Sentiment analysis on Bahasa Indonesia tweets using Unibigram models and machine learning techniques

B H Iswanto\textsuperscript{1}\textsuperscript{*} and V Poerwoto\textsuperscript{2}

\textsuperscript{1}Complex Systems Research Group, Department of Physics, Universitas Negeri Jakarta, Indonesia
\textsuperscript{2}Graduate Program in Information Technology, Bina Nusantara University, Indonesia

*\texttt{bhi@unj.ac.id}

Abstract. Sentiment analysis on English tweets has its challenges. In addition to frequent use of the informal language, the words used are usually less consistent, contain abbreviations, and mixed with local languages. In this study, we combined n-grams feature selection models, i.e., unigrams, bigrams, and unibigram (1+2-grams) to analyze the public opinion of the Bahasa Indonesia tweets about the presidential candidates of the Republic of Indonesia in the 2014 presidential election. The experiment was carried out using Naive Bayes classifiers, Maximum Entropy classifiers, and Support Vector Machines with and without stop words removal and stemming in pre-processed tweet documents. The experiment results show the best performance is achieved by Naïve Bayes classifiers with unibigram feature models without removing the stop words and stemming process. This method achieved high performance with the precision and recall up to 85.50\% indicating that automatic sentiment analysis of tweet documents using well-known supervised learning methods is feasible for the Indonesian language. More interestingly is the fact that stop words removal and stemming process on the corpus made the classification performances worse compared with the corpus that experienced cleansing only.

1. Introduction

Sentiment analysis on Bahasa Indonesia tweets has its challenges since social media generally does not have standard language rules for its users. In addition to frequent use of informal language, unstructured sentences, contains abundant sentiments, and the words used are also less consistent in expressing opinions [1, 2]. Also, the Indonesian tweet is generally written in a mixed language with local or foreign languages, contains slang words, uses abbreviations [3]. This habit can be found almost on all tweet channels in Twitter; a microblogging service enables its users to send and read messages quickly; even it's limited to a maximum of 140 characters, which is called tweets. They use it for various purposes, such as communicating, comment on events, and expressed political opinions [4].

With the growing availability of opinion-rich resources, new opportunities and challenges arise to seek out and understand the public sentiment or opinions [5]. In recent years a lot of works have been done in the field of sentiment or opinion analysis. Several techniques have been developed using machine learning algorithms, both unsupervised learning as well as supervised learning methods [6]. They used some feature spaces to represent documents, such as unigram n-gram feature models, the part-of-speech (POS) [7], and terms frequency [8]. There is comparatively less research on sentiment
analysis of Bahasa Indonesian tweet documents. Generally, they use well known machine learning algorithms, such as Naive Bayes classifiers, Maximum Entropy (MaxEnt) classifiers, Decision Tree, and Support Vector Machines (SVM) with various feature models. Barfian & Iswanto [9] proposed unigram and bigram feature models for classification of negative content tweets. The other authors proposed text normalization algorithm to analyze complaint category [2], POS tagging for brand reputation classification [10] and combining unigram and frequency values as the features for movie review analysis [11]. However, they have not explored sufficiently, especially on combining feature models and important preprocessing techniques on Bahasa Indonesia tweets documents.

For the recent study, we combined n-grams feature selection models for sentiment analysis of Bahasa Indonesia tweet documents using machine learning techniques. Moreover, the influence of stop words and stemming in preprocessed documents also will be investigated on the classification performance with various feature selection models. The corpus domain in this research is a public opinion about the presidential candidates of the Republic of Indonesia in the 2014 presidential election and only tweets with Bahasa Indonesia that will be processed.

2. Methodology
The idea of sentiment classification is to assign the correct sentiment to each document using supervised machine learning models trained and tested using a set of training and testing data. In this work, we followed the approach taken by Pang et al. [12] in using the supervised machine learning techniques, namely Naive Bayes, MaxEnt classifiers, and SVM and data preparation for classification purposes. Overall, in our research, the sentiment classification are implemented through four steps: (i) data collection, (ii) preprocessing, (iii) feature selection, and (iv) classification.

