Combining Likes-Retweet Analysis and Naive Bayes Classifier within Twitter for Sentiment Analysis

Rizal Setya Perdana and Aryo Pinandito
Information System Department, Computer Science Faculty, Universitas Brawijaya, Malang, Indonesia.
rizalespe@ub.ac.id

***Abstract*—Sentiment analysis is a research study that aims to extract subjectivity of opinions. Due to massive growth number of user generated content in social media, Twitter is one of the most popular microblogging application which user is freely to discuss and share opinions about specific topic or entity. Twitter have several features that potentially can be used to improve sentiment analysis such as like and retweet. Like and retweet are mechanism in Twitter to propagate or share and to show appreciation of other user posting. This paper proposes a combination of textual and non-textual features to improve performance of sentiment prediction. In this research we apply Naïve Bayes for textual classification and Fisher Score to determine non-textual (like and retweet) features. By combining two kinds of features, our experimental find the optimal value of α and β. The evaluation performance using F1-measure gives 0.838 of accuracy with α and β are 0.6 and 0.4 respectively.**

***Index Terms*—Sentiment Analysis; Twitter; Naive Bayes; Retweet-Like.**

I. INTRODUCTION

The massive growth and utilization of internet application such as social media activity has change the way of people in communicate. Such condition has led to generate large amount of data and boost the information spread among people through the mechanism of social media. Twitter is one of the most popular social media microblogging sites which users use it to share their opinions, attitudes and emotions of an entity in 140 characters' length of text. Currently, Twitter has 320 million active users that generate over 500 million users generated text (Tweet) per day [1]. Large amount of Tweets data can be analyzed to generate new information for building application such as crowd traffic monitoring, detecting an event, sentiment analysis [1][2][3], etc. Twitter is considered as a worthy subject for sentiment and subjective opinions analysis because people free to discuss and share their opinion about specific topic, entities, or event. Sentiment Analysis or opinions mining is a study of calculating and finding the polarity of people opinions and emotions toward events or topics [4].

There are several applications that implement sentiment analysis process such as product review, market prediction, political sentiment determination, equality value prediction, box office prediction [5]. Several recent works in sentiment analysis using statistical approach or machine learning algorithms applied to annotated that a document or a Tweet has positive or negative sentiment based on containing words. Such approaches have some problems in the limited availability of known sentiment word and different subjectivity opinion between people.

Tweets or status updates are basic object of all things in Twitter which contains text, images, videos, link, hashtag and mentions. Beside its text, a Tweet has many other attributes or features that can be used as a mining object to gain new considerations such as information about who is the contributors, the language used by the user, time the Tweet was created, user who share the Tweet, entities, marker that Tweet is liked, number of user like the Tweet, geolocation (longitude and latitude) or location where the Tweet is posted, identity number of a Tweet, indication that the Tweet is shared or not, number of times the Tweet shared, text, user etc. From all these attributes we analyzed that some attributes have possibility to improve the classification process in sentiment analysis in Twitter document.

Retweeting is an activity to propagate or share a Tweet from other user's Tweet to our home timeline, the effect of this activity is people who follow us will also read the related tweet. This is a repost feature from Twitter that helps the user quickly and easily share that Tweet with all of people who follow the user. It is possible to retweet user's own tweets or tweets from someone else [6]. Retweet is a Twitter's feature which is represented by a small arrows-shaped icon, to retweet or repost a Tweet user can tap or click this button. The small arrows-shaped icon will turn to green which indicate that the user has repost or Retweeted this Tweet. For some purposes, Retweet is a tools for propagation of information through mechanism which causes increase of popularity of the Tweet [7]. High popularity of a Tweet is indicated by the number of Retweet which can be used to improve detecting task of sentiment in a Tweet.

Likes is a new feature in Twitter which is represented by a small heart-shaped icon and used to show appreciation and agreement for an event or Tweet [8]. Tweets that appear in home timeline will be read and user has their own perception about a status updates. To like a Tweet, user can click or tap the heart-shaped icon and it will turn the color to red, confirming that user has like the Tweet. The number of Likes indicate the number of user that like or agree with the content of the Tweet. The heart-shaped icon button to express like or agreement is a universal symbol that has same meaning across countries, cultures, languages and time zones.

