Indonesian Social Media Sentiment Analysis with Sarcasm Detection

Edwin Lunando[#1], Ayu Purwarianti[#2]

School of Electrical Engineering and Informatics, Institute Technology of Bandung,
10th Ganeca Street, Bandung, Indonesia.
Email: [1]edwinlunando@gmail.com, [2]ayu@stei.itb.ac.id

Abstract— **Sarcasm is considered one of the most difficult problem in sentiment analysis. In our observation on Indonesian social media, for certain topics, people tend to criticize something using sarcasm. Here, we proposed two additional features to detect sarcasm after a common sentiment analysis is conducted. The features are the negativity information and the number of interjection words. We also employed translated SentiWordNet in the sentiment classification. All the classifications were conducted with machine learning algorithms. The experimental results showed that the additional features are quite effective in the sarcasm detection.**

Keywords— **Sentiment analysis, sarcasm, classification, SentiWordNet**

I. INTRODUCTION

Recently, the topic of sentiment analysis, automatic classification of the opinion or sentiment conveyed by a text towards a subject, is highly researched. There are a lot of researches in this area which improves the technique in building a sentiment analysis system. Meanwhile, in the business application industry, sentiment analysis application has been used comprehensively to give more information about the user insight about one topic to help them while making decision. Despite the high usage of the sentiment analysis application, there are still rooms for improvements. One of the problems that still become a challenge in sentiment analysis is sarcasm.

Sarcasm is using irony to mock or convey contempt. Sarcasm transforms the polarity of the text into its opposite while, the text itself looked like the original sentiment. From our observation from 100 microblogging text with simple conversation topics such as food, life, and health, we found that sarcasm text is very rare. There were only 2 from 100 texts that contain sarcasm. But for sensitive topics such as government, brand, or politic, the quantity is greatly rose. There were 18 texts out of 100 texts containing sarcasm.

Sarcasm or irony is an old and well researched phenomenon in the field of linguistic and psychology [1]. Unfortunately, in the field of text mining or more specifically: sentiment analysis, detecting sarcasm automatically is still considered a difficult problem because lexical features do not give enough information to detect sarcasm. One approach to solve this problem was by employing lexical features such as unigram and pragmatic factors such as smileys or emoticons [5]. An interesting analysis from the field of linguistic tells us that lexical factors like presence of adjectives and adverbs, presence of interjections, and use of punctuations play a quite significant role in sarcasm text [4].

In the basic sentiment analysis, many researches employed machine learning techniques such as Naïve Bayes, Maximum Entropy, and Support Vector machine because these algorithms are tend to outperform the other algorithm in the context of text classification [2][3][7]. Other than that, there is also a lexical resource, SentiWordNet [6], developed as a resource of word with sentiment score. These two techniques are also employed in our sentiment analysis system of Indonesian social media. This is different with other researches on sentiment analysis of Indonesian social media [2] [3] which didn't address the sarcasm detection and didn't employ sentiment score in the SentiWordNet.

II. SENTIMENT ANALYSIS WITH SARCASM CLASSIFICATION

An easy way to comply with the conference paper formatting requirements is to use this document as a template and simply type your text into it.

Similar with other sentiment analysis system, our sentiment classification system consists of pre-processing, feature extraction and classification. The system architecture is as shown in Fig 1.

Fig 1. Architecture of the Sentiment Analysis System

A. Pre-processing component

The pre-processing aims to minimize the vocabulary of terms used in the text message. In Indonesian text message of social media such as Twitter or Facebook, people tend to use slank words than the formal one such as using numeric to replace alphabet, repeating vocal characters, and using common informal words to replace the formal words [2]. To process such words, we employed the preprocessing such as follow:

1. Converse numeric character into alphabet, such as "ga2l" into "gagal" (fail)
2. Remove vocal repetition, such as "cemunguuuudh" into "cemungudh"
3. Translate informal words into formal words using dictionary, such as "cemungudh" into "semangat" (high spirit).

Here, even though some informal words are misspelled from formal words, but some other are completely different lexical than the formal words, therefore the strategy is to build a dictionary and use it to translate informal words into formal words.

