I J A B E R, Vol. 14, No. 11, (2016): 8211-8223

NAÏVE BAYES IMPLEMENTATION INTO BAHASA INDONESIA STEMMER FOR CONTENT BASED WEBPAGE CLASSIFICATION

Andreas[*] and Lusia Permata Sari Hartanti[**]

Abstract: *A lot of information can be gathered by using internet technology. Internet may give out positive and negative impacts at once. There is a lot of content that not suitable to be consumed, especially for children, which is contain pornography. There are several ways have been done to block webpages containing pornography, but none has done it by reading the content. It is necessary to embed the machine learning role in the web browser so that the blocking process can be executed in real time. Blocking process performed by classifying web page based on its content into two groups, that is pornography or not. A good stemming process is needed in order to provide a good result of the web page reading. This study focuses on web pages in Bahasa Indonesia. Implementing Naive Bayes into Bahasa Indonesia stemmer algorithm in this study can classify web pages into several class with an accuracy of 84.03%.*

Keywords: *information retrieval, Naïve Bayes, web content filtering*

1. INTRODUCTION

Internet is used for information retrieval widely today. Unfortunately, not all of the information in the internet is valid and good consumed. Information on the internet needs to be selected and sorted wisely.

The ease of finding information on the internet lead to internet users increased rapidly in recent years. In late 2014 the International Telecommunication Union (ITU) noted an increase in the number of internet users in one year was 6.6%. Until now there are more than 3 billion internet users worldwide.

Content that is not suitable consumed feared would adversely affect users. The impacts include crime rising, the decline of morality, security threats, and others. This negative impact will be more worrisome if it occurs to children which do not have extensive knowledge and strong faith to select and sort the information obtained from the internet. Currently the important matters relating to children in conjunction with the internet is pornography.

[*] Information System, *E-mail: andreas.jodhinata*

[**] Industrial Engineering, Universitas Pelita Harapan Surabaya, Indonesia, *E-mail:lusia.hartanti@uph.edu*

The development of information technology (internet) which is so rapidly, plus the information available on the internet is “uncensored”, then the use of the internet as a medium for the dissemination of pornography content becomes more fertile. There are many ways to hide the content by giving domain name and metadata that is far from being pornography.

Based on this case then comes some applications that are used to block pornography from the internet known as content filtering. Unfortunately, content filtering applied today mostly still be less flexible. Content filtering only works based on the examination of the existence of given keywords. If the content contained words that are included in the keyword then the web page will automatically be blocked. Yet in reality, not all the web pages that contain those words can be classified as pornography.

According to Kamus Besar Bahasa Indonesia (Indonesian dictionary), pornography is the depiction of erotic behavior with paintings or writings to arouse lust; reading material that is deliberately and solely designed to arouse lust in sex. The word pornography itself comes from the Greek, namely porne (prostitutes) and graphos (image or text). Peter Webb as quoted by Rizal Mustansyir complete the definition of pornography, adding that it was related to obscenity more than just eroticism. Then Ernst and Seagle added as “Any matter or exhibiting odd thing visually representing persons or animals performing the sexual act, whatever normal or abnormal”. According to Black’s Law Dictionary quoted by Adami Chazawi in his book entitled “Crime Pornography”, pornography is defined as “material (such as Writings, photographs, erotic movies) depicting sexual activity or erotic behavior in a way that is designed to arouse sexual excitement”.

One example of web content that contains elements in the keyword but cannot be classified into pornography website category is sex education website. Surely there are many words relating to sex or sexuality will be listed within this website, but all of them are used as learning material and not as sexual exploitation or for sexual appetite, so it cannot be categorized as pornography.

Based on the problem above, it needed an application that can blocking web pages based on content better. It required the proper method in classifying web pages based on overall content, not just partially based on the given keywords. This study modifies Bahasa Indonesia stemmer algorithms combined with Naïve Bayes method in order to classify properly. Additionally, applications are built must also be able to hold the web content while the whole content is read and determined the category. Thus web page content will not be displayed on the browser screen during the classification process.

2. STEMMING BAHASA INDONESIA

In linguistic morphology and information retrieval, stemming is the process of reducing inflected (or sometimes derived) words to their word stem, base or root form. Based on the assumption that words that have the same stem generally have a similar meaning, stemming process widely used in information retrieval as one way to improve retrieval performance.