Preprocessing data is important since many non-standard text spelling words as found in Bahasa Indonesia tweet documents. The preprocessing technique in this research involves: (i) cleansing documents such as converting all the letters to the lower case; removing the URLs, email addresses, hashtag, username, non-letter characters, and all single characters; replacing two or more repeating letters with a single letter; replacing two or more repeating spaces with a single space; and removing the searching keywords. (ii) removing the words which do not exist in Kamus Besar Bahasa Indonesia (KBBI), the official Indonesian language dictionary; and (iii) removing the stop words and stemming to a part of the documents.

Thereafter, feature selection was conducted to prepare to represent a document as a vector of features. In this study, we used the standard bag-of-features framework, n-gram language model, and their variance. It is a set of co-occurring words within a given window, and when computing the n-grams, you typically move one word forward. For example, for the sentence "presiden harapan rakyat", for $n = 1$ (unigrams) then would be ‘presiden’, ‘harapan’ and ‘rakyat’, for $n = 2$ (bigrams) would be ‘harapan rakyat’. The combination of the unigrams and bigrams forms 1+2-gram (unibigrams). It will be formed by 1-2 co-occurring works, then for the example text would be ‘presiden’ and ‘harapan rakyat’.

In sentiment classification, we experimented three supervised machine learning methods: Naïve Bayes, EntMax classifiers, and SVM with unigrams (as a baseline), bigrams, and unibigrams. In addition, experiments were conducted by removing stop words and stemming of tweet documents to study their effect on the classification performance.

For each experimental variation described above, 10-fold cross-validation was performed by partitioning the sentiment corpus into ten folds, where each fold contains an equal number of each class. As for the results of the classification was tabulated in the confusion matrix. Therefore the performance of the classifiers could be evaluated by measuring recall and precision. The recall is a performance measure of the whole positive part of a dataset, whereas precision is a performance measure of positive predictions.
3. Experiment results

For all experiments, datasets were collected through Tweets crawling process by using Twitter4J API to retrieve tweet documents. According to the aims of the study, a set of keywords was selected to retrieve only the desired tweets that discussed two presidential candidates of the Republic of Indonesia on the 2014 presidential election: first candidate (Prabowo Subianto) and the second one (Joko Widodo). The keyword set for retrieving desired tweets for the first candidate is \textit{\{prabowo subianto hatta rajasa, prabowo subianto, hatta rajasa, Prabowo, Hatta, hatta, Prbw\}}, while for the second one is: \textit{\{joko widodo jusuf kalla, joko widodo, jusuf kalla, jokowi jk, Jkw, Kalla, Jokowi\}}. Once it’s crawled, the tweet will be filtered using Language Detection Java API developed by [13], then only Bahasa Indonesia tweets would be stored in datasets. The total number of tweet documents after filtering is 26,764 documents: 10,768 documents about first president candidate and 15,996 about the others stored as PRAB and JKW dataset respectively.

For comparing the effect of cleansing data, removing stop word and stemming on sentiment classification, we experimented with varying treatments:

- CL1: cleansing data only
- CL2: cleansing and removing terms which are not found in KBBI
- CL3: cleansing, removing terms which are not found in KBBI, removing stop words and stemming.

The stop words removal and stemming were conducted using IndonesianStemmer Apache Lucene API package developed by [14]. Each treatment was applied to the PRAB and JKW dataset then each stored as PRAB\_CL1, PRAB\_CL2, PRAB\_CL3, JKW\_CL1, JKW\_CL2, and JKW\_CL3. For example, preprocessor results on a tweet document using the treatments are as follows:

Tweet: RT @yeoliday: Jokowi-JK bukan pemimpin biasa. Indonesia butuh pemimpin yang berbeda, kami merindukan pemimpin yg merakyat!!