Naive Bayes is a machine learning classification method which is utilized for supervised and statistical learning concept. This algorithm works based on Bayesian Theorem with high independent assumption and simple probabilistic classification [9]. Classification algorithm uses pre labeled data or training set data which is labeled as positive, negative,

or neutral to learn the characteristic of data groups or classes. In sentiment analysis, class label for each tweet is used to differentiate positive from negative sentiment for predicted or non-labeled Tweets. The feature set we considered for the classification of Tweet is text of Twitter.

The purpose of this paper is to improve sentiment analysis task by combining Naive Bayes Tweet classification and Like-Retweet features analysis to determine the polarity of the tweets. Like and Retweet features are not included in classification process due to the specific of these features characteristic in sentiment analysis.

This paper is organized as follows. Section II addresses to describe the theory and concept that is used behind the developed work about main concept sentiment analysis, text mining techniques process that preparing the data from raw Twitter data until ready to mining, Naive Bayes classification process, and the detail about Like and Retweet in Twitter. Section III illustrates the proposed combining method and respective function of each part. Section IV proposes the validation procedure used to validate the proposed combining strategy. Section V gives a brief statement from the provided work, conclusion, and proposal for the future work.

II. Literature Review

A. Sentiment Analysis

Available online textual information is classified in two categories: fact data and sentiment data [10]. Fact data are the objective type of information that is the result of observation, measurement, or capturing an event, while sentiment data is subjective term of individual's opinion. Analysis of sentiment is a computation research area of extracting the polarity of opinions between classes (positive, negative, or neutral) from text document. This process tries to recognize and classify different sentiment on textual document which is applied to specific product, event or topic as positive, negative, or neutral. Various text documents that potentially contains useful subjective information such as generated text by social media users, internet forums, discussion groups, product reviews, blogs, etc. Sentiment analysis task of text document has three main block component such as the subject of analysis, the sentiment, and the topic about the subject will talk about. The main task of sentiment analysis in textual document is to classify each member of document set $D = \{d_1, d_2, d_3, \dots, d_n\}$ into defined class $C = \{positive, negative\}$. But in other research there is multi-class classification that have more than two classes used to classify the sentiment textual information. Sentiment analysis research area is formed by multi-disciplined researches including machine learning, information retrieval and natural language processing. In this paper, sentiment analysis will utilize supervised machine learning algorithm and combined with analysis of specific features in Twitter.

B. Text Mining

In this research, textual data from Twitter will be processed in text mining steps manners. Text mining is important step of data mining or knowledge discovery process which the resources are from semi-structured to unstructured data to extract previously unknown information [11]. Standard text mining process starts with data acquisition or collecting document from various resources than resulting document collection. Preprocessing textual data in document collection is needed to normalize the document format and to clean the text contained. Several processes in text preprocessing such as tokenization, case folding, stemming and stop word removal. Then, applied text mining techniques will generate model that is ready to be used for analysis of text. Analysis of text can be done by machine learning algorithm or lexicon based approach. The result of overall text mining process is the discovery of unknown information from collection of document. Figure 1 shows overall text mining workflow.

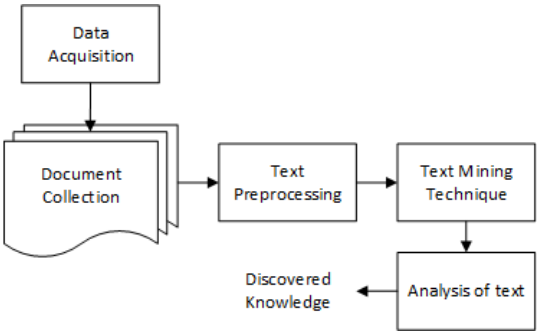

Figure 1: Text mining process

C. Naive Bayes Classification

An increasing number of machine learning algorithm for classification is broadly used for document classification problem. Naive Bayes is one of machine learning classification algorithms which apply Bayes theorem to determine unknown class label based on set of data with predefined class label. It has been widely used for text classification due to its efficiency and simplicity for computation [12]. Naive Bayes work with conditional probability of $P(c_i \mid d_j)$, document d_j has been assigned by defined class c_i. According to Bayesian theorem, the probability of document d_j has class c_i can be calculated in Equation (1) as follows:

$$P(c_i \mid d_j) = \frac{P(d_j \mid c_i)P(c_i)}{P(d_j)} \quad (1)$$

In classification, text document d_j can be regarded as a tuple of words $\langle w_1, w_2, w_3, \dots, w_n \rangle$ which the word occurrence frequency is assumed as random variable with particular probability distribution. The goal of text document classification is finding maximum values of probability of each word w_i in class c_j as follows in Equation (2):

$$C_{MAP} = \arg\max_{c_j \in C} P(c_j | w_1, w_2, \dots, w_n) \quad (2)$$

Bayesian theorem in Equation (1) can be applied to Equation (2) to comply the word containing it as follow:

$$C_{MAP} = \arg\max_{c_j \in C} \frac{P(w_1, w_2, \dots, w_n \,|c_j)\, P(c_j)}{P(w_1, w_2, \dots, w_n)} \quad (3)$$

