Personality Classification based on Facebook status text using Multinomial Naïve Bayes method

Y B N D Artissa¹, I Asror² and S A Faraby³
School of Computing, Telkom University, Bandung

E-mail: ¹ ybnunungda@gmail.com, ² iasror@telkomuniversity.ac.id, ³ said.al.faraby@gmail.com

Abstract. Psychological research shows that the way people write is influenced by their personalities. Words that are often used can describe someone's personality. Facebook is one of the social networking sites for users to express themselves. User status posts can be used to identify the user's personality. To find out the personality of a person based on the statuses they write on Facebook use text classification techniques using Multinomial Naïve Bayes method. The language is English. The test result shows that accuracy increases after reducing the number of word variations and variations of prior probability values. The accuracy obtained by using stemming in the preprocessing process is 59.9% and using uniform prior increases 0.3%.

1. Introduction
Personality is about the personal self of an individual as a whole. Psychological research shows that the way a person writes or speaks is influenced by their personality. People with the same personality traits tend to use similar words to express themselves [1]. The choice of words that often used can describe individual personalities [2]. Facebook has become one of the most widely used social networking sites for users to express themselves [3]. The status posted by Facebook users can show their personal information. The writings of Facebook user status can use to find out their personalities. One of the benefits of knowing someone's personality based on Facebook status is for employee selection process in a company. Because Facebook status is public, the company can check the history of an employee's Facebook status without having to ask directly or without conducting a personality test to find out the personality of the employee.

To find out someone's personality based on the statuses they wrote on Facebook, text classification techniques were used using the Multinomial Naïve Bayes (MNB) method. MNB is a model used by calculating the frequency of each occurrence of a word in a document with its probability [4]. With MNB, the words on the status written by Facebook users can be calculated according to the number of occurrences. MNB also pays attention to the probability of a document that occurs in a particular class or commonly called a prior probability. Previous research shows that variations in prior probability values can affect classification performance [5]. However, the research was conducted on the translation data of Al-Qur'an verses, so the authors wanted to do the same research using Facebook status data. Before entering the classification phase using the MNB method, a preprocessing process is carried out. In the same study, the process of reducing the number of word variations was carried out using the stemming process, but the process produced poor classifier performance compared to the classification process without using the stemming process. So that in this personality classification process will be done to reduce the number of word variations in the preprocessing process.

2. Related work
2.1. Text mining
Text Mining is a field that extracts meaningful information from natural language texts. This can be interpreted as a process of analysing text to extract information that is useful for a particular purpose.
Text in text mining is not structured, ambiguous, and difficult to process such as e-mail, text documents and others [6]. However, text is the most common way to exchange formal information.

2.2. Preprocessing

Preprocessing is the process of changing an unstructured word form into a structured word form to facilitate the classification process [7].

- Lower case is change all letters to lowercase.
- Remove symbol is remove all characters other than letters such as numbers and punctuation marks.
- Remove link is remove the links in the document.
- Tokenization is a word separation phase based on each word that composes it, for example like I eat ice cream to be [i, eat, ice, cream].
- Stopword removal is the stage of removing stopword words like the, in, this, you, etc.
- Stemming is the process of cutting the additions or returns of the inflected word into the word stem. The stemming algorithm used in this study is the Porter Stemmer Algorithm in which the basic word formation process by eliminating the suffixes for English words. Examples of words such as helping become help, happy become happi.
- Lemmatization is a process to find the basic form of each word according to the existing dictionary and analyze morphology. Lemmatization changes the basic form of the word while preserving the meaning of the word. Examples such as worker doesn’t change to work because the worker is a subject work is a predicate.

