International Research Journal of Advanced Engineering and Science
ISSN (Online): 2455-9024

Sentiment Analysis of Comments on Instagram Gramedia Pustaka Accounts (GPU) Using Lexicon Based and Naive Bayes Classifier Methods

Murniyati[1], Tristyanti Yusnitasari[2*], Dwieky Iskandar Muhammad Santoso[3*]
[1,2,3]Department of Information System, Universitas Gunadarma, Depok, Jawa Barat, Indonesia - 16431
Email address: [1]murnirk77(at)gmail.com, [2]tyusnitasari(at)gmail.com, [3]dwiekyiskandar31(at)gmail.com

Abstract*— Every company needs public feedback on the performance and services provided by the company. However, the assessment carried out manually causes the assessment of public opinion to be complex, and takes a lot of time. One of the companies in Indonesia that runs in the printing and publishing sector is Gramedia Pustaka Utama (GPU). In this study, sentiment analysis was carried out on comments on the Gramedia Pustaka Utama (GPU) Instagram account. Sentiment analysis is carried out by combining data mining and text mining, or a method used to process various opinions given by the public to the performance given by Gramedia Pustaka Utama (GPU) in the comments column of the company's Instagram account. The comments given can be about the assessment of a product, service or an agency which will later be used to understand, extract opinion data, and process textual data automatically to get a sentiment contained in an opinion. For the method used, the researcher uses the Lexicon based method and the Naive Bayes Classifier (NBC).*

Keywords*— Sentiment Analysis, Naive Bayes Classifier, RStudio, Instagram.*

I. Introduction

Instagram is one of the social media that is widely used by companies to provide information to the general public, with various features that make it easier for people to give comments and likes to company posts, and is widely used as a promotional media. One of them is the biggest book publisher in Indonesia, namely Gramedia Pustaka Utama (GPU) is a subsidiary of Kompas Gramedia. As a company in the field of book publishing, positive and negative assessments are needed from public feedback on the performance and services provided by Gramedia Plibrary Utama (GPU). However, the assessment that is still done manually causes the assessment of public opinion to be complex, and takes a lot of time. To get feedback from the public on the performance of Gramedia Pustaka Utama (GPU), the right technique to implement it is to use the Sentiment Analysis technique. Sentiment analysis or opinion mining is the process of understanding, extracting and processing textual data automatically to obtain sentiment information contained in an opinion sentence. Sentiment analysis is carried out to see opinions or opinion tendencies towards a problem or object by someone, whether they tend to have negative or positive views or opinions. One example of the application of sentiment analysis is when a company issues a product and the company provides services to receive opinions from consumers about the product. Sentiment analysis is used to classify positive and negative opinions from consumers who use these products so as to speed up and simplify the company's task to review product deficiencies. There are sentiment analysis methods, for example, classification methods that can be used, namely Machine Learning such as Naive Bayes, Support Vector Machine, Logistic Regression and Lexicon-Based are often used to get the best results. Based on the problem above, the researcher intends to use the Naive Bayes Classifier (NBC) method for the classification method because it only requires a small amount of training data to estimate the parameters (average and variance of the variables) needed for classifying other methods. Naive Bayes Machine learning classifier that has a model in the form of probability or probability. According to previous research, the advantage of the Naive Bayes Classifier is its ability to classify documents with its simplicity and computational speed. In addition, the Naive Bayes Classifier requires only a small amount of training data to estimate the parameters (variants of the class) required for classification.

II. Literature Review

Text mining is an interdisciplinary field that refers to information retrieval, data mining, machine learning, statistics, and computational linguistics. Since most information (common estimates say more than 80%) is currently stored as text, text mining is believed to have high commercial value potential. Text mining is a technique used to deal with classification, clustering, information extraction and information retrieval problems. Text mining (text mining) is mining carried out by computers to get something new, something that was not known before or rediscover implicitly implied information, which comes from information extracted automatically from different text data sources. (Feldman & Sanger, 2007). Text mining refers to the process of extracting high quality information from text. High quality information is usually obtained through forecasting patterns and trends through means such as statistical pattern learning. Typical text mining processes include text categorization, text clustering, concept/entity extraction, granular taxonomy production, sentiment analysis, document inference, and entity relationship modeling (i.e., learning relationships between named entities). The laboratory-intensive manual text mining approach first emerged in the mid-1980s, but advances in technology have allowed the field to expand over the last decade. Text mining typically involves the process of structuring the input text (usually parsing, along with the addition of some derived linguistic features and the removal of some of them, and subsequent insertion into the database), determining patterns in the structured data, and finally evaluating and interpreting the output. High quality in the field of text mining usually refers to some combination of relevance, novelty, and interestingness. The stages of text mining in general are text preprocessing and feature selection