Which we can ignore the probability value of words occurrence $P(w_1, w_2, \dots, w_n)$ in a class due similar value in every class, so we can modify the Equation (3) as follow:

$$C_{MAP} = \arg\max_{c_j \in C} P(w_1, w_2, \dots, w_n \,|c_j)\, P(c_j) \quad (4)$$

With the assumption that each word in $\langle w_1, w_2, w_3, \dots, w_n \rangle$ are independent, $P(w_1, w_2, \dots, w_n \,|c_j)$ in Equation (4) can be rewritten as:

$$P(w_1, w_2, \dots, w_n \,|c_j) = \prod_i P(w_i|c_j) \tag{5}$$

Using Equations (4) and (5) we obtain:

$$C_{MAP} = \arg\max_{c_j \in C} P(c_j) \prod_i P(w_i|c_j) \tag{6}$$

Probability of number of document in each class with all available pre-labeled training document $P(c_j)$ can be written in formula as follow:

$$P(c_j) = \frac{|doc_j|}{|D|} \tag{7}$$

where:

$|doc_j|$: Number of document which has class j

$|D|$: Total number of all document in training set which used to generate classifier model.

Probability of word i in class j, $P(w_i|c_j)$ is calculated as:

$$P(w_i|c_j) = \frac{n_k + 1}{n + |vocabulary|} \tag{8}$$

where:

n_k : Frequency of word w_i occurrence in a document with class c_j

n : Number of all distinct words in all document which has class c_j

$|vocabulary|$: Number of all distinct words in all training document

Generally, text document classification workflow is depicted in Figure 2, started by generating classification model (training). This process will generate term or word vector for classifying or predicting the label for new non-labeled document.

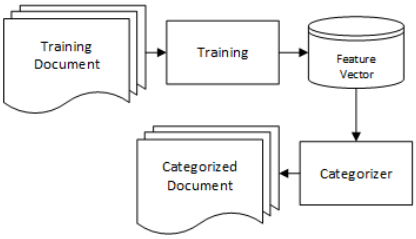

Figure 2: Document classification workflow

D. Like and Retweet in Twitter

Retweet feature enables Twitter's user to propagate their tweet or other user tweet, this mechanism also known as information diffusion between users. Most users tend to share their favorite tweet to their followers, therefore retweet can also be viewed as an important signal of user interest and needs [13]. In this research, retweet number will be analyzed due to the reason of why user retweeting a tweet to improve the traditional classification algorithm. Different from retweet, like feature show the agreement about the content and we can see the list of tweets which we like before. The number of like which symbolized by heart shaped icon has meaning that the tweet is liked by some number of Twitter users. Figure 3 represent a tweet from user @Thomas1774Paine has 998 of retweet and 1.4 K number of user that like this tweet. These two features will be used in this research to improve sentiment analysis classification due to limited number of labeled sentiment tweet.

Figure 3: Screenshot of a tweet with the number of retweet and like

III. METHODOLOGY

A. Data Collection

We obtain 17349 tweets collected from Twitter by writing Python script to access raw data through API (Application Programming Interface) GET search/tweets that is provided for researcher or developer. Tweets are extracted from several keywords related to nominees of Jakarta Gubernatorial Election on February 2017 in Bahasa Indonesia. Data collected during in range one month on October 2016 for 4 nominees which is we get approximately 3500 tweets each nominees. Manual annotating by human labeled each tweet as positive, negative, and unknown. The 'unknown' label indicates that the process of manual annotation is not clear between negative nor positive label sentiment. From manual annotation we use only 14367 number of positive and negative label only for generating classification model. The dataset was split randomly for each nominee into two groups of ratio 3:1 for training and test set respectively.

