# **LAPORAN PROYEK AKHIR**

**PRAKTIKUM DATA SCIENCE**

**Analisis Sentimen Ulasan Google Map Candi Prambanan dengan Metode Naïve Bayes**



|  |  |
| --- | --- |
| Alfian Saputra | 123200007 |
| Yuni Safitri | 123200088 |

**PROGRAM STUDI INFORMATIKA**

**JURUSAN TEKNIK INFORMATIKA**

**FAKULTAS TEKNIK INDUSTRI**

**UNIVERSITAS PEMBANGUNAN NASIONAL “VETERAN” YOGYAKARTA**

**2022**

**HALAMAN PENGESAHAN**

**LAPORAN PROYEK AKHIR**

****

Disusun Oleh:

*Alfian Saputra 123200007  
Yuni Safitri 123200088*

Telah Diperiksa dan Disetujui oleh Asisten Praktikum pada tanggal:…………………

|  |  |
| --- | --- |
| **Asisten Pratikum** | **Asisten Pratikum** |
| Dhea Anggita  123190046 | Rico Aminanda  123190079 |

**KATA PENGANTAR**

Puji syukur kami panjatkan kepada Tuhan Yang Maha Esa yang senantiasa mencurahkan rahmat dan hidayah-Nya sehingga saya dapat menyelesaikan praktikum Data Science serta laporan proyek akhir praktikum yang berjudul “Analisis Sentimen Ulasan Google Map Candi Prambanan dengan Metode Naïve Bayes”. Adapun laporan ini berisi tentang proyek akhir yang saya pilih dari hasil pembelajaran selama praktikum berlangsung.

Tidak lupa ucapan terimakasih kepada asisten dosen yang selalu membimbing dan mengajari kami dalam melaksanakan praktikum dan dalam menyusun laporan ini. Laporan ini masih sangat jauh dari kesempurnaan, oleh karena itu kritik serta saran yang membangun kami harapkan untuk menyempurnakan laporan akhir ini.

Atas perhatian dari semua pihak yang membantu penulisan ini, kami ucapkan terimakasih. Semoga laporan ini dapat dipergunakan seperlunya.

Yogyakarta, 25 November 2022

Penyusun

**DAFTAR ISI**

**Cover i**

**Halaman Pengesahan .......ii**

**Kata Pengantar. iii**

**Daftar Isi iv**

**BAB I. Judul Proyek Akhir v**

**BAB II. Pedahuluan 6**

2.1. Latar Belakang Masalah 6

2.2. Tujuan Proyek Akhir 6

**BAB III. Pembahasan 7**

3.1. Metode 7

3.2. Hasil dan Pembahasan 9

**BAB IV. Kesimpulan dan Saran 22**

5.1. Kesimpulan 22

5.2. Saran 22

**BAB I**

**JUDUL PROYEK AKHIR**

**Analisis Sentimen Ulasan Google Map Candi Prambanan dengan Metode Naïve Bayes**

**BAB II**

**PENDAHULUAN**

* 1. **Latar Belakang**

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 melakukan analisis data yang dimana dapat memberikan sebuah data hasil analisis yang dapat berguna untuk kelompok atau individu yang terkait. 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 Naive Bayes Classifier dan KNN untuk mengolah datanya dan Bahasa R yang digunakan untuk membuat aplikasinya.

* 1. **Tujuan Proyek Akhir**

Tujuan dari proyek Data Science ini adalah untuk mengetahui berapa banyak ulasan yang ada di google map mengenai candi prambanan dan kata apa yang banyak muncul dalam ulasannya.

**BAB III**

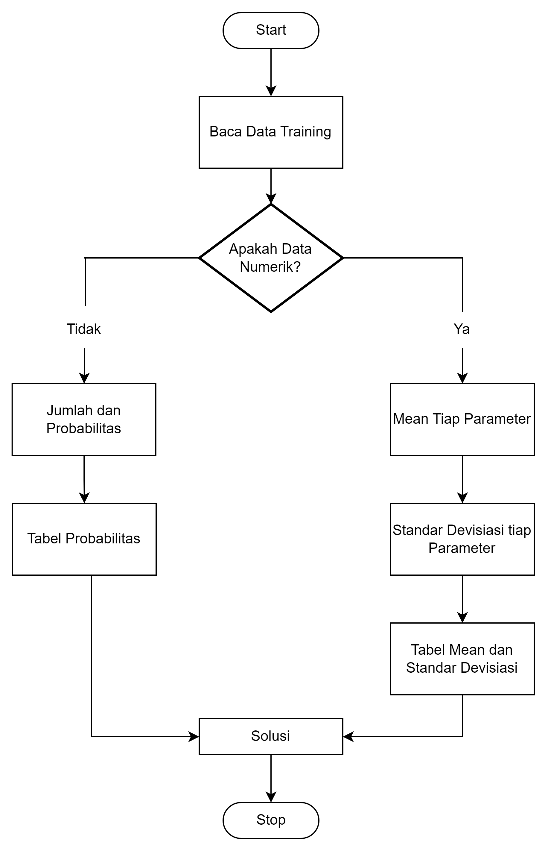
**PEMBAHASAN**

* 1. **Metode**

Data yang kita dapat berasal dari scrapping hasil ulasan digoogle map mengenai candi prambanan untuk mendapatkan dataset tentang Candi Prambanan. 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 mengunakan 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 paremeter 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.



