Negation handling in sentiment classification using rule-based adapted from Indonesian language syntactic for Indonesian text in Twitter

Rizkiana Amalia, Moch Arif Bijaksana, Dhinta Darmantoro
School Of Computing, Telkom University, Indonesia

munief.rizkiana@gmail.com, arifbijaksana@telkomuniversity.ac.id, dhinta@telkom.co.id

Abstract. The presence of the word negation is able to change the polarity of the text if it is not handled properly. It will affect the performance of the sentiment classification. Negation words in Indonesian are 'tidak', 'bukan', 'belum' and 'jangan'. Also, there is a conjunction word that is able to reverse the actual values, as the word 'tetapi', or 'tapi'. Unigram has shortcomings in dealing with the existence of negation because it treats negation word and the negated words as separate words. A general approach for negation handling in English text gives the tag 'NEG_' for following words after negation until the first punctuation. But this may give the tag to un-negated, and this approach does not handle negation and conjunction in one sentence. The rule-based method to determine what words are negated by adapting the rules of Indonesian language syntactic of negation to determine the scope of negation was proposed in this study. With adapting syntactic rules and tagging “NEG_” using SVM classifier with RBF kernel has better performance results than the other experiments. Considering the average F1-score value, the performance of this proposed method can be improved against baseline equal to 1.79% (baseline without negation handling) and 5% (baseline with existing negation handling) for a dataset that all tweets contain negation words. And also for the second dataset that has various numbers of negation words in document tweet. It can be improved against baseline at 2.69% (without negation handling) and 3.17% (with existing negation handling).

1. Introduction
Social media such as blogs, Facebook, discussion forums, Twitter, Instagram, has large users. Twitter as a social media states in their website that the users of Twitter until 30 June 2016 are approximately 1 billion unique visits monthly to the sites with embedded Tweets, 313 million active users each month, and 82% are active users on mobile [1]. However, the structure of the text on Twitter is unstructured texts, in the sense that words are not formal, abbreviations, slang, and misspellings. Sentiment Analysis (SA) can be framed as a text classification task where the categories are polarities such as positive and negative. Many companies use sentiment analysis approach to finding out what is happening with the market. For example the feedback of customers on products released, problems or disorders are often experienced by customers. They give a potential comparison to competitors of the same product and others. Preceded by Pang Lee et.al. [2] perform classification on movie review into two classes, the positive reviews and negative reviews, the results of experiments showing that the use
unigram based (a bag of individual word) as a classification feature goes well with classifiers such as Multinomial Naive Bayes or Support Vector Machine.

Besides the unstructured text, there are several challenges in sentiment analysis fields, one of those is a negation. The important part when handling negation is to determine correctly what part of the meaning expressed is modified by the presence of the negation, this is called the scope of negation. The negations can appear in various forms in opinion reviewed by customers. For example from previous work in Indonesian Twitter Supriyono, Budhi [3], from 20844 tweets there are 4672 tweets or 22.41% from data containing at least one negation term in a tweet. It can be seen that the negation is often used in the tweet opinion, therefore, important to note the effect on the polarity of the tweet.

The most common way is modeling document into a vector space of numerical linear algebra and perform operations on these vectors, such representation is known as the "Bag-Of-Word" or BOW. It uses frequencies of words or terms ("word occurrences") on a document. So it is closely related to the number of occurrences of words in a document. With traditional methods, unigram features, it can not deal the existence of negation, because unigram treats negation words with words that negated as a separate word.

Motivated by this problems, this research is to investigate the influence of negation handling for sentiment classification in Indonesian Twitter, also to handle the existence of transitional words in a sentence, and to enhance the sentiment analysis with the proposed negation handling method at this study. With adapting Indonesian syntactic rules as negation representation capable to handling negation as negated representation in a sentence and can enhance the sentiment classification process. This study is conducted based on the Twitter dataset in telecommunication product domain, specifically, they are the product of PT. Telkom Indonesia (Speedy, Indihome, and UseeTV) and used two classifiers, Multinomial Naive Bayes and SVM.