2.3. Multinomial Naïve Bayes (MNB)

Some text. MNB is a variation of Naive Bayes designed to complete the classification of text documents. MNB uses a multinomial distribution with the number of words appearing or the word weight as a classification feature. MNB calculates the frequency of each word that appears on the document. For example, there are documents d and class c. To calculate the class of document d, it can be calculated by the formula [4]:

$$ P(t|d) = P(C) \times P(t_1|c) \times P(t_2|c) \times P(t_3|c) \times ... \times P(t_n|c) $$  \hspace{1cm} (1)

- $P(t|d)$ = Probability of a document that occurs in a particular class
- $P(C)$ = Prior probability of class c
- $t_n$ = Document word n
- $P(t_n|c)$ = Probability of the nth word with known class c

Prior class C probability is determined by the formula:

$$ P(C) = \frac{N_c}{N} $$  \hspace{1cm} (2)

- $N_c$ = Number of classes c in all documents
- $N$ = Number of all documents

The probability of the nth word is determined using Laplace smoothing:

$$ P(t_n|c) = \frac{\text{count}(t_n, c) + 1}{\text{count}(c) + |V|} $$  \hspace{1cm} (3)

- $\text{count}(t_n, c)$ = The number of terms found in all training data with class c
- $\text{count}(c)$ = Number of terms in all training data with class c
- $|V|$ = Number of vocab in the training data
2.4. Performance evaluation

Precision and recall is a technique that can be used to calculate the performance value of a text processing system. Precision and recall calculations require four components, namely TP (True Positive), TN (True Negative), FP (False Positive) and FN (False Negative) [8]. Figure 1 is a confusion matrix table which is a table that states the number of test data that is correctly classified and the number of test data that is incorrectly classified.

| Prediction Class |   |
|------------------|--|
| Actual Class     | TP| FN|
| Class            | + | FP| TN|

**Figure 1.** Confusion matrix table.

The following performances will be tested include:

- Precision can be interpreted as the ratio of the number of precision predictions of a class to the total number of predictions classified into that class.
- Recall can be interpreted as the ratio of the number of precision predictions of a class to the total number of facts classified into that class.
- F1-measure is the average result between precision and recall. This F1-measure appears to equalize precision and recall values that are often far away. F1-measure is also interpreted as equalizing precision and recall values.
- Accuracy is a calculation of measuring the correct prediction ratio regarding the total number of cases evaluated.

Evaluation for multi-label classification uses micro-macro average. Micro average can calculate the average value by giving the same weight to each class. For Macro average calculations are calculated based on the number of TP, TN, FP, FN. Micro average can calculate the average value by giving weight to each document classification decision. Because the size of F1 ignores TN and is largely determined by the number of TP values, it is more favorable to the dominant class.

2.5. Personality, Facebook and its association

Psychological research shows that the way people write or speak shows a person's personality traits, and many associations have been identified between personality, emotional manifestations and the use of linguistic words. Therefore, people with the same personality traits tend to use similar words to express their sentiments. The words a person uses in everyday life can reveal important aspects of their social and psychological life. Among all personality models, the Big Five Personality model is widely adopted by the study of computational personality [9]. This model comes from the analysis of natural languages that people use to describe themselves and others. Here are the five dimensions of personality [10]:

- O (Openness): Artistic, curious, imaginative, etc.
- C (Conscientiousness): Efficient, organized, etc.
- E (Extraversion): Energetic, active, assertive, etc.
- A (Agreeableness): Merciful, cooperative etc.
- N (Neuroticism): Anxious, tense, self-pitying, etc.

Facebook has become one of the most widely used social networking sites for someone to show themselves, reveal information about their lives, exchange ideas and express emotions. Facebook accounts are personal, so they can reflect their personal lives. Activities on Facebook such as posting status can reveal personal information.

3. Development

3.1. Dataset

The dataset used is MyPersonality dataset. MyPersonality Project is a Facebook application that is used to predict personality based on an online questionnaire. The dataset contains Facebook status in raw text, personality labels including values and classes, status dates, and some social network
measurements such as network size, betweenness centrality, density, brokerage and transitivity. But the dataset used for this research is the status of Facebook users and their personality class. The dataset consists of 10000 status updates from 250 users who have been labelled into the Big Five Personality dimension. The number of statuses per user ranges from 1 - 223 status. The original myPersonality dataset is slightly modified. All statuses from one user are put together into one long string document, so the final dataset consists of 250 documents from 250 users.