Gambar 1.1

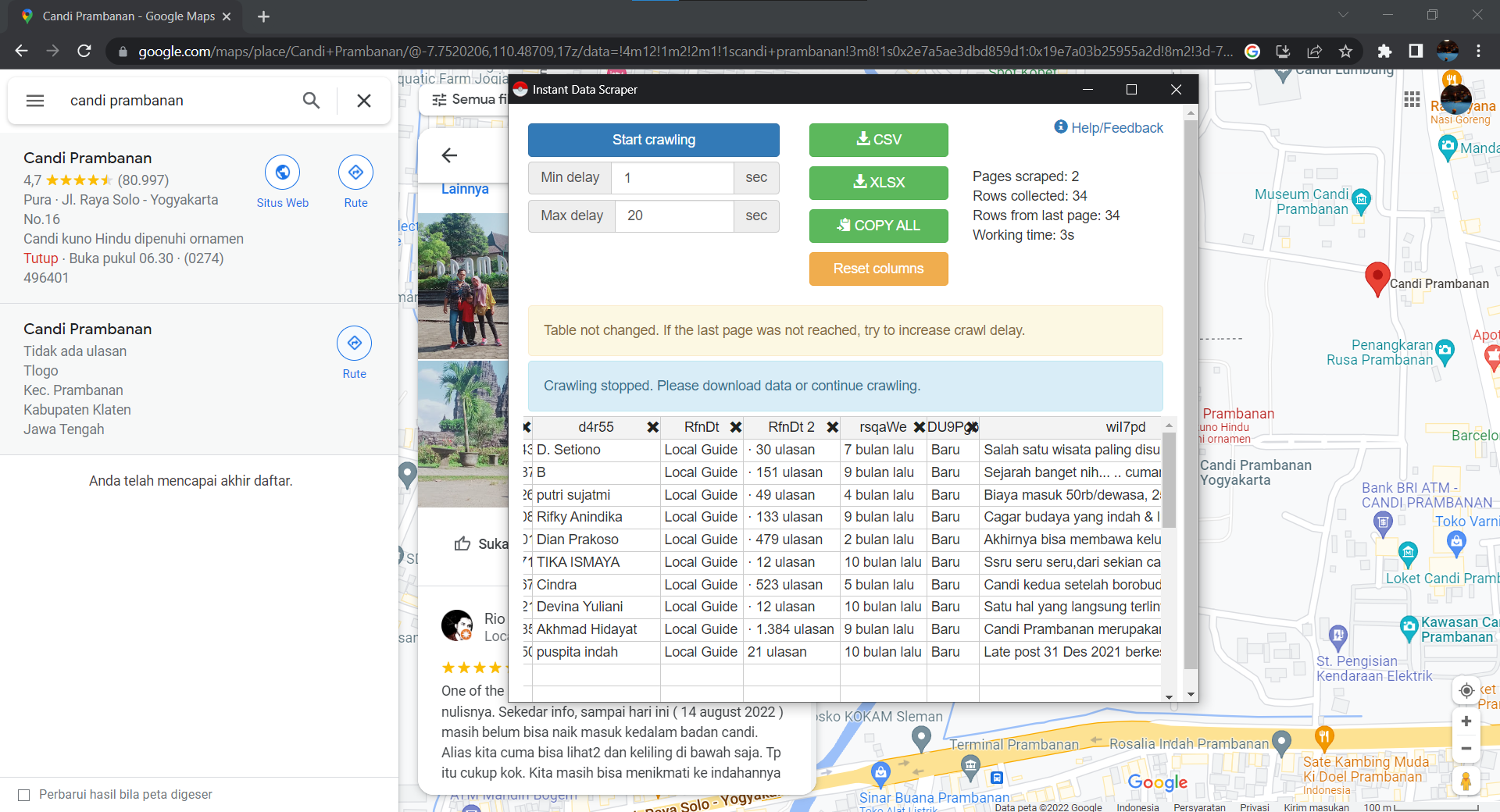
Keterangan:

* Membaca data training.
* Menghitung Jumlah dan Probabilitas dengan cara menghitung jumlah data yang sesuai dari kategori yang sama dibagi dengan jumlah data pada kategori tersebut.
* Mendapatkan nilai dalam table mean, Standart Deviasi dan Probabilitas.
* Menghasilkan solusi.
  1. **Hasil dan Pembahasan**

1. **Listing Program**

Berikut adalah source code dari program Analisis Sentimen Ulasan Google Map Candi Prambanan dengan Metode Naïve Bayes.

Berikut listing program aplikasi instant data scraper.



