

**LAPORAN PROYEK DATA SCIENCE  
ANALISIS SENTIMEN REVIEW APLIKASI BIBIT DENGAN METODE  
NAÏVE BAYES**



**DISUSUN OLEH:**

<b>MUHAMMAD FARHAN HARVITO</b>	<b>123200029</b>
<b>ALFIAN RIZKI MUGHNI AZIZ</b>	<b>123200050</b>
<b>MUHAMMAD IKHWAN HANIF</b>	<b>123200096</b>

**PROGRAM STUDI INFORMATIKA  
JURUSAN INFORMATIKA  
FAKULTAS TEKNIK INDUSTRI  
UNIVERSITAS PEMBANGUNAN NASIONAL "VETERAN"  
YOGYAKARTA  
2022**

## **DAFTAR ISI**

<b>DAFTAR ISI</b>	<b>2</b>
<b>1. PENDAHULUAN</b>	<b>3</b>
<b>2. METODE</b>	<b>3</b>
<b>3. HASIL DAN PEMBAHASAN</b>	<b>4</b>
<b>3.1 Code Program</b>	<b>4</b>
<b>3.2 Tampilan GUI</b>	<b>11</b>
<b>4. KESIMPULAN</b>	<b>13</b>
<b>5. PEMBAGIAN TUGAS</b>	<b>13</b>

## **1. PENDAHULUAN**

Pada era teknologi yang semakin meningkat dan yang semakin modern berpengaruh terhadap kehidupan individu maupun kelompok. Perkembangan TI dapat membantu dalam mengembangkan tugas-tugas baru pada perusahaan berskala pasar global atau pada instansi pemerintah, TI dimanfaatkan untuk mengatasi meningkatkan kemampuan seseorang dalam meraih keunggulan dan kesuksesan yang handal serta canggih.

Pemanfaatan teknologi informasi juga digunakan dalam aplikasi Bibit yang dapat digunakan disegala aspek dengan menggunakan data yang telah tersedia, data ini akan diolah dengan teknologi yang akan menghasilkan informasi transparan, akurat dan mudah dipahami oleh semua kalangan.

Salah satu kegunaan teknologi informasi dalam museum yaitu untuk mengetahui penilaian dari pengunjung setelah berkunjung ke museum. Analisis tersebut biasa disebut sebagai Analisa Sentimen atau Sentiment Analysis ini adalah proses penggunaan text analisis untuk mendapatkan berbagai sumber data dari internet dan beragam platform media sosial. Tujuannya adalah untuk memperoleh opini dari pengguna yang terdapat pada platform tersebut.

Dengan menggunakan teknologi dan sekumpulan data yang ada, dapat menghasilkan informasi yang akurat juga mudah dipahami tanpa perlu waktu yang lama. Menggunakan metode Naïve Bayes Classifier untuk mengolah datanya dan Bahasa R yang digunakan untuk membuat aplikasinya.

## **2. METODE**

Data yang kita dapat berasal dari scrapping Aplikasi Bibit di Google Play dengan bahasa python untuk mendapatkan dataset tentang Aplikasi Bibit. Metode yang kami gunakan yaitu Naïve Bayes Classifier merupakan merupakan sebuah pengklasifikasian probabilistik sederhana yang menghitung sekumpulan probabilitas dengan menjumlahkan frekuensi dan kombinasi nilai dari dataset yang diberikan. Algoritma menggunakan teorema Bayes dan mengasumsikan semua atribut independen atau tidak saling ketergantungan yang diberikan oleh nilai pada variabel kelas.

Naïve Bayes didasarkan pada asumsi penyederhanaan bahwa nilai atribut secara kondisional saling bebas jika diberikan nilai output. Dengan kata lain, diberikan nilai output, probabilitas mengamati secara bersama adalah produk dari probabilitas individu. Keuntungan penggunaan Naïve Bayes adalah bahwa metode ini hanya membutuhkan jumlah data pelatihan (Training Data) yang kecil untuk menentukan estimasi parameter yang diperlukan dalam proses pengklasifikasian. Naïve Bayes sering bekerja jauh lebih baik dalam kebanyakan situasi dunia nyata yang kompleks dari pada yang diharapkan Naïve Bayes Classifier dinilai bekerja sangat baik dibanding dengan model classifier lainnya, yaitu Naïve Bayes Classifier memiliki tingkat akurasi yg lebih baik dibanding model classifier lainnya.

