# **LAPORAN PROYEK AKHIR**

**PRAKTIKUM DATA SCIENCE**

**ANALISIS SENTIMEN MENGENAI PENDAPAT PENGGUNA TWITTER PERIHAL KASUS WADAS**



|  |  |
| --- | --- |
| MUHAMMAD RIFQY | 123200046 |
| AFRIEN KHOIRUNNISA SHOBAR | 123200093 |

**PROGRAM STUDI INFORMATIKA**

**JURUSAN INFORMATIKA**

**FAKULTAS TEKNIK INDUSTRI**

**UNIVERSITAS PEMBANGUNAN NASIONAL “VETERAN” YOGYAKARTA**

**2022**

# **HALAMAN PENGESAHAN PEMBIMBING**

**Analisis Sentimen Mengenai Pendapat Pengguna Twitter Perihal Kasus Wadas**

|  |
| --- |
| Disusun oleh: |

|  |  |
| --- | --- |
| Muhammad Rifqy | 123200046 |
| Afrien Khoirunnisa Shobar | 123200093 |

|  |  |
| --- | --- |
| Telah Diperiksa dan Disetujui oleh Asisten Praktikum pada | |
| tanggal: 1 Desember 2022 | |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
|  |  |
| Menyetujui, | |
|  | |
| **Asisten Praktikum** | **Asisten Praktikum** |
|  |  |
|  |  |
|  |  |
|  |  |
| **Rico Aminanda** | **Dhea Anggita** |
| **NIM. 123190079** | **NIM. 123190046** |

****

# **PENDAHULUAN**

Wadas adalah nama sebuah desa di Kecamatan Bener, Kabupaten Purworejo, Jawa Tengah. Desa ini menjadi sorotan banyak pihak ketika konfliknya menjadi buah bibir dengan tagline #wadasmelawan.

Hal ini bermula ketika adanya rencana pemerintah mengenai salah satu Proyek Strategis Nasional (PSN), yaitu pembangunan bendungan yang akan memasok sebagian besar kebutuhan air ke Bandara Yogyakarta International Airport (YIA) di Kabupaten Kulon Progo, Yogyakarta.

Menurut data yang tercatat di laman Komite Percepatan Penyediaan Infrastruktur Prioritas (KPPIP), Bendungan Bener rencananya akan memiliki kapasitas 100,94 meter kubik. Dengan kapasitas tersebut, bendungan ini dapat mengairi lahan seluas 15.069 hektar dan mengurangi debit banjir hingga 210 meter kubik per detik. Bendungan ini juga dapat menyediakan pasokan air baku hingga 1,60 meter per detik, serta menghasilkan listrik sebesar 6 MW. Bendungan Bener dibangun menggunakan APBN dengan nilai total proyek mencapai Rp 2,060 triliun.

Dengan kapasitas yang sebesar itu, tentunya perlu bahan baku yang banyak untuk mencukupi kebutuhan dalam pembangunan bendungan ini. Dan untuk memenuhi kebutuhan tersebut, pemerintah bermaksud untuk mendirikan tambang batu andesit di wilayah desa Wadas.

Akan tetapi, hal tersebutlah yang memicu perseteruan antara pihak warga dengan pemerintah. Dari laman petisi terungkap, luas lahan di Desa Wadas yang akan dikeruk untuk penambangan andesit mencapai 145 hektare. Sebagian warga pun menolak rencana penambangan tersebut. Sebab, hal itu dikhawatirkan akan merusak 28 titik sumber mata air warga desa. Rusaknya sumber mata air akan berakibat pada kerusakan lahan pertanian dan lebih lanjut warga kehilangan mata pencaharian. Penambangan tersebut juga dikhawatirkan menyebabkan Desa Wadas semakin rawan longsor. Apalagi, berdasarkan Peraturan Daerah tentang Rencana Tata Ruang Wilayah Kabupaten Purworejo 2011-2031, Kecamatan Bener, termasuk di dalamnya Desa Wadas, merupakan bagian dari kawasan rawan bencana tanah longsor.

Dengan adanya pro kontra tersebut, tentunya hal ini mengundang masyarakat untuk beropini dan mengemukakan pendapatnya sendiri, terutama melalui aplikasi Twitter. Berdasarkan hal tersebut, kami berniat untuk melakukan sebuah analisis sentimen mengenai pendapat pengguna Twitter mengenai kasus wadas ini.