**Listing Program 3.2.1** instant data scraper

Berikut listing program cleaning data

library(tm) #Untuk membersihkan data

library(vroom) #Load Dataset

library(here) #Menyimpan Dataser

d <- vroom(here('ulasan\_prambanan.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("&amp", "", 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 = 'ulasan\_prambanan\_bersih.csv')

**Listing Program 3.2.2** Cleaning dataset

Berikut listing program utama

Library yang digunakan

```{r}

library(tm)

library(wordcloud2)

library(vroom)

library(here)

library(RTextTools)

library(dplyr)

library(wordcloud)

library(shiny)

library(ggplot2)

library(plotly)

```

**Listing Program 3.2.3** library yang digunakan

Listing program sentiment analisis naïve bayes classifier

```{r Sentimen Analisis Naive Bayes Classifier}

library(e1071) #Untuk Naive Bayes

library(caret) #untuk Klasifikasi Data

library(syuzhet) #untuk membaca fungsi get\_nrc

data <- read.csv("ulasan\_prambanan\_bersih.csv", stringsAsFactors = FALSE)

review <- as.character(data$text) #merubah text menjadi char

s <- get\_nrc\_sentiment(review)

review\_combine <- cbind(data$text,s) #klasifikasi Data

par(mar=rep(3,4))

a <- barplot(colSums(s), col=rainbow(10),ylab='count',main='Sentiment Analisis Prambanan')

brplt <- a

```

**Listing Program 3.2.4** sentiment analisis naïve bayes classifier

Listing program pemodelan

```{r}

require (corpus)

df<-read.csv("ulasan\_prambanan\_bersih.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", "prambanan", "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,]

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 = 4,max.words=100,random.order=F,colors=brewer.pal(8,"Dark2"))

kalimat2<-read.csv("ulasan\_prambanan\_bersih.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")