There are two preliminary process should be done in text processing before the stemming process begun, tokenization and stopword removal. Both of these processes are done to split a document into words and to simplify the words of the splitting results. Stemming process diagram can be seen in Figure 1.

Tokenization is a separation process of a sequence of characters based on the spacing character, and may also do the removal process of certain characters at the same time, such as punctuation.

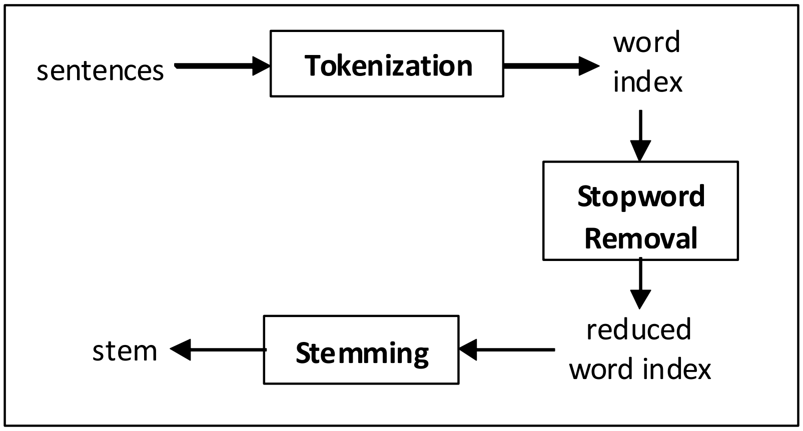

Figure 1: Stemming process diagram

In the process, there are various things that need to be considered in doing tokenization. For a complex tokenization, there are several issues that must be considered, for example in English, there are words such as “Finland’s capital”. Options that occurs are to turn that word into “Finland”, or “Finlands”, or “Finland’s”. Another issue is that there are two words that have one meaning, such as “DI Jogjakarta”. The word “DI Jogjakarta” can be considered as one or two tokens. When considered as one token, it would appear the question of how to decide that “DI Jogjakarta” is one token. This question is an issue in tokenization. A simple tokenization process, for example which is only based on spacing character or punctuation, called segmentation.

Stopword removal is the process of removing the words that are commonly used and do not have any valuable information in a context. This process requires a list of words that frequently appear (but less important) to be excluded from the index. Some examples of the stopword list are:

1. Conjunction word such as after, while, later;
2. Preposition such as for, from, to, with;
3. Adverb such as very, only, more;
4. Numbers such as some, few, a lot;
5. Pronoun such as us, we, they, it;
6. etc.

Stopwords elimination is useful because it reduces the size of the number of words up to 40%. As it aims to reduce the index size, then verbs, adjectives and other adverb can also be included into the stopword list. In another words, stopword list can be increased or decreased as needed.

There are several kinds of stemming techniques, namely:

1. Stemming with reference table: done by removing word's affixes according to the used reference table.
2. Stemming using a dictionary: a dictionary of root word is used as a reference for the stemming results when the process has done. Stemming results from the process with this method must exist on the root word dictionary, if not then the entered term is considered as the stem.
3. Corpus (collection of documents) based stemming: stemming result using this method is affected by the collection of documents used in the testing process. The formed stem class is influenced by co-occurrence statistical value of each term on the stem class. This method was developed from an initial hypothesis that two terms with the same root word would often appear in the collection of documents used in testing phase. Value of how often these words appear simultaneously is calculated using co-occurrence statistics.

Stemming process is very dependent on morphology of a language. Since Bahasa Indonesia has a different morphology with English, then the stemming process has the difference too. In general, Bahasa Indonesia stemming process relatively more difficult than English. This is due to the combinations of affixes in Bahasa Indonesia.

Based on its position, affixes in morphology Bahasa Indonesia divided into three types, namely:

1. Prefix: affix in the front of the word;
2. Infix: affix in the middle of the word;
3. Suffix: affix in the end of the word.

