

**LAPORAN PROYEK AKHIR
PRAKTIKUM DATA SCIENCE**

**Analisis Sentimen Review Aplikasi Bibit dengan
Metode Support Vector Machine**



Muhammad Ikhwan Hanif	123200096
Taufik Sahid Fadhil	123200028

**PROGRAM STUDI INFORMATIKA
JURUSAN TEKNIK INFORMATIKA
FAKULTAS TEKNIK INDUSTRI
UNIVERSITAS PEMBANGUNAN NASIONAL “VETERAN”
YOGYAKARTA
2021**

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.

Bibit adalah aplikasi jual beli reksa dana online. Dalam aplikasi Bibit, pengguna bisa membelo reksa dana pasar uang, obligasi, saham, syariah, serta Surat Berharga Negara. Fitur fitur yang disediakan dari bibit juga dapat memudahkan pengguna untuk mendapat rekomendasi pilihan reksa dana terbaik. Rekomendasi tersebut diberikan berdasarkan profil risiko yang telah diisi pengguna disaat membuat akun bibit. Itulah beberapa alasan mengapa Bibit disebut sebagai aplikasi investasi yang cocok bagi pemula.

Analisis sentiment dibutuhkan untuk mendapatkan sentiment dari pengguna bibit melalui ulasan pada aplikasi Bibit. Analisis Sentimen 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 analisis sentimen yaitu untuk mengetahui penilaian dari ulasan atau review aplikasi Bibit. 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 Support Vector Machine untuk mengolah datanya dan Bahasa R yang digunakan untuk membuat aplikasinya

2. METODE

Dalam penelitian proyek menggunakan Teknik data mining yaitu Supervised learning bahwa Supervised Learning merupakan metode belajar dimana terdapat pelatih atau data testing dan juga Latihan atau data training yang kemudian menjadi output dari proses training tersebut. Kemudian dari data training dan testing dihasilkan analisis sentiment dengan menerapkan Algoritma Support Vector Machine. Analisis sentiment dilakukan untuk menentukan kategori komentar positif dan negative.

2.1 Pengumpulan Data

Dataset yang akan digunakan dalam analisis sentiment ini adalah dataset berbahasa Indonesia yang merupakan hasil ulasan seseorang mengenai Aplikasi Bibit. Dataset didapatkan dengan melakukan scraping menggunakan bahasa pemrograman python pada playstore kemudian disimpan dalam bentuk csv.

2.2 Preprocessing Data

Proses preprocessing ini dilakukan melalui beberapa proses agar mendapatkan data ulasan yang bersih, dan akan digunakan dalam pengolahan data ulasan. Preprocessing dilakukan dalam bahasa R melalui aplikasi Rstudio.

2.3 Proses Klasifikasi

Metode klasifikasi yang dipakai adalah Support Vector Machine karena diketahui algoritma ini sangat baik untuk dilakukan pada klasifikasi teks dan tidak memerlukan kemampuan komputasi yang berat, metode ini sangat mudah di implementasi pada perangkat yang tidak terlalu mumpuni untuk melakukan pembelajaran mesin dan sangat sering menjadi benchmark untuk metode metode lain.

3. HASIL DAN PEMBAHASAN

Pada proyek ini data diolah menggunakan algoritma Support Vector Machine . Berikut adalah hasil dan pembahasan dari Langkah uji coba

3.1 Scraping Aplikasi Bibit

Pengambilan data melakukan proses scraping secara manual menggunakan bahasa python diambil id dari aplikasi bibit yang didapatkan melalui website play.google.com

3.2 Preprocessing Data.

Setelah memperoleh data kotor dari proses scraping, maka tahapan selanjutnya adalah proses preprocessing data. Melalui tahapan ini dataset diolah melalui berbagai macam proses seperti penhilangan URL, enter, koma, titik koma, titik dua, symbol, dan menghilangkan kata sambung pada ulasan agar kata yang diproses adalah inisi dari ulasan. Juga mengubah teks menjadi huruf kecil untuk semua ulasan.