```

**Listing Program 3.2.5** program pemodelan

Listing program pemodelan pada shiny

```{r}

library(syuzhet) #untuk membaca fungsi get\_nrc

dataLabel<- read.csv("datalabel.csv")

ui <- fluidPage(

titlePanel("Sentimen Analisis Prambanan"),

mainPanel(

tabsetPanel(type = "tabs",

tabPanel("Plot", plotOutput("scatterplot")),

# Plot

tabPanel("Skor", 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("ulasan\_prambanan\_bersih.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=c('gray'),ylab='count',main='Sentimen Analisis Prambanan')

}, 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"))

})

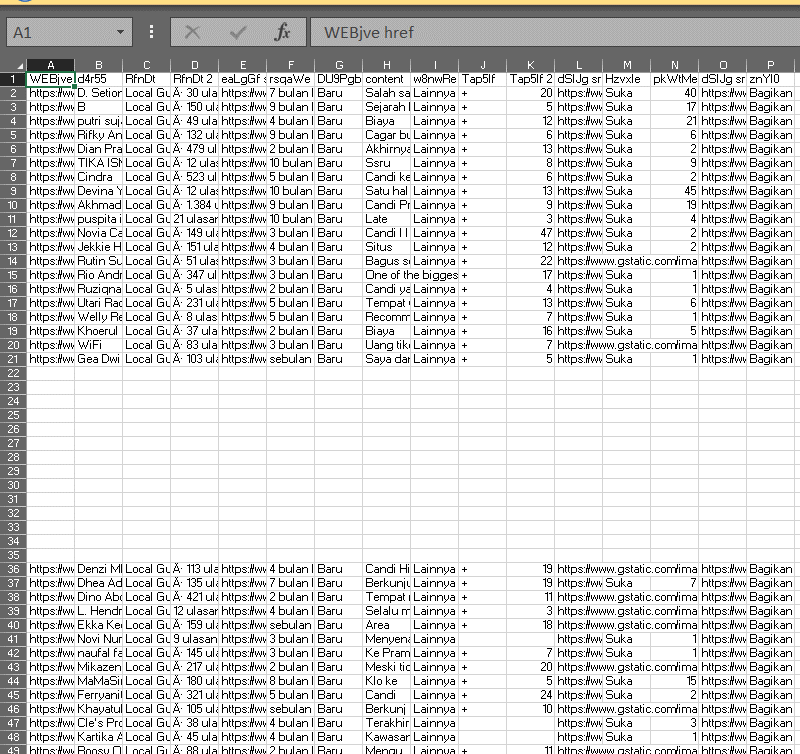
}

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

**Listing Program 3.2.6** pemodelan pada shiny

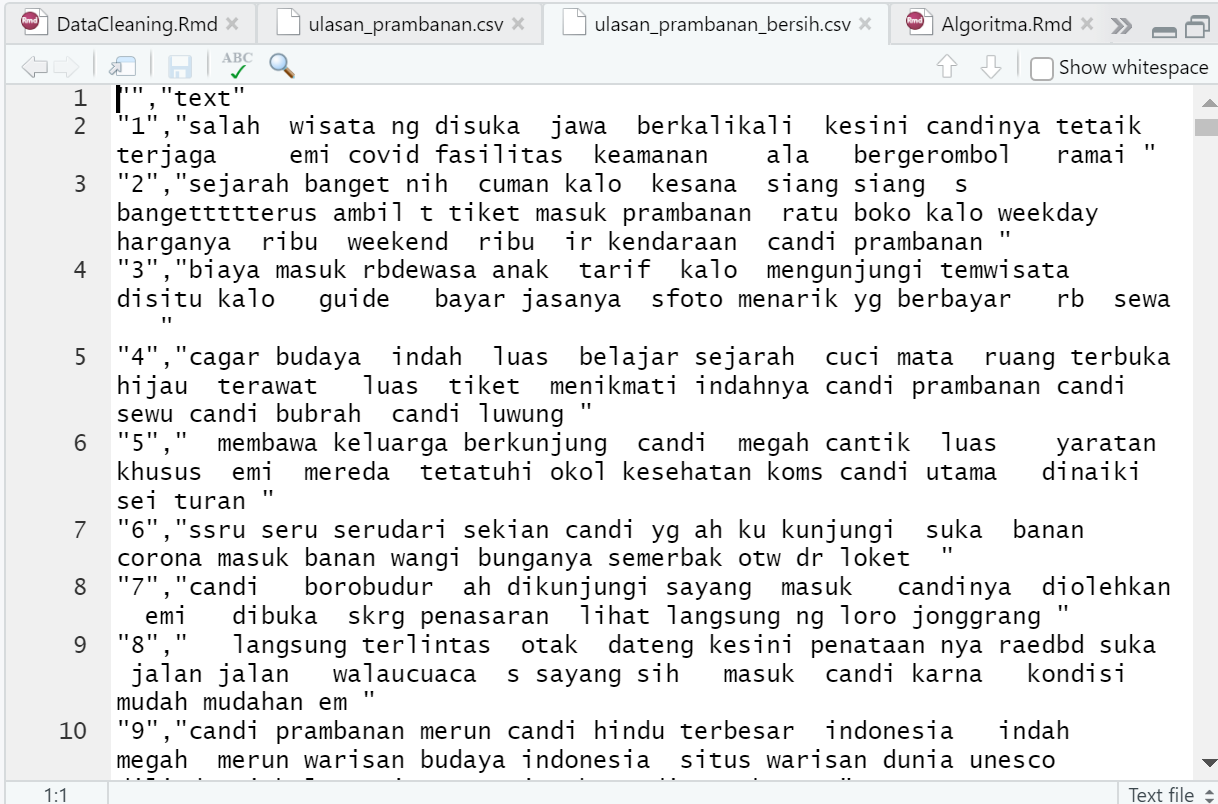
1. **Output Program**

Berikut tampilan scrapping web untuk dataset



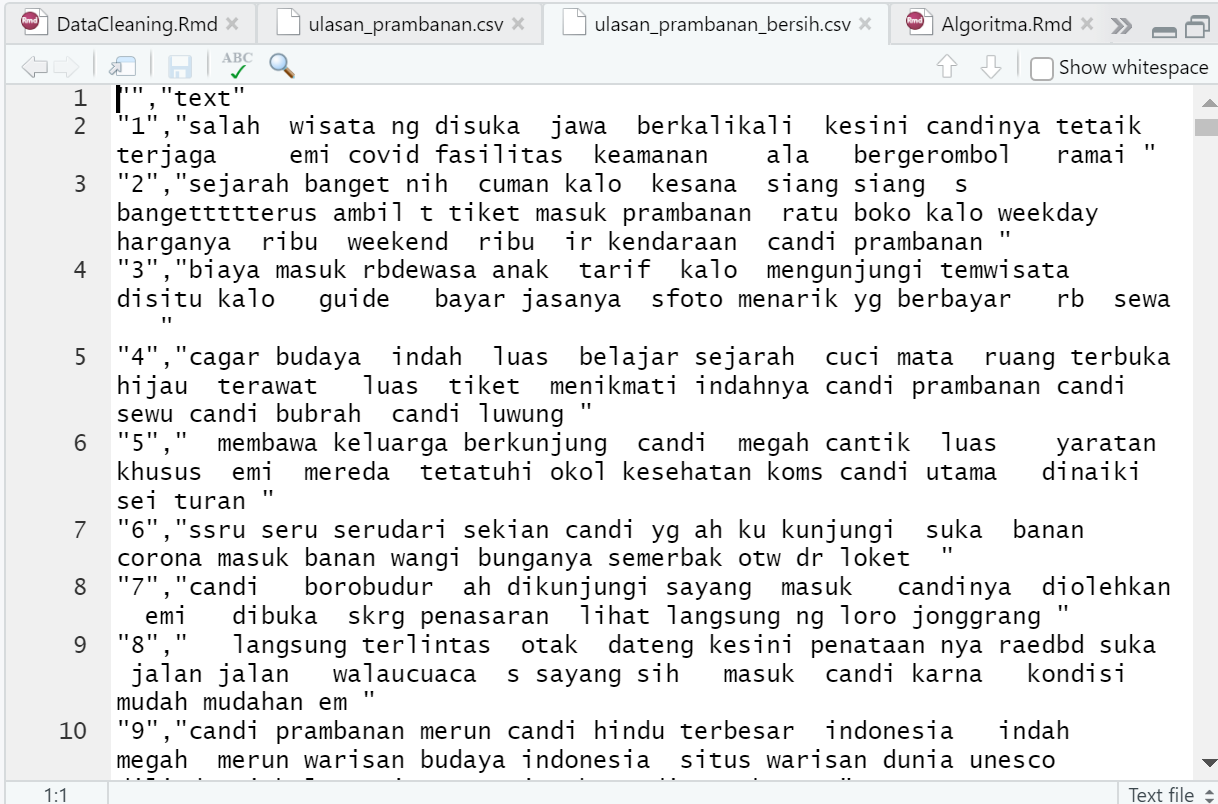
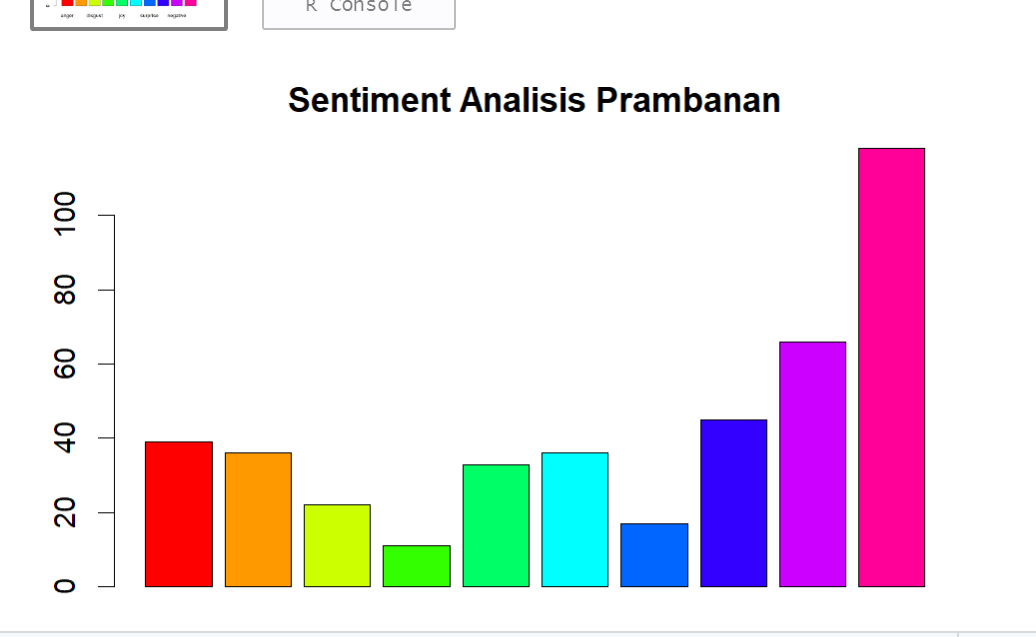