2. Literature Review

2.1. Related Works

Preceded by Pang Lee et. al [2], they classify documents not by topics but by sentiments, e.g determining whether the review is positive or negative. For negation handling, if a word \( x \) follows the negation word then a new feature ‘NOT_\( x \)’ created tag every word from \( x \) until first punctuation mark. But this method cannot model the scope of negation, because it is heuristically tagging all word until it finds the mark, without concerning with negation words or not. Addition in preprocessing task, mostly the punctuation marks are removed; this is for simplification in preprocessing stage.

For the negation in sentiment, there are some of the researchers that focus on the impact of the negation in sentiment sentences. A survey conducted by Wieghan et. al [4], they survey for negation role in sentiment analysis. They state that effective negation model for sentiment analysis usually requires the knowledge of polar expression. Jia et. al. [5] studied the impact of each occurrence of a negation term in a sentence on its polarity and introduced the concept of scope of the negation term \( t \).

As research in Indonesia, Bojar [6] who conducted research about the resources of the lexicon for Indonesian sentiment also did the negation handling. By adapting the technique from Das and Chen [7] handled the negation of sentiment caused by a negation word. Bojar uses negation words such as ‘tidak’, ‘tak’, ‘tanpa’, ‘belum’, and ‘kurang’. The words that occur between the negation words and the first punctuation after the negation word are tagged with ‘NOT_’. Example, there is a sentence: ‘kameranya kurang bagus gambarnya’ became ‘kameranya kurang NOT_bagus NOT_gambarnya’.

Several studies also concern with the scope of negation, Moral Dadvar et. al.[8], conduct a study dealing with different negation scopes to investigate how it affects the polarity identification of the sentences and assume that opinions are mostly expressed by the use of adjective and adverbs. Hogenboom et. al. [9] they state that for English review sentences, the best performing method is considering 2 words following the negation to be negated. Not only in the English language, in Chinese [10] they have constructed a Chinese corpus and a new syntactic structure-based framework
to identify the linguistic scope of a cue, instead of the traditional chunking-based framework. In Hindi [11], initially annotated dataset to test the proposed algorithm was conducted then the rules are devised for handling negation with static windows (3 to 5 words). And in Arabic [12], they stated that sentiment is related to culture and language morphology, so they also defined a set of rules that capture the morphology of negations in Arabic. These rules are then used to detect sentiment taking care of negated words. It can be seen that linguistic features that define the structure or syntactic from languages may influence to determine the polarity of sentiment.

2.2. Bag-of-Words and TFIDF Weight
Model representation of a document or text most frequently and widely used is the VSM (Vector Space Model) or known as BOW (Bag Of Words). BOW ignores the order of the words in a document and disregards all grammar. The exact ordering of the terms in a document is ignored but the number of occurrences of each term is material [13]. However, it performs generally very well in text-based classification. Another approach besides calculates term frequency, there are Term Frequency-Inverse Document Frequency (TF-IDF). Since each word has a different degree of importance in the document, for each word it is given an indicator, i.e the term weight. It can be used for reduction of the term dominance that often appears in various documents. This is necessary because the term that appears in many documents, can be considered as a common term so that it does not matter its value. Conversely, the term scarcity factor in the document collection should be considered in weighting.

$$\text{Weight}(i) = \text{tf} \cdot \text{idf}(i) = \text{tf} \cdot \log \frac{N}{df(w)}$$

Where $\text{tf}$ is the term frequency or a number of a term $(w)$ in a document, $\text{idf}$ is the frequency of the document or the number of documents that contained the term $(w)$ in it, $N$ is total documents, and $Df$ is the number of the document that contained the term $(w)$.