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

Sentiment analysis is a combination of data mining and text mining, or a method used to process various opinions given by consumers or experts through various media, regarding a product, service or an agency. Sentiment analysis is a method used to understand, extract opinion data, and process textual data automatically to get a sentiment contained in an opinion. Sentiment analysis consists of 3 types of opinions, namely positive opinions, negative opinions and neutral opinions, so that with sentiment analysis the company or related agencies can find out the community's response to a service or product, through feedback from the community or experts. Sentiment refers to the focus of a particular topic, statements on a topic may have different meanings with the same statement on different subjects, therefore in some studies, especially in product reviews, work is preceded by determining the elements of a product being discussed before starting the process. Sentiment Analysis. (Afrizal, 2019)

III. Research Methodology

In the research conducted, it begins with the preprocessing stage, namely eliminating noise, uniforming word forms and reducing vocabulary volume. This stage includes Case Folding, Normalization, Filtering, Changing Standard Words, Stopwords and Stemming. The following is the application at the document preprocessing stage in the classification system:

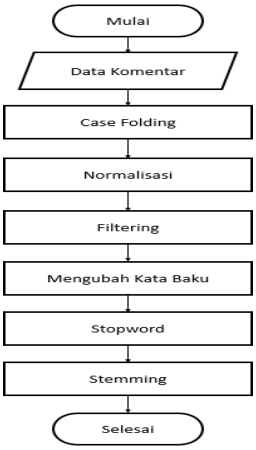


Fig. 1. Preprocessing Step

Case Folding at this stage is done by changing the letters in the review to lowercase. Only letters 'a' to 'z' are accepted. Characters other than letters are omitted and are considered as delimiters. Normalization at the Normalization stage is carried out to clean comment data from unnecessary components such as URLs, hashtags (#), usernames, numbers, and emojis. The normalization stage is carried out because these components have no meaning and have no effect in the sentiment analysis process. The filtering stage is continued to prevent duplicate data or duplication of comment data that has been taken at the data collection stage. Comment data taken sometimes is duplicate data or duplication. Then the filtering stage is carried out to eliminate the duplicate data. Changing standard words, namely changing non-standard words or everyday words (slang) in the comment data into standard words according to the Big Indonesian Dictionary. This is done to make it easier to search for words on the engine. Stopword at this stage, every word in the comments will be checked. If there are conjunctions, prepositions, pronouns or words that are not related in the sentiment analysis, those words will be removed. Stemming is the process of changing words to their basic form by removing affixes to words in the document. The Stemming Algorithm used in this study is the Nazief Adriani Algorithm:

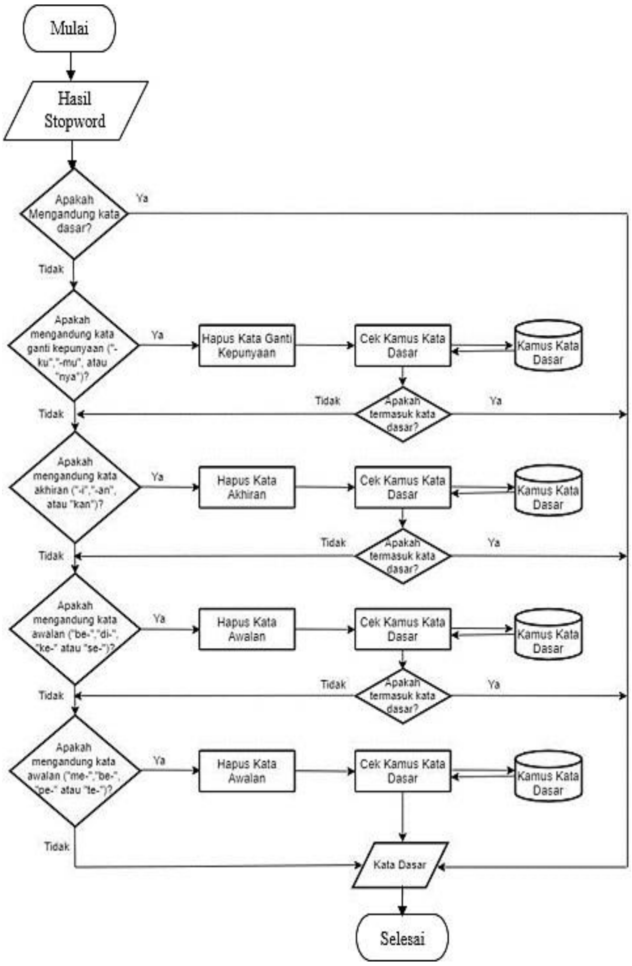

Mulai
Hasil Stopword
Apakah Mengandung kata dasar?
Ya
Tidak
Apakah mengandung kata ganti kepunyaan ("-ku","-mu", atau "nya")?
Ya
Hapus Kata Ganti Kepunyaan
Cek Kamus Kata Dasar
Kamus Kata Dasar
Apakah termasuk kata dasar?
Tidak
Ya
Tidak
Apakah mengandung kata akhiran ("-i","-an", atau "kan")?
Ya
Hapus Kata Akhiran
Cek Kamus Kata Dasar
Kamus Kata Dasar
Apakah termasuk kata dasar?
Tidak
Ya
Tidak
Apakah mengandung kata awalan ("be-","di-", "ke-" atau "se-")?
Ya
Hapus Kata Awalan
Cek Kamus Kata Dasar
Kamus Kata Dasar
Apakah termasuk kata dasar?
Tidak
Ya
Tidak
Apakah mengandung kata awalan ("me-","be-", "pe-" atau "te-")?
Ya
Hapus Kata Awalan
Cek Kamus Kata Dasar
Kamus Kata Dasar
Tidak
Kata Dasar
Selesai

Fig. 2. Stemming Flow

IV. Results and Discussion

A. Lexicon Based Classification

Sentiment classification with Lexicon Based is a classification based on positive and negative words in the comment data that has done the preprocessing stage. This classification has been matched with words found in the Lexicon Indonesian dictionary. If the comment has a positive word, it will be classified as positive sentiment. If the comment has a negative word it will be classified as negative sentiment. The flow of the classification of the Lexicon Based method can be seen in Figure 3.

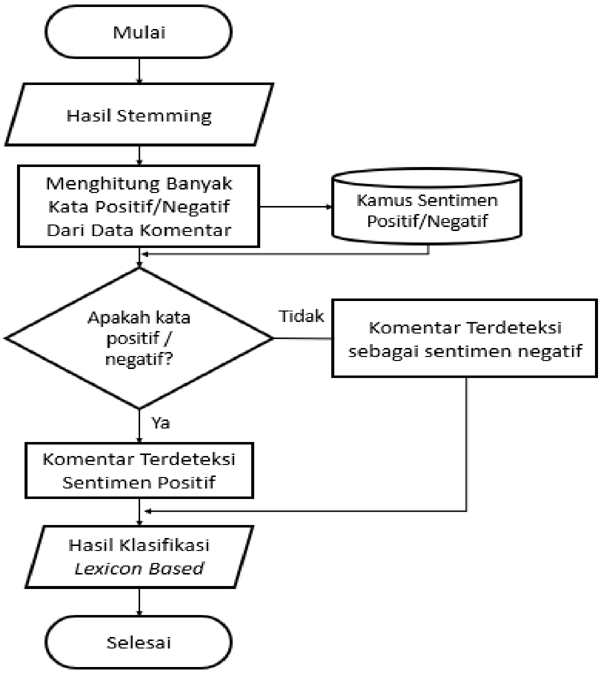

Mulai
Hasil Stemming
Menghitung Banyak Kata Positif/Negatif Dari Data Komentar
Kamus Sentimen Positif/Negatif
Apakah kata positif / negatif?
Tidak
Komentar Terdeteksi sebagai sentimen negatif
Ya
Komentar Terdeteksi Sentimen Positif
Hasil Klasifikasi *Lexicon Based*
Selesai

Fig. 3. Lexicon Based Classification Flow

After going through the text preprocessing stage, the comment data will be read word by word and matched with a dictionary of sentimental words, if a sentence has more positive sentiment words than the number of negative sentiment words, then the sentence is classified as positive sentimental, and vice versa.