# **METODE**

Dalam penelitian proyek menggunakan metode *data mining* yaitu *supervised learning*. *Supervised learning* merupakan sebuah pemodelan dimana memerlukan *data training*, *data training* sendiri digunakan dalam memprediksi maupun mengklasifikasikan data. Kemudian dari *data training* dihasilkan analisis sentimen dengan menerapkan algoritma *support vector machine* (SVM). Analisis sentimen dilakukan untuk menentukan kategori pendapat positif dan negatif.

1. Pengumpulan Data

*Dataset* yang digunakan dalam analisis sentimen ini adalah *dataset* berbahasa Indonesia yang merupakan hasil dari pendapat pengguna Twitter mengenai “Wadas”. *Dataset* didapatkan dengan melakukan *scrapping* menggunakan bahasa pemrograman R yang kemudian disimpan dalam format csv.

1. *Preprocessing* Data

Proses *preprocessing* ini dilakukan dengan beberapa proses agar didapatkan data pendapat yang bersih dan siap digunakan untuk diolah. *Preprocessing* dilakukan dengan bahasa R melalui aplikasi Rstudio.

1. Proses Klasifikasi

Metode klasifikasi yang dipakai adalah *support vector machine* karena algoritma ini sangat baik untuk mengklasifikasi teks dan tidak memerlukan kemampuan komputasi yang berat. Metode ini sangat mudah di implementasikan pada perangkat standar untuk melakukan pembelajaran mesin.

# **HASIL DAN PEMBAHASAN**

1. Twitter *Scrapping*

Pengambilan data dilakukan dengan proses *scrapping* Twitter menggunakan bahasa R dengan aplikasi Rstudio.

1. Preprocessing Data

Setelah memperoleh data kotor dari proses *scrapping*, tahapan selanjutnya adalah proses *preprocessing* data. Pada tahap ini berbagai macam proses seperti penghilangan URL, *new line*, koma, *retweet*, titik koma, titik dua, titik tiga, dan menghilangkan simbol-simbol pada *tweet* agar kata yang diproses berupa inti dari *tweet*. Selain itu pada tahap ini juga dilakukan pengubahan teks menjadi huruf kecil untuk seluruh *tweet*.

**Tabel 3.1** Hasil *Preprocessing* Data

|  |  |
| --- | --- |
| No | Text |
| 1 | masih percaya wadas saja blom beres |
| 2 | warga wadas sudah menobatkan ganjar penjahat lingkungan |
| 3 | pie wadase tambang ilegalkatanya bekingannya kuate iki itu aja ngeluh gmn mau |
| 4 | masalah wadas blm selesai rakyat sdh antiti sm ganjarngeri kalo dimn rambut ubanan tandanya s |
| 5 | selama ini kok gak digrebek yakena kok waktu akan milua ini skenario waktu makan diwarung |
| 6 | dkt dgn rakyat lalu bgmn dgn kasus warga wadas |
| 7 | rusak |
| 8 | kartika kalau ganjar yg jd esiden berikutnya d disimlkan kondisi indonesia tdk akan terjadi rubahan kemajuan gak usah di |
| 9 | si ganjar pranowo hanya dekat rakyat jateng kalau rakyat jawa barat dan luar jawa jauh rakyat wadas saja |

1. Klasifikasi Data

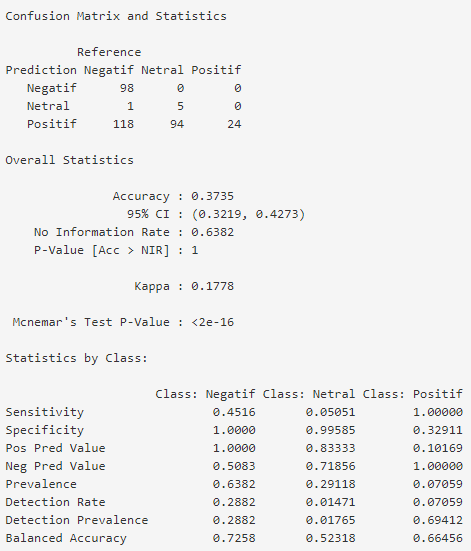
Pada tahap ini dillakukan klasifikasi data dengan tujuan untuk mendapatkan nilai pendapat positif dan negatif. Tahap ini menggunakan algoritma *support vector machine*.