2.3. Multinomial Naive Bayes

Naive Bayes classification is one of classification methods based on applications of simple possibility Bayes theorem. It is assumed that the presence or absence of a particular occurrence of a group is not associated with the presence or absence of other events. Bayes theorem is used to calculate the probability of occurrence of an event based on the effect obtained from the observation. Bayesian theory can also be used as a decision-making tool to renew the confidence level of the information. They can be formulated as,

$$p(C_k|x) = \frac{p(x|C_k) \cdot p(C_k)}{p(x)}$$

Where $x$ is a vector that representing some $n$ features (independent variables), it assigns to this instance probability $p(C_k|x_{1}, \ldots, x_{n})$ for each of $k$ possible outcomes or classes $C_k$. In text classification, our goal is to find the best class for the document. The best class in Multinomial Naive Bayes classification is the most likely or maximum a posteriori (MAP) decision rule. A Bayes classifier is the functions that assign a class label $\hat{y} = C_k$ for some $k$ as follow,

$$\hat{y}(\text{predicted class}) = \arg\max_{k \in \{1, \ldots, K\}} p(C_k) \prod_{i=1}^{n} p(x_i | C_k)$$

Naive Bayes Classifier is a general term which refers to the conditional independence of each of the features in the model, while Multinomial Naive Bayes classifier is a specific instance of a Naive Bayes classifier which uses a multinomial distribution for each of the features. The term Multinomial Naive
Bayes simply lets us know that each \( p(x|\mathcal{L}_k) \) is a multinomial distribution, rather than some other distribution. This works well for data which can easily be turned into counts, such as word counts in the text. Multinomial Naive Bayes can be used for various purposes including for the classification of documents, spam detection or filtering spam, and sentiment classification [14].

2.4. Support Vector Machine
The aim of SVM is to find the best classification function to distinguish between classes in the training data and to produce a model based on training data which predicts the target values of the test data given only the test data attributes [15]. In linear data set, where the data can be separated linearly, example with training data set of the instance label pair \( (x_i, y_i), i = 1, \ldots, l \) where \( x_i \in \mathbb{R}^n \) and \( y \in \{1, -1\}^l \) is the label class from \( x_i \). Because there are many such linear hyperplanes, SVM additionally guarantees that the best function is found by maximizing the margin between the two classes. Mathematically, the SVM optimization problem formulation for the case of linear classification in primal space is:

\[
\min \frac{1}{2}\|w\|^2
\]

subject to

\( y_i(w x_i + b) \geq 1, i = 1, \ldots, n \)

To solve nonlinearity problems that often occur in real cases, the kernel method is applied. The kernel method provides an alternative approach by mapping data \( x \) from the input space to feature space \( F \) through a function \( \varphi \) then \( \varphi: x \rightarrow \varphi(x) \). The kernels in this are used linear and RBF. Much like research on text classification using SVM before [16]. And also in [15] stated that RBF kernel function has the same performance with linear SVM on certain parameters.

2.5. Negation in Indonesian
The negations words (tidak, bukan, belum, jangan) not only used to express the denial or inverse the sentences but also in conjunction with the yes or no questions. Besides the direct negation marker, there are also another words that indirectly can invert the meaning of sentences even it is not there negation marker, such tetapi or tapi.

2.6. Negation Handling
From literature survey, many studies have been performed with respect to sentiment classification problem at the sentence or phrase level approach, including handling the negation in subtasks of sentiment analysis. It becomes important because, in traditional methods such unigram, two sentences can end up with same VSM representation, for example, sentences “responnya cepat sekali” with “responnya gak cepat sama sekali”. This two sentences may be represented with same feature vector, and it can be leading to misclassified. Some ways in the previous work to handle negation in English are by determining the scope of negation first that have been done, in [8], [7] Das and Chen proposed using tagging “NOT_”, or using static windows [2], [9]. For scope of negation using fixed static window has been tested in various number, start with simplest way to invert the polarity of the sentiment word directly following the negation word [17], 5 next words to the negation are inverted [18], or from three to six words following after negation [19]. Not only in English language, research about negation effect in sentiment classification also has been done for another language [10],[11],[12].

