Sentiment Analysis of Movie Review using Naïve Bayes Method with Gini Index Feature Selection

Riko Bintang Purnomoputra¹, Adiwijaya², Untari Novia Wisesty³

School of Computing, Telkom University
Jalan Telekomunikasi No.1 Bandung, Indonesia

Rikostars@students.telkomuniversity.ac.id
Adiwijaya@telkomuniversity.ac.id
Untarinw@telkomuniversity.ac.id

Abstract

In movie reviews, there is information that determines whether the movie is good or bad. Sentiment analysis is used to process information to determine the polarity of the sentence. With unstructured reviews and a lot of data attributes so that it requires much time and computational capabilities that become a problem in the classification process. To process a lot of data selection features becomes a solution to reduce dimensions so it accelerate the classification process and reduce the occurrence of misclassification. The first Gini Index Text feature selection used to classify documents and successfully enhanced the classifier performance. Multinomial Naïve Bayes (MNNB) is a popular classifier used for document classification however, will the Gini Index Text feature selection able to improve MNNB classification performance. Therefore in this study the author aims to use the Gini Index Text (GIT) for text feature selection with MNNB classifier to classify movie review into positive and negative classes. The data used is IMDB dataset that contains reviews in English sentences, the data will be divided into two parts, training data is 90% and data testing is 10%. The test results prove that the Gini index as a selection feature can increase accuracy where accuracy without feature selection is 56% and with feature selection of 59.54% with an increase of 3.54%.

Keywords: Sentiment Analysis, Movie Review, Multinomial Naïve Bayes (MNNB), Gini Index Text (GIT)

I. INTRODUCTION

With the current technological advances many sites that inform about movies that are currently or will be aired such as IMDB, Rotten Tomatoes, Metacritic, and so on. To determine whether a movie is good or bad, it is necessary to look at the reviews of viewers of the movie so that the movie can attract attention to watch. Some movie reviewers pour their reviews expressing their opinions based on the persona and there were differences of opinion in the movie review. Some reviews may appear clearly included in positive or negative reviews, but there are still reviews that are not clearly categorized.

One technique for classifying opinions is sentiment analysis or known as opinion mining with this technique can determine whether the review is positive or negative, there are 2 approaches to classify them, the first by machine learning and the second by the lexicon approach. Both approaches classify text into positive and negative classes depending on the polarity of the sentence, the lexicon approach generally engages with dictionaries of opinion words or known as sentiment dictionaries, to define the sentiment...
orientation as positive or negative. While the machine learning approach uses manual data classification from the dataset and trains the classifier of the sample or called training data which will later be tested in data testing [1]. Based on the results of research that has been done [2] shows the results of various methods for sentiment analysis and can be seen that the machine learning approach produces the highest accuracy. But with so many data attributes can be a problem in the classification process that causes slowness of the process and misclassification. In research [9] feature selection can improve the accuracy of MNNB performance however, MNNB is more sensitive to the feature selection method therefore some feature selection may not able to improve the classification performance.

In research [14] the K-Nearest Neighbor (KNN) method and Support Vector Machine (SVM) with enhanced Gini Index feature selection to work better for classifying text known as Gini Index Text (GIT). The results of this study prove the performance of GIT can reduce irrelevant features and still maintain representative features so that it can improve classification performance however, whether GIT can improve performance with other classification techniques such as Multinomial Naïve Bayes (MNNB).

Based on the problem, the research that will be conducted by the author focuses on the Multinomial Naïve Bayes classification technique by using the Gini Index Text feature selection used in classifying movie reviews and conducting tests to determine the performance of the model based on the test scenario. The limitation of the problem with this study is the review data used is sample data derived from polarity data on research-based Kaggle [13] which has been added from the IMDB website totaling 4000 reviews consisting of 2000 positive reviews and 2000 negative reviews that have been labeled. The results of this study are the performance of the MNNB with GIT feature selection by calculating the accuracy of classification.

In Section 2, we discuss some research that related to this study, the Gini-Index Text feature selection, along with calculating the GIT score, dataset and preprocessing that used in this study, the MNNB Classifier Theory and measuring performance used. In Section 3, we present our proposed system. In Section 4, by means of experimental results, we compare and discuss the classification performances based on test scenario. In Section 5, we draw conclusions and contemplate future studies.