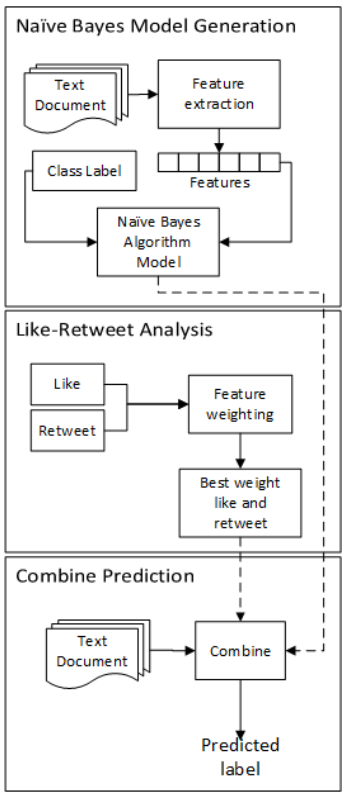

Figure 4: Flow diagram of combining process of Naïve Bayes and Like-Retweet

B. Feature Extraction

In classification of text mining, features of data is represented by the number of distinct words or terms in all training documents. Feature extraction is a process which is defying big amount of feature that causing low performance and quality of prediction [14]. Useless feature or words will be reduced at classification model generation stage. There are several steps in process of feature extraction, we use in this research as follows:

1. **Cleaning –** All text contained in tweet is standardized by character standard encoding like ANSI and Unicode UTF-8, otherwise will be removed from this step. Punctuation, number, and special tags are also removed in this step for further process.
2. **Tokenization –** In statistical classification, term is recognized as symbol of feature which we do not consider the meaning of word. Tokenization has aim to separate or distinct every term (separated by blank space) in a document for statistical manners, so at the end of this process we have arrays of term.
3. **Case folding** – This process is aimed to convert capitalized letter of words to lower case.
4. **Stop word removal** – Stop word list is an array of common word or term which occurrence will not represent the meaning of a document. In this research we have 758 stop word from previous research in [15].
5. **Stemming** – The function of stemming is to decrease relevant term or word into a single form. Stemming is a process to reform a word into its root form. The main procedure of stemming is recognizing and eliminating prefix or suffix of a word. In this research, we use stemming algorithm from a research proposed by Arifin and Setiono for Bahasa Indonesia in [16].
6. **Term weighting** – This research uses term frequency (tf) combining with inverted document frequency (idf) for weighting term in a document. The principal process of $tf - idf$ is scoring each term by the occurrence frequency in a document and the distribution level of term in some documents. $tf_{t,d}$ is number of occurrence of term t in document d, tf weight is calculated as follows:

$$tf_{t,d} = \begin{cases} 1 + \log_{10} tf_{t,d}, & tf_{t,d} > 0 \\ 0, & otherwise \end{cases} \quad (9)$$

idf_t is degree of term distribution across document, it also defining the probability number of document that containing term t against the number of all documents in collection of tweet N, which can be write as formula as follow:

$$idf_t = \log_{10} \frac{N}{df_t} \quad (10)$$

Using Equations (9) and (10) the formula of term weight W_t of term t can be obtain as:

$$W_t = tf_{t,d} \times idf_t \quad (11)$$

7. **Index generation** – In this process, all training data will processed from step 1-6 to generate matrix term weight w_t for each term t in documents d.

After the extraction of the features, we apply algorithm for generating model classification to classify tweets document.

C. Naïve Bayes Model Generation

In standard Naïve Bayes document classification, $P(w_i|c_j)$ the probability of occurrence word i has class j in Equation (8), the value of n_k is equal to term frequency tf. Standard Naïve Bayes can be improved by applying $tf - idf$ transformation to word transformation instead of using tf [17]. The result of applying modification of Equation (8) is formulated as follow:

$$P(w_i|c_j) = \frac{tf - idf(n_k) + 1}{n + |vocabulary|} \quad (12)$$

where:

$n_k, n, |vocabulary|$: Explained before in Equation 8 respectively.

The representation of classification model generation based on Equation (12) will generate matrices of term and class which the value is probability occurrence of term i in class j. Actually, this standard Naïve Bayes is not the final result of predicting sentiment analysis of a tweet. We need to get the separate probability score of a tweet from each class $C = \{positive, negative\}$ for further process, so we need to skip calculating argument to maxima $argmax$ in Equation (6).