3. Baseline & Proposed Methods
3.1. Baseline System
There was two baselines system used as a comparison to see how far the handling effect of negation. First the baseline without negation handling. It was the experiment where there was not a treatment for any negation in the text, and it was called B-NEG for the experiment. The second baseline with existing negation handling method, it was the experiment to determine the scope of negation in order to negation handling using tag 'NEG_' for words between negation and first punctuation mark (period, comma, question marks). And it was called B+NEG for the experiment.

3.2. Dataset
There were two datasets crawled and labeled manual from Twitter using the keyword. First dataset (all_negation), were all tweets containing negation or conjunction at least one word. A total number of negation tweets 300 tweets, consists of 128 negative tweets, 100 non-opini tweets, and 72 positive tweets. The second dataset (training_negation) contained tweets that did not all contain negations or conjunctions. Consists of 100 negative tweets, 102 non-opini tweets, and 98 positive tweets. With 30% (~90 tweets) contains negation words. The domain was in telecommunication Telkom product, Speedy, and Indihome. For testing data, we applied split in a dataset with 80:20 composition. 80% for the training set and 20% for testing data for each dataset. Negation handling applied after preprocessing phase.

3.3. Proposed Methods
The proposed method experiments were determined the scope of negation using linguistic approach, by defining the set of rules based on Indonesian syntactic (see table 1). In Indonesian grammars referring to [20], there are four negative words: ‘tidak’, ‘bukan’, ‘belum’ and ‘jangan’. And also from a research about comparison negation for English and Indonesia, [21] and [22] stated that Indonesian negation markers come after various syntactic categories such as noun, adjective, adverb, prepositional phrase. In addition, heuristic rules based on Part-of-Speech from [23] (JJ_NN and VB_NN forms) were used. The syntactic grammars were adapted from these references. We used this rules with 2 different experiments way to determine and treat negations in the text (see Figure 1).

- Proposed rules as negation term, called it with Neg_Term for the experiment, a pair or set of words that obtained from rules were inserted into the text as a presentation from negation.
- Proposed rules as the scope of negation, called it with Neg_Term+NEG for the experiment, a pair or set of words that obtained from rules were given the tag 'NEG_' as a presentation from negation.

Example raw text: ‘@TelkomCare Bukan salah Telkom Speedy tapi PLN yang padam kebetulan UPS dan Genset ku rusak.’

Preprocessing: ‘bukan salah telkom speedy tapi padam genset rusak’
Negation handling (Neg_Term): ‘bukan_salah telkom speedy tapi_padam genset rusak’
Negation handling (Neg_Term+NEG): ‘bukan NEG_salah telkom speedy tapi NEG_padam genset rusak’

| Negation Words      | Adjacent  | Examples           |
|---------------------|-----------|--------------------|
| Tidak               | Adjective – JJ | ('tidak', u'NEG'), ('adil', u'JJ')) |
|                     | Verb – VB  | ('tidak', u'NEG'), ('beri', u'VB')) |
|                     | Noun – NN  | ('tidak', u'NEG'), ('masalah', u'NN')) |
|                     | Indefinite numbers – CD | ('tidak', u'NEG'), ('banyak', u'CD')) |
|                     | Adjective followed by Noun – JJ_NN | ('tidak', u'NEG'), ('jelas', u'JJ'), ('terimakasih', u'NN')) |
|                     | Verb followed by Noun – VB_NN | ('tidak', u'NEG'), ('paham', u'VB'), ('sistem', u'NN')) |
|                     | Modals (bisa,dapat, | ('tidak', u'NEG'), ('dapat', u'MD'), |
4. Result
The final result of the experiment was the classification process done by machine learning approach, using Multinomial Naive Bayes and Support Vector Machine. Performance measurements performed measuring the average of F1-score (percentage) and accuracy (percentage).

