation is Vector Space Model [14]. In the model, each document *dj* is transformed into a corresponding vector

$$d_j = (w_{1j}, w_{2j}, ..., w_{ij}) \quad (1)$$

where w_{ij} is the weight of the i^{th} term that exists in document *j*.

The weight of each term can be represented by several ways such as binary representation, Terms Frequency or Terms Frequency and Inverse-Document Frequency (TF-IDF) [2][15][16]. The classic TF-IDF methods shows a better performance compared to the binary and frequency method. The method is defined as:

$$w_{ij} = tf_{ij} . \log_2 \left(\frac{N}{df_i} \right), \quad (2)$$

where w_{ij} is the weight of i^{th} term in document *j*, whereas tf_{ij} is the frequency of the i^{th} term in document *j*. *N* is the total number of document processed, and df_j is the number of documents in which term *i* exists. The overall process of keywords selection is shown in Figure 1.

Instead of using Bracewell's multilingual single document keyword [17], the method for the keywords selection is ranking the words previously weighted using the top-*n* selection method, where the words are weighted using Equation 1 and 2. The top-*n* words are retrieved to represent the keywords of the article, with *n* being a number of words more than one and less than the total weighted words in the article. The Bracewell's method is not used in this stage because the particular method focuses more to the multilingual capability of the extractor which fits a rather more complicated language. Thus, due to the simplicity of Indonesian language, the top-*n* selection is used instead. This method has been commonly used for Indonesian text classification as seen in [10] and [11].

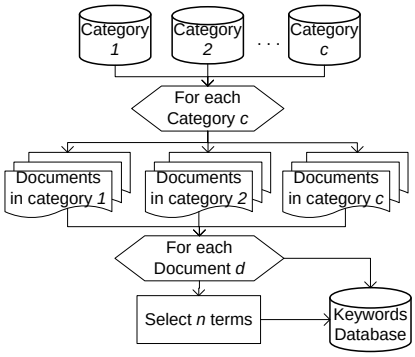


Figure 1. Keywords Selection Overview

The fitting number of top keywords to be taken is also a subject to be tested. It is assumed that after going through the preprocessing step, the words in the articles left are only the ones that are highly contributive towards the content of the article. The method is also used because it doesn't require an elaborate selection method that might lengthen the computation time and can be easily adjusted to meet the news content volume.

C. Bracewell's Algorithm

The first step in the category classification algorithm is determining the primitive categories. These primitive

categories are predefined categories. As explained in [4], a category is described as the higher level of groupings for news of which content generally has the similar idea from one another. Topic, on the other hand, is the main theme of an article. These groupings make a drill down navigation, with one category having more than one topics. In this paper, the categories used are a result of an analysis of several well-known Indonesian news sites like *www.kompas.com*, *www.antaranews.com* and *www.tempointeraktif.com*.

The category classification process can be divided into two main stages: training and classification [4]. The result from the training process is a descriptor for each category which then will be used in the classification stage. This descriptor is built for each category and contains the category name, respective number of documents and the keywords.

1) Training: The training phase trains the classifier category-based. To train the classifier, a set of predefined category and training data is needed. The Indonesian news articles are acquired, transferred into corpus format and kept in the database. For each category, the news articles are kept along with the total number of documents. As described in [4], this count indicates how many training documents have been seen for this category. These articles then underwent two processes explained before, which are preprocessing and keywords selection. The preprocessing phase yields a set of stemmed term for each documents in each category, while the keywords selection yileds a set of keywords for each category. These keywords are then stored in the database according to each predefined category. Although the workflow of this training phase applies the Bracewell's algorithm [4], the method for preprocessing and keywords selection are different. This is because we wanted to adjust the classifier to Indonesian language documents.

2) Category Classification: When a news article is classified, it undergoes four major steps. First, the keywords of the given articles are selected using the same method mentioned in the training phase. Next, the likelihood between the articles and each category is calculated. After that, a dynamic threshold is created. Finally, the best category is selected and the article is assigned a category.