Based on its composition, morphology Bahasa Indonesia also divide affixes into three types, namely:

1. Inflection suffix which is suffix group that does not alter the root word. For example, word "duduk" is given the suffix "-lah" would be "duduklah". This group can be divided into two:
   a. Particle (P), including "-lah", "-kah", "-tah", and "-pun".
   b. Possessive Pronoun (PP), including "-ku", "-mu", and "-nya".
2. Derivation Suffixes (DS) is a group of original suffixes in Bahasa Indonesia which added directly to the root word like "-i", "-kan", and "-an".
3. Derivational Prefixes (DP) is a group of prefixes that could be added into pure root word or to root word which already added up to two prefixes. Derivational prefixes including:
   a. Prefix that can morphologies ("me-", "be-", "pe-", and "te-")
   b. Prefix that cannot morphologies ("di-", "ke-" and "se-")

Based on the affixes classification above, the form of affixed words in Bahasa Indonesia can be modeled as:

[DP+ [DP+ [DP+]]] Root_Word [[+DS] [+PP] [+P]]

Until now there has been some Bahasa Indonesia stemming algorithms were developed, among other things: Nazief and Andriani, confix stripping (CS), Enhanced confix stripping (ECS), and Sastrawi.

3. NAÏVE BAYES METHOD

Naïve Bayes algorithm is one of algorithms that is widely used for classification using statistics and probabilities stated by Thomas Bayes. Bayes' theorem is used to predict future change based on the preceded experience. This theorem combined with Naïve algorithm which assume that the condition between the attributes are independent. Naïve Bayes classification has a rule that a specific characteristic of a class has no relation with others.

Common forms of Bayes' theorem can be formulated as:

$$p\left(C_j \mid d\right) = \frac{p(c_j)p\left(d \mid c_j\right)}{p(d)} \tag{1}$$

Where c_j is the text category that will be classified, and $p(c_j)$ is probability of text category c_j. While d is text document that represented as a set of words (w_1, w_2,…, w_n), where w_1 is the first word, w_2 is the second and so on. During the text documentation classification process, Bayes approach will select text categories that have the highest probability (c_{MAP}):

$$c_{MAP} = \arg\max \frac{p(c_j)p(d \mid c_j)}{p(d)} \tag{2}$$

The value of $p(d)$ can be ignored since it is constant for each c_j, so Formula 2 can be simplified as:

$$c_{MAP} = \arg\max p(c_j)p(d \mid c_j) \tag{3}$$

The probability of $p(c_j)$ can be estimated by counting the number of training documents in each text category c_j. While to calculate the distribution of $p(d \mid c_j)$ will be difficult to do, especially in classifying process with a large amount of text documents due to the number of terms $p(d \mid c_j)$ can be very large. It is because the number of terms is equal to the sum of all word position combinations multiplied by the number of categories to be classified With Naïve Bayes approach that assumes that every word in each category are independent on each other, then the calculation can be simplified and can be written as follows:

$$p(d \mid c_j) = \Pi_i \, p(w_i \mid c_j) \tag{4}$$

Thus:

$$c_{MAP} = \arg\max \, p(c_j) \, \Pi_i \, p(w_i \mid c_j) \tag{5}$$

The value of $p(c_j)$ and $p(w_i \mid c_j)$ will be calculated during training process is executed as follows:

$$p(c_j) = \frac{n(c_j)}{n(sample)} \tag{6}$$

$$p(w_i \mid c_j) = \frac{1 + p(w_i, c_j)}{|V| + count(c)} \tag{7}$$

where $n(c_j)$ is the number of documents in j category and $n(sample)$ is the number of sample documents that used in training process. While $p(w_i, c_j)$ is an occurrence number of the word w_i in c_j category, $|V|$ is the number of all words in c_j category and $count(c)$ is the number of unique words in all training data.

4. WEBPAGE CLASSIFICATION

Applications built in this study consists of two major groups, which are web services and stemmer. Application that placed on the web service will do the cleaning process, tokenization, stopword, classifying and weighting, and Naïve Bayes. While stemmer application will do the Bahasa Indonesia stemming process. Figure 2 shows a flowchart of the application is built.