B. *Naïve Bayes Clasifier Classification*

Next is data classification using the Naive Bayes Classifier method, namely the stage of classifying whether the data to be tested is included in positive or negative sentiment. At this stage, labeling is carried out first with the Naive Bayes Classifier method. There are two processes in this method, namely the process of training and testing. The following are the stages of the classification process using the Naive Bayes Classifier method in Fig. 4:

C. *Data Labeling*

At the data labeling stage, a labeling process will be carried out to determine the range of data that falls into the category of training data and test data. The proportion of the distribution of training data and test data is 70:30 percent of the total data. The field used is the sentiment field because in this study, the subject of the calculation is positive or negative sentiment.

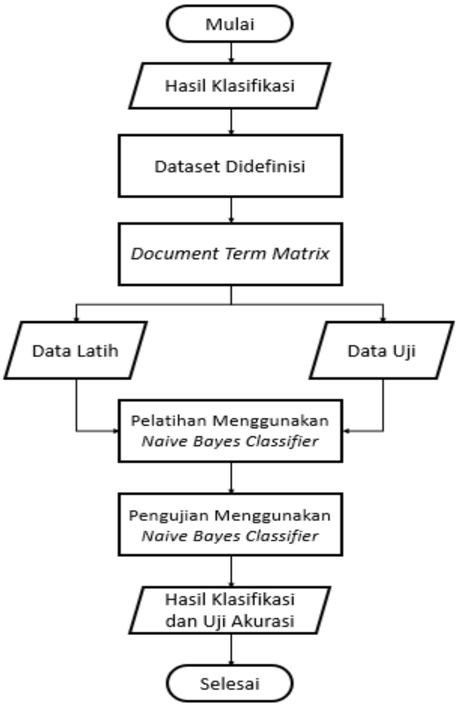

Mulai
Hasil Klasifikasi
Dataset Didefinisi
Document Term Matrix
Data Latih
Data Uji
Pelatihan Menggunakan *Naive Bayes Classifier*
Pengujian Menggunakan *Naive Bayes Classifier*
Hasil Klasifikasi dan Uji Akurasi
Selesai

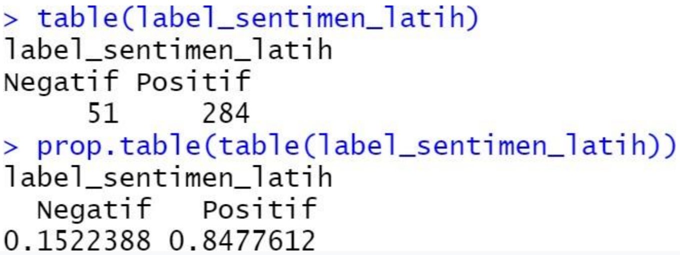
Fig. 4. Classification Using the Naives Bayes Classifier Method

Training Data

In training data labeling, the proportion of training data is 80 percent of the total data, namely from data 1 to data 335. Training data is data used as a learning system. To carry out the labeling process and find out the number of positive and negative sentiments and their probabilities, it can be written with the following command fragment:

```
#data latih sentimen label_sentimen_
tabel_akhir[1:335,]$Sentimen                    <
label_sentimen_latih
table(label_sentimen_latih)
prop.table(table(label_sentimen_lati
```

Label_sentimen_train positive_or_negatif [1:335,] Sentimen is the command used to define the sentiment field for data 1-335 as training data with the function name label_sentimen_train. Below is an output image of the training data labeling and the number of positive or negative sentiments and their probabilities :



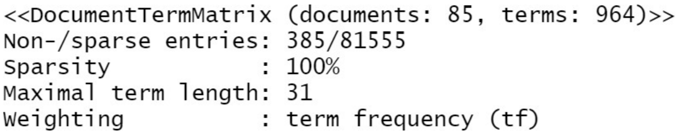
```
> table(label_sentimen_latih)
label_sentimen_latih
Negatif Positif
     51     284
> prop.table(table(label_sentimen_latih))
label_sentimen_latih
  Negatif   Positif
0.1522388 0.8477612
```