The keywords are used to describe article. Thus, a likelihood can be calculated for each category. The likelihood of a category given an article is defined in Equation 3.

$$\text{Likelihood}\ (c_j \mid A=\{k_1, k_2, \ldots, k_n\}) = -\sum_{i=1}^{n} P(k_i \mid c_j) \log\left(P(k_i \mid c_j)\right) \quad (3)$$

In this equation, c_j is the $_j$th category. *A*, is the given article defined by a set of keywords and $P(k_i \mid c_j)$ is calculated using the in-document and the total number of documents count. After all the likelihoods between article and each category have been calculated, the next step is to calculate a dynamic threshold. The calculation is shown in Equation 4, where *L* is the list of all likelihoods and l_i is the likelihood of category *i*.

$$\text{Threshold} = \frac{\sum_1^{|L|} l_i}{|L|} + \sqrt{\frac{\sum (l_i - \frac{\sum_1^{|L|} l_i}{|L|})^2}{|L|}} \quad (4)$$

The mean and standard deviation of all the likelihoods are used to decide the dynamic threshold. The categories that have a likelihood greater than the mean plus one standard deviation are assigned to the article. With this method, an article can have more than one categories assigned to them. Figure 2 shows an overview of the algorithm.

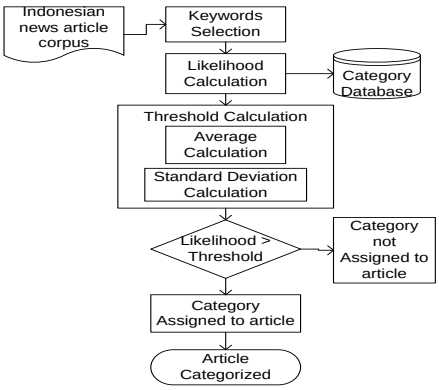


Figure 2. Category Classification

3) Topic Identification: The algorithm first classifies a given article as previously seen topic. Classification is done by finding the most similar topic to the article. Problem arises when the article has an unseen topic. This creates a requirement for the algorithm to be able to detect whether the previously seen topic assigned to the article is a true match for the article. The method divides the topic identification step into two sub-steps. The first is the topic identification where the article will be given the previously seen topic. The second is the topic discovery where the assigned topic will be further analyzed to detect whether the article requires a new topic to be created. Figure 3 shows an example of this transformation in a vector format.

Topic Vector	Kurs	Dollar	Saham	→ Kurs	Dollar	Saham	Valuta
	5	10	3	→ 5	10	3	0
Article Vector	Valuta	Kurs	Dollar	→ Kurs	Dollar	Saham	Valuta
	2	3	7	→ 3	7	0	2

Figure 3. Vector Transformation

After both vectors have been normalized, the similarity can be computed. Equation 5 shows the cosine similarity. t_i is topic i vector, while A is the article's vector. The Cosine Similarity is calculated for each topic, and the topic with the highest similarity is assigned to the news.

$$\text{CosSim}\ (t_i, A) = \frac{t_i \cdot A}{|t_i||A|} \quad (5)$$

Topic discovery determines if the topic assigned in the topic identification is suitable for the article already or if it should be assigned a new topic. This substep is done by dynamic thresholding. The thresholds are defined in figure 4.