The cleaning process do the following things:

1. Take all heading tag (H1 until H7) then remove all tags in it;
2. Remove all STYLE tags;

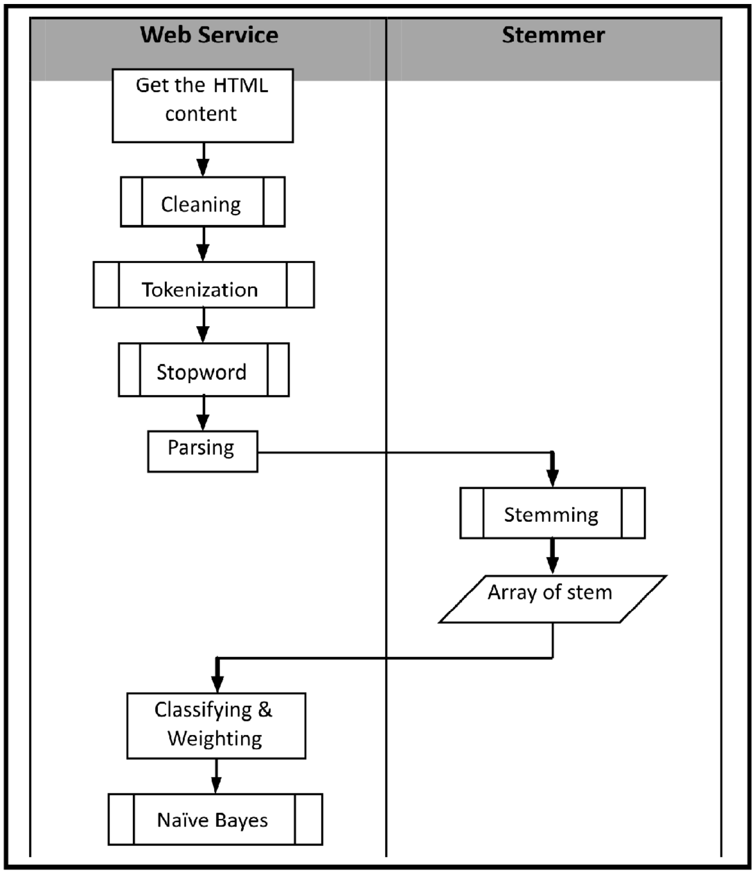

Figure 2: Webpage content classification process diagram

3. Remove all comment;
4. Remove all FORM tags;
5. Remove all entity characters;
6. Remove all non-alphabet characters;
7. Remove all additional space characters (trim);
8. Convert all characters to lowercase;

There are two things to do on tokenization stage:

1. Split the content of H1 to H7 tag with space character as the delimiter;
2. Save the result into the proper array.

The next step is stopword. Stopword is a removal process of the common words which are not considered as keywords. Two files are required at this stage, common word taken from the IndoDic website and additional common word that added as a supplement that contains words from testing results. Both of these files are used to remove all common words in heading tags. Next, all words that composed of two or less characters will also be deleted.

Once common word is removed, the remaining words would be entered as an input to the stemming process. Repetitive words will be split into two parts and then each section will be processed separately. The stemming process only enforced to the words that composed more than three characters.

Stemming process consists of the following steps:

1. Remove suffix –lah, –kah, –tah, and –pun;
2. Remove suffix –ku, –mu, and –nya;
3. Remove suffix –i, –kan, and –an;
4. Remove prefix di–, ke–, and se–;

Whenever suffix and prefix of a word are removed, the remaining word will be checked for its existence in the dictionary. After that, regular expression process will be executed based on Table 1.

Table 1
Rules of affixes separation

<table>
<tr><th><em>No</em></th><th><em>Word Format</em></th><th><em>Separation Rule</em></th></tr>
<tr><td>1</td><td>berV…</td><td>ber-V… | be-rV…</td></tr>
<tr><td>2</td><td>berCAP…</td><td>ber-CAP… ; C≠′r′, P≠′e′</td></tr>
<tr><td>3</td><td>berCAerV…</td><td>ber-CAerV…; C≠′r′</td></tr>
<tr><td>4</td><td>belajar</td><td>bel-ajar</td></tr>
<tr><td>5</td><td>beC_1erC_2…</td><td>be-C_1erC_2…; C_1≠{‘r′, ‘l′}</td></tr>
<tr><td>6</td><td>terV…</td><td>ter-V… | te-rV…</td></tr>
<tr><td>7</td><td>terCerV…</td><td>ter-CerV…; C≠′r′</td></tr>
<tr><td>8</td><td>terCP…</td><td>ter-CP…; C≠′r′, P≠′er′</td></tr>
<tr><td>9</td><td>teC_1erC_2…</td><td>te-C_1erC_2…; C_1≠′r′</td></tr>
<tr><td>10</td><td>me{l|r|w|y}V…</td><td>me-{1|r|w|y}V…</td></tr>
<tr><td>11</td><td>mem{b|f|v}…</td><td>mem-{b|f|v}…</td></tr>
<tr><td>12</td><td>mempe…</td><td>mem-pe…</td></tr>
<tr><td>13</td><td>mem{rV|V}…</td><td>me-m{rV|V}… | me-p{rV|V}…</td></tr>
<tr><td>14</td><td>men{c|d|j|s|z}…</td><td>men-{c|d|j|s|z}…</td></tr>
</table>