**Tabel 3.2** Hasil Klasifikasi Data

|  |  |  |  |
| --- | --- | --- | --- |
| No | Klasifikasi | Score | Text |
| 1 | Negatif | -1 | masih percaya wadas saja blom beres |
| 2 | Negatif | -1 | warga wadas sudah menobatkan ganjar penjahat lingkungan |
| 3 | Netral | 0 | pie wadase tambang ilegalkatanya bekingannya kuate iki itu aja ngeluh gmn mau |
| 4 | Negatif | -2 | masalah wadas blm selesai rakyat sdh antiti sm ganjarngeri kalo dimn rambut ubanan tandanya s |
| 5 | Netral | 0 | selama ini kok gak digrebek yakena kok waktu akan milua ini skenario waktu makan diwarung |
| 6 | Netral | 0 | dkt dgn rakyat lalu bgmn dgn kasus warga wadas |
| 7 | Negatif | -1 | rusak |
| 8 | Positig | 2 | kartika kalau ganjar yg jd esiden berikutnya d disimlkan kondisi indonesia tdk akan terjadi rubahan kemajuan gak usah di |
| 9 | Negatif | -1 | si ganjar pranowo hanya dekat rakyat jateng kalau rakyat jawa barat dan luar jawa jauh rakyat wadas saja |

1. Melakukan Uji SVM dari Data

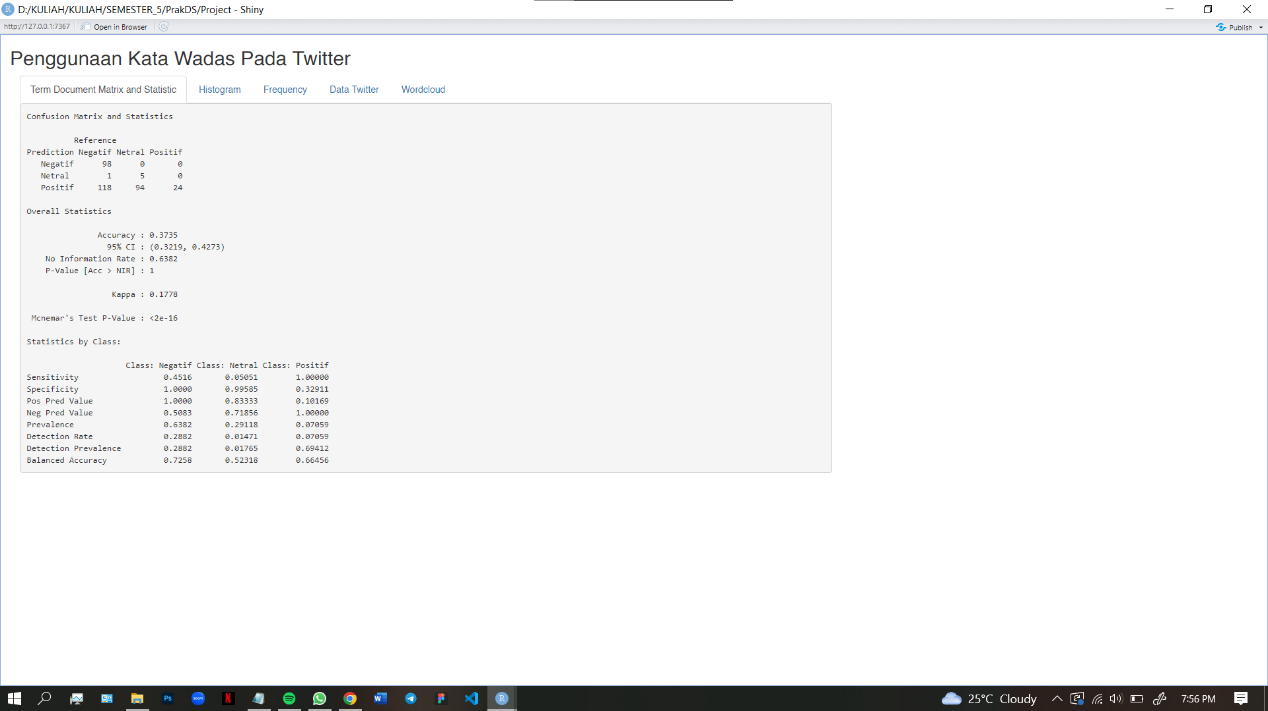
Dari pengujian SVM dapat dilihat yaitu hasil pengujian untuk *data* *training* yaitu pada nilai *p-value* (*Mcnemar’s Test P-Value*) sebesar <2e-16, *balanced accuracy* atau akurasi seimbangnya sebesar 0,7258 (72,6%). *Sensitivity* atau persentase untuk *true-positive* atau yang di prediksikan benar adalah 0,32911 (32,9%), dan *specificity* atau persentase *true negative*-nya sebesar 1,0000 (100%).