Table 2. Performance based on average F1-score

| Experiments      | All_negation M-NB | SVM | Training_negation M-NB | SVM |
|------------------|-------------------|-----|------------------------|-----|
| B - Neg          | 85.03%            | 84.95% | 83.08%                | 83.73% |
| B + Neg          | 80.01%            | 81.74% | 82.22%                | 83.25% |
| Neg_Term         | 81.67%            | 81.83% | 84.93%                | 85.23% |
| Neg_Term + Neg   | 84.96%            | 86.74% | 86.26%                | 86.42% |

Table 3. Performance Based on Accuracy

| Experiments      | All_negation M-NB | SVM | Training_negation M-NB | SVM |
|------------------|-------------------|-----|------------------------|-----|
| B - Neg          | 85.00%            | 85.00% | 83.33%                | 83.33% |
| B + Neg          | 80.00%            | 81.67% | 83.33%                | 83.33% |
| Neg_Term         | 81.67%            | 81.67% | 85.00%                | 85.00% |
| Neg_Term + Neg   | 85.00%            | 86.67% | 86.67%                | 86.67% |

4.1. Classification Performance
In Table 2, it can be seen in all_negation dataset for our proposed methods there are no increases in average F1-score when compared to the baseline without negation handling B-NEG with Naive Bayes classifier, but compared to baseline negation handling B+NEG our proposed method there is an increase of 1.66% (Neg_Term) and 4.95% (Neg_Term+NEG). While with SVM classifier our proposed method Neg_Term+NEG seen there was a better improvement when compared with both baseline, B-NEG (1.79%) and B+NEG (5%).

And for training_negation, in this type of data, both of our proposed methods can increase average F1-score performance using Naive Bayes classifier at 1.85% (Neg_Term) and 3.18% (Neg_Term+NEG) against B-NEG and 2.71% (Neg_Term) and 4.04% (Neg_Term+NEG) against B+NEG. The same result for both our proposed method and SVM classifier compared with baseline B-NEG was 1.5% (Neg_Term) and 2.69% (Neg_Term+NEG), and compared against with baseline B+NEG was 1.98% (Neg_Term) and 3.17% (Neg_Term+NEG). From experiment results based on F1-score and accuracy, our proposed methods have no significant effect on Naive Bayes classifier in all_negation data. But has improved with using SVM classifier, it also for another dataset.

5. Discussion
In the first dataset, all_negation, each tweet document contained a component of negation word, so negation had an almost be spread evenly distributed across the document tweets. It is possible that the negation words forming a new form (tagging 'NEG_' or negation_term) will be present in multiple document tweets and distributed to in 3 classes. For Multinomial Naive Bayes, it has to deal with the count of the word for each class and the total count off of all words in the class, which is a calculation
of conditional probability or likelihood events for each term. TFIDF weighting is closely related to the number of terms in a document belonging to a particular class (TF) and the number of terms in the entire document (IDF). Like the word ‘bisa’, there are 98 words ‘bisa’ in 300 document tweets in all_negation dataset. For example, in B+NEG experiment, there were 55 ‘bisa’ and Neg_Term+NEG had 49 ‘bisa’ which was negated into ‘NEG_bisa’ which could be in positive, negative, and non_opinion tweets. The calculation of the total TFIDF weight of the words ‘bisa’ and ‘NEG_bisa’ in the entire training document would be not much different. So when there was data testing that contained the word ‘NEG_bisa’, the term weight value in conditional probability for each given class had no significant effect. As if the process without negation handling with negation handling result was not significant.