3.2. System design
The system that will be built in this research is a system that can classify personality on Facebook status with the method used is Multinomial Naive Bayes. The programming language used is Java. The data consists of five classes, each class has a yes and no sub-class, so the resulting model has five models.

![System design diagram](image)

**Figure 2.** System design diagram.

The process of determining the class on Facebook status can be done in various stages as shown in figure 2.

- **Split data**
  The dataset MyPersonality is broken down into training data and testing data.

- **Preprocessing**
  The stages of the preprocessing process are lower case, remove symbol, remove links, tokenization, stopword removal and stemming or lemmatization. In the training data preprocessing process, data is preprocessed based on the yes and no labels. Data with the label 'y' is put together, as well as data labeled 'n'. So there are 2 documents, namely 'y' and 'n' documents. This process applies to all classes.

- **TF-IDF**
  In the TF-IDF method, the calculation is done by weighting the term t in a document. The equation used for the calculation of TF-IDF is equation 4:

  \[ tfidf_t = f_{t,d} \times \log \frac{N}{df_t} \tag{4} \]

  - \( tfidf_t \) = term weight t
  - \( f_{t,d} \) = term t in document d
  - \( N \) = total document
  - \( df_t \) = the number of documents containing the term t
In the preprocessing process, the data with the label 'y' is put together, the data labeled 'n' is put together. So there are 2 documents, 'y' and 'n'. So, in this process it produces TF-IDF weight on the label 'y' and 'n' for each word.

- Calculate conditional probability
  The process of making a classifier model is a conditional probability calculation. Naïve Bayes Multinomial Formula with the TF-IDF weighting is as follows (5):
  \[
  (t_n|c) = \frac{W_{ct} + 1}{\sum_{W'\in V} W'_{ct} + B'}
  \]

  \(W_{ct}\) = The weighting value of tfidf or W from term t in category c
  \(\sum_{W'\in V} W'_{ct}\) = The total amount of W from the entire term in category c
  \(B'\) = Number of unique W words (idf value not multiplied by tf) in all documents.

  After the process of making a one-class model is complete, then it is repeated for making other class models.

- Calculation of prior probability
  The equation used is equation 3. Examples of prior probability calculation processes such as openness personality class, this calculation process produces a number of prior probability values for the yes openness class and the prior probability value for the no openness class.

- Choosing class
  In this process what is done is calculating the probability of personality of a status in a class, for example whether a status is classified as openness yes or openness no. This classification process is multiplying the value of each word contained in that status, the value obtained from the results of the conditional probability calculation (equation 5) multiplied by the prior probability value (equation 3). If the probability value 'y' is greater than 'n' then the status includes the category 'y', and vice versa. After the selection process for one class is complete, proceed with another class.

- Evaluation
  After the classification results are obtained, the next step is evaluation. Evaluation is a process to measure how accurately a method is implemented. The following are the formulas for the evaluation process:

  \[
  \text{Precision} = \frac{TP}{TP + FP} \quad (6)
  \]
  \[
  \text{Recall} = \frac{TP}{TP + FN} \quad (7)
  \]
  \[
  F1 = \frac{2(\text{precision} \times \text{recall})}{\text{precision} + \text{recall}} \quad (8)
  \]
  \[
  \text{Accuracy} = \frac{TP + TN}{TP + FP + TN + FN} \quad (9)
  \]
  \[
  \text{Precision Micro} = \frac{TP + FP1 + TP2 + FP2}{TP1 + TP2} \quad (10)
  \]
  \[
  \text{Recall Micro} = \frac{TP1 + FN1 + TP2 + FN2}{\text{Precision1} + \text{Precision2}} \quad (11)
  \]
  \[
  \text{Precision Macro} = \frac{\text{Precision1} + \text{Precision2}}{2} \quad (12)
  \]
  \[
  \text{Recall Macro} = \frac{\text{Recall1} + \text{Recall2}}{2} \quad (13)
  \]

  The first step is calculate the precision (equation 6), recall (equation 7), f1 score (equation 8) and accuracy (equation 9) on the results of classifications per class. Because the personality
classification process is a multi-label classification, then the micro-macro average calculation includes micro precision (equation 10), micro recall (equation 11), micro f1 score, macro precision (equation 12), recall macro (equation 13), F1 score macro in the whole class. For the F1 score on micro and macro average it can be calculated like equation 8. After that, macro accuracy is calculated because the testing process uses the 10 Fold Cross-Validation method.