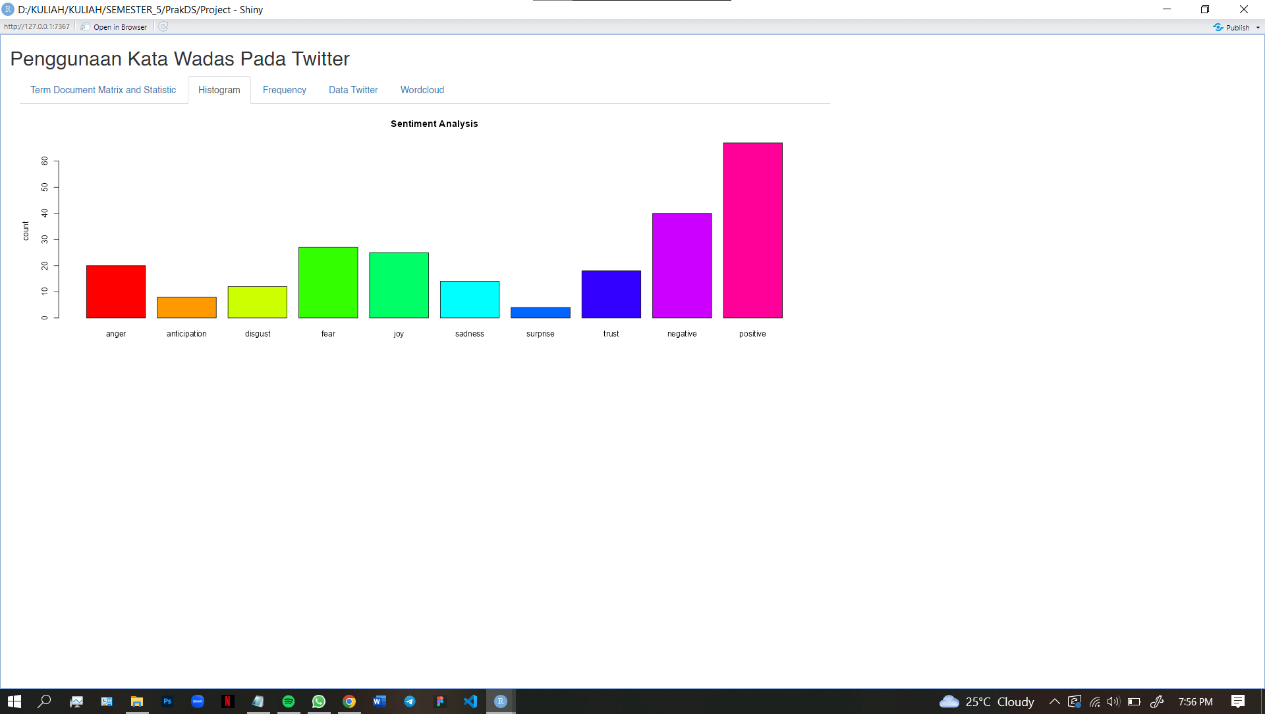
**Gambar 3.1** *Confusion Matrix*

1. Visualisasi dengan Shiny

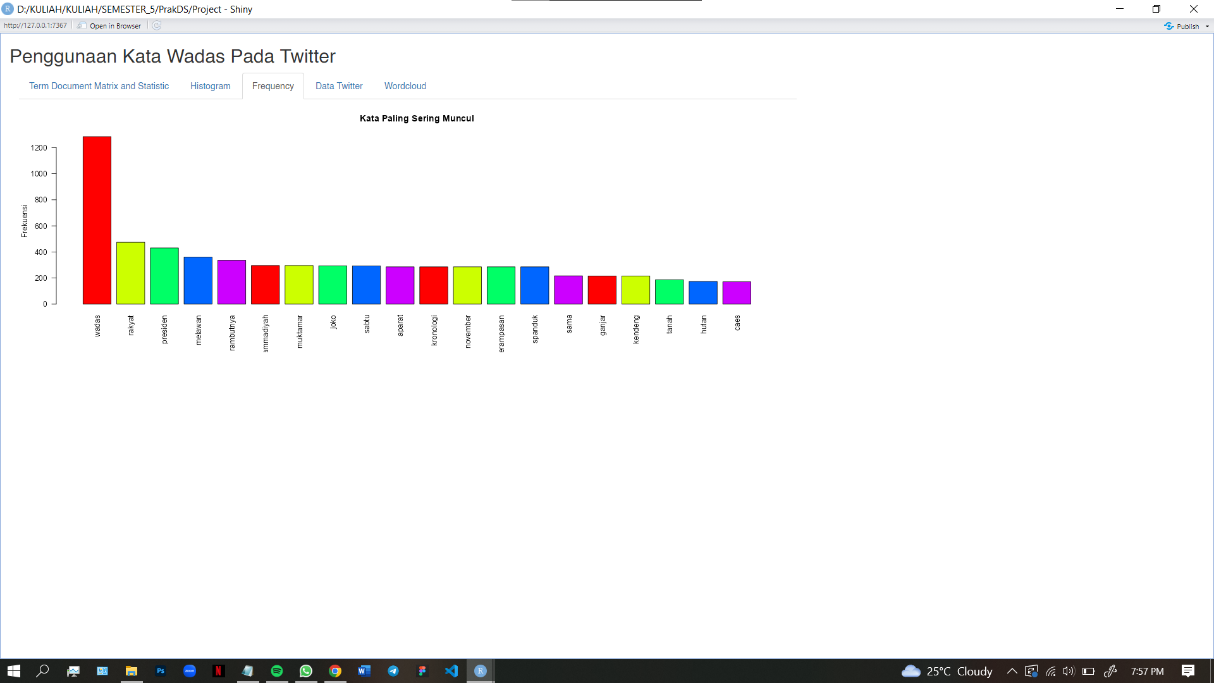