While in SVM, although experiment with negation handling (B + NEG and Neg_Term) the F1-score average score was lower than experiment without negation handling, but the F1-score average on Neg_Term + NEG was better than the other three. Through TF-IDF the important words which have used to train SVM model were obtained. The accuracy of the model generated from the training process with SVM was highly dependent on the kernel function and the parameters used. It was different from Multinomial Naive Bayes which was treated the features as independent, whereas SVM looks at the interactions between them to a certain degree. But in the second dataset, training_negation, the more varied data the number of negations in the tweets, negation handling more visible influence. Since the negation representation word distribution had a smaller DF if it was compared to all_negation data. So the difference in the weight of the word representation negation of the document looks more significant for a particular class.

In addition, when reanalyzed on the implementation of syntactic rules for proposed methods on both datasets, there were some tweets that despite the word negation but unsuccessful syntactic rules apply. These were 32 tweets (10.67%) in the all_negation dataset and 8 tweets (2.67%) in the training_negation dataset. The tweets that were not applied by syntactic rules were caused by the following:

- The position of the word negation was at the end of the sentences.
- The word after the negation had a POS-Tag which was not included in the syntactic rules.
- The unigram POS-tag had default tag ‘NN’ for unrecognized tokens through tagging training.

6. Conclusions & Future Work

6.1. Conclusions
The traditional method of negation handling using the ‘NEG_’ tag for words after negation until the first punctuation, there will be irrelevant words that were negated. In this study, two ways to handle negation were proposed namely by representing negation and word negated to determine the scope of negation. Based on experiment our proposed methods, did not work well with Multinomial Naive Bayes in the data that had a negation word distribution almost across the tweet document, but it had increased compared to the baselines (without negation handling and negation handling existing) in the data whose amount of negation did not dominate the overall tweets (in this study, percentage of the word negation is 30% tweet of the document tweet). For overall, the second proposed methods with adapting syntactic rules and tagging “NEG_” using SVM classifier with RBF kernel had better performance results than the other experiments. The RBF kernel hyperparameters from 10-fold cross validation auto-tuning process were obtained (C = 100 and gamma = 0.001) for both datasets. Considering the best average F1-score value was 86.74% for the first dataset that all tweets contain negation words and 85.42% for the second dataset that had the various number of negation words in document tweet. The performance of proposed method can be improved against the baseline, equal to 1.79% (without negation handling) and 5% (with existing negation handling) for the first dataset. And also for the second dataset, it can be improved against baseline at 2.69% (without negation handling) and 3.17% (with existing negation handling).
6.2. Future Work
Based on the findings and the conclusion of the study, there are some recommendations. In the future, the handling of negations may be done on another domain for the Indonesian language. But with resource adjustments such as dictionaries and datasets used. And also by using a complete dictionary or automatic translator for deep preprocessing. Because a good preprocessing process can improve the performance of negation handling better. Meanwhile, POS-tagging is an important support task in the implementation of syntactic rules, in the future it can be tested POS-tagging method other than unigram tagger for Bahasa Indonesia.

References

[1] “Twitter Corp.,” [Online]. Available: https://about.twitter.com/company.

[2] B. Pang, L. Lee and S. Vaithyanathan, “Thumbs Up? Sentiment Classification Using Machine Learning Techniques,” Proceedings of the ACL-02 conference on Empirical methods in natural language processing - EMNLP '02, vol. 10, pp. 79-86, 2002.

[3] B. Supriyono, “WEB DATA MINING FOR CUSTOMER’S SENTIMENT CLASSIFICATION FOR TELKOM SPEEDY USING TWITTER IN INDONESIAN,” no. August 2015.

[4] M. Wiegand, A. Balahur, B. Roth, D. Klakow and A. Montoyo, “A Survey on the Role of Negation in Sentiment Analysis,” Proceedings of the Workshop on Negation and Speculation in Natural Language Processing, no. July, pp. 60-68, 2010.

[5] L. Jia, C. Yu and W. Meng, “The effect of negation on sentiment analysis and retrieval effectiveness,” Proceeding of the 18th ACM conference, no. c, pp. 1827-1830, 2009.