4. Evaluation
4.1. Testing 1
The method used in the testing process is 10 Fold Cross-Validation with 25 documents as test data and 225 as training data per fold. This testing scenario aims to find out whether there is an influence on the results of accuracy if the form of a word is changed. In this test 1, three testing scenarios were carried out. The first test scenario is to do the classification without changing the basic word form. The second test scenario is to do classification using the stemming process in the preprocessing process. The third test scenario is to do classification using lemmatization in the preprocessing process.

| Scenario | Without changing into the basic word | Stemming | Lemmatization |
|----------|------------------------------------|----------|--------------|
| Average AGR accuracy | 56 ± 0.132 | 56 ± 0.161 | 56 ± 0.158 |
| Average CON accuracy | 58.8 ± 0.12 | 58.4 ± 0.087 | 58.8 ± 0.076 |
| Average EXT accuracy | 52.8 ± 0.109 | 54.8 ± 0.108 | 53.6 ± 0.091 |
| Average NEU accuracy | 59.2 ± 0.101 | 60.8 ± 0.101 | 60.4 ± 0.085 |
| Average OPN accuracy | 69.2 ± 0.067 | 69.6 ± 0.109 | 0.7 ± 0.104 |
| Average accuracy | 59.2 ± 0.06 | 59.9 ± 0.055 | 59.8 ± 0.062 |

In table 1 above is the result of the average value of each evaluation performance produced by each test scenario. Based on table 1, it can be seen that the results of the macro accuracy performance value generated by the classification without changing to the basic word form is 59.2%. The accuracy value will be used as a baseline in analysing the influence of each other scenario. Accuracy results show that the classification without changing to the basic form produces the lowest results compared to the classification using stemming or lemmatization although the results are not significant. The following is the results and analysis of each scenario that is carried out as follows:

- Analysis of the classification process without changing the basic word form
  Preprocessing stages in this process are lower case, remove symbol, remove links, tokenization, stopword removal. At the time of the choosing class process, words are found a little. Examples of words such as wishing in the training data and wishes in testing data. The two words are the same, but they will not be suitable because there is no change in the form of the basic word wish. Because there is no process that forms the basic word, although it multiplies the word list for the classifier model, the word found in the choosing class process is small, resulting in a smaller accuracy compared to the process that uses stemming or lemmatization.

- Analysis of the classification process using stemming
  Preprocessing stages in this process are lower case, remove symbol, remove links, tokenization, stopword removal then stemming. The resulting accuracy is 59.9% greater than 0.7% of the test scenario without changing to the basic word form. The stemming algorithm used is the porter stemmer. In the English-language porter stemmer algorithm, it will change the word into its basic word form by eliminating the suffix because usually English does not recognize the prefix. As the previous example is word wishes, wishing and wished. In the porter stemmer algorithm, there are rules to eliminate the suffix -es, -ing and -ed, so that it forms the base word wish. Other examples like helping to be help and goodness become good. So the word found in the choosing class process is more so that it produces greater accuracy than the process without changing to
the basic word form. This shows that reducing the number of word variations can improve accuracy.

- Analysis of the classification process using lemmatization

Preprocessing stages in this process are lower case, remove symbol, remove links, tokenization, stopword removal and then lemmatization. The resulting accuracy is 59.8% greater than 0.6% of the test scenario without changing the basic word form. Like stemming, lemmatization also forms into basic word forms such as wishes, wishing and wished so that the words found in the choosing class process are many. This is why accuracy in lemmatization is greater than the classification process without changing to the basic word form because it reduces the number of word variations. However, according to their understanding, lemmatization is changing words into basic words according to existing dictionaries. An example is the word effect and effective. The effective word for lemmatization does not turn out to be an effect because it is effective in having its own meaning. So the list of words for the formation of the classifier model increases but the words found in the choosing class process are little compared to stemming. Therefore the classification process using lemmatization results in 0.1% smaller accuracy compared to the process using stemming.

