

**LAPORAN PROYEK AKHIR
PRAKTIKUM DATA SCIENCE**

**Analisis Sentimen dari Dampak Kebijakan
Pemberian Bantuan Sosial di Indonesia
Menggunakan Scrapping Twitter**



Helma Liana Putri 123200030

Danica Kirana 123200055

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

HALAMAN PENGESAHAN
LAPORAN PROYEK AKHIR

Disusun oleh :

<i>Helma Liana Putri</i>	<i>123200030</i>
<i>Danica Kirana</i>	<i>123200055</i>

Telah Diperiksa dan Disetujui oleh Asisten Praktikum Data Science

Pada Tanggal :

Asisten Praktikum

Asisten Praktikum

Ega Erinovian Megananda Putra

123190113

Luckman Nathan Syarif Aljustin

123190151

DAFTAR ISI

COVER	1
HALAMAN PENGESAHAN	2
DAFTAR ISI	3
1. PENDAHULUAN	4
1.1. Latar Belakang	4
1.2. Rumusan Masalah	4
1.3. Tujuan	4
2. METODE	5
2.1. Analisis Sentimen	5
2.2. Naïve Bayes	5
2.3. Data Collection	6
2.4. Data Preprocessing	6
2.4.1. Case Folding.....	6
2.4.2. Data Cleaning	6
2.4.3. Filtering	6
2.5. Data Classification.....	7
2.5.1. Labeling.....	7
2.5.2. Transformasi.....	7
2.5.3. Modeling	8
3. HASIL DAN PEMBAHASAN	8
4. KESIMPULAN.....	23

1. PENDAHULUAN

1.1 Latar belakang

Kebijakan pemerintah mengenai pemberian bantuan sosial telah lama dilakukan di Indonesia. Bantuan sosial atau yang biasa disebut bansos biasanya diberikan pemerintah kepada masyarakat kurang mampu ataupun kepada masyarakat korban bencana alam yang membutuhkan bantuan. Namun, pemberian bantuan sosial kadang menuai pro dan kontra di kalangan masyarakat. Maka dari itu, dalam project ini kami mencoba untuk melakukan analisis sentimen dari dampak kebijakan Pemerintah Indonesia dalam pemberian bantuan sosial. Kami melakukan analisis sentimen menggunakan scrapping twitter.

Twitter merupakan suatu layanan jejaring sosial yang memungkinkan penggunaanya untuk mengirim dan membaca pesan berbasis teks. Twitter merupakan salah satu media sosial yang banyak digunakan masyarakat untuk menyalurkan opini. Fitur pada twitter yang memungkinkan pengguna mengirimkan unggahan berupa teks menjadikan twitter digemari penggunaanya karena memudahkan untuk melakukan diskusi ataupun sekedar mengirimkan suatu komentar atau opini. Dengan banyaknya opini masyarakat yang dapat ditemukan di twitter, kami memutuskan menggunakan aplikasi twitter sebagai sumber data utama dalam project kami.

1.2 Rumusan Masalah

Berdasarkan latar belakang di atas, dapat kami rumuskan masalah yang akan kami bahas yaitu bagaimana respon masyarakat terhadap kebijakan pemerintah mengenai pemberian bakti sosial di Indonesia.

1.3 Tujuan

Tujuan dari project ini yaitu untuk mendapatkan analisa sentimen dari opini masyarakat mengenai dampak kebijakan pemerintah dalam pemberian bantuan sosial di Indonesia.

2. METODE

Dalam penelitian proyek ini kami menggunakan teknik *data mining* Naïve Bayes. Penelitian yang dilakukan pada kali ini menghasilkan analisis sentiment dengan menerapkan Algoritma Naïve Bayes untuk menentukan kategori dari komentar positif dan negatif berdasarkan data yang telah didapatkan.

2.1 Analisis Sentiment

Analisis sentiment atau yang biasa dikenal sebagai *opinion mining* atau *emotion artificial intelligence* merupakan teknik pengolahan bahasa alami, komputasi linguistik, dan text mining yang berguna untuk mendeteksi suatu subyek seperti pendapat, sentimen, evaluasi, sikap, penilaian dan emosi seseorang apakah pembicara atau penulis berkenaan dengan suatu topik, produk, layanan, organisasi, individu, ataupun kegiatan tertentu.

2.2 Naïve Bayes

Naive Bayes adalah metode yang cocok untuk klasifikasi biner dan multiclass. Metode yang juga dikenal sebagai Naive Bayes Classifier ini menerapkan teknik supervised klasifikasi objek di masa depan dengan menetapkan label kelas ke *instance* atau catatan menggunakan probabilitas bersyarat. Probabilitas bersyarat adalah ukuran peluang suatu peristiwa yang terjadi berdasarkan peristiwa lain yang telah (dengan asumsi, praduga, pernyataan, atau terbukti) terjadi.

Nah, model *machine learning* yang diterapkan pada program tersebut menggunakan teorema Bayes yang dirumuskan sebagai berikut:

$$P(A \mid B) = P(B \mid A)P(A)P(B)$$

Keterangan:

$P(A \mid B)$: Probabilitas A terjadi dengan bukti bahwa B telah terjadi (probabilitas superior).