### 3. HASIL DAN PEMBAHASAN

#### 3.1 Code Program

**Scraping Data Review Ulasan Bibit di Playstore.** Dilakukan proses scraping data melalui bahasa python untuk mendapatkan hasil review aplikasi Bibit di Google Play Store.

```
!pip install google-play-scraper

from google_play_scraper import app

import pandas as pd #untukmembuatdataframe

import numpy as np #untukoperasivektroataumatriks

from google_play_scraper import Sort, reviews_all

us_reviews = reviews_all(

    'com.bibit.bibitid',

    sleep_milliseconds=0, # defaults to 0

    lang='id', # defaults to 'en'

    country='id', # defaults to 'us'

    sort=Sort.NEWEST, # defaults to Sort.MOST_RELEVANT

    count=250,

)

df_data = pd.DataFrame(np.array(us_reviews),columns=['review'])

df_data = df_data.join(pd.DataFrame(df_data.pop('review').tolist()))

df_data.head()

len(df_data.index)

df_data[['userName', 'score','at', 'content']].head()

review = df_data[['userName', 'score','at', 'content']]

sorted_df = review.sort_values(by='at', ascending=False)
```

```
sorted_df.head()

ss = sorted_df[['userName', 'score', 'at', 'content']]

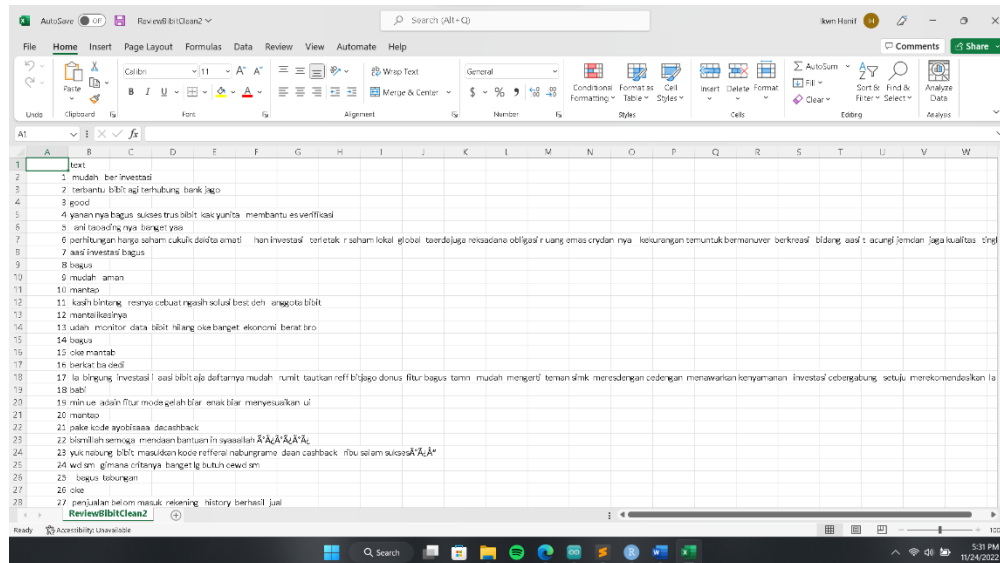
ss.head()

ss.to_csv("ReviewBibit.csv", index = False)
```

**Bagian 1. Preprocessing Data.** ReviewBibit.csv dilakukan pembersihan data untuk mendapatkan hasil review pengguna saja, tanpa tanda baca, huruf besar, titik, titik koma, dan sebagainya. Lalu disimpan pada file ReviewBibitClean.csv

```
```{r}
library(tm) #Untuk membersihkan data
library(vroom) #Load Dataset
library(here) #Menyimpan Dataset