CL1: \textit{bukan pemimpin biasa indonesia butuh pemimpin yang berbeda kami merindukan pemimpin yg merakyat dan jujur}

CL2: \textit{bukan pemimpin biasa indonesia butuh pemimpin yang berbeda kami merindukan pemimpin merakyat dan jujur}

CL3: \textit{impin indonesia butuh impin beda rindu impin rakyat jujur}

The unigrams, bigrams and the unibigrams feature model were implemented on the tweet documents and stored in the different datasets as in Table 1. Note that UNI, BI, and UBI stand for unigrams, bigrams, and unibigrams respectively, and \#Feat represents the number of feature for each dataset.

| Dataset       | \#Feat. | Dataset       | \#Feat. |
|---------------|---------|---------------|---------|
| PRAB\_CL1\_UNI | 775    | JKW\_CL1\_UNI | 819    |
| PRAB\_CL1\_BI  | 298    | JKW\_CL1\_BI  | 364    |
| PRAB\_CL1\_UBI | 1071   | JKW\_CL1\_UBI | 1182   |
| PRAB\_CL2\_UNI | 1100   | JKW\_CL2\_UNI | 592    |
| PRAB\_CL2\_BI  | 164    | JKW\_CL2\_BI  | 230    |
| PRAB\_CL2\_UBI | 696    | JKW\_CL2\_UBI | 821    |
| PRAB\_CL3\_UNI | 368    | JKW\_CL3\_UNI | 419    |
| PRAB\_CL3\_BI  | 89     | JKW\_CL3\_BI  | 121    |
| PRAB\_CL3\_UBI | 454    | JKW\_CL3\_UBI | 538    |

Upon finishing preprocessing the data, Term Frequency-Inverse Document Frequency (TF-IDF) method is applied to score the weight of terms in a document based on how frequently they appear across multiple documents [15]. The TF-IDF method also was successfully applied in many text mining tasks [16].
Therefore, the experiments using machine learning methods given the dataset were carried out to achieve the objectives of this study. The experiment scenario is as follows. First, all tweet corpus in the datasets is labeled manually according to sentiment polarity: positive or negative on a presidential candidate. This work was done to prepare the dataset for the classification task. We used a balanced dataset.

Therefore, supervised machine learning methods, i.e., Naïve Bayes classifiers, Entropy Maximum classifiers, and SVM were trained using the various datasets with different feature models: unigrams (as a baseline), bigrams, and unibigrams. Experiments also were conducted by removing stop words and stemming (CL3) of tweet documents to study their effect on the classification performances. The experiments were conducted intensively in WEKA, an open source data mining software. Hence, the machine learning algorithms could be applied directly to classify the datasets in Table 1.

For each experimental variation, we performed 10-fold cross-validation by partitioning the sentiment corpus into ten folds, where each fold contains an equal number of each class. As for the results of the classification was tabulated in the confusion matrix. Therefore the performance of each classifier could be evaluated by measuring recall and precision [17]. The recall is a performance measure of the whole positive part of a dataset, whereas precision is a performance measure of positive predictions.

The experiment results using PRAB and JKW dataset is presented as in Table 2 and Table 3. The tables show the average precision score and the average recall score obtained by each classifier in classifying the sentiment of test objects using the different feature selection models. Note that NB stands for Naive Bayes classifiers, MaxEnt stands for Maximum Entropy classification, and SVM stands for Support Vector Machines.