II. LITERATURE REVIEW

There are many studies on sentiment analysis on topics such as social media comments, products, politics, and more. There are so many techniques for classifying text, many researchers are trying to do a combination of techniques in order to achieve better performance. As in research [5] compared to the various feature selection for the Naïve Bayes algorithm the goals of the research is to determine the selection features that are accurate and stable using Newsgroups and Reuters-21578 data. In this study pre-processing data are not displayed so that the data processing is not explained. The highest accuracy results obtained with the Chi feature of 81% for Reuters-21578 data and 84% for Newsgroups and for stable results obtained using the Term Frequency (TF) and Domain Frequency (DF) techniques, but are inefficient compared to other techniques. In Research [8] comparing selection features of Gini Index, Weight Formula (NG), and Word Frequency Mutual Information (MIDF) with KNN and Naïve Bayes classification techniques, the purpose of this study is to compare and validate with new weight features based on the Gini Index on the performance of classification techniques. The results of this study, the Gini Index gets a very good performance that is 75% with KNN and 95% with Naïve Bayes. In research [9] comparing the Multinomial Naïve Bayes (MNNB) and SVM classification techniques with a variety of selection features as many as 18 feature selection, for pre-processing data is not explained in detail what the process is done. The results showed the same performance using the Gini Index, Weighted Log-Likelihood Ratio (WLLR) and Cross-Entropy for Text (CET) on MNNB. Based on the results of this study concluded that SVM requires fewer features to achieve maximum results while Naïve Bayes requires more features to achieve optimal results and is more sensitive to the feature selection method.

In [11] researchers compared the Gini Index Text with various feature selection on IMDB dataset data, Spanish dataset, and Portuguese dataset data extracted in advance with n-grams and using MNNB, SVM, and Weighted SVM (WSVM) classification techniques. From the conclusion, it was found that the selected features of the Gini Index, Domain Frequency, CET, and CHI got the best performance, 90% with SVM, and 92% with MNNB. The SVM classification technique is a fast algorithm and can work with high dimensions while MNNB works better on short documents or to classify sentences, WSVM works better than SVM if it does not use the feature selection. In research [14] the K-Nearest Neighbor (KNN) method and Support
Vector Machine (SVM) with Gini Index Text to work better for text classification produces an average performance of 97% in Reuters-21578 data with GIT can eliminate many features irrelevant and redundant subset of features and retain many representative features that will improve overall classification performance.

A. Gini Index Text

The Gini Index is used to separate attributes, this technique is generally used in decision trees and successfully improves the precision of classifications. There have been many studies to improve the Gini Index method, one of which is the Gini Index Text (GIT) which was introduced in [14], which was made to work on documents with very many features. For each feature in the film, the review will be calculated based on GIT A, GIT B, and GIT C in the positive and the negative class. This method can reduce the features of a subset of features while also retaining many feature representative GIT is defined as:

$$\text{GiniTextA}(w, c_i) = P(c_i|w)^2$$

$$\text{GiniTextB}(w, c_i) = \frac{|P(c_i|w)^2|}{\log P(w)}$$

$$\text{GiniTextC}(w, c_i) = \frac{P(c_i|w)^2}{[\log P(w|c_i)^2]}$$

With:

$P(w|c_i)$ is the probability of movie reviews feature $w$ for $c_i$ classes (positive and negative).

$P(c_i|w)$ is the probability for $c_i$ classes (positive and negative) in the appearance of the movie reviews feature $w$.

$P(w)$ is the probability of movie reviews feature $w$ with the total number of words in the movie review document.

Gini Index Text A will produce a high score if the word $w$ only appears in one class, but if $w$ is evenly distributed among classes it has a low score. Gini Index Text B has the $P(w)$ is normalized with logarithms and absolute values to increase its deviation so that it can calculate specific features and general features more efficiently. Gini Index Text C The $P(w)$ is normalized as $P(w|c_i)^2$ with logarithms and absolute values to increase its deviation so that if a feature $w$ is specific and large class, $P(w|c_i)^2$ has very little value and can be excluded from the process comparison with threshold.