Pada tahap visualisasi menampilkan berbagai *output* dengan shinyApp.



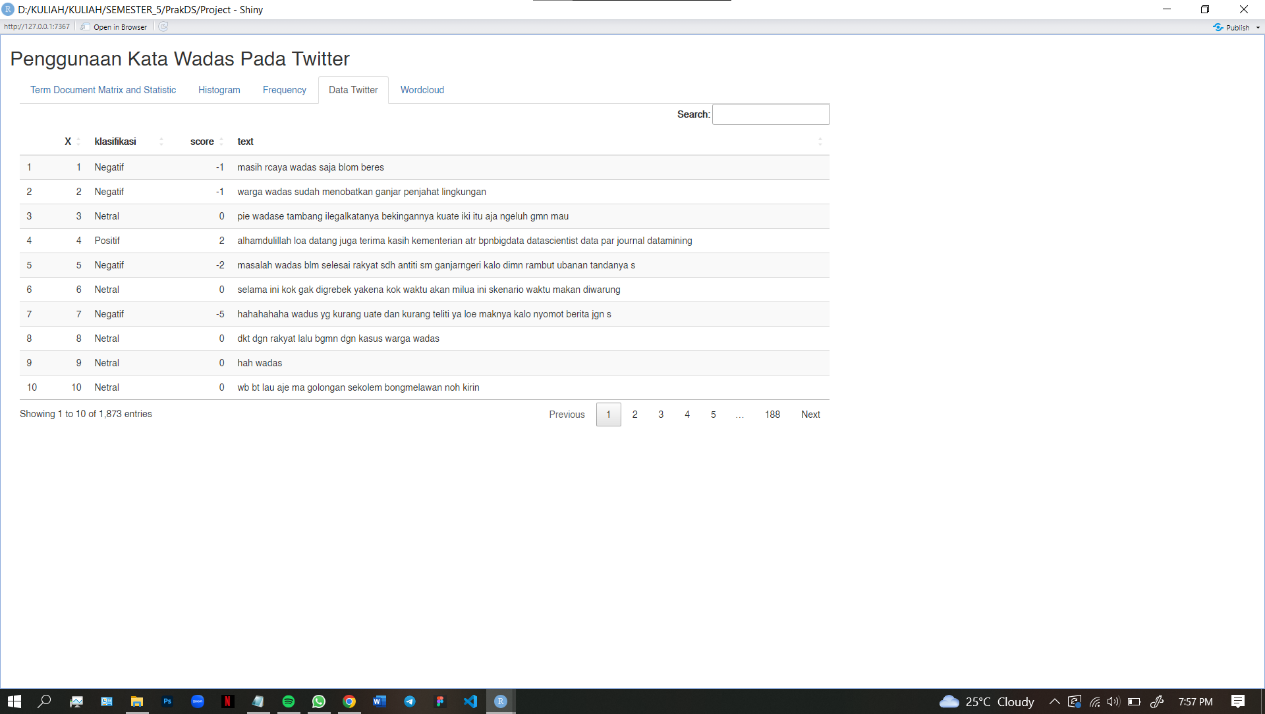
**Gambar 3.2** *Term Document Matrix and Statistic*