**Table 2. Performance of classifiers against feature selection for the PRAB dataset.**

|                | CL1  |          | CL2  |          | CL3  |          |
|----------------|------|----------|------|----------|------|----------|
|                | Prec. | Recall   | Prec. | Recall   | Prec. | Recall   |
| Unigram (775 features) |      |          |      |          |      |          |
| NB             | 0.798 | 0.798    | 0.748 | 0.748    | 0.740 | 0.740    |
| MaxEnt         | 0.624 | 0.624    | 0.615 | 0.615    | 0.697 | 0.697    |
| SVM            | 0.756 | 0.754    | 0.719 | 0.718    | 0.739 | 0.737    |
| Bigram (298 features) |      |          |      |          |      |          |
| NB             | 0.726 | 0.667    | 0.698 | 0.617    | 0.695 | 0.585    |
| MaxEnt         | 0.715 | 0.662    | 0.690 | 0.612    | 0.703 | 0.586    |
| SVM            | 0.742 | 0.555    | 0.681 | 0.575    | 0.693 | 0.578    |
| Unibigram (1071 features) |      |          |      |          |      |          |
| NB             | 0.805 | 0.805    | 0.754 | 0.754    | 0.751 | 0.750    |
| MaxEnt         | 0.691 | 0.691    | 0.593 | 0.593    | 0.649 | 0.649    |
| SVM            | 0.767 | 0.743    | 0.735 | 0.730    | 0.741 | 0.734    |

Table 2 presents the experiment results using PRAB dataset for each corpus variation by implementing different treatment: CL1, CL2, and CL3. From the table, clearly shows the best classification performance is achieved Naïve Bayes classifiers with precision and recall of 80.50% and 80.50% respectively when unibigram models were used as feature model on tweet corpus without removing stop words and stemming. Applying the same feature model and the preprocessing technique, we also observe an increase in performance of other classifiers, both EntMax classifiers, and SVM.
Table 3. Performance of classifiers against feature selection for JKW dataset.

|                | CL1         | CL2         | CL3         |
|----------------|-------------|-------------|-------------|
|                | Prec. | Recall | Prec. | Recall | Prec. | Recall |
| **Unigrams (819 features)** |      |         |      |         |      |         |
| NB             | 0.854 | 0.851 | 0.826 | 0.825 | 0.805 | 0.805 |
| MaxEnt         | 0.752 | 0.752 | 0.732 | 0.732 | 0.714 | 0.714 |
| SVM            | 0.832 | 0.809 | 0.783 | 0.762 | 0.775 | 0.768 |
| **Bigram (364 features)** |      |         |      |         |      |         |
| NB             | 0.751 | 0.701 | 0.731 | 0.624 | 0.722 | 0.60  |
| MaxEnt         | 0.750 | 0.697 | 0.753 | 0.681 | 0.743 | 0.636 |
| SVM            | 0.764 | 0.553 | 0.762 | 0.546 | 0.763 | 0.55  |
| **Unibigram (1182 features)** |      |         |      |         |      |         |
| NB             | 0.855 | **0.853** | 0.836 | 0.835 | 0.801 | 0.801 |
| MaxEnt         | 0.777 | 0.777 | 0.735 | 0.735 | 0.65 | 0.65 |
| SVM            | 0.820 | 0.785 | 0.784 | 0.738 | 0.771 | 0.754 |

Table 3 shows the experiment results using JKW dataset. We observed similar results as in previous experiments using the PRAB database. Using JKW dataset, the best classification performance also is achieved Naïve Bayes with precision and recall of 85.50% and 85.30% when unibigram were used on tweet corpus without removing stop words and stemming process. Performance of EntMax classifiers and SVM also increased by applying unibigram and same preprocessing technique.

Moreover, from Table 2 and Table 3 we observed that removing stop words and stemming in preprocessed data decreased the classification performance for all classifiers by comparing scores in CL1 and CL3. We noted the preprocess decreased the recall score of all Naïve Bayes, MaxEnt classifiers, and SVM significantly. The effect of stop words removal and stemming on a classifier is different, depending on the feature models used. The classification performance of Naïve Bayes classifiers decreased up to 14.41% (using bigrams, CL2), Entropy Maximum classifiers 16.34% (using unigrams, CL3), and SVM up to 5.07% (using unigrams, CL1). This seems to confirm our intuition that Indonesian language tweets typically non-formal, contain slang words, mixed languages with local languages, and abbreviations, such as blas, jkw, gak beradat, di pimpin, ngember, and so on. Thus, when the words are removed due to considered to have no meaning, then the preprocessed documents make a lot of difference to the performance. Indeed, stemming in Bahasa Indonesian tweets also can impact on the meaning of the word, as well as removing prefix. This result is in accordance with the study by [18] which concludes that effectiveness of a stemming process is determined by the morphological complexity of the language.