B. Multinomial Naïve Bayes

The Naïve Bayes Classifier is a simple probability classifier that applies the Bayes theorem with a high assumption of independence. Bayes' theorem is a theorem used in statistics to calculate the probability of a hypothesis [12]. Generally there are three distribution models Bernoulli, Multinomial, and Poisson. These three models are used as classifiers, namely Bernouli Naïve Bayes, Multinomial Naïve Bayes and Poisson Naïve Bayes and not as an independent document.

Multinomial Naïve Bayes (MNNB) calculates the number of words in a document so that it assumes the independence of the appearance of words in the document. This assumption shows that the likelihood that each word event in a document is free does not take into account word order and word context in documents [12]. MNNB will be used to classify movie review documents into positive and negative classes MNNB defined as:

$$\arg\max P(C_j) \prod_i P(w_i|C_j)$$

$$P(w_i, c_k) = \frac{1 + \text{Count}(w, c_k)}{|V| + \text{Count}(c_k)}$$

With:

$\text{Count}(w, c_k)$ is the sum of movie review feature $w$ that appear in a $c_k$ classes (positive and negative)

$\text{Count}(c_k)$ is the sum of all movie review feature $w$ in the $c_k$ classes (positive and negative)
$|V|$ is all the vocabulary that appears in movie review documents

C. Measuring Performance

Confusion Matrix is a method for measuring performance for classification in machine learning confusion matrix taking into account 4 terms TP (True Positive) is the number of positive reviews classified positive, TN (True Negative) is the number of negative reviews classified as negative, FP (False Positive) is the number of negative reviews classified positive and FN (False Negative) is the number of positive reviews classified as negative.

| TABLE I
CONFUSION MATRIX |
<table>
<thead>
<tr>
<th></th>
</tr>
</thead>
<tbody>
<tr>
<td>Predicted</td>
</tr>
<tr>
<td>Positive</td>
</tr>
<tr>
<td>Positive</td>
</tr>
<tr>
<td>Negative</td>
</tr>
</tbody>
</table>

Using a confusion matrix will be very useful for calculating the performance of classification results such as accuracy of precision, recall, and F1-score. The results of the testing data prediction will be validated by using accuracy testing to measure the classification performance of the model based on the confusion matrix.

Accuracy is a general measurement method that is often used to see the level of success of experiments conducted using accuracy. Accuracy is calculated based on the correctness of the classification results of all documents, the number of true predictions divided by the total number of true and false predictions defined as:

$$\frac{TP + TN}{TP + FP + FN + TN}$$ (6)

III. RESEARCH METHOD

The system design used for this research is the data that has been obtained will be processed to reduce the number of movie reviews features with Pre-processing namely Tokenization, Case Folding, Stopword Removal, and Lemmatization. Data will be divided for training data and testing data, for training data features will be selected based on the Gini Index Text (GIT) score. After features are selected, Multinomial Naïve Bayes (MNNB) will be trained on selected features. Testing data will be classified with MNNB that have been trained and then the accuracy of the classification results will be calculated.

A. Dataset

The beginning of the system design is data retrieval, data used is a dataset used in research [13] with the title Sentiment Polarity Dataset Version 2.0 on Kaggle data has been updated at...
https://www.kaggle.com/nltkdata/movie-review, with total files containing 32967 positive review files and 31752 negative review files. The data are divided into three parts: 2000 files, 3000 files and 4000 files, each section have 50% positive review files and 50% negative review files and is divided into 90% training data and 10% testing data.

B. Preprocessing

At this process the data will be processed and the removal of a word that has no value to reduce the number of features so as to increase the effectivenes and efficiency of the classification process. Following are the stages of pre-processing

- Tokenization
- Data Cleaning
- Case Folding
- Stopwords Removal
- Lematization

C. Calculates the Gini Index Text Score

The features of the training data will be calculated using the Gini Index Text (GIT) formula GIT A, GIT B, and GIT C. Each feature will have a score that will be used to select features with the highest score. Here is the example of 25 positive features in 4000 data:

Fig. 2 50 Best Positive Features in 4000 Data Using GIT A

Fig. 2 shows 25 movie review features with the highest score in the positive class. Based on the calculation of features scores with GIT A get a maximum score of 1 this proves that these features only appear in one class thus achieve the highest score.