***Gambar 3.3*** *Barplot* Analisis Sentimen Pengguna Twitter Perihal Wadas



***Gambar 3.4*** *Frequency* Kata



***Gambar 3.5*** Klasifikasi Data



**Gambar 3.6** *Wordcloud*

1. *Listing* Program

|  |
| --- |
| ```{r}  library(tm) #data cleaning (corpus)  library(twitteR) #akses twitter APIs  library(rtweet) #collect and organize twitter data  library(shiny) #shiny  library(syuzhet) #baca fungsi get\_nrc  library(wordcloud) #wordcloud  library(vroom) #load dataset  library(here) #menyimpan dataset  library(dplyr) #manipulasi data frame  library(ggplot2) #visualisasi data (barplot, grafik)  library(RColorBrewer) #pengaturan warna  library(RTextTools) # naive bayes  ``` |

***Listing* 3.1** Deklarasi *library*

|  |
| --- |
| ```{r}  reqURL <- "http://api.twitter.com/oath/request\_token"  accessURL <- "https://api.twitter.com/oauth/access\_token"  CUSTOMER\_KEY <- "nSDC4fxZukXOSoC0xop56vONe"  ACCESS\_TOKEN <- "810446010-y9D6fi8x3aQeFzcLqUZ7KqwEFQOqdp0rk6EXZEuc"  CUSTOMER\_SECRET <- "NRbJuadU2BooR0IBKOlIrfER2q73baExtEBjv0ZJabxjFZMsVi"  ACCESS\_secret <- "TQCLLwlvkFWhx42KoIh9GJsKDJHBTaGWDQL0WSFy7Qn8j"  setup\_twitter\_oauth(CUSTOMER\_KEY, CUSTOMER\_SECRET, ACCESS\_TOKEN, ACCESS\_secret)  ``` |

***Listing* 3.2** Autentikasi Twitter API

|  |
| --- |
| ```{r}  tweets = searchTwitter('Wadas', n = 2000, retryOnRateLimit = 10e5, lang = "id") #retryOnRateLimit untuk looping  text <- do.call("rbind", lapply(tweets, as.data.frame))  write.csv(text, file = 'wadas.rds')  ``` |

***Listing* 3.3** *Scrapping* data

|  |
| --- |
| ```{r}  set <- read.csv('wadas.rds')  d = twListToDF(tweets)  corpus <- d$text  corpus <- Corpus(VectorSource(corpus))  #hapus URL  removeURL <- function(x) gsub("https\\S\*", "", x)  twtclean <- tm\_map(corpus, removeURL)  #hapus New Line  removeNL <- function(y) gsub("\n", "", y)  twtclean <- tm\_map(twtclean, removeNL)  #hapus koma  replacecomma <- function(y) gsub(",", "", y)  twtclean <- tm\_map(twtclean, replacecomma)  #hapus retweet  removeRT <- function(y) gsub("^RT:?", "", y)  twtclean <- tm\_map(twtclean, removeRT)  #hapus titik koma  removetitikkoma <- function(y) gsub(";", " ", y)  twtclean <- tm\_map(twtclean, removetitikkoma)  #hapus titik dua  removetitik2 <- function(y) gsub(":", "", y)  twtclean <- tm\_map(twtclean, removetitik2)  #hapus titik tiga  removetitik3 <- function(y) gsub("p.", "", y)  twtclean <- tm\_map(twtclean, removetitik3)  #hapus &amp  removeamp <- function(y) gsub("&amp;", "", y)  twtclean <- tm\_map(twtclean, removeamp)  #hapus mention  removeUN <- function(z) gsub("@[A-Za-z0-9]+", "", z)  twtclean <- tm\_map(twtclean, removeUN)  #hapus space dll  remove.all <- function(xy) gsub("[^[:alpha:][:space:]]\*", "", xy)  twtclean <- tm\_map(twtclean,stripWhitespace)  inspect(twtclean[1:10])  twtclean <- tm\_map(twtclean,remove.all)  twtclean <- tm\_map(twtclean, removePunctuation) #tanda baca  twtclean <- tm\_map(twtclean, tolower) #mengubah huruf kecil  textClean <- data.frame(text=unlist(sapply(twtclean, paste, collapse = " ")), stringsAsFactors=F)  View(textClean)  write.csv(textClean,'wadasClean.csv') |