Fig. 5. Sentiment Labeling Results and Probability of Training Data

The training data in this study consisted of 335 comment data with the number of positive sentiments as much as 284 and the number of negative sentiments as much as 51. For the positive probability to the number of training data is 0.8477612, while the negative probability to the number of training data is 0.1522388. Next, create a Document Terms Matrix (DTM) for training data labels using the following command

```
dtm_komentar_latih<-
table_final[1:335,  ]
dtm_komentar_latih
```

dtm_komentar_train table_final [1:335,] is a command to create a DTM with training data objects, namely data 1 to 335 which is stored in the dtm_komentar_train variable:



```
<<DocumentTermMatrix (documents: 85, terms: 964)>>
Non-/sparse entries: 385/81555
Sparsity           : 100%
Maximal term length: 31
Weighting          : term frequency (tf)
```

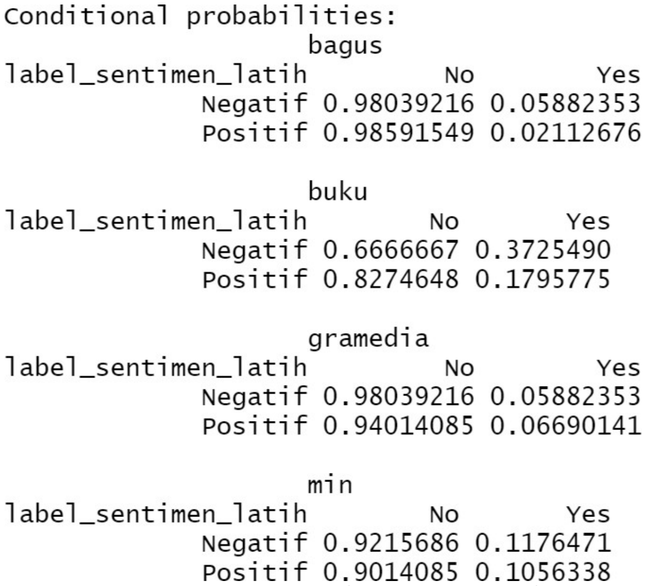
Fig. 6. Document Terms Matrix on data test

Training Data Set

The training stage for training data sets using the Naïve Bayes Classifier is included in the next stage, namely the learning process. The input data used is training data. The learning process of training data sets using the Naïve Bayes Classifier algorithm consists of 2 calculations, namely calculations to find the probability of the occurrence of sentiment labels and calculations to find the probability of occurrence of each term for each classification. For program execution on Rstudio, package "e1071" is needed. Package "e1071" is used for the implementation of the Naïve Bayes Classifier in the R programming language. The following is an excerpt of the command for the learning phase using *Naïve Bayes Classifier*:

```
library(“e1071”)
gpu_classifier<-
  naiveBayes(dtm_komentar_latih,
label_sentimen_latih, laplace=1)
system.time(gpu_classifier
naiveBayes(dtm_komentar_latih,
label_sentimen_latih, laplace=1))
gpu_classifier class(gpu_classifier)
  laplace=1))gpu_classifier
class(gpu_classifier)
```

The library function (e1071) for package e1071, which is a package that contains functions to perform machine learning work including Naïve Bayes classification, gpu_classifier naiveBayes(dtm_komentar_train, label_sentimen_train, laplace=1) is the main function of learning using the Naïve Bayes Classifier method with create a gpu_classifier function, then call back the dtm_komentar_train and label_sentimen_train functions that have been defined previously and use laplace = 1. System.time is a function to see the speed of processing time carried out by the system. The class is to see what methods are used. The results of the learning process and the probability table for the first term sample in the training data set are shown in Fig. 7



```
Conditional probabilities:
                     bagus
label_sentimen_latih         No        Yes
             Negatif 0.98039216 0.05882353
             Positif 0.98591549 0.02112676

                     buku
label_sentimen_latih        No       Yes
             Negatif 0.6666667 0.3725490
             Positif 0.8274648 0.1795775

                     gramedia
label_sentimen_latih         No        Yes
             Negatif 0.98039216 0.05882353
             Positif 0.94014085 0.06690141

                     min
label_sentimen_latih        No       Yes
             Negatif 0.9215686 0.1176471
             Positif 0.9014085 0.1056338
```

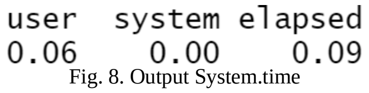
Fig. 7. Learning Outcomes of Naïve Bayes Classifier

In the table of possible samples, it can be seen that there are "no" and "yes" columns. The purpose of the table is in the "yes" column, the term "good" has the possibility of appearing in the "positive" classification of 0.02118676% and the probability of not appearing in the "negative" classification is 0.05882353%. 4.4.4 Testing with Naïve Bayes Classifier