Tabel Hasil Preprocessing

No	Ulasan
1.	mudah ber investasi
2.	terbantu bibit lagi terhubung bank jago
3.	good
4.	yang bagus sukses trus bibit kak yunita membantu es verifikasi
5.	ani taoding nya banget yaa
6.	perhitungan harga saham cukuik dakita amati han investasi terletak saham lokal global taerda juga reksadana obligasi ruang emas crydan kekurangan temuntuk bermanuver berkreasi bidang acungi jempol jaga kualitas tingkatkan terimakasih
7.	investasi bagus
8.	bagus
9.	mudah aman
10.	mantap

3.3 Klasifikasi Data

Pada proses klasifikasi data dilakukan dengan tujuan mendapatkan nilai klasifikasi ulasan positif dan negatif. proses ini menggunakan algoritma Support Vector Machine.

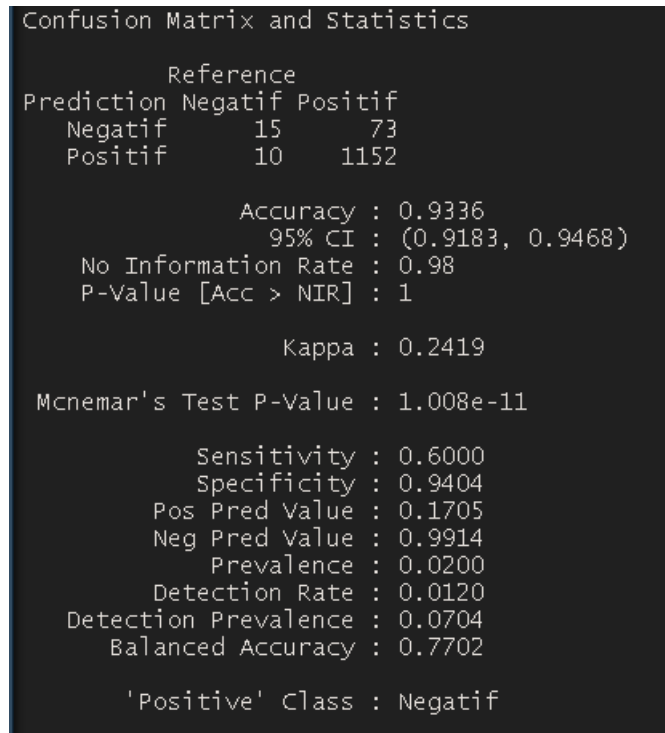
Tabel Hasil Klasifikasi

No	Klasifikasi	Score	Ulasan
1	Positif	0	mudah ber investasi
2	Positif	0	terbantu bibit lagi terhubung bank jago
3	Positif	0	good
4	Positif	3	yang bagus sukses trus bibit kak yunita membantu es verifikasi
5	Positif	1	ani taoding nya banget yaa
6	Positif	3	perhitungan harga saham cukuik dakita amati han investasi terletak saham lokal global taerda juga reksadana obligasi ruang emas crydan kekurangan temuntuk bermanuver berkreasi bidang acungi jempol jaga kualitas tingkatkan terimakasih
7	Positif	1	investasi bagus
8	Positif	1	bagus
9	Positif	1	mudah aman
10	Positif	1	mantap