D. Like-Retweet Analysis

Like (LK) and retweet (RT) are non-textual features in Twitter data objects. These two numerical features are potentially used in classification through sentiment analysis task due to its function related to a tweet. We decide to not join *LK* and *RT* as features together with textual classification due to the difference of such of two approaches. We assumed that the rise of number LK and RT of a tweet reinforce tweet has positive sentiment. To measure the weight of LK and RT feature, we use Fisher Score (FS) statistic tool to estimate score or weight in each feature which indicates the degree of features importance [18]. In this research, FS is calculated independently for each feature using formulation as follow [19]:

$$FS(x_j) = \frac{\sum_{k=1}^{c} n_k(\mu_k^j - \mu^j)^2}{(\sigma^j)^2} \quad (13)$$

where:

n_k : Size of class k
μ_k^j : Mean of feature j in class k
μ^j : Mean of all class in feature j
σ^j : Standard deviation of whole data in feature j

The result of this process are W_{LK}, W_{RT} weight of like and retweet feature respectively, non-textual feature score $NonTxt$ for each unlabeled $tweet_j$ which has number of like n_{LK} and number of retweet n_{RT} can be calculated as follows:

$$NonTxt(tweet_j) = (W_{LK} \times n_{LK}) + (W_{RT} \times n_{RT}) \quad (14)$$

E. Combine Prediction

The last block process is about combining the two kind of feature, textual and non-textual. Consider score is composite result from classification textual document and score from non-textual Like-Retweet analysis. For an unlabeled tweet tw_i, the result of prediction from combination CP is labeled as class $k \in \{positive, negative\}$. For each class k in CP,

we can determine the class positive or negative by find maximum value of CP, the proposed combining formula is defined as follows:

$$CP(tw_i) = \arg\max_{k \in C}(\alpha \times Txt_k(tw_i) + \beta \times NonTxt(tw_i)) \quad (15)$$

where:

α : Weight for textual result from Naïve Bayes Txt from class k

β : Weight for non-textual $NonTxt$ features which the value is $1 - \alpha$, with $\alpha,\beta \in [0,1]$

The consequences of such condition if α increase, β will decrease (and vice versa).

IV. Results and Discussion

In this section, we evaluate the performance of textual feature Naïve Bayes classification that is combined with non-textual feature like (LK) and retweet (RT). To prove the accuracy and the performance of the combination of textual and non-textual, we evaluate several variables such as the best value of Fisher Score (FS) for each non-textual feature like and retweet, optimal combination of α and β. The outcome of this proposed combining method have 4 probabilities for predicting a tweet is being positive or negative sentiment such as: a tweet is classified as positive sentiment when it truly is a positive sentiment (true positive, TP); a tweet is classified as positive sentiment when it actually has negative sentiment (false positive, FP); it can be classified as negative sentiment when it actually has positive sentiment (false negative, FN); or it can be classified as negative sentiment when it actually has negative sentiment (true negative, TN). Based on the 4 probabilities outcomes, we can evaluate the prediction system performance using precision, recall and F1-measure. From Table 1, first we decide to arrange the distribution of collected dataset positive and negative class for training and testing purpose.

Table 1
Distribution Training and Testing of Collected Dataset Class

Tweet Groups	Training Dataset		Testing Dataset		Total
	Positive	Negative	Positive	Negative	
#1 nominee	1083	1140	791	820	3834
#2 nominee	1026	991	763	745	3525
#3 nominee	973	996	736	748	3453
#4 nominee	1018	1019	759	759	3555

For non-textual features (Like and Retweet), we try to compare each feature weight from Fisher Score by averaging 10 times randomly generated from training data. Table 2 presents the detail of each number of weight like feature W_{LK} and weight retweet feature W_{RT}.

From Table 2 we can see that the average of W_{LK} and W_{RT} are 0.656 and 0.344 respectively. These values are used for estimating non-textual features score in Equation (14) which represent the non-textual features score individually.

As mentioned before, performance evaluation of this proposed research is affected by values of α (weight of textual) and β (weight of non-textual). Table 3 presents the detail result precision, recall and F1-measure from combination value of α and β. **Precision** is the proportion of tweets that are correctly labeled as positive (negative) among those labeled as same with prediction, this can be calculated as follows:

Table 2
Weight of Non-Textual Features (Like and Retweet) using Fisher Score (FS)

#Random	W_{LK}	W_{RT}
1	0.565	0.435
2	0.471	0.529
3	0.678	0.322
4	0.978	0.022
5	0.541	0.459
6	0.636	0.364
7	0.748	0.252
8	0.626	0.374
9	0.765	0.235
10	0.557	0.443
Average	0.656	0.344