At this testing stage, the core process of this research will be carried out, namely the process of testing test data based on the results of the training process using the Naïve Bayes Classifier method. The following is an excerpt of the command for the testing phase using the Naïve Bayes Classifier:

```
gpu_uji_pred <- predict(gpu_classifier,
newdata =dtm_komentar_uji)
system.time(gpu_uji_pred
predict(gpu_classifier, dtm_komentar_uji))
gpu_uji_pred table(gpu_uji_pred)
gpu_uji_predict <- predict(gpu_classifier,
newdata = dtm_komentar_uji)
gpu_uji_predict
system.time(pred <- predict(gpu_classifier,
newdata = dtm_komentar_uji))
table(gpu_uji_pred,
label_sentimen_uji)
prop.table(table(gpu_uji_pred,
label_sentimen_uji))
```

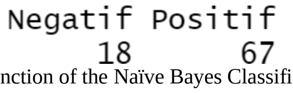
In the above syntax, the function of gpu_test_pred predict (gpu_classifier, newdata = dtm_komentar_uji) is the main function for the test process using the Naïve Bayes Classifier. The gpu_test_pred function is a function to perform classification by calling back the gpu_classifier function and the processed data is test data, not training data. System.time is a function to see the speed of processing time carried out by the system. The function of the table is to display the number of positive and negative sentiments on the test data. Then prop.table(table(gpu_uji_pred, label_sentimen_uji)) is the syntax to display the probability of the number of positive or negative sentiments by the number of test data. The following is the output of the system.time function which can be seen in Fig.8 :



```
user  system elapsed
0.06    0.00    0.09
```

Fig. 8. Output System.time

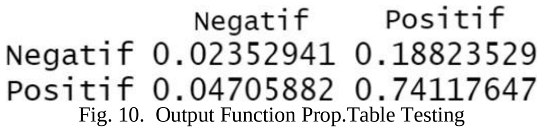
The following is the output of the table function to see the number of sentiments in the positive and negative categories which can be seen in Fig. 9.



```
Negatif Positif
     18      67
```

Fig. 9. Output Function of the Naïve Bayes Classifier Testing Table

The following is the output of the prop.table function which can be seen in Fig. 10 :



```
           Negatif    Positif
Negatif 0.02352941 0.18823529
Positif 0.04705882 0.74117647
```

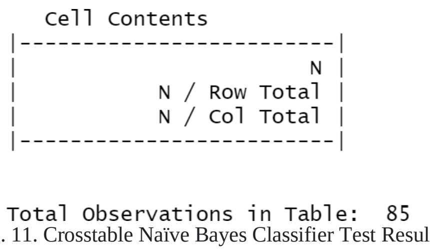
Fig. 10. Output Function Prop.Table Testing

System Accuracy

The next stage will be a process to display the accuracy of the system. The packages used are “gmodels” to create a probability crosstable and “caret” to use the Confusion Matrix function. The following syntax is used:

```
library(“gmodels”)
CrossTable(gpu_uji_pred,
      label_sentimen_uji, prop.chisq =
      F, prop.t= F,
      dnn=c(“Actual”,”Predicted”))
table(“Sebenarnya” = label_sentimen_uji,
“Prediksi”= gpu_uji_pred)
```

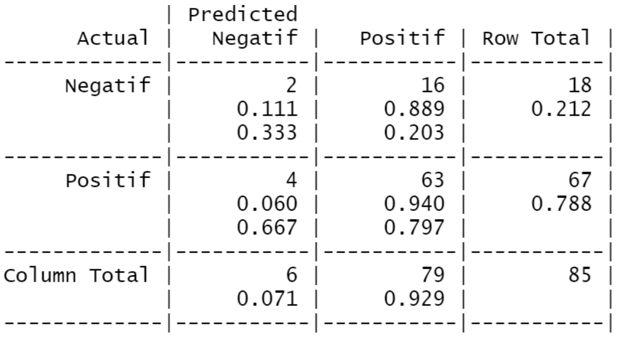
The gmodels function is a package for creating matching models or commonly called model fittings. CrossTable serves to create a table and calculate its probabilities vertically and horizontally. The table is for displaying the number of positive and negative sentiments. The following is the output of the crosstable function can be seen in Fig. 11:



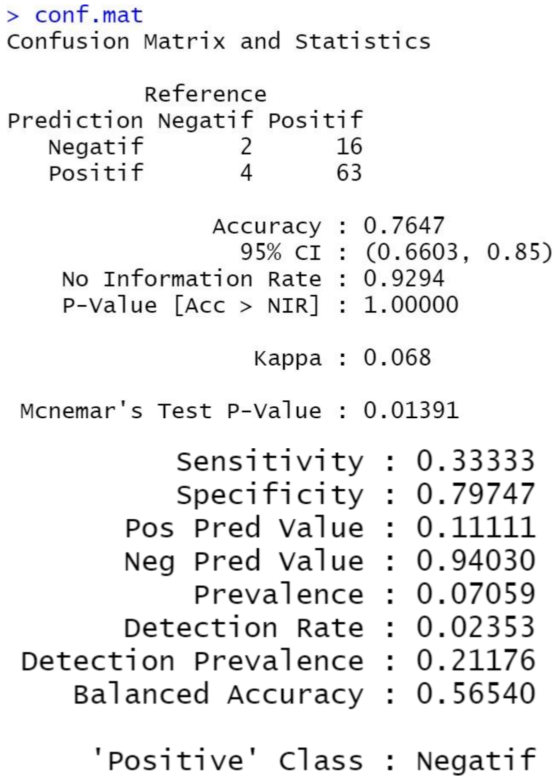
```
   Cell Contents
|-------------------------|
|                       N |
|           N / Row Total |
|           N / Col Total |
|-------------------------|


Total Observations in Table:  85
```

Fig. 11. Crosstable Naïve Bayes Classifier Test Results

The following in Fig. 12. is the accuracy of the test system using the confusion Matrix function



```
             | Predicted
      Actual |   Negatif |   Positif | Row Total |
-------------|-----------|-----------|-----------|
     Negatif |         2 |        16 |        18 |
             |     0.111 |     0.889 |     0.212 |
             |     0.333 |     0.203 |           |
-------------|-----------|-----------|-----------|
     Positif |         4 |        63 |        67 |
             |     0.060 |     0.940 |     0.788 |
             |     0.667 |     0.797 |           |
-------------|-----------|-----------|-----------|
Column Total |         6 |        79 |        85 |
             |     0.071 |     0.929 |           |
-------------|-----------|-----------|-----------|
```



```
> conf.mat
Confusion Matrix and Statistics

          Reference
Prediction Negatif Positif
   Negatif       2      16
   Positif       4      63

               Accuracy : 0.7647
                 95% CI : (0.6603, 0.85)
    No Information Rate : 0.9294
    P-Value [Acc > NIR] : 1.00000

                  Kappa : 0.068

 Mcnemar's Test P-Value : 0.01391

            Sensitivity : 0.33333
            Specificity : 0.79747
         Pos Pred Value : 0.11111
         Neg Pred Value : 0.94030
             Prevalence : 0.07059
         Detection Rate : 0.02353
   Detection Prevalence : 0.21176
      Balanced Accuracy : 0.56540

       'Positive' Class : Negatif
```

Fig. 12. System Accuracy Results Using Naive Bayes Classifier

The confusion Matrix function is a syntax for displaying system accuracy results which can be executed by calling the “caret” package. It can be seen in the picture above that the value of the system calculation accuracy with the Naive Bayes method is 0.7647 or 76.5%.

System testing with confusion matrix

Based on the accuracy test, the results obtained from the accuracy of the comment data classification from the sentiment analysis system using Labeling and tested with the Naive Bayes Classifier system of 76.47% with system errors of 23.53%, 94.02 % for Recall, and 79.74% for Precision.

The results of the accuracy of sentiment analysis in this study can be concluded that public opinion through Instagram comments on the Gramedia Pustaka Utama (GPU) Instagram account using 2 methods, namely Lexicon Based and Naive Bayes Classifier, has a positive tendency.

TABLE 4.1 Table *Confusion Matrix* Labelisasi dengan Sistem

Amount of Test Data 85		Reference / Actual Class	
		Negative	Positive
Predicted Class	Negative	TN = True Negative 2	FP = False Positive 16
	Positive	FN = False Negative 4	TP = True Positive 63

V. CONCLUSION

Based on the results of trials conducted by researchers, several conclusions can be drawn in this study, including:

1. The results of preprocessing and initial classification using the Lexicon Based method in this study resulted in 419 data comments from 506 data. The 419 data consists of 362 positive sentiments and 57 negative sentiments.
2. Of the 419 available commentary data for the next stage, it is divided for the training and testing stages. The division is 80% data for training, 20% data for testing.
3. Performance results using the Naive Bayes Classifier method, get a percentage value of 76.47 % for Accuracy, 94.02 % for Recall, and 79.74 % for Precision.