d <- vroom(here('ReviewBibit.csv'))
ulasan <- d$content
ulasan1 <- Corpus(VectorSource(ulasan))
removeURL <- function(x) gsub("http[^\s:space:]*", "", x)
reviewclean <- tm_map(ulasan1, removeURL)
removeNL <- function(y) gsub("\n", " ", y)
reviewclean <- tm_map(ulasan1, removeNL)
replacecomma <- function(y) gsub(",", "", y)
reviewclean <- tm_map(reviewclean, replacecomma)
removetitik2 <- function(y) gsub(".", "", y)
reviewclean <- tm_map(reviewclean, removetitik2)
removetitikkoma <- function(y) gsub(";", " ", y)
reviewclean <- tm_map(reviewclean, removetitikkoma)
removetitik3 <- function(y) gsub("p...", "", y)
reviewclean <- tm_map(reviewclean, removetitik3)
removeamp <- function(y) gsub("&", "", y)
reviewclean <- tm_map(reviewclean, removeamp)
removeUN <- function(z) gsub("@\\w+", "", z)
reviewclean <- tm_map(reviewclean, removeUN)
remove.all <- function(xy) gsub("[^\s:alpha:][\s:space:]*", "", xy)
reviewclean <- tm_map(reviewclean, remove.all)
reviewclean <- tm_map(reviewclean, removePunctuation)
reviewclean <- tm_map(reviewclean, tolower)
MyStopWords <- readLines("stopwords-id.txt")
reviewclean <- tm_map(reviewclean, removeWords, MyStopWords)
dataframe <- data.frame(text=unlist(sapply(reviewclean, `[ ])), stringsAsFactors = F)
View(dataframe)
write.csv(dataframe, file = 'ReviewBibitClean2.csv')```
```



**Gambar 3.1.1 Data frame hasil cleaning**

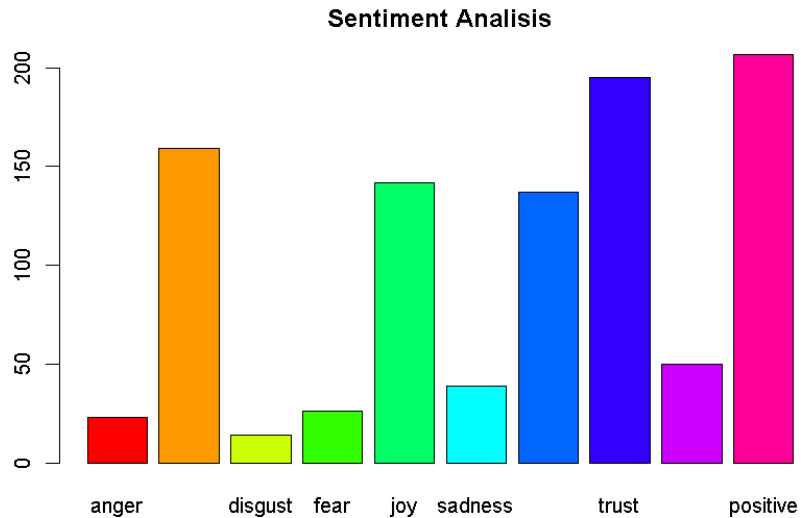
**Bagian 2.** Setelah dilakukan data Cleaning, proses selanjutnya adalah Sentimen Analisis dengan Metode Naïve Bayes.

```

```{r Sentimen Analisis Naive Bayes Classifier}
library(e1071) #Untuk Naive Bayes
library(caret) #untuk Klasifikasi Data
library(syuzhet) #untuk membaca fungsi get_nrc
library(ggplot2)
library(plotly)
bibitdata <- read.csv("ReviewBibitClean2.csv", stringsAsFactors = FALSE)
review <- as.character(bibitdata$text) #merubah text menjadi char
s <- get_nrc_sentiment(review)

review_combine <- cbind(bibitdata$text,s) #klasifikasi Data
par(mar=rep(3,4))
a <- barplot(colSums(s), col=rainbow(10),ylab='count',main='Sentiment Analisis BIBIT')
brplt <-

```



**Gambar 3.1.2 Hasil Sentimen Analisis Berupa BarPlot**

**Bagian 3.** Visualisasi data berupa word cloud yang akan menampilkan seberapa sering kata itu terinput sistem berdasarkan ulasan yang diberikan. Dilakukan pengaturan seed generator bilangan acak R, yang berguna untuk membuat simulasi atau objek acak yang dapat direproduksi.

```

```{r}
library(wordcloud2)
library(vroom)
library(here)
require (corpus)
df<-read.csv("ReviewBibitClean2.csv",stringsAsFactors = FALSE)
glimpse(df)
set.seed(20)
df<-df[sample(nrow(df)),]
df<-df[sample(nrow(df)),]
glimpse(df)
corpus<-Corpus(VectorSource(df$text))
corpus
inspect(corpus[1:10])
#fungsinya untuk membersihkan data data yang tidak dibutuhkan
corpus.clean<-corpus%>%
  tm_map(content_transformer(tolower))%>%
  tm_map(removePunctuation)%>%
  tm_map(removeNumbers)%>%
  tm_map(removeWords, c("yang", "dan", "dari", "aasi", "ini", "kita", "untuk"
, "nya"))%>%
  tm_map(removeWords, stopwords(kind="en"))%>%
  tm_map(stripWhitespace)
dtm<-DocumentTermMatrix(corpus.clean)
inspect(dtm[1:10, 1:20])
df.train<-df[1:50,]
df.test<-df[51:100,]

```





**Bagian 4.** Selanjutnya melakukan skoring dari ulasan pengguna Bibit. Apakah sentimen termasuk netral, negatif, atau positif