B. Feature extraction component

In the feature extraction, there are several features taken from the pre-processed text:

1. Unigram

We analysed that unigram is more suitable for Indonesian social media text since the grammars used in Indonesian social media texts are various and informal. The unigram taken from the text is only the term that exists in our translated SentiWordNet. We translated English SentiWordNet into Indonesian using an available statistical machine translation (Google). In the translation, one Indonesian word with more than one English translation is given the average score of all the English translations.

For text "Pasangan Rieke dan Teten cocok untuk memajukan Jawa Barat!" (Rieke and Teten are suitable to bring forward West Java!), the unigram features are taken for "cocok" (suitable) and "memajukan" (bring forward) which gives SentiWordNet score of 0.5 and 0.375.

There are several phenomena in using unigram:
a. Negation

Negation words (such as "no" or "not") tend to change sentiment score of a certain word. For example, in the text "Televisi jaman sekarang tidak begitu bagus karena mahal" (nowadays televisions are not good because they are expensive), the sentiment word "bagus" (good) is changed into negative because it is preceded by a negation word ("tidak" or "not"). The negation words usually reside before the sentiment word and it can be located two or three words away from the sentiment word. We handled this by multiplying the score of sentiment word closest to the negation word by 1.
b. Word context

One word sentiment may change depends on its word context. For example, the word "mahasiswa" (student) is basically a neutral word but if this word is preceded by "harga" (price) which makes it into "harga mahasiswa" (low price) then the word becomes a positive word. To handle this problem, we used a special list of word such as used in [2].
c. Affix

Word with different affix may have different sentiment. For example "murah" (low) has a positive sentiment, while "murahan" (twopenny) has a negative

sentiment. Since not all suffix "an" have negative sentiment, we handled this problem by using a list of word such as used in [2].

2. Negativity

This feature represents the percentage of the negative sentiment in the topic of the text message. This feature is intended to catch global information. It gives information about the real sentiment of a certain topic. In order to get this feature, the topic of the text message should be extracted first. In this research, we did the topic extraction manually to determine the topic for each text.

For example, in a topic about Indonesian singer, "Rhoma Irama wants to be Indonesia's President" has 80% negative sentiment texts in the public. Then, the negativity value of the topic is 80% or 0.8. After determining each topic negativity value, the input text negativity value can be determined by extracting the topic of input text and get the negativity value from the topic.

3. Number of interjection words

This feature shows the number of interjection words from the text message. Example of interjection words are "aha", "bah", "nah", "wew", "wow", "yay", "uh", etc. We employ this feature based on our observation that among 100 sarcasm text, there are 20 text with interjection words. Below are the examples of sarcasm text with interjection. If the text is using interjection words, the text has more tendency to be classified into sarcasm text.

"Wow kk wow. hebat banget rhoma irama berani nyapres"
"jir, Polri Indonesia makin jago aja. Kapolda sendiri buron gitu."
"wah… cantik sekali cara telkomsel melayani pelanggannya, dikacangin coy."

4. Question word

This feature is used to classify neutral text. By detecting the question word like "who", "what", "when", "how", "where", and "why" it will show that the text has no sentiment value. Almost all text with question words are classified into neutral text. This feature works like Boolean value. If there are question word at a text, this feature will get "true" and vice versa.

C. Classification component

In the classification component, there are two classification steps. The first classification is to classify each text into three sentiment classes: positive, negative and neutral. The second classification is to classify the sarcasm of the positive text. The flow is as shown in Fig 2. We only classify positive sarcasm text is because almost all of the sarcasm text is looked like

positive text, while the real value of the text is negative. Although it may be possible to encounter sarcasm text that has the positive value, the quantity is very low.
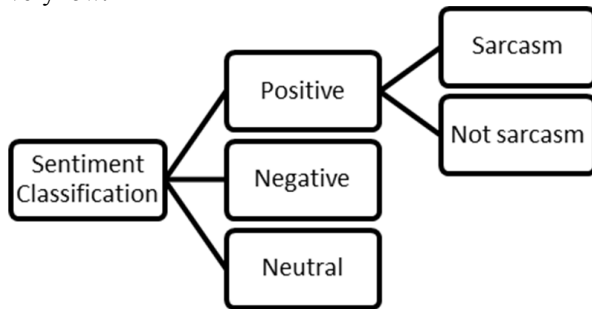

Fig 2. Classification Flow in the Sentiment Analysis System

All the classifications are conducted using several machine learning algorithms such as Naïve Bayes, Maximum Entropy, and Support Vector Machine. These algorithms were chosen because they have shown good accuracy in many text classification task [7].