3.4 Melakukan Uji SVM dari Data

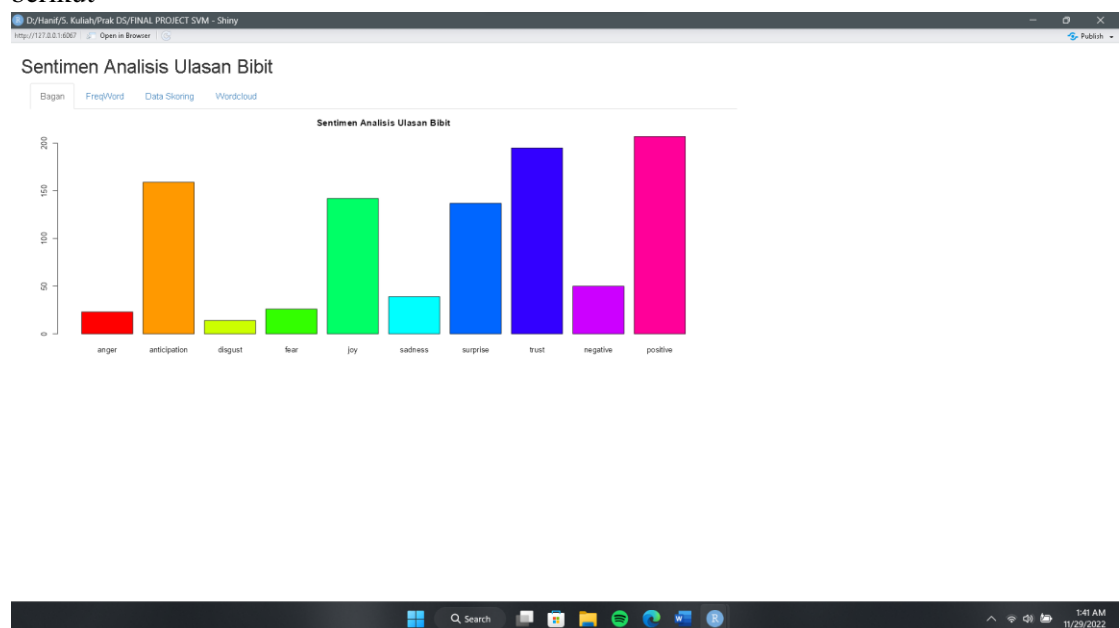
Dari Pengujian SVM dari data dapat dilihat yaitu hasil untuk pengujian untuk data train. Dapat dilihat pada nilai p-value(McNemar's Test P-Value) sebesar 1.008×10^{-11} , Balanced Accuracy atau tingkat akurasi dari hasil prediksi dengan data testing sebesar 0.7702 (77.02%). Sensitivity atau presentasi true positif atau yang di prediksi benar adalah 0.6000 (60%), dan specificity atau nilai presentasi true negative yaitu sebesar 0.9404 (94.04%).

Gambar Confusion Matrix



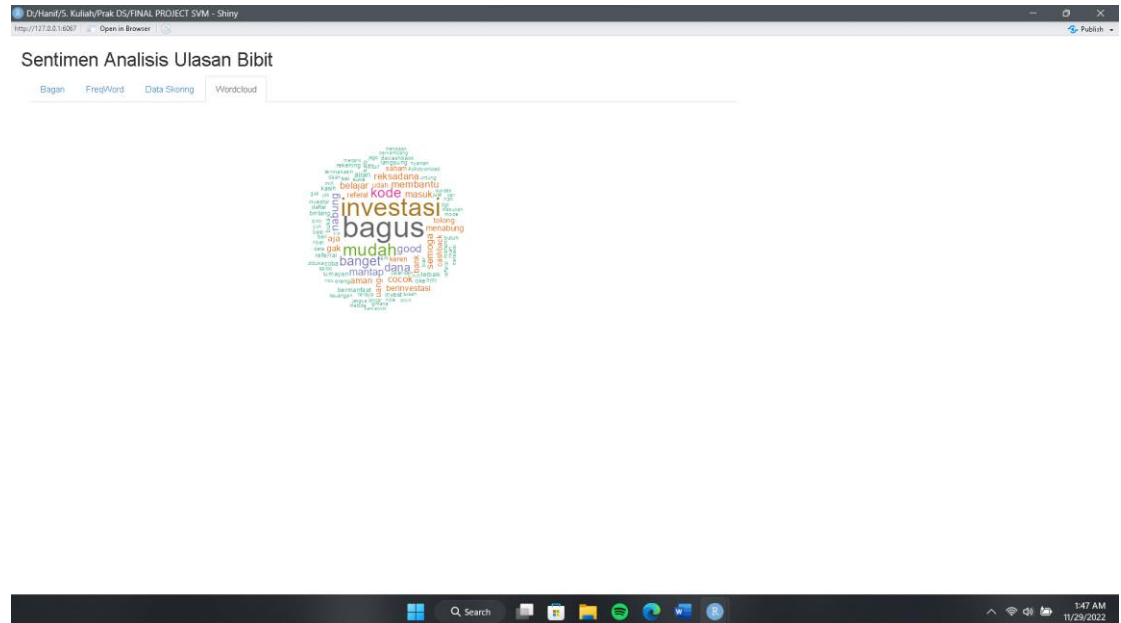
3.5 Visualisasi dengan Shiny

Pada tahap visualisasi ini menampilkan output dengan shinyApp, seperti pada gambar berikut