***Listing* 3.4** *Data cleaning*

|  |
| --- |
| ```{r}  wadasClean <- read.csv("wadasClean.csv", header=T)  #scoring  kata.positif <- scan("positive-words.txt",what="character",comment.char=";")  kata.negatif <- scan("negative-words.txt",what="character",comment.char=";")  score.sentiment = function(sentence, positif, negatif, .progress='none') {  require(plyr)  require(stringr)  scores = laply(sentence, function(kalimat, positif, 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=sentence)    return(scores.df)  }  hasil = score.sentiment(wadasClean$text, kata.positif, kata.negatif)  #convert score to sentiment  hasil$klasifikasi <- ifelse(hasil$score<0, "Negatif", ifelse(hasil$score==0,"Netral","Positif"))  hasil$klasifikasi  View(hasil)  data <- hasil[c(3,1,2)] #ubah urutan kolom  View(data)  write.csv(data, file = "wadasLabel.csv")  ``` |

***Listing* 3.5** *Scoring*

|  |
| --- |
| ```{r}  library(e1071) #library yang terdapat sebuah algoritma naivebayes  library(caret) #library yang terdapat sebuah algoritma naivebayes  library(syuzhet) #library yang terdapat sebuah algoritma naivebayes  #digunakan untuk membaca file csv yang sudah di cleaning data  wadas\_dataset <- read.csv("wadasClean.csv", stringsAsFactors = FALSE)  #digunakan untuk mengeset variabel kolom text menjadi char  review <- as.character(wadas\_dataset$text)  #memanggil sentimen dictionary untuk menghitung presentasi dari beberapa emotion dan mengubahnya ke dalam text file  get\_nrc\_sentiment('happy')  get\_nrc\_sentiment('excitement')  s <- get\_nrc\_sentiment(review)  review\_combine <- cbind(wadas\_dataset$text,s)  par(mar=rep(3,4))  barplot(colSums(s), col=rainbow(10), ylab='count',main='Sentiment Analysis')  ``` |

***Listing* 3.6** *Lexicon* dan *barplot*

|  |
| --- |
| ```{r}  require(corpus)  data.frame <- read.csv("wadasLabel.csv", stringsAsFactors = F)  data.frame$klasifikasi <- factor(data.frame$klasifikasi)  glimpse(data.frame)  set.seed(20)  data.frame <- data.frame[sample(nrow(data.frame)),]  data.frame <- data.frame[sample(nrow(data.frame)),]  glimpse(data.frame)  corpus <- Corpus(VectorSource(data.frame$text))  corpus  inspect(corpus[1:10])  #membersihkan data yang tidak diperlukan  corpus.clean <- corpus %>%  tm\_map(content\_transformer(tolower)) %>% #mengubah string huruf besar menjadi string huruf kecil  tm\_map(removePunctuation) %>% #menghapus tanda baca  tm\_map(removeNumbers) %>% #menghapus nomor  tm\_map(removeWords, stopwords(kind="en")) %>% #menghapus stopwords  tm\_map(stripWhitespace)    dtm <- DocumentTermMatrix(corpus.clean)  inspect(dtm[1:10,1:20]) |

***Listing* 3.7** *Naive Bayes*

|  |
| --- |
| ```{r}  require(corpus)  data.frame <- read.csv("wadasLabel.csv", stringsAsFactors = F)  data.frame$klasifikasi <- factor(data.frame$klasifikasi)  glimpse(data.frame)  set.seed(20)  data.frame <- data.frame[sample(nrow(data.frame)),]  data.frame <- data.frame[sample(nrow(data.frame)),]  glimpse(data.frame)  corpus <- Corpus(VectorSource(data.frame$text))  corpus  inspect(corpus[1:10])  #membersihkan data yang tidak diperlukan  corpus.clean <- corpus %>%  tm\_map(content\_transformer(tolower)) %>% #mengubah string huruf besar menjadi string huruf kecil  tm\_map(removePunctuation) %>% #menghapus tanda baca  tm\_map(removeNumbers) %>% #menghapus nomor  tm\_map(removeWords, stopwords(kind="en")) %>% #menghapus stopwords  tm\_map(stripWhitespace)    dtm <- DocumentTermMatrix(corpus.clean)  inspect(dtm[1:10,1:20]) |