contd. table 1

No	*Word Format*	*Separation Rule*
15	menV…	me-nV… | me-tV…
16	meng{g|h|k|q}…	meng-{g|h|k|q}…
17	mengV…	meng-V… | meng-kV… | (meng-V… jika V='e')}
18	menyV…	meny-sV…
19	mempA…	mem-pA… ; A≠'e'
20	pe{w|y}V…	pe-{w|y}V…
21	perV…	per-V… | pe-rV…
22	perCAP…	perCAP…; C≠'r', P≠'er'
23	perCAerV…	per-CAerV…; C≠'r'
24	pem{b|f|V}…	pem-{b|f|V}…
25	pem{rV|V}…	pe-m{rV|V}… | pe-p{rV|V}…
26	pen{c|d|j|z}…	pen-{c|d|j|z}…
27	penV…	pe-nV… | pe-tV…
28	peng{g|h|q}…	peng-{g|h|q}…
29	pengC…	peng-C…
No	Word Format	Separation Rule
30	pengV…	peng-V… | peng-kV… | (peng-V… jika V='e')}
31	penyV…	peny-sV…
32	pelV…	pe-lV… (except 'pelajar' à 'ajar')
33	peCerV…	per-erV…; C≠{l|m|n|r|q|y}
34	peCP…	peCP…; C≠{l|m|n|r|w|y}, P≠'er'
35	terC_1erC_2…	ter-C_1erC_2…; C_1≠'r'
36	peC_1erC_2…	pe-C_1erC_2…; C_1≠{l|m|n|r|w|y}

Description:

C : Consonant

V : Vowel

A : Consonant or vowel

P : particle or fragment of a word

After do the stemming stage, words in the array are grouped, occurrence frequency is counted and weighted. Giving weight will follow Table 2.

Table 2
Giving weight table

Tag Name	*Weight per word*
H1	8
H2	7
H3	6
H4	5
H5	4
H6	3
H7	2
Other tags	1

As a final step of the webpage classification process is the use of Naïve Bayes algorithm. Naïve Bayes algorithm must go through the training and testing phases.

Probability calculation of the training document class has not been done at training phase. Training phase only calculate the probability of words and classes in the training data. Training phase is conducted several times and carried out with the training dataset.

Once training phase is completed, the process will proceed to testing phase. The training process consists of several steps as follow:

1. Calculate the power of $p(w \mid c)$ of every word in every training document;
2. If the value is not zero, then that word will be included into class calculation process;
3. Calculate the value of $p(c)$ of every class;
4. Determine the class of each document.

Classification process of the real documents is similar to the training data. In order to speed up the process then only words that are not exist in the training document will be calculated its $p(w \mid c)$ value. After that, the value of $p(w \mid c)$ of each word will be exponent to the amount/weight of the word. Then the results will be added in the calculation of its class.

Since the value of $p(w \mid c)$ produce too many decimal digits, then to simplify the class calculating process it use a logarithmic function as in Formula 8.

$$c_{MAP} = \arg\max[\log p(c_j)\, \Pi_i \log\, p(w_i \mid c_j)] \tag{8}$$

Thus the class calculation results will be negative. Classes with value closest to zero will be the results.

Training and testing process in this study use 120 webpage document with 7:3 ratio, so 84 documents being used for training process and 36 documents for testing process. Those document are taken from various categories, which are: economics, health, politics, culture, and education.