Fig. 3 50 Best Positive Features in 4000 Data Using GIT B
Fig. 3 shows 25 movie review features with the highest score in the positive class. Based on the calculation of features scores with GIT B get a maximum score of 0.06 this proves that these features only appear in one class and are divided by \( \log_2 P(w|c) \) so that they can achieve better deviations.

![Fig. 3](image)

**Fig. 3** 25 Best Positive Features in 4000 Data Using GIT B

Fig. 4 shows 25 movie review features with the highest score in the positive class. Based on the calculation of features scores with GIT C get a maximum score of 0.035, this proves that these features only appear in one class and are divided by \( |\log_2 P(w|c)|^2 \) the goal is almost the same as GIT B in order to achieve a better deviation.

Based on the fig. 2, fig. 4, and fig. 4 above points out that GIT A gets a score of 1, GIT B gets a score of 0.06, and GIT C gets a score of 0.035 for each of the 50 best features. After getting a score on each feature, the next step is to select the feature, which is to select the features with a high GIT score.

D. Multinomial Naïve Bayes Classification

In the Multinomial Naïve Bayes classification process the features that have been selected based on the best Gini Index Text score of k features will be recognized by the Multinomial Naïve Bayes for the model training process. After that MNNB who have been trained will predict positive or negative labels from the test data. Testing will be conducted periodically by adding features with the highest score every iteration to be able to see the performance of the selected features. The next process will calculate the accuracy of the classification, prediction which will be explained in the next chapter.

IV. RESULTS AND DISCUSSION

In the testing phase, two testing scenarios are performed, the first is tested using the Multinomial Naïve Bayes method only without using feature selection. The second is testing using the Multinomial Naïve Bayes by selecting the Gini Index Text feature, meaning that this test will reduce features based on the Gini score Index Text.

A. Testing without Selection Features

In this test the data will be classified using the Naïve Bayes Multinomial method to determine the level of accuracy in the prediction classification of movie reviews without using feature selection. A collection of feature generated from the pre-processing process will be directly classified without being reduced. The data will be divided into 3 parts, namely using 2000, 3000, and 4000 data. The results of this test are the accuracy of the classification:

**TABLE II**

**MULTINOMIAL NAÏVE BAYES PERFORMANCE RESULTS**

<table>
<thead>
<tr>
<th>Data</th>
<th>Accuracy</th>
<th>Features</th>
</tr>
</thead>
<tbody>
<tr>
<td>2000</td>
<td>62.5%</td>
<td>6968</td>
</tr>
<tr>
<td>3000</td>
<td>54.84%</td>
<td>8825</td>
</tr>
<tr>
<td>4000</td>
<td>56%</td>
<td>10336</td>
</tr>
</tbody>
</table>
Based on Table II shows the results of performance using 2000, 3000, and 4000 data with an average accuracy of 57.78%. The best results were obtained using 2000 data, with an accuracy of 62.5%.

B. Testing with Selection Features

In this test the data will be classified using the Naïve Bayes Multinomial method to determine the level of accuracy in the prediction classification of film reviews with the best k features. Testing phase will be conducted periodically in the amount of 100 positive features and 100 negative features per iteration to get the best results based on the features selected. The data will be divided into 3 parts using 2000, 3000, and 4000 data. The results of this test are the accuracy and the following F1-scores are the results of the test:

![Figure 5: Accuracy Results Using 2000 Data](image1)

In Figure 5 shows the accuracy and F1-score line graphs using 2000 data on the number of features used with a total of 6968 features. The highest accuracy obtained by GIT of 62.5% this indicated that GIT A, B, and C did not improve accuracy result.

![Figure 6: Accuracy Results Using 3000 Data](image2)

In Figure 6 shows the accuracy and F1-score line graphs using 3000 data on the number of features used with a total of 8825 features. It can be seen that the performance of GIT A tends to be lower than GIT B and C by using 6000 to 8600 features. Feature selection with GIT achieves the highest accuracy on 8795 features.