The results of the study can determine the sentiment towards the object conveyed in the comments on the Gramedia Pustaka Utama (GPU) Instagram account in Indonesian which has a positive tendency with the Naive Bayes Classfier method.

REFERENCES

[1]. Adriani, M., J. Asian, B. Nazief, S. M. Tahaghoghi, and H. E. Williams (2007). *Stemming indonesian: A confix-stripping approach*. ACM Transactions on Asian Language Information Processing (TALIP) 6(4), 1-33.

[2]. Afrizal, H. Irmanda. (2019). *Implementasi Metode Naive Bayes untuk Analisis Sentimen Warga Jakarta Terhadap Kehadiran Mass Rapid Transit.* "Jurnal Informatika". Vol. 15 No. 13, hlm 1-12.

[3]. Antinasari, P., Perdana, R. S., & Fauzi, M. A. (2017). *Analisis Sentimen Tentang Opini Film Pada Dokumen Twitter Berbahasa Indonesia Menggunakan NaïveBayes Dengan Perbaikan Kata Tidak Baku.*

[4]. Ding, X., Liu, B., & Yu, Philip S. (2008). *A Holistic Lexicon-Based Approach to Opinion Mining. WSDM*.

[5]. Feldman, R., J. Sanger, et al. (2007) *The text mining handbook: Advanced Approaches in Analyzing Unstructured Data*. Cambridge university press.

[6]. Gunawan, H. Sastypratiwi. 2018. *Sistem Analisis pada Ulasan Produk Menggunakan Metode Naive Bayes.* "Jurnal Edukasi dan Penelitian Informatika". Vol. 4 No. 2, hlm 1-6.

[7]. Gusti Nur Aulia dan Eka Patriya. (2019) *Implementasi Lexicon Based dan Naive Bayes pada Analisis Sentimen Pengguna Twitter Topik Pemilihan Presiden 2019*.

[8]. Han, J., J. Pei, J., & Kamber, M. (2011). *Data mining: concepts and techniques.* Elsevier.

[9]. Kusumawati, I. (2017). *Analisa Sentimen Menggunakan Lexicon Based untuk Melihat Persepsi Masyarakat Terhadap Kenaikan Harga Rokok pada MediaSosial Twitter*.

[10]. Mindrawa, R. (2017). *Penerapan Algortima KNN dan K-Means untuk penggolongan Pendapat Masyarakat di Twitter Terhadap PT POS Indonesia*.

[11]. Munir, M. M., Fauzi, M. A., & Perdana, R. S. (2018). *Implementasi Metode Backpropagation Neural Network Berbasis Lexicon Based Features dan Bagof Words untuk Identifikasi Ujaran Kebencian pada Twitter.*

[12]. Prihatiningsih W. (2017). *Motif Pengguna Media Sosial Instagram Di Kalangan Instagram.* hlm 1-15.

[13]. Ravindran, Sharan Kumar & Garg, Vikram. (2015). *Mastering Social Media Mining with R*. Packt Publishing Ltd. UK.

[14]. Siti Mujilahwati. "*Pre-Processing Text Mining* Pada Data Twitter". Seminar Nasional Teknologi Informasi dan Komunikasi 2016 (SENTIKA 2016).

[15]. Supriyatna Adi. (2018). *Komparasi Algoritma Naive Bayes dan SVM untuk Mempresiksi Keberhasilan Imunoterapi pada Penyakit Kutil.* "Jurnal Sains Komputer dan Informatika". Vol. 15 No. 3, hlm 1-11.

[16]. Tala, F. Z. (2003). *A Study of Stemming Effects on Information Retrieval in* BahasaIndonesia. M.Sc. *Thesis,* Appendix D, pp : 39-46.

[17]. Umi Rofiqoh, Rizal Setya Perdana dan M. Ali Fauzi. (2017) *Analisis Sentimen Tingkat Kepuasan Pengguna Penyedia Layanan Telekomunikasi Seluler Indonesia Pada Twitter Dengan Metode Support Vector Machine dan Lexicon Based Features*.