Table 3
Estimation Value α and β for Optimizing Combination Performance

#Test	α	β(1-α)	Precision	Recall	F1-measure
1	0.0	1.0	0.364	0.544	0.436
2	0.1	0.9	0.466	0.846	0.601
3	0.2	0.8	0.614	0.641	0.627
4	0.3	0.7	0.633	0.669	0.651
5	0.4	0.6	0.563	0.686	0.618
6	0.5	0.5	0.545	0.52	0.532
7	0.6	0.4	0.853	0.824	0.838
8	0.7	0.3	0.68	0.883	0.768
9	0.8	0.2	0.904	0.656	0.760
10	0.9	0.1	0.567	0.678	0.618
11	1.0	0.0	0.686	0.644	0.664

$$P = \frac{TP}{TP + FP} \quad (16)$$

Recall is the proportion of a tweet is correctly labeled sentiment, this can be calculated as follows:

$$R = \frac{TP}{TP + FN} \quad (17)$$

F1-score is the performance evaluation based on both value of precision and recall, this can be calculated as follows:

$$F = \frac{2 \times P \times R}{P + R} \quad (18)$$

The results from Table 3 show that the best combination of two weight between textual and non-textual appear in the value of α is 0.6 and β is 0.4 (7^{th} combination). The 1^{st} combination represents the condition if we only consider non-textual features in predicting sentiment, otherwise if we only consider the textual features, this is represented by 11^{th} combination.

This research result show that the potential of combining textual and non-textual features of Twitter is very possible due to the specific characteristic of like and retweet in term of sentiment analysis. Other features are also potential to support or combine with the textual features of Twitter for special cases. In the experiment of estimating value of weight in textual and non-textual features finding the best proportion of α (textual features) and β (non-textual features). We also depict the degree of importance between textual and non-textual in Figure 5.

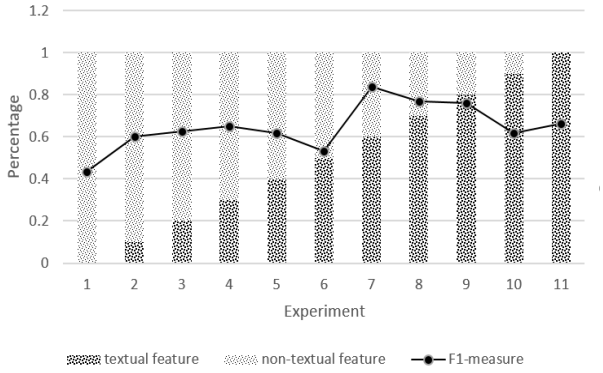

Figure 5: F1-measure based on the percentage of textual and non-textual features

Figure 5 show that textual features more affecting rather than non-textual, but if we only consider to textual features, the F1-measure can only reach 0.664. So, the combining of such two features (textual and non-textual) can improve the classification of sentiment.

V. CONCLUSION

In this research, we propose a hybrid approach which combines textual and non-textual features in predicting Twitter sentiment analysis. There are 3 main stages: textual features mining, non-textual features analysis, and combining stage. In the textual mining stage, we apply text mining processes and Naïve Bayes classification mining techniques to extract each score of a tweet are belonging to positive and negative sentiment class. In the non-textual features (like and retweet), we apply feature weight scoring algorithm Fisher Score to find the weight of each numerical features. Main task of combining stage is finding the best weight of both textual and non-textual for best sentiment analysis prediction performance respectively. We perform experiments on 14367 Twitter datasets which potentially contain user sentiment. The experiment evaluation result shows that the proposed combining method can achieve F1-measure 0.838, which overcome the textual mining value of 0.436. The experiment result shows the best combination of textual and non-textual for predicting Twitter sentiment are 0.6 and 0.4 respectively, which indicate textual feature is more important than non-textual. In the future research, we can consider popularity aspect of a tweet such as number of reply, interaction, and its impression value.

ACKNOWLEDGMENT

This research was partially supported by Intelligence Computation, a research group in Faculty of Computer Science Brawijaya University (Filkom UB). We also thankful to all support at Information Technology and Communication Division (Unit TIK UB) who provide their facilities and expertise that assist this research.