```
```{r}
kalimat2<-read.csv("ReviewBibitClean2.csv",header=TRUE)
#skoring
kata.positif <- scan("kata-pos.txt",what="character",comment.char=";")
kata.negatif <- scan("kata-neg.txt",what="character",comment.char=";")
score.sentiment = function(kalimat2, kata.positif, kata.negatif,
                           .progress='none')
{
  require(plyr)
  require(stringr)
  scores = laply(kalimat2, function(kalimat, kata.positif,
                                   kata.negatif) {
    kalimat = gsub('[:punct:]', "", kalimat)
    kalimat = gsub('[:cntrl:]', "", kalimat)
    kalimat = gsub('\\d+', "", kalimat)
    kalimat = tolower(kalimat)
    list.kata = str_split(kalimat, "\\s+")
    kata2 = unlist(list.kata)
    positif.matches = match(kata2, kata.positif)
    negatif.matches = match(kata2, kata.negatif)
    positif.matches = !is.na(positif.matches)
    negatif.matches = !is.na(negatif.matches)
    score = sum(positif.matches) - (sum(negatif.matches))
    return(score)
  }, kata.positif, kata.negatif, .progress=.progress )
  scores.df = data.frame(score=scores, text=kalimat2)
  return(scores.df)}
hasil = score.sentiment(kalimat2$text, kata.positif, kata.negatif)
#mengubah nilai score menjadi sentimen
hasil$klasifikasi<- ifelse(hasil$score<0,
"Negatif",ifelse(hasil$score==0,"Netral","Positif"))
hasil$klasifikasi
#menukar urutan baris
data <- hasil[c(3,1,2)]
#View(data)
write.csv(data, file = "datalabel.csv")```
```

**Bagian 5.** Selanjutnya, dibuat sebuah *user interface* sederhana berdasarkan dataset yang telah diolah dan dilakukan visualisasi tersebut di atas.

```
```{r}
library(syuzhet) #untuk membaca fungsi get_nrc
library(shiny)
dataLabel<- read.csv("datalabel.csv")
ui <- fluidPage(
  titlePanel("Sentimen Analisis Ulasan Bibit"),
  mainPanel(

    tabsetPanel(type = "tabs",
```

```

        tabPanel("Bagan", plotOutput("scatterplot")),
        # Plot
        tabPanel("Data", DT::dataTableOutput('tbl1')),
        # Output Data Dalam Tabel
        tabPanel("Wordcloud", plotOutput("Wordcloud"))
    )
)
# SERVER
server <- function(input, output) {

    # Output Data
    output$tbl1 = DT::renderDataTable({
        DT::datatable(dataLabel, options = list(lengthChange = FALSE))
    })

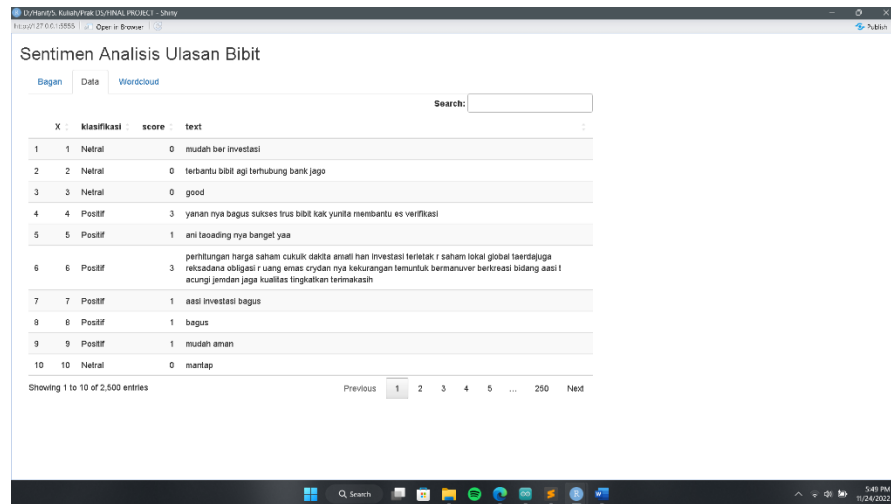
    output$scatterplot <- renderPlot({produk_dataset<-
read.csv("ReviewBibitClean2.csv",stringsAsFactors = FALSE)
    review <-as.character(produk_dataset$text)
    s<-get_nrc_sentiment(review)
    review_combine<-cbind(produk_dataset$text,s)
    par(mar=rep(3,4))
    barplot(colSums(s),col= rainbow(10),ylab='count',main='Sentimen Analisis Ulasan
Bibit')
    }, height=400)
    output$Wordcloud <- renderPlot({
        set.seed(20)
        df<-df[sample(nrow(df)),]
        df<-df[sample(nrow(df)),]
        glimpse(df)
        inspect(dtm[1:10,1:20])
        df.train<-df[1:50,]
        df.test<-df[51:100,]
        dtm.train<-dtm[1:50,]
        dtm.test<-dtm[51:100,]
        dim(dtm.train)
        fivefreq<-findFreqTerms(dtm.train,5)
        length(fivefreq)
        dtm.train.nb<-DocumentTermMatrix(corpus.clean.train,control =
list(dictionary=fivefreq))
        #dim(dtm.train.nb)
        dtm.test.nb<-DocumentTermMatrix(corpus.clean.test,control =
list(dictionary=fivefreq))
        dim(dtm.test.nb)