4. Conclusion
In this research, we studied the combination of n-grams feature models, i.e., unigram, bigram, and unibigram, and impact of stop words removal and stemming on the sentiment classification of Bahasa Indonesia tweets using supervised machine learning methods: Naïve Bayes, MaxEnt classifiers, and SVM. In general, the Naïve Bayes tests yielded the best results among the various classification techniques employed with unigrams models, without removing stop words and stemming process. Moreover, the study also found that stop words removal and stemming in preprocessed tweet documents impacted significantly on the decrease of classification performances. Therefore, stop word and stemming algorithms should be considered if it is necessary to do in the sentiment analysis, especially for Bahasa Indonesia tweet documents.
References

[1] Zhang S, Wei Z, Wang Y and Liao T 2018 Sentiment analysis of Chinese micro-blog text based on extended sentiment dictionary Future Generation Computer Systems 81 p 395-403

[2] Hanafiah N, Kevin A, Sutanto C, Fiona, Arifin Y and Hartanto J 2017 Text Normalization Algorithm on Twitter in Complaint Category Procedia Computer Science 116 p 20-26

[3] Hidayatullah F 2015 Language tweet characteristics of Indonesian citizens in 2015 International Conference on Science and Technology (TICST) Patham Thani

[4] Öztürk N and Ayvaz S 2018 Sentiment analysis on Twitter: A text mining approach to the Syrian refugee crisis Telematics and Informatics 15 p 136-147

[5] Pang B and Lee L 2008 Opinion mining and sentiment analysis Foundations and Trends in Information Retrieval 2 1-2 p 1–135

[6] Liu L and Zhang L 2014 A survey of opinion mining and sentiment analysis in Mining Text Data (Springer US) p 415-463

[7] Agarwal, Xie B, Vovsha I, Rambow O and Passonneau R 2011 Sentiment Analysis of Twitter Data in Workshop on Language in Social Media Oregon

[8] Liu L 2012 Sentiment Analysis and Opinion Mining (Morgan & Claypool Publishers)

[9] Barfian E, Iswanto B H and Isa S M 2017 Twitter Pornography Multilingual Content Identification Based on Machine Learning Procedia Computer Science 116 p 129-136

[10] Vidya N A, Fanany M I and Budi I 2015 Twitter Sentiment to Analyze Net Brand Reputation of Mobile Phone Providers Procedia Computer Science 72 p 519-526

[11] Franky and Manurung R 2008 Machine Learning-based Sentiment Analysis of Automatic Indonesian Translations of English Movie Reviews in Proceedings of the International Conference on Advanced Computational Intelligence and Its Applications (ICACIA)

[12] Pang B, Lee L and Vaithyanathan S 2002 Thumbs up?: sentiment classification using machine learning techniques in Proceedings of Conference on Empirical Methods in Natural Language Processing (EMNLP-2002)

[13] Tala F Z 2003 A Study of Stemming Effects on Information Retrieval in Bahasa Indonesia.” Master of Logic Project, Institute for Logic, Language and Computation (Netherlands: Universiteit van Amsterdam)

[14] Shuyo N Language Detection Library for Java Retrieved Mei 10, 2014, from https://code.google.com/p/language-detection/

[15] Salton G and Buckley C 1988 Term-weighted approaches to automatic text retrieval Information Processing and Management 24 5 p 513–523

[16] Kuncoro B A and Iswanto B H 2015 TF-IDF method in ranking keywords of Instagram users’ image captions in International Conference on Information Technology Systems and Innovation (ICITSI) Bandung

[17] Saito T and Rehmsmeier M 2015 The Precision-Recall Plot Is More Informative than the ROC Plot When Evaluating Binary Classifiers on Imbalanced Datasets PLoS ONE 10 3 e0118432

[18] Popovič M and Willett P 1992 The effectiveness of stemming from natural- language access to Slovene textual data J. Am. Soc. Inf. Sci. 43 p 384-390