<table>
<thead>
<tr>
<th>Method</th>
<th>Accuracy</th>
<th>Features</th>
</tr>
</thead>
<tbody>
<tr>
<td>MNNB with GIT A, B, C</td>
<td>55.51%</td>
<td>8795</td>
</tr>
<tr>
<td>MNNB</td>
<td>54.84%</td>
<td>8825</td>
</tr>
</tbody>
</table>

TABLE III

<table>
<thead>
<tr>
<th>HIGHEST ACCURACY RESULTS ON 3000 DATA</th>
</tr>
</thead>
</table>

In Table III shows the accuracy results using 8795 features in 3000 data, the accuracy of GIT A, B, and C are higher than not using feature selection. This shows that the features selected can improve classification performance, the accuracy result are 55.51%.
Figure 7 shows the accuracy line graphs using 4000 data on the number of features used with a total of 10336 features. The classification results show that accuracy with feature selection is higher than not using feature selection. The performance of GIT A achieves higher accuracy compared to GIT B and C, this shows that the features selected in GIT A have more influence on the classification of data.

**TABLE IV**  
**HIGHEST ACCURACY RESULTS ON 4000 DATA**

<table>
<thead>
<tr>
<th>Method</th>
<th>Accuracy</th>
<th>Features</th>
</tr>
</thead>
<tbody>
<tr>
<td>MNNB with GIT A</td>
<td>59.54%</td>
<td>8312</td>
</tr>
<tr>
<td>MNNB with GIT B, C</td>
<td>59.29%</td>
<td>8217</td>
</tr>
<tr>
<td>MNNB</td>
<td>56%</td>
<td>10336</td>
</tr>
</tbody>
</table>

In Table IV shows the accuracy results on 4000 data, the accuracy of GIT A, B, and C are higher than not using feature selection. This shows that the features selected can improve classification performance, the accuracy result are 59.54% with GIT A and 59.29% with GIT B and C.

From the test results it can be seen that there is an increase in accuracy by using the feature selection, as in Table 3 with the results of 54.84% without feature selection and 55.51% with features selected by 8795 features, an increase of 0.67% with GIT A, B and, C. In Table 5 shows accuracy results of 56% without feature selection and 59.54% with GIT A using 8312 features an increase of 3.54% and 59.29% with GIT B and C an increase of 3.29%.

This test also proves different accuracy results using GIT A, GIT B, and GIT C as in Tables 4 show that GIT A gets a higher accuracy compared to GIT B and GIT C at 59.54% using 8312 features while GIT B and GIT C got 59.29% using 8217 features with a difference of 0.25%.

Based on the results Multinomial Naïve Bayes with Gini Index Text can work well, this proves that by using feature selection can make accuracy better like 4000 data (which should have 10336 features), and with only 8795 features it produces better accuracy so that selecting features can reduce misclassification. The bigger the data, the better the features will be compared to the smaller data so the classification performance can look better. To see the results of deeper feature reduction the authors suggest using the threshold in feature selection so that the performance of GIT A, B and C can be seen clearly.
V. Conclusion

Using the Naïve Bayes Multinomial Method with Gini Index Text Feature Selection can improve the accuracy of classification results as in 4000 data with an increase in accuracy of 3.54% with 8312 features. In 3000 data the accuracy increased by 0.67% with 8795 features. Gini Index Text feature selection produces different performance as in GIT A has a better performance compared to GIT B and GIT C as seen in 4000 data accuracy result are 59.54%, while GIT B and GIT C is 59.29% with a difference of 0.25%. GIT B and GIT C produce the same performance in every test this happens due to the features selected are not different. Feature selection can be seen to be more influential on the classification of data using 4000 data compared to 3000 data, and 2000 data so that shows the bigger the data resulting a better features. The result of this study can be used to analyze the effect of Gini Index Text feature selection on Multinomial Naïve Bayes to classify documents. We recommend using different approaches on selecting features such as threshold so feature reduction can be seen more clearly. Thus, it can be concluded that the Gini Index Text Feature Selection can improve the performance of the Multinomial Naïve Bayes and can be an alternative solution to classify movie reviews data quite accurately. In the future works to see a more detailed performance from the Gini Index Text feature selection is by benchmarking with other popular feature selection, such as Chi, Mutual Information and TF-IDF and adding more classes in the data.

REFERENCES

[12] Butar Thio, Faizi Mochammad, Indriati, “Penentuan Rating Review Film Menggunakan Metode Multinomial Naïve Bayes Classifier dengan Feature Selection berbasis Chi-Square dan Galvotti- Sebastiani-Simi Coefficient” Universitas Brawijaya”. 2018