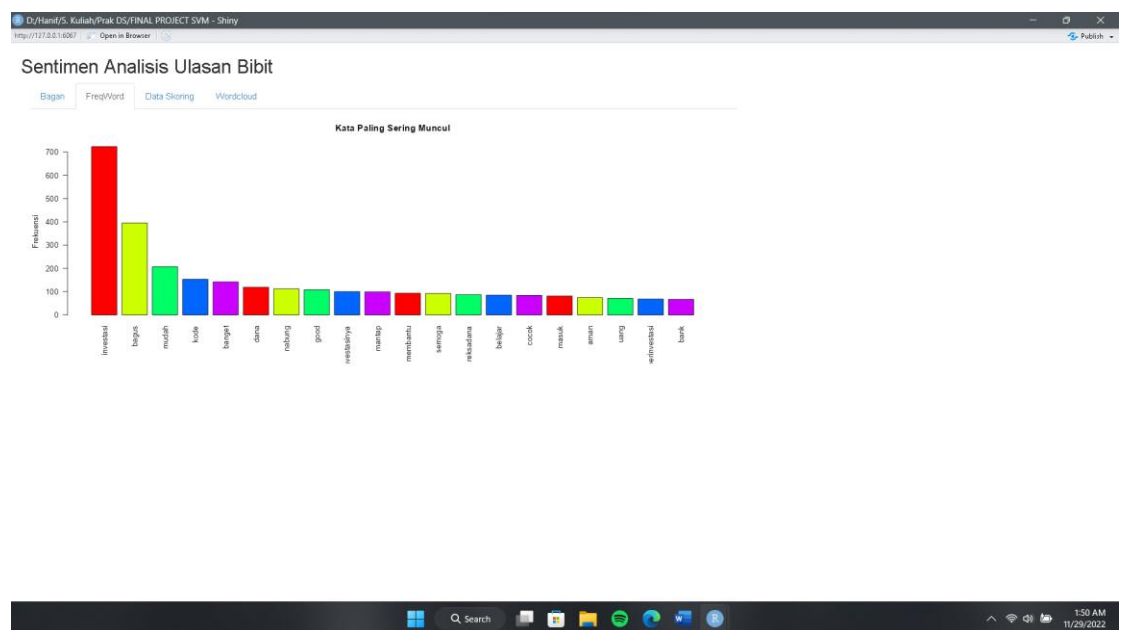


Gambar 1 Barplot Analisis Sentimen ulasan untuk Review Bibitdata

Dari Visualisasi mengenai review ulasan Bibit, ditemukan jumlah ulasan positif lebih banyak dibandingkan dengan Negatif. Menandakan orang-orang yang telah menggunakan aplikasi bibit mempunyai pengalaman yang positif dengan aplikasi Bibit.



Gambar 2 Wordcloud



Gambar 2 Frekuensi Plot

Analisis berikutnya adalah wordcloud dan barplot frekuensi. Dibuat menggunakan kata kata yang sering dituliskan dan emosi yang terkait dengan mereka. Dapat dilihat kata investasi dan bagus menjadi kata yang paling banyak dijadikan kata ulasan dalam Aplikasi Bibit.

3.6 Listing Program

Tabel Data Cleaning

```
---
title: "Project Cleaning Data"
author: "Muhammad Ikhwan Hanif/123200096 - Taufik Sahid
Fadhil/123200028"
date: "2022-11-20"
output: html_document
---
```{r setup, include=FALSE}
knitr::opts_chunk$set(echo = TRUE)
```
```{r}
library(tm) #Untuk membersihkan data
library(vroom) #Load Dataset
library(here) #Menyimpan Dataser
```
```{r load dataset}
d <- vroom(here('ReviewBibit.csv'))
ulasan <- d$content
ulasan1 <- Corpus(VectorSource(ulasan))
removeURL <- function(x) gsub("http[^\[: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("[^\[:alpha:]\[: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')
```
```

Tabel Skoring

```
---
title: "SentimenAnalisis Metode SVM"
author: "Muhammad Ikhwan Hanif/123200096 - Taufik Sahid
Fadhil/123200028"
date: "2022-11-20"
output: html_document
---
```{r setup, include=FALSE}
knitr::opts_chunk$set(echo = TRUE)
```
```{r}
library(tm)
library(wordcloud2)
library(vroom)
library(here)
library(RTextTools)
library(dplyr)
library(wordcloud)
library(shiny)
library(ggplot2)
library(plotly)
```
```