    convert_count <- function(x){
        y<-ifelse(x>0,1,0)
        y<-factor(y,levels=c(0,1),labels=c("no","yes"))
        y
    }
    trainNB<-apply(dtm.train.nb,2,convert_count)
    testNB<-apply(dtm.test.nb,1,convert_count)
    wordcloud(corpus.clean,min.freq =
4,max.words=100,random.order=F,colors=brewer.pal(8,"Dark2"))
    })
}

```

```
}
shinyApp(ui = ui, server = server)``
```

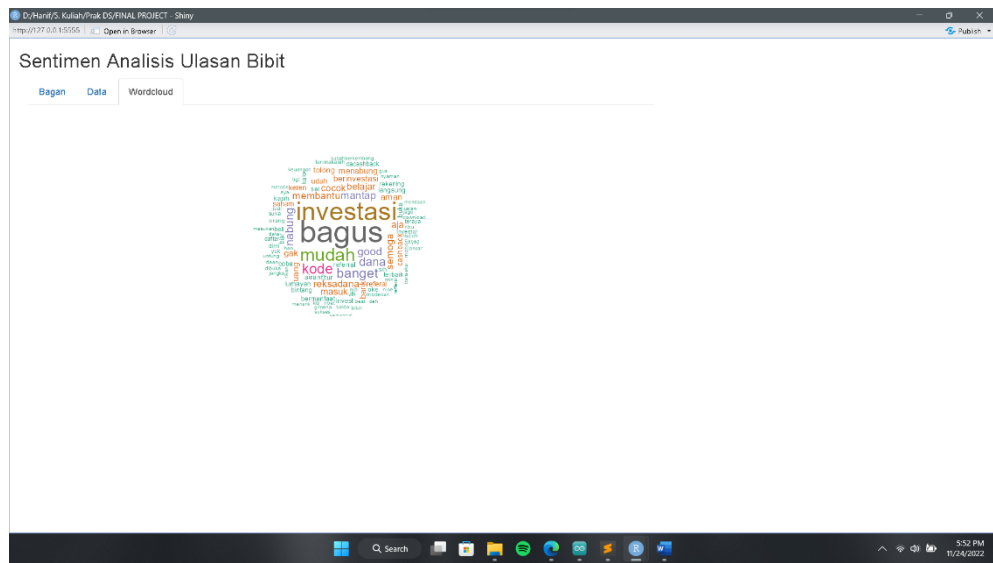
## 3.2 Tampilan GUI



Gambar 3.2.1 Data Sentimen



Gambar 3.2.2 Tampilan Sentimen dalam Barplot



**Gambar 3.2.3 Tampilan WorldCloud pada shiny**

#### 4. KESIMPULAN

Dengan adanya Sentiment Analysis ini, pihak developer Bibit dapat mengetahui penilaian dari para pengguna aplikasi Bibit. Dari data yang diperoleh oleh praktikan menyimpulkan bahwa Sebagian besar menilai positif setelah atau saat menggunakan aplikasi Bibit dan ada juga yang merasa kurang puas. Dengan adanya program ini semoga dapat lebih berkembang dan memudahkan developer Bibit.

#### 5. PEMBAGIAN TUGAS

Pembagian tugas proyek mata kuliah Data Science Analisis Sentimen Review Aplikasi Bibit dengan Metode Naïve Bayes, sebagai berikut:

**Tabel 4.2 Tabel Pembagian Tugas**

<b>No.</b>	<b>Aktivitas</b>	<b>Pelaksana</b>
1.	Penentuan judul	Harvito
2.	Perancangan konsep proyek	Alfian
3.	Pencarian dataset	Hanif
4.	Pembuatan program	Hanif, Alfian, Harvito
5.	Penyempurnaan	Hanif, Alfian, Harvito
6.	Pembuatan laporan	Hanif, Alfian, Harvito