REFERENCES

[1] Yang, L. C., Selvaretnam, B., Hoong, P. K., Tan, I. K. T., Howg, E. K., & Kar, L. H. (2016). Exploration of road traffic tweets for congestion monitoring. Journal of Telecommunication, Electronic and Computer Engineering, 8(2), 141–145. Retrieved from https://www.scopus.com/inward/record.uri?eid=2-s2.0-84984848398 partnerID=40&md5=aa3409237b2ff2788facabd0f6edd723

[2] Chierichetti, F., Kleinberg, J., & Kumar, R. (2014). Event Detection via Communication Pattern Analysis, 51–60.

[3] Mahadzir, N. H., Omar, M. F., & Nawi, M. N. M. (2016). Towards sentiment analysis application in housing projects. Journal of Telecommunication, Electronic and Computer Engineering, 8(8), 20060. https://doi.org/10.1063/1.4960900

[4] Medhat, W., Hassan, A., & Korashy, H. (2014). Sentiment analysis algorithms and applications: A survey. Ain Shams Engineering Journal, 5(4), 1093–1113. http://doi.org/10.1016/j.asej.2014.04.011

[5] Ravi, K., & Ravi, V. (2015). A survey on opinion mining and sentiment analysis: Tasks, approaches and applications. Knowledge-Based Systems, 89, 14–46. https://doi.org/10.1016/j.knosys.2015.06.015

[6] FAQs about Retweets. 2016. Help Center Twitter. Retrieved Feb 1, 2017 from https://support.twitter.com/articles/77606

[7] Wu, B., & Shen, H. (2015). Analyzing and predicting news popularity on Twitter. International Journal of Information Management, 35(6), 702–711. http://doi.org/10.1016/j.ijinfomgt.2015.07.003

[8] Liking A Tweet or Moment. 2016. Help Center Twitter. Retrieved Feb 1, 2017 from https://support.twitter.com/articles/20169874

[9] Mertiya, M., & Singh, A. (2016). Combining Naive Bayes and Adjective Analysis for Sentiment Detection on Twitter. Inventive Computation Technologies (ICICT), International Conference on, 3. Retrieved from 10.1109/INVENTIVE.2016.7824847

[10] Bing Liu, N. Indurkhya and F. J. Damerau, Handbook of Natural Language Processing, Second Edition, 2010, pp. 1-3860-68.

[11] S. Vijay Gaikwad, P. D. Y Patil, and P. Patil, "Text Mining Methods and Techniques," Int. J. Comput. Appl., vol. 85, no. 17, pp. 975–8887, 2014.

[12] Tang, B., Kay, S., & He, H. (2016). Toward Optimal Feature Selection in Naive Bayes for Text Categorization. IEEE Transactions on Knowledge and Data Engineering, 28(9), 2508–2521. https://doi.org/10.1109/TKDE.2016.2563436

[13] Z. Xu and Q. Yang, "Analyzing user retweet behavior on twitter," Proc. 2012 IEEE/ACM Int. Conf. Adv. Soc. Networks Anal. Mining, ASONAM 2012, pp. 46–50, 2012.

[14] I. Guyon and A. Elisseeff, "An Introduction to Variable and Feature Selection," J. Mach. Learn. Res., vol. 3, no. 3, pp. 1157–1182, 2003.

[15] Tala, F. Z. (2003). A Study of Stemming Effects on Information Retrieval in Bahasa Indonesia. M.S. thesis. M.Sc. Thesis. Master of Logic Project. Institute for Logic, Language and Computation. Universiteti van Amsterdam the Netherlands.

[16] Zainal, A., & Novan, A. (2002). "Klasifikasi Dokumen Berita Kejadian Berbahasa Indonesia dengan Algoritma Single Pass Clustering", Prosiding Seminar on Intelligent Technology and its Applications (SITIA), Teknik Elektro, Institut Teknologi Sepuluh Nopember Surabaya

[17] Kibriya A.M., Frank E., Pfahringer B., Holmes G. (2004) Multinomial Naive Bayes for Text Categorization Revisited. In: Webb G.I., Yu X. (eds) AI 2004: Advances in Artificial Intelligence. AI 2004. Lecture Notes in Computer Science, vol 3339. Springer, Berlin, Heidelberg

[18] Q. Gu, Z. Li, and J. Han, "Generalized Fisher Score for Feature Selection," CoRR, vol. abs/1202.3, no. August, pp. 327–330, 2012.

[19] B. Singh, "Optimization of Feature Selection Method for High Dimensional Data Using Fisher Score and Minimum Spanning Tree," IEEE India Conf. Optim., 2014.