Tabel Listing Program SVM

```
library(e1071)
library(caret)
library(devtools)
cf = read.csv("D:/Hanif/5. Kuliah/Prak DS/FINAL PROJECT
SVM/pelabelan.csv")
cf$klasifikasi = factor(cf$klasifikasi)
corpus2 = Corpus(VectorSource(cf$text))
dtm = DocumentTermMatrix(corpus2)

dtMatrix = create_matrix(cf["text"], language = "id",
removeStopwords = FALSE,
                        removeNumbers = FALSE, stemWords
                        = FALSE, tm::weightTfIdf)
#mat = as.matrix(dtMatrix)
container = create_container(dtMatrix, cf$klasifikasi,
trainSize = 1251:2500, testSize = 1:1250, virgin = FALSE)
#Modeling
model = train_model(container, "SVM", kernel = "linear", cost
= 1)
result = classify_model(container, model)
result$SVM_LABEL = factor(result$SVM_LABEL)
confussionMatrix = confusionMatrix(cf$klasifikasi[1:1250],
result[, "SVM_LABEL"])
confussionMatrix
```
```

## Tabel Listing Wordcloud & Freqplot

```
```{r}
```



```

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","aasinya","ini", "kita", "untuk" ,"nya",
    "bibit","öy"))%>%
  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,]
dtm.train<-dtm[1:50,]
dtm.test<-dtm[51:100,]
corpus.clean.train<-corpus.clean[1:50]
corpus.clean.test<-corpus.clean[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 =
20,max.words=70,random.order=F,colors=brewer.pal(8,"Dark2"))
```


```

```{r freq}
data1 = read.csv("pelabelan.csv")
corpus = Corpus(VectorSource(data1$text))
corpus<- tm_map(corpus, gsub, pattern="aasi","aasinya",
replacement="investasi")

```


```

```

corpus <- tm_map(corpus, removeWords,"bibit")
corpus <- tm_map(corpus, removeWords,"nya")
corpus <- tm_map(corpus, removeWords,"aja")
corpus <- tm_map(corpus, removeWords,"gak")
corpus <- tm_map(corpus, removeWords,"ðý")
dtm <- TermDocumentMatrix(corpus)
m <- as.matrix(dtm)
v <- sort(rowSums(m),decreasing=TRUE)
d <- data.frame(word = names(v),freq=v)
barplot(d[1:20,]$freq, las = 2, names.arg = d[1:20,]$word,
col=rainbow(5),
      main = "Kata Paling Sering Muncul", ylab =
      "Frekuensi")
      ...

```

Tabel Listing ShinyApp

```

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

 tabsetPanel(type = "tabs",
 #Output Bagan
 tabPanel("Bagan",
plotOutput("scatterplot")),
 #Output Frekuensi
 tabPanel("FreqWord",
plotOutput("freqplot")),
 # Plot
 tabPanel("Data Skoring",
DT::dataTableOutput('tbl1')),
 # Output Data Dalam Tabel
 tabPanel("Wordcloud",
plotOutput("Wordcloud"))
)
)
)

SERVER
server <- function(input, output) {
 # Output Data Skoring
 output$tbl1 = DT::renderDataTable({
 DT::datatable(dataLabel, options = list(lengthChange
= FALSE))
 })
 #Barplot
 output$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 Freq Plot
output$freqplot <- renderPlot({data1 =
read.csv("pelabelan.csv")
corpus = Corpus(VectorSource(data1$text))
corpus<- tm_map(corpus, gsub, pattern="aasi", "aasinya",
replacement="investasi")
corpus <- tm_map(corpus, removeWords, "bibit")
corpus <- tm_map(corpus, removeWords, "nya")
corpus <- tm_map(corpus, removeWords, "aja")
corpus <- tm_map(corpus, removeWords, "gak")
corpus <- tm_map(corpus, removeWords, "ðy")

dtm <- TermDocumentMatrix(corpus)
m <- as.matrix(dtm)
v <- sort(rowSums(m), decreasing=TRUE)
d <- data.frame(word = names(v), freq=v)

barplot(d[1:20,]$freq, las = 2, names.arg = d[1:20,]$word,
col=rainbow(5),
 main = "Kata Paling Sering Muncul", ylab =
"Frekuensi")
 }, height=400)

#WordCloud
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)
``,`
```

#### 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.