(i) $CosSim(t_c,A) > 0.1 \wedge CosSim(t_c,A) > NewTSim(t_c,A)$
(ii) $NumTopics > 10 \wedge CosSim(t_c,A) > (2 \times StdDev(AllTopicSims) + Mean(AllTopicSims))$

Figure 4. Dynamic Thresholds

$CosSim(t_c,A)$ is the highest cosine similarity computed using equation 5, while *NewTSim* is calculated using equation 6 where $|t_c|$ is the length of topic vector, *Mean(A)* is the mean of article vector *A*, *StdDev.(A)* is the standard deviation of

article vector *A*, *Mean(t_c)* is the mean of topic vector, and |*A*| is the length of vector *A*.

$$\text{NewTSim}(t_c, A) = \frac{(0.05 \times |t_c|) \times \left(\text{Mean}(A) - \text{StdDev.}(A)\right) \times \text{Mean}(t_c)}{\left(|A| \times \left(\text{Mean}(A)\right)^2\right) \times \left(|t_c| \times \left(\text{Mean}(t_c)\right)^2\right)} \quad (6)$$

The first threshold is used to compare the cosine similarity of the conditionally classified topic (t_c) and the article (*A*) to the cosine similarity of the article and a hypothetical topic as calculated by *NewTSim* in Equation 6. *NewTSim* uses the information from the conditionally classified topic and the article to try and determine the cosine similarity between the article and the hypothetical topic that is similar to it. The second threshold is useful when enough topics have been discovered, which in the experiment determined to be 10. It checks that the cosine similarity of the conditionally classified topic is much greater than the cosine similarity of the other known topics. If both of the thresholds are met, then the conditionally classified topic becomes officially assigned to the article. Otherwise, a new topic can be input by the user and the article is the first source of training data.

IV. EXPERIMENTATION

A. Dataset

Corpora is retrieved from *http://www.kompas.com* (KOMPAS). Datasets are retrieved according to the primitive categories written in *kompas*. Total number of documents for each category for the training and testing may slightly differ from one another. The primitive category is later used as the ground truth for the testing experiment. The total number of training documents in this paper is 932, while the total number of testing documents is 455 documents. The training documents downloaded are those of the year 2011, on the other hand, the testing documents vary from the year 2011 and 2012. The datasets are shown in Table 1.

TABLE I. TRAINING AND TESTING DATASETS

Category	Training Documents	Testing Documents
Nasional (*National*)	105	50
Regional (*Regional*)	106	51
Internasional (*International*)	104	51
Metropolitan (*Metropolitan*)	106	50
Bisnis dan Ekonomi (*Business and Economy*)	101	51
Olahraga (*Sports*)	110	51
Sains dan Teknologi (*Science and Technology*)	90	51
Edukasi (*Education*)	109	50
Pariwisata (*Tourism*)	101	50
Total	**932**	**455**

B. Performance Measurements

To evaluate the utility of the methods, *accuracy* has been used. Accuracy combines the retrieved and relevant approach described in Table 2.

TABLE II. RETRIEVED AND RELEVANT

	Relevant	Not Relevant
Retrieved	TP	FP
Not Retrieved	FN	TN

As shown in Table 2, a document that is relevant and is retrieved by the classifier is considered as True Positive (TP). While a document that is not relevant yet is still retrieved by the classifier is named False Positive (FP). False Positives are considered as the noise of the documents. False Negative (FN) is the number of documents that are relevant but are not retrieved, while the number of documents that are not relevant and not retrieved by the classifier is a True Negative (TN). The components are then used to calculate the Accuracy as described in Equation 7.

$$\textit{Accuracy}\ (A) = (TP + TN) / (TP + FP + FN + TN) \quad (7)$$

C. Result

1) Category Classification

The category classification test was conducted to discover the overall performance of the classifier using the training data and to know the number of keywords that should be selected to achieve optimum performance. This test was conducted on two manners: offline and online. The number of keywords used for the experiment was 5, 10, 15 and 20. These ranges are selected due to the observation in a series of experiments that poor content representation is shown by using below 5 keywords whilst high computation time was recorded when more than 20 keywords is used.

Although the results show a satisfying result as the number of keywords increases, the classifier shows quite a substantial difference on each category. The classifier produced a high result on some categories, yet low on other. Category "*Olahraga*" (Sports) in this case, always generate high accuracy, but the "*Regional*" (Regional) category always yields the lowest score form the other categories. Figure 5 shows the result of the offline and online classification.