There are several training and testing datasets that implemented in this study:

Table 3
The usage of training and testing documents

I	Training	Document 1-84
	Testing	Document 85-120
II	Training	Document 37-120
	Testing	Document 1-36
III	Training	Document 1-36 and 73-120
	Testing	Document 37-72
IV	Training	Document 1-72 and 109-120
	Testing	Document 73-108

Here is a confidence level table (confusion matrix) of each training and testing dataset:

Table 4
Confusion matrix set I

		Actual class				
		C_1	C_2	C_3	C_4	C_5
Expected class	C_1	6	0	0	1	0
	C_2	0	6	0	1	0
	C_3	0	1	5	0	1
	C_4	0	0	0	7	0
	C_5	1	2	0	0	5

Table 5
Confusion matrix set II

		Actual class				
		C_1	C_2	C_3	C_4	C_5
Expected class	C_1	5	0	0	1	1
	C_2	0	6	1	0	0
	C_3	0	1	5	0	1
	C_4	0	0	0	7	0
	C_5	0	1	0	0	7

Table 6
Confusion matrix set III

		Actual class				
		C_1	C_2	C_3	C_4	C_5
Expected class	C_1	6	0	1	0	0
	C_2	0	6	1	0	0
	C_3	0	0	7	0	0
	C_4	1	0	0	6	0
	C_5	0	1	0	0	7

Table 7
Confusion matrix set IV

		Actual class				
		C_1	C_2	C_3	C_4	C_5
Expected class	C_1	6	0	1	0	0
	C_2	0	6	1	0	0
	C_3	1	0	6	0	0
	C_4	0	1	0	5	1
	C_5	0	1	0	0	7

Table 8
Overall experiment result

Dataset	*Accuracy level*
I	80.56%
II	83.33%
III	88.89%
IV	83.33%

Table 8 shown that the average level of overall accuracy is 84.03%. The errors that occur mostly caused by many words that are not keywords enter into the database so that it be included in the calculation process.

5. CONCLUSION

This study concludes several things:

1. Stemming process can be accelerated by reducing the dictionary reading and adding more rules that fits with Indonesian morphology;
2. The accuracy level of webpage classification process is highly dependent on stopword list and keyword that used;
3. Less level of confidence in this study is also influenced by many foreign languages in Bahasa Indonesia website.

References

Arifin, A.Z., I.P.A.K. Mahendra and H.T. (2009). Ciptaningtyas. 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).

Bustami. (2013). Penerapan Algoritma Naive Bayes Untuk Mengklasifikasi Data Nasabah Asuransi. Jurnal Teknik Informatika, 2013:1-10.

Jelita, Asian. (2007). Effective Techniques for Indonesian Text Retrieval. PhD thesis, RMIT University Australia.

Manning, Christoper D., Prabhakar Raghavan and Hinrich Schütze. (2009). Introduction to Information Retrieval. London: Cambridge University Press.

Natalius, Samuel. (2010). Metode Naive Bayes Classifier dan Penggunaannya pada Klasifikasi Dokumen. Makalah Teknik Informatika, 2010:1-5.

Putra, Tegar Riyono, Daniel Oranova, and Umi Laili Yuhana. (2010). Klasifikasi Severity dari Bug untuk Proyek Perangkat Lunak. Jurnal Teknik Informatika, 2010:1-9.

Samodra, Joko, Surya Sumpeno, and Mochamad Hariadi. (2009). Klasifikasi Dokumen Teks Berbahasa Indonesia dengan Menggunakan Naïve Bayes. Electrical National Seminar, Informatics, and It's Educations.

Tahitoe, Andita Dwiyoga and Diana Purwitasari. (2010). Implementasi Modifikasi Enhanced Confix Stripping Stemmer untuk Bahasa Indonesia dengan Metode Corpus Based Stemming. Jurnal Teknik Informatika, 2010:10-15.

Wibisono, Yudi. (2005). Klasifikasi Berita Berbahasa Indonesia menggunakan Naive Bayes Classifier. Mathematics National Seminar UPI Bandung.

Wibowo, Jati Sasongko and Sri Hartati. (2011). Text Document Retrieval in English Using Keywords of Indonesian Dictionary Based. Jurnal Teknik Informatika, 2011:26-32.