***Listing* 3.8** *Lanjutan* *Naive Bayes*

|  |
| --- |
| ```{r}  library(wordcloud2)  data1 <- read.csv('wadasLabel.csv')  text <- data1$text  docs <- Corpus(VectorSource(text))  docs <- tm\_map(docs, removeWords,"yang")  docs <- tm\_map(docs, removeWords,"oleh")  docs <- tm\_map(docs, removeWords,"tih")  dtm <- TermDocumentMatrix(docs)  matrix <- as.matrix(dtm)  words <- sort(rowSums(matrix),decreasing=TRUE)  df <- data.frame(word = names(words),freq=words)  tweets\_words <- data1 %>%  select(text) %>%  unnest\_tokens(word, text)  words <- tweets\_words %>% dplyr::count(word, sort=TRUE)  p <- wordcloud2(data=df, size=1.6, color='random-dark')  ``` |

***Listing* 3.9** *Wordcloud*

|  |
| --- |
| ```{r freq}  data1 = read.csv("wadasLabel.csv")  corpus = Corpus(VectorSource(data1$text))  corpus <- tm\_map(corpus, removeWords,"yang")  corpus <- tm\_map(corpus, removeWords,"oleh")  corpus <- tm\_map(corpus, removeWords,"tih")  dtm <- TermDocumentMatrix(corpus)  m <- as.matrix(dtm)  v <- sort(rowSums(m),decreasing=TRUE)  d <- data.frame(word = names(v),freq=v)  barplot(d[1:20,]$freq, las = 2, names.arg = d[1:20,]$word, col=rainbow(5),  main = "Kata Paling Sering Muncul", ylab = "Frekuensi")  ``` |

***Listing* 3.10** *Barlot* dan *Frequency*

|  |
| --- |
| ```{r}  #shiny  #membuka file csv  twitter <- read.csv(file="wadasClean.csv", header=TRUE)  #membuka text file pada data frame twitter  tweet <- twitter$text  #mengatur tampilan web  ui <- fluidPage(  titlePanel("Penggunaan Kata Wadas Pada Twitter"), #judul  mainPanel( #tab  #plot output : untuk scatterplot  tabsetPanel(type = "tabs",  tabPanel("Term Document Matrix and Statistic", verbatimTextOutput("result")),  tabPanel("Histogram", plotOutput("scatterplot")), #tab berupa histogram  tabPanel("Frequency", plotOutput("freqplot")), #tab berupa frequency  tabPanel("Data Twitter", DT::dataTableOutput('tbl')), #tab berupa data cleaning twitter & skoring  tabPanel("Wordcloud", wordcloud2Output("Wordcloud2")) #tab berupa worldcloud  )  )  ) |

***Listing* 3.11** *User Interface*

|  |
| --- |
| ```{r}  #tempat data akan dianalisis dan diproses, hasilnya ditampilkan/diplotkan pada bagian mainpanel() ui  server <- function(input, output) {  #output Data  output$result <-renderPrint({  conf.matNB  })  wadasLabel <- read.csv('wadasLabel.csv')  output$tbl = DT::renderDataTable({  DT::datatable(wadasLabel, options = list(lengthChange = FALSE)) #data ditampilkan dalam beberapa halaman  })    #barplot  output$scatterplot <- renderPlot({  barplot(colSums(s), col=rainbow(10), ylab='count',main='Sentiment Analysis')  }, height = 400)    #freq Plot  output$freqplot <- renderPlot({  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$Wordcloud2 <- renderWordcloud2({  p  })  }  ``` |

***Listing* 3.12** *Server*

|  |
| --- |
| ```{r}  shinyApp(ui = ui, server = server)  ``` |

***Listing* 3.13** *Run Shiny*

# **KESIMPULAN**

Dengan adanya analisis sentimen ini, pihak warga maupun pemerintah dapat mengetahui penilaian dari para pengguna Twitter mengenai wacana pembangunan di wilayah desa Wadas. Sebagian besar pengguna Twitter berpendapat positif kepada pembangunan tersebut. Dengan adanya program ini semoga dapat memberikan masukan untuk warga Wadas maupun pemerintah.