**Output Gambar 3.2.7** Tampilan scrapping web untuk dataset

Berikut tampilan hasil cleaning data



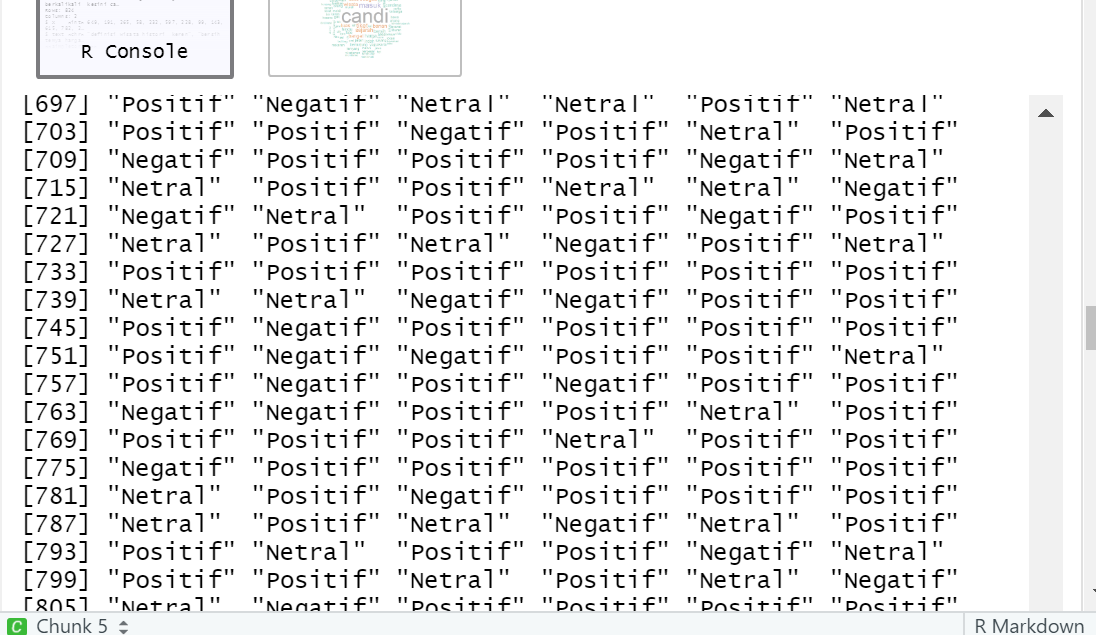
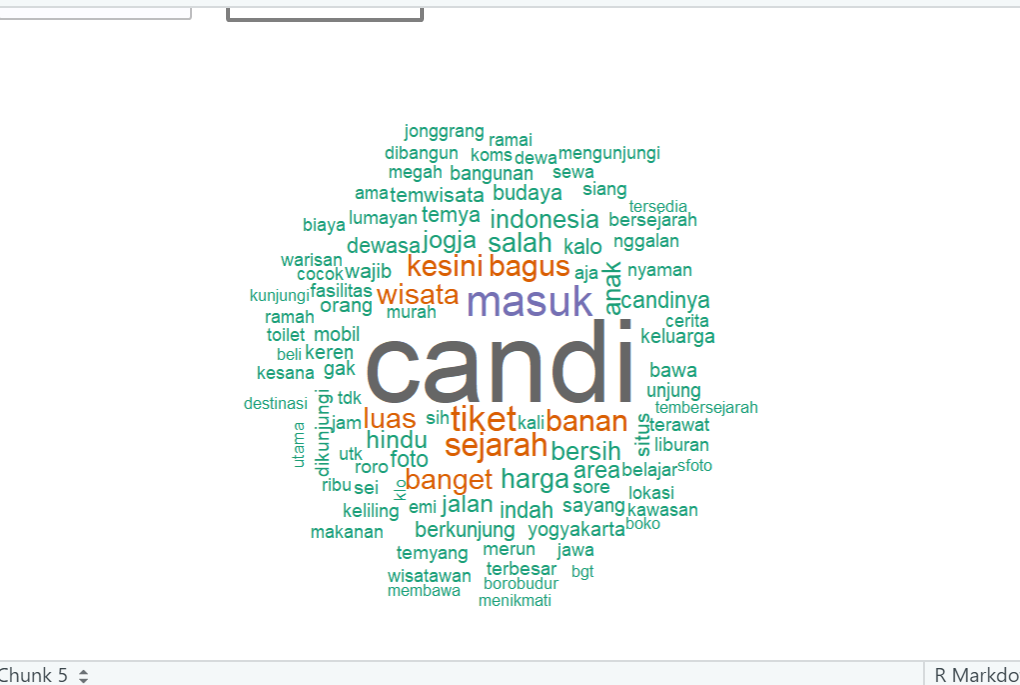
**Output Gambar 3.2.8** Tampilan data yang sudah bersih

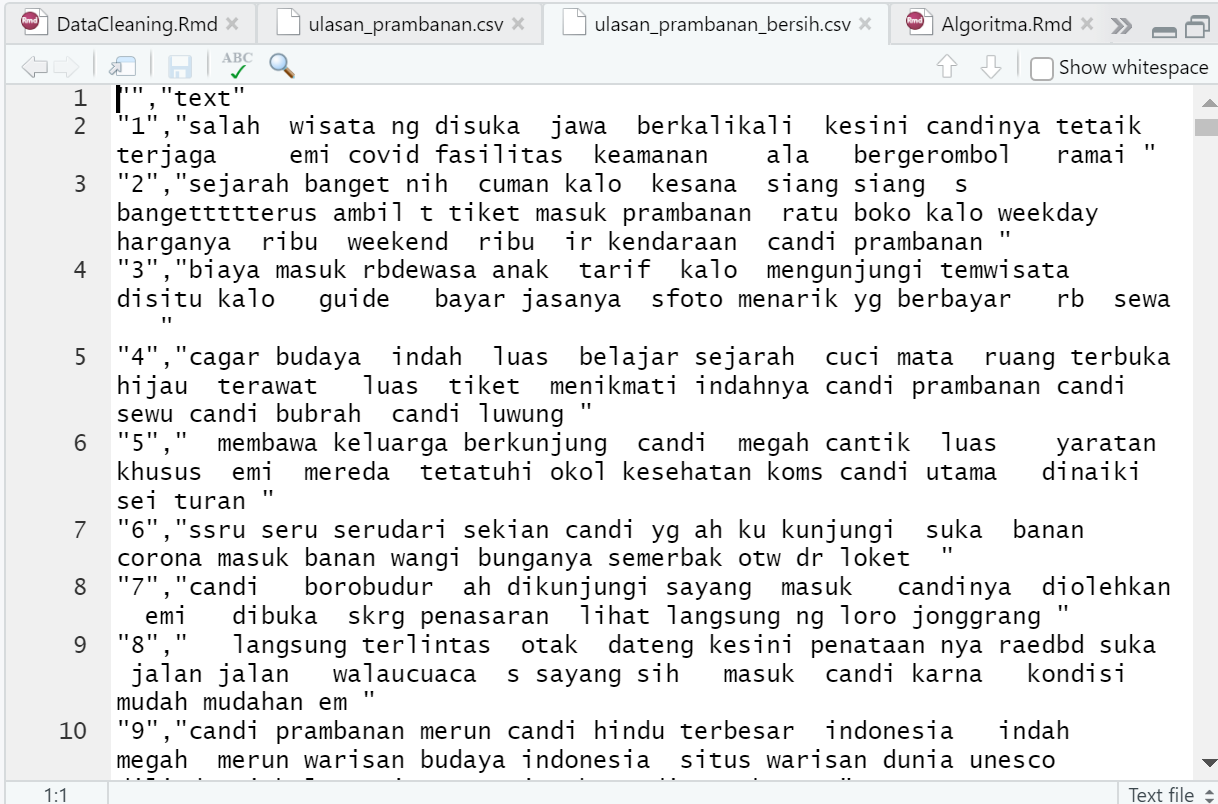
Berikut tampilan hasil sentiment analisis



**Output Gambar 3.2.9** tampilan hasil sentiment analisis

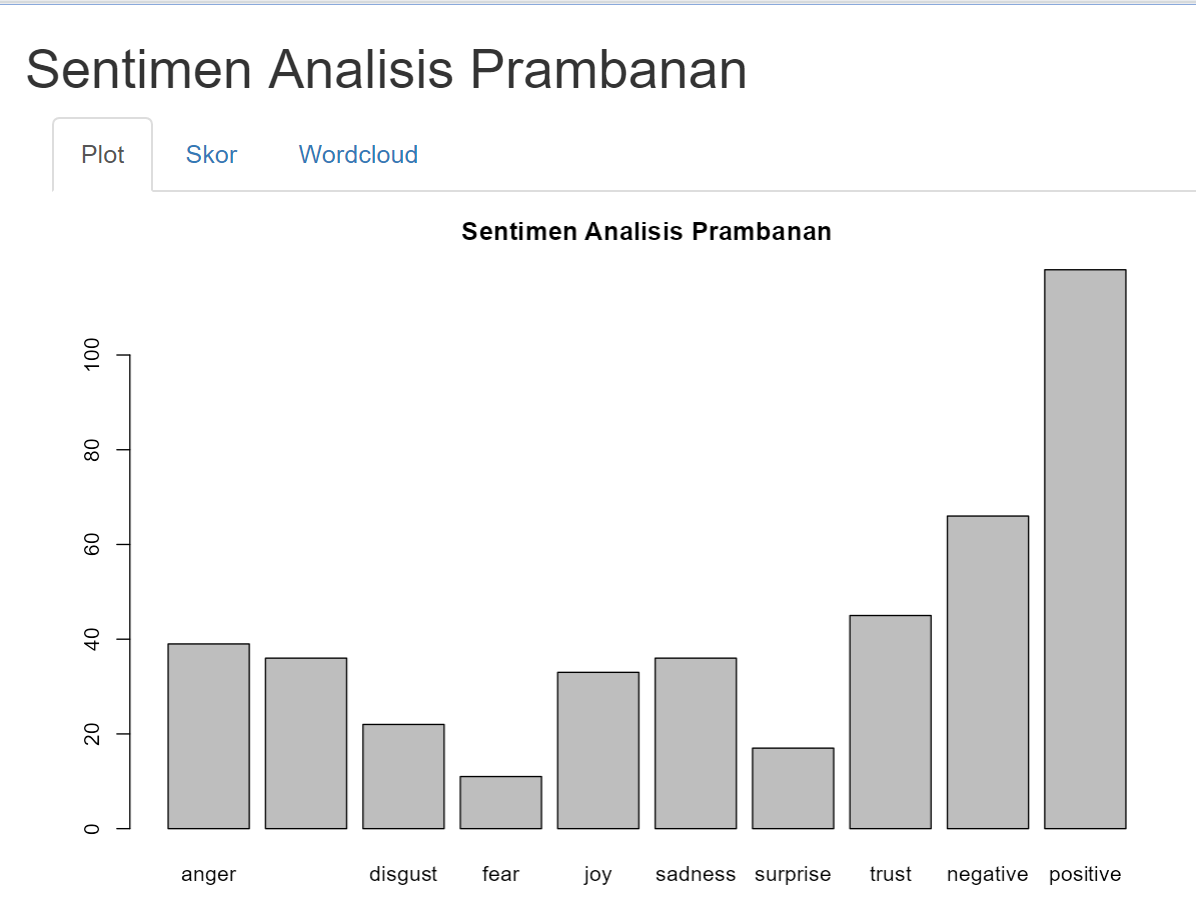
Berikut tampilan hasil pengolahan dan pemodelan

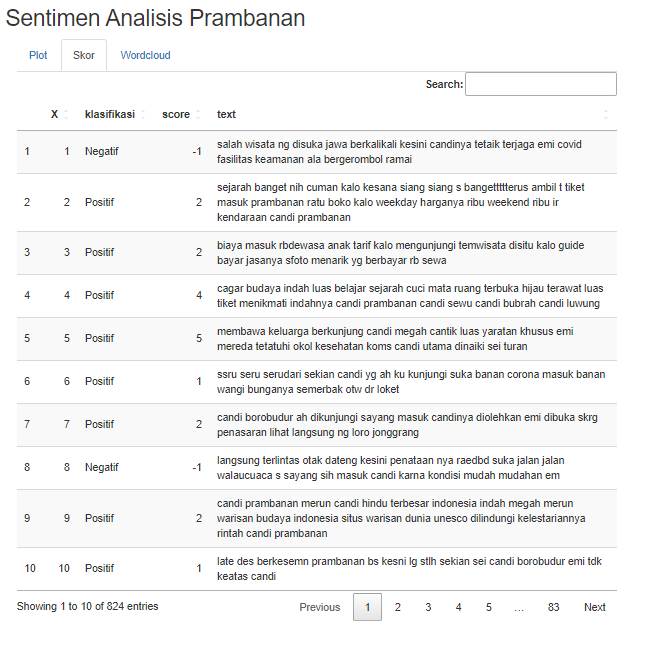