Although there are a lot of machine learning algorithm has been used, automated sentiment analysis is still a major research area and still have a lot of room of improvement. Naïve Bayes is often used in automated text classification because it is simple and did not need a lot of data compared to another machine learning algorithm [8]. Maximum Entropy algorithm construct a stochastic model that accurately represents the behaviour of the process. Such a model is a method estimating the conditional probability that, given a context, what will the output from the process. Maximum Entropy find the most uniform model for each context [9]. Support Vector Machine maps every feature set into a two dimensional plane and construct a model that based on a linear line that separate the class from the mapped feature set [10].

In the first classification, the feature is the unigram, while in the second classification, the features are unigram, negativity and the number of interjection words.
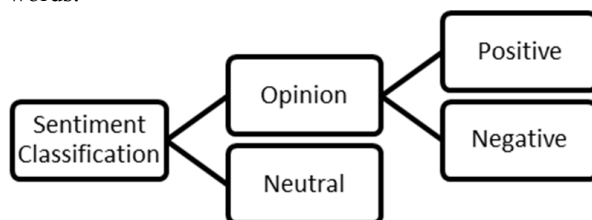

Fig 3. Levelled Method in Sentiment Analysis

Fig 3 describe one of the classification method in sentiment analysis. Rather than directly classify a text into three classes, at first, this method classify a text into opinion and neutral text. After that, the opinion text will be classified into positive or negative class. [3] Has tried to use this method, but they haven't compare those two methods.

III. EXPERIMENTS

A. Experimental data

The data used in the experiment was gathered manually from Twitter. The size of training data is 980 which consist of 502 neutral texts, 250 positive texts and 228 negative texts. The size of testing data is 300 which consist of 200 neutral texts, 60 positive texts and 40 negative texts. The training and testing data were collected from various topics such as politic, food, movie, and public figure.

In order to get the list of the negativity value, we generated it by using the search API from Twitter. We gathered 100 tweets from each topic and labelled the sentiment manually to get the number of positive and negative tweets.

B. Experimental result and analysis

There were three experiments conducted. The first experiment is on the first classification step, and the second experiment is on the second classification step (classification method), and the third classification is on the third classification step (sarcasm classification).

1) Experiments on Sentiment Score

As the first experiment, we evaluated the usage of the translated SentiWordNet in the first classification type which classify each text into 3 classes: positive, negative and neutral. We compared two things in the experiment: using only the lexical of sentiment word; and using the score of the sentiment word.

TABLE 1. Experimental Result on Usage of Translated SentiWordNet

Algorithm	Lexical value	Sentiment Score
Naïve Bayes	73.1%	77.4%
Maximum Entropy	73.2%	78.4%
Support Vector Machine	74.3%	77.8%

The experimental results on Table 1 strengthened other research results on the usage of SentiWordNet which showed that using sentiment score in the classification gave higher accuracy than only using the lexical words. The sentiment score can differentiate the word with low score of certain sentiment and the word with high score of sentiment. By using sentiment score, word with low sentiment score might have been ignored and give result of neutral or opposite class, while for the lexical value, word listed in the vocabulary can't be ignored and may give incorrect result.

For example, in the text "Berlibur di Hanoi Vietnam juga bisa menjadi liburan yang berbeda saat Tahun Baru Cina nanti Travelers" (Vacation at Hanoi, Vietnam can be a different vacation while Chinese New Year), the word "bisa" (can) has a sentiment score of 0.125 which shows a low valued positive sentiment. By using the sentiment score and term weight, the word "bisa" is ignored and it gives the neutral sentiment, which is a correct one. But, by using

only the lexical value, the word "bisa" can't be ignored and it gives the positive sentiment, which is an incorrect result.