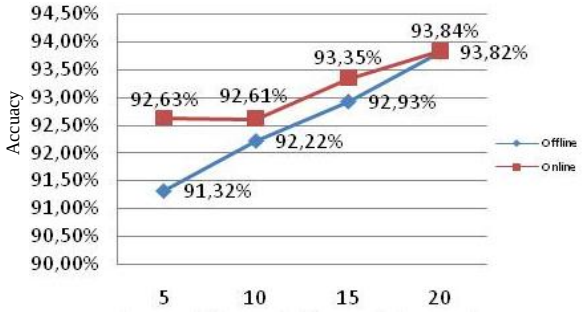


Figure 5. Category Classification Offline and Online

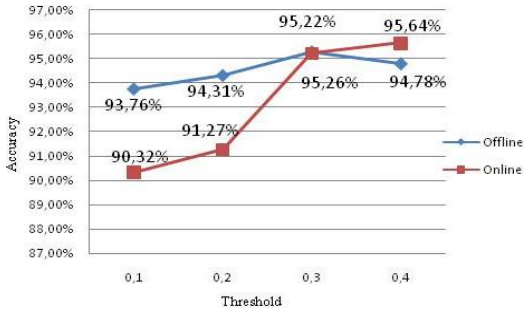
The result shows that the classifier yielded an overall higher result when the classification is done online. The possible reason for this is because when retrieving the data straight from the internet, the classifier requires less memory space, since the news article is directly grabbed and classified. The possible reason for a slight decrease on 10 keywords is because in online classification, there are more error factors that can affect the classifier performance, such as the grabber capability, the browser's HTML reading capability, etc.

2) Topic Identification

The topic identification experiment is done to figure out the performance of the classifier, whether it is able to assign the fitting topic and to detect if it needs a new topic, and to figure the optimum threshold value. Since the category classification showed that the best number of keywords to be selected is 20, the topic identification experiment used 20 as

the number of keywords and 0.1, 0.2, 0.3 and 0.4 as the thresholds parameter. The result is illustrated in figure 6.

Figure 6. Topic Identification Offline and Online



The results were less stable than the category classification, and although the accuracy were lower than those of the category classification, both the offline and online topic identification algorithm achieved high note on the 0.3 threshold value.

3) Grabber - Parser

The grabber and parser used in this paper is custom made for the *kompas* website, and holding an important role during the experiment. This is because the result of the downloaded data through the grabber and parser is used in the next steps. The grabber function is to download the news article from the website, while the parser is to parse the downloaded article and in this case, transforming it into a corpus. The result of the test is shown in table 3.

TABLE III. GRABBER AND PARSER RESULTS

Error Types	Case	
	Error	Cause
HTML 2.0 character reading	&gt;	>
	&mdash;	—
	&quot;	"
	ldquo;	“
Documents not completely downloaded	Documents only partially downloaded	Documents fully downloaded

The results show some characters that were not removed during the downloading process. This is due to the technical issue encountered during the implementation. The programming language used to implement the algorithm had some limitation in terms of HTML reading. While the HTML version used in the website was more advanced, the compiler could only cope with the version slightly lower than the one that the website used.

V. CONCLUSION

The paper proposed a text categorization of Indonesian language text documents using Bracewell's algorithm. The algorithm does not only categorize text to a predefined category, but also identify, classify and detect the topic. The results showed that the algorithm were able to perform both category classification and topic identification for Indonesian text documents with an accuracy as high as 95.64% and 93.84%.

While the original algorithm used the Multilingual Keywords Extraction for the feature selection, this paper implemented the keyword selection phase based on the TF-IDF weighing scheme and top-*n* ranking.