**Output Gambar 3.2.10** Tampilan data yang sudah bersih

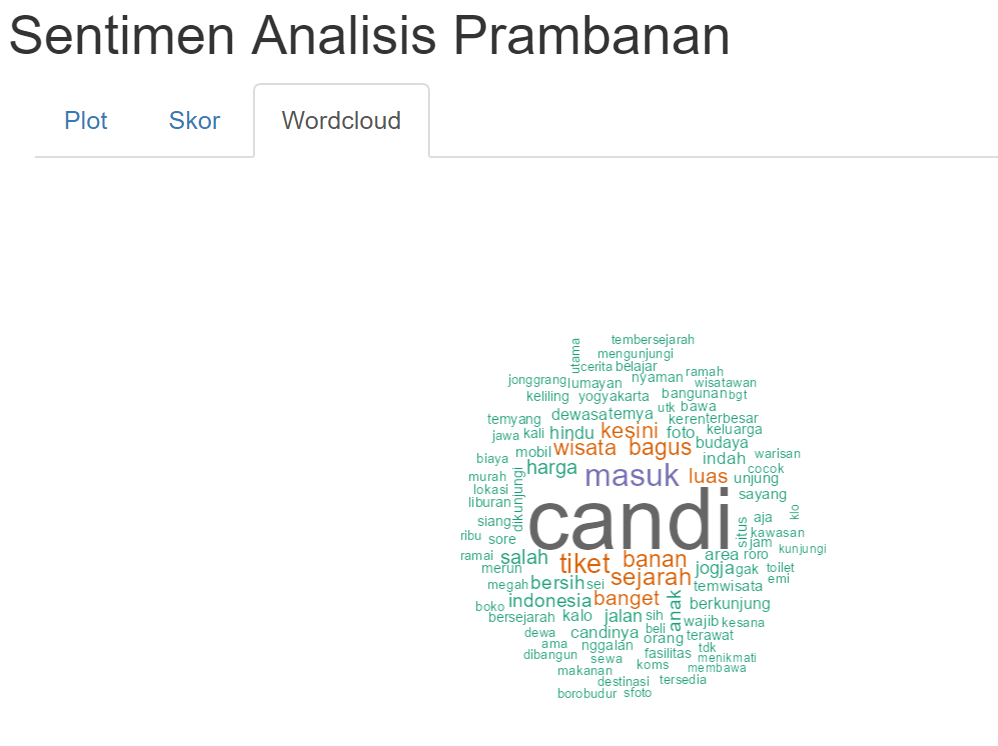
Berikut tampilan hasil pemodelan pada shiny



**Output Gambar 3.2.11** Tampilan pemodelan shiny Plot



**Output Gambar 3.2.12** Tampilan pemodelan shiny skor



**Output Gambar 3.2.13** Tampilan pemodelan shiny worldcloud

**BAB IV**

**KESIMPULAN DAN SARAN**

* 1. **Kesimpulan**

Dengan adanya Sentiment Analysis ini, pihak penjaga atau instasi terkait dapat mengetahui mengetahui penilaian pengunjung untuk candi prambanan dan dapat digunakan sebagai bahan pertimbangan dalam mengembangkan pelayanan atau fasilitas yang berhubungan. Dari data yang diperoleh oleh praktikan menyimpulkan bahwa Sebagian besar menilai positif setelah atau mengunjungi candi prambanan dan ada juga yang merasa kurang puas

* 1. **Saran**

Program yang dibuat oleh penulis masih dapat dikembangkan menjadi lebih komplit dan rapi, karena penulis menyadari bahwa program yang dibuat masih terdapat banyak kekurangan.