[6] Franky, O. Bojar and K. Veselovská, “Resources for Indonesian Sentiment Analysis,” The Prague Bulletin of Mathematical Linguistics, vol. 103, no. 1, pp. 21-41, 2015.

[7] S. Das and M. Chen, “Yahoo! for Amazon: Sentiment Extraction from Small Talk on the Web,” Management science, vol. 53 (9), 2004.

[8] M. Dadvar, C. Hauff and F. D. Jong, “Scope of Negation Detection in Sentiment Analysis,” Proceedings of the Dutch-Belgian Information Retrieval Workshop (DIR 2011), pp. 16-20, 2011.

[9] A. Hogenboom, P. Van Iterson, B. Heerschop, F. Frasincar and U. Kaymak, “Determining negation scope and strength in sentiment analysis,” Conference Proceedings - IEEE International Conference on Systems, Man, and Cybernetics, pp. 2589-2594, 2011.

[10] B. Zou, Q. Zhu and G. Zhou, “Negation and Speculation Identification in Chinese Language,” Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing, pp. 656-665, 2015.
[11] N. Mittal and B. Agarwal, “Sentiment Analysis of Hindi Review based on Negation and Discourse Relation,” Sixth International Joint Conference on Natural Language Processing, pp. 57-62, 2013.

[12] R. M. Duwairi and M. A. Alshboul, “Negation-Aware Framework for Sentiment Analysis in Arabic Reviews,” Proceedings - 2015 International Conference on Future Internet of Things and Cloud, FiCloud 2015 and 2015 International Conference on Open and Big Data, OBD 2015, pp. 731-735, 2015.

[13] C. D. Manning, P. Raghavan and H. Schutze, “Chapter 6: Scoring, term weighting, and the vector space model,” in An Introduction to Information Retrieval, Cambridge, England, Cambridge Press, 2009, pp. 109 - 134.

[14] S. Raschka, “Naive Bayes and Text Classification: Introduction and Theory,” 2014.

[15] W. H. Chih, C. C. Chih and J. L. Chih, “A Practical Guide to Support Vector Classification,” BJU International, vol. 101, no. 1, pp. 1396-400, 2008.

[16] T. Joachims, “Text Categorization with Support Vector Machines: Learning with Many Relevant Features,” Machine Learning, vol. 1398, no. LS-8 Report 23, p. 137–142, 1998.

[17] P. V. Iterson, B. Heerschop, A. Hogenboom, F. Frasincar and U. Kaymak, “Accounting for Negation in Sentiment Analysis,” DIR, no. February 2011.

[18] G. Grefenstette, Y. Qu, J. Shanahan and D. Evans, “Coupling Niche Browsers and Affect Analysis for an Opinion Mining Application,” In Proceedings of the 12th International Conference Recherche d'Information Assistee par Ordinateur, pp. 186-194, 2004.

[19] R. Narayanan, B. Liu and A. Choudhary, “Sentiment analysis of conditional sentences,” Proceedings of the 2009 Conference on Empirical Methods in Natural Language Processing Volume 1 EMNLP 09, no. August, p. 180, 2009.

[20] J. N. Sneddon, Indonesian A Comprehensive Grammar, 1996.

[21] D. N. Syafar, “NEGASI DALAM BAHASA INDONESIA DAN BAHASA INGGRIS,” Jurnal Arbitrer, vol. 3, no. 1, April 2016.

[22] R. Rakhmania, “A COMPARISON BETWEEN ENGLISH AND INDONESIAN NEGATION MARKERS: A SYNTACTIC STUDY,” 2012.

[23] U. Farooq, H. Mansoor, A. Nongaillard, Y. Ouzrout and M. Abdul, “Negation Handling in Sentiment Analysis at Sentence Level,” Journal of Computers, vol. 12, no. 5, pp. 470-478, 2016.