In the future work, we would like to integrate the Latent Semantic Indexing feature selection method with the classifier. The idea would be to integrate the two algorithms in order to reduce the high dimensionality issue encountered when dealing with the keywords. Also, in order to evaluate the classifier better, in the future we would like to compare the result computed using accuracy with human judgment.

VI. REFERENCES

[1] Mowery, D. C., & Simcoe, T. (2002). Is the Internet a US invention? an economic and technological history of computer networking. Research Policy , Vol I , 1369-1387.

[2] Li, Y. H., & Jain, A. K. (1998). Classification of Text Documents. The Computer Journal . Vol 41, 537-546

[3] Figueiredo, F., Rocha, L., Couto, T., Salles, T., Goncalves, M. A., Wagner Jr, M. (2011). Word co-occurence features for text classification. Information Systems , Vol 36, 843-858.

[4] Bracewell, D., Yan, J., Ren, F., & Kuroiwa, S. (2009). Category Classification and Topic Discovery of Japanese and English News Articles. Electronic Notes in Theoretical Computer Science , 51-65.

[5] Urguz, H. (2011). A two-stage feature selection method for text categorization by using information gain, principal component analysis and genetic algorithm. Knowledge-Based Systems , Vol 24, 1024-1032.

[6] Lewis, D. D., Yang, Y., Rose, T. G., & Li, F. (2004). RCV1: A New Benchmark Collection for Text Categorization Research. Journal of Machine Learning Research, Vol 5 , 361-397.

[7] Rish, I. (2011). An empirical study of the naive Bayes classifier. RC 22230 .

[8] Johnson, D. E., Oles, F. J., Zhang, T., & Goetz, T. (2002). A decision-tree-based symbolic rule induction system for text categorization. IBM Systems Journal .

[9] Yu, X.-P., & Yu, X.-G. (2007). Novel Text Classification Based on K-Nearest Neighbor. Sixth International Conference on Machine Learning Kwon, O.-W., & Lee, J.-H. (2003). Text categorization based on k-nearest neighbor approach for Web site classification. Information Processing and Management, Vol 39, 25-44.

[10] Arifin, A. Z., & Setiono, A. N. (2002). Klasifikasi Dokumen Berita Kejadian Berbahasa Indonesia dengan Algoritma Single Pass Clustering. Proceeding of Seminar on Intelligent Technology and Its Applications (SITIA). Surabaya.

[11] Asy'arie, A. D., & Pribadi, A. W. (2009). Automatic News Articles Classification In Indonesian Language By Using Naive Bayes Classifier Method. Integration and Web-based Applications & Services (iiWAS) 2009 ERPAS, (pp. 658-662). Kuala Lumpur.

[12] Arifin, A. Z., Mahendra, I. P., & Ciptaningtyas, H. T. (2008). Enhanced Confix Stripping Stemmer and Ants Algorithm For Classifying News Documents in Indonesian Language. The 5th International Conference on Information & Communication Technology & Systems . 149-158.

[13] Arifin, A. Z., Roby, D., Navastara, D. A., & Ciptaningtyas, T. H. (2008). Klasifikasi Online Pada Dokumen Berita Berbahasa Indonesia Dengan Algoritma Suffix Tree Clustering. Seminar Sistem Informasi Indonesia (SESINDO 2008). Surabaya.

[14] Soucy, P., & Mineau, G. W. (2005). Beyond TFIDF Weighting For Text Categorization In The Vector Space Model. In International Joint Conference on Artificial Intelligence (Vol. 19, p. 1130). Lawrence Erlbaum Associates Ltd.

[15] Yates, R. B., & Neto, B. R. (1999). Modern Information Retrieval. New York: ACM Press.

[16] Skiba, M. J. (2010). Text Preprocessing in Programmable Logic. Canada: University of Waterloo.

[17] Bracewell, D. B., Ren, F., & Kuroiwa, F. (2005). Multilingual Single Document Keyword Extraction for Information Retrieval. NLP-KE (pp. 517-522). IEEE.