2) Experiments on classification method:

The second experiment evaluated the direct and levelled method in sentiment classification. Both of the classification use the sentiment score feature as one of the base feature. The results in Table 2 showed that direct classification gave higher accuracy than levelled method. Both of the methods use the same feature. Here are the list of the feature that used by both method:

1. Unigram
2. Sentiment Score
3. Question words.

TABLE 2. Experimental Result on Direct and Levelled Classification

Algorithm	Levelled	Direct
Naïve Bayes	76.5%	77.4%
Maximum Entropy	76.7%	78.4%
Support Vector Machine	77.3%	77.8%

Table 2 shows that the levelled method give lower result in sentiment classification. If we break down the classification, the positive and negative classification give more than 95% accuracy. The problem lies in the opinion and neutral classification that only give 78% accuracy. In order to improve the accuracy, more feature to detect neutral text is required.

3) Experiments on sarcasm detection.

The third experiment evaluated the additional features of negativity and interjection number in the sarcasm classification accuracy. The results in Table 3 showed that the additional features are effective in sarcasm detection.

TABLE 3. Experimental Result on Usage of Negativity and Interjection for Detecting Sarcasm

Algorithm	Unigram	Unigram + Negativity + Interjection
Naïve Bayes	45.7%	53.1%
Maximum Entropy	47.1%	53.8%
Support Vector Machine	48.5%	54.1%

The additional features also showed that Indonesian people tend to write their critics using sarcasm. As for the low accuracy, we found that there are many sarcasm texts have no global topic. For example, in the text "Men, lu ganteng banget kalo pake dress.p" (Man, you are so handsome using dress). Here, the text topic is not widely known, thus, the negativity feature is useless. There are a quite a lot of text with personal message that can only be analysed if the reader know about the context or know about the writer.

IV. CONCLUSIONS

We have shown that by using the sentiment score, the accuracy of general sentiment analysis is improved for about 4%.And then, the result shown that the negativity feature for detecting sarcasm are quite effective since it increased the accuracy by 6%. Based on our observation of the sarcasm data, we added the features of negativity and number of interjection words. The negativity feature tried to catch the global sentiment value, while the interjection feature represents the lexical phenomena of the text message. In our next research, we will evaluate the method to detect the sarcasm text without global topic and adding some new features to leveled classification to improve the result.

REFERENCES

[1] Gibbs, R. W. and Colston, H. L. 2007. *Irony in Language and Thought*. Routledge (Taylor and Francis), New York.

[2] Naradhipa, A. R. and Purwarianti, A. 2011. *Sentiment Classification for Indonesian Messages in Social Media*. International Conference on Electrical Engineering and Informatics (ICEEI). Bandung. Indonesia

[3] Harlili and Wibisono, Yudi. 2012. *Sistem Analisis Opini Microblogging Berbahasa Indonesia*. Konferensi Nasional Sistem Informasi. Bali. Indonesia

[4] Kreuz, R. J and Caucci, G. M. 2007. *Lexical Influences on the Perception of Sarcasm*. Proceedings of the Workshop on Computational Approaches to Figurative Language. Rochester, New York.

[5] Gonzales-Ibanez, R., Muresan, S., and Wacholder N.. 2011. *Identifying Sarcasm in Twitter: a closer look*. Proceedings of the 49th Annual Meeting of the Association for Computational Linguistics: Human Language Technologies. Portland, Oregon, USA.

[6] Baccianella, S., Esuli, A. and Sebastiani, F.. 2010. *SentiWordNet 3.0: An Enhanced Lexical Resource for Sentiment Analysis and Opinion Mining*. Proceedings of LREC-2010. Valletta.

[7] Pang, Bo., Lee, L., and Vaithyanathan, S.. 2002. *Thumb's up? Sentiment Classification using machine learning Techniques*. Proceeding of the 7th Conference on Empirical Methods in Natural Language Processing (EMNLP-02). Stroudsburg, PA, USA.

[8] Kim S. B., Rim H. C., Yook D. S. and Lim H. S., "*Effective Methods for Improving Naive Bayes Text Classifiers*", LNAI 2417, 2002, pp. 414-423

[9] Berger A. L., Pietra V. J. D., and Della S. A. P., "*A Maximum Entropy Approach to Natural Language Processing*", ACL

[10] Joachims, T. 1998. "*Text Categorization with Support Vector Machine: Learning with Many Relevant Features*". European Centre for Modern Languages.