4.2. Testing 2

In this test 2, a test scenario is performed on the prior probability value where the prior probability value is 0.5 or commonly called the uniform prior. This test aims to determine whether there is a change in accuracy if variations of prior probability values are made for the labels "y" and "n". This test is done because in the preprocessing process split data is done for yes and no, documents with the label "y" are put together as well as the document labeled "n". So that the total documents used for the TF-IDF calculation process are 2. So that based on the TF-IDF calculation the prior probability value is 0.5 for each "y" and "n".

| Scenario   | Prior probability values from stemming process data training | Prior probability values 0.5 |
|------------|-------------------------------------------------------------|-------------------------------|
| Average AGR accuracy | 56 ± 0.161                                                  | 56.8 ± 0.147                   |
| Average CON accuracy  | 58.4 ± 0.087                                               | 59.2 ± 0.098                   |
| Average EXT accuracy  | 54.8 ± 0.108                                               | 53.6 ± 0.107                   |
| Average NEU accuracy  | 60.8 ± 0.101                                               | 61.6 ± 0.104                   |
| Average OPN accuracy  | 69.6 ± 0.109                                               | 69.6 ± 0.128                   |
| Average accuracy      | 59.9 ± 0.055                                               | 60.2 ± 0.059                   |

The third column of table 3 shows the test results by changing the prior probability values using uniform prior. These results are compared with the results of accuracy in test 1 where the test uses the prior value of training data where the prior probability value can be seen in table 2. The test results show that the prior probability value with uniform prior produces an accuracy of 0.3% higher than using prior values from the calculation of training data. In this modelling that using uniform prior, 3 out of 5 classes increased accuracy, 1 class did not change and 1 class experienced a decrease in accuracy. This shows
that variations in prior probability values have an important role in producing a better classifier and can affect the accuracy of a system.

5. Conclusion
Based on the results and analysis that have been carried out in this research, it can be concluded that average accuracy of the classification process using stemming in the preprocessing process produces the greatest accuracy of 59.9% because the stemming process changes the form to the base based on the stemming algorithm used. This shows that reducing the number of word variations can improve accuracy. And personality classification with variations in prior probability values using uniform prior produces greater accuracy than using prior probability values from training data that is equal to 60.2%. This shows that variations in prior probability values have an important role in producing a better classifier and can affect the accuracy of a system.

References
[1] Lin J and Mao W 2015 Personality based public sentiment classification in microblog IEEE 978 151
[2] Mairesse F, Walker M A, Mehl M R and Moore R K 2007 Using linguistic cues for the automatic recognition of personality in conversation and text J. of Artificial Intelligence Research 30 pp 457-500
[3] Pratama B Y and Riyanarto S 2015 Personality classification based on twitter text using naive bayes, KNN and SVM International Conference on Data and Software Engineering IEEE 978 170
[4] Rahman A, Wiranto and Doewes A 2017 Online news classification using multinomial naive bayes Jurnal Ilmiah Teknologi dan Informasi 6 1
[5] Pane R A, Mubarok M S and Adiwijaya 2018 Klasifikasi multi-label pada topik ayat al-quran terjemahan bahasa inggris menggunakan multinomial naive bayes e-Proceeding of Engineering 5 1
[6] Kumar L and Bathia P K 2013 Text mining: concept, process and applications Journal of Global Research in Computer Science 4 3
[7] Suyanto 2017 Data mining untuk klasifikasi dan klasifikasi data (Bandung: Penerbit Informatika)
[8] Goutte C and Gaussier 2005 A probabilistic interpretation of precision, recall and F-score, with implication for evaluation Proceedings of the European Colloquium Springer pp 345-359
[9] Pennebaker J W, Mehl M R and Niederhofer K G 2013 Psychological aspects of natural language use: our words, our selves Annual Review of Psychology vol 54 pp 547-577
[10] Alam F, Stepanov E A and Riccardi G 2013 Personality Traits Recognition on Social Network – Facebook Association for the Advancement of Artificial Intelligence