$P(B \mid A)$: Probabilitas B terjadi dengan bukti bahwa A telah terjadi

$P(A)$: Peluang terjadinya A

$P(B)$: Peluang terjadinya B

2.3 Data Collection

Data Collection merupakan proses mengumpulkan, mengukur, dan menganalisis data yang akurat dari berbagai sumber yang relevan untuk menemukan jawaban atas masalah penelitian, menjawab pertanyaan, mengevaluasi hasil, dan memperkirakan tren dan probabilitas. Data Collection yang dilakukan pada project akhir ini dilakukan melalui proses scrapping yang diambil sosial media Twitter. Tahap pertama yang dilakukan pada proses data collection adalah mengambil tweet menggunakan tweet API yang telah disediakan oleh Twitter Developer. Untuk mendapatkan tweet tersebut, diperlukannya user membuat App pada Twitter Developer untuk mendapatkan *key*, *secret*, *access token*, dan *access secret* sehingga user dapat membuat token API. Tahap selanjutnya adalah pengambilan tweet berdasarkan kata kunci yang dicari. Karena dalam projek ini yang akan dibahas adalah bantuan sosial atau bansos. Maka, kata kunci yang digunakan adalah “bansos” sebagai kata kunci, dan menambahkan lang = “id” untuk mendapatkan tweet berbahasa Indonesia.

2.4 Data Preprocessing

Data Preprocessing merupakan proses untuk mendapatkan data text atau ulasan yang bersih yang melakukan beberapa tahapan didalamnya, yaitu :

2.4.1 Case Folding merupakan proses yang diterapkan pada urutan karakter teks pada suatu dokumen dengan merubah bentuk standar huruf besar-kecil dengan mengurangi semua huruf menjadi huruf kecil atau *lowercase*.

2.4.2 Data cleaning merupakan proses menyiapkan data untuk menganalisis data lebih lanjut yang berguna untuk menghasilkan data dengan kualitas yang tinggi. Data cleaning dilakukan dengan cara menghapus atau memodifikasi data yang salah, tidak relevan, duplikat, dan tidak berformat dengan melakukan proses penghilangan komponen tertentu yang terdapat dalam data tweet yang telah didapatkan yakni *Uniform Resource Locator* (URL), *username*, RT (*Retweet*), karakter HTML, *emoticon*, hashtag, serta tanda baca dan kata-kata yang tidak berguna.

2.4.3 Filtering merupakan tahap mengambil kata-kata penting dari hasil token dengan menggunakan algoritma stoplist (*stopword*) untuk membuang

kata kurang penting atau wordlist (menyimpan kata penting) dengan dilakukannya penghapusan kata (*removewords*).

2.5 Data Classification

Data *Classification* merupakan salah satu contoh dari metode analisis data. *Classification* adalah teknik di mana kita bisa mengkategorikan data ke dalam sejumlah kelas yang telah ditentukan sebelumnya. Tujuan utama dari klasifikasi adalah untuk membantu praktisi data dalam menentukan kelas atau kategori dari data baru berdasarkan karakteristik data yang telah ada sebelumnya. Klasifikasi tidak hanya bisa dilakukan di data yang terstruktur, namun juga bisa digunakan untuk data yang tidak terstruktur.

Classification juga merupakan salah satu algoritma Machine Learning yang bersifat Supervised Learning, dimana data akan dikelompokkan berdasarkan label yang ada dengan melalui beberapa proses, diantaranya adalah :

2.5.1 Labeling merupakan pemberian label pada data. Proses labeling dilakukan menggunakan metode Lexical Based yang merupakan proses labeling berdasarkan kata positif, kata negatif, maupun kata netral yang terdapat pada data tweets yang telah dilakukan cleaning. Klasifikasi labeling ini dilakukan dengan pencocokkan dengan kata-kata yang terdapat dalam kamus Lexicon. Dimana, jika data tweets memiliki kata positif, maka akan digolongkan pada sentimen tweet positif. Sedangkan, jika terdapat data pada tweets yang memiliki kata negatif, maka data tersebut akan digolongkan pada sentimen tweet negatif. Namun pada kasus lain, jika kedua kata ini bernilai sama, maka digolongkan dalam tweet netral.

2.5.2 Transformasi merupakan proses pelabelan pada teks tweets yang akan diubah menjadi bentuk yang dapat digunakan oleh komputer. Pengubahan bentuk tersebut dikenal dengan vektorisasi. Vektorisasi merupakan proses membuat vektor dengan nilainya berupa angka-angka kemunculan term (kata unik) dalam dokumen. Teks ini akan diubah perkata. Pemisahan teks tweets menjadi kata-kata disebut dengan tokenisasi. Tokenisasi merupakan proses untuk membagi teks yang dapat berupa kalimat, paragraf atau dokumen, menjadi token-

token/bagian-bagian tertentu. Dalam hal ini komponen individual tersebut yaitu token adalah kata dari tiap teks tweets. *Project* ini token-token dibuat dalam bentuk dokumen term matrix.

2.5.3 Modeling merupakan tahap pengklasifikasian untuk menentukan apakah data yang akan diuji termasuk kedalam sentimen positif atau negatif. Pada tahap ini menggunakan metode Naïve Bayes Classifier dengan pelabelan terlebih dahulu. Metode ini terdiri dari 2 proses, yaitu setelah data latih dan data uji dibuat maka dilanjutkan ke proses latih. Pada proses latih diimplementasikan algoritma pengklasifikasi Naïve Bayes. Algoritma pengklasifikasi Naïve Bayes akan menggunakan kemunculan tiap kata unik dalam teks tweets untuk menghitung pembuatan set data latih dan set data uji, melakukan data latih yaitu dengan membangun model pengklasifikasi menggunakan algoritma Naïve Bayes, dan diakhiri dengan klasifikasi yang mengimplementasikan model pengklasifikasi pada set data uji teks tweets.

3. HASIL DAN PEMBAHASAN

a. Import library yang dibutuhkan

```
**Deklarasi library**
```{r rlib}
library(tm) #data cleaning (corpus)
library(tidytext)
library(twitterR) #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) #buat naive bayes
library(devtools)
```
```


b. Akses Twitter API

Penggunaan API key untuk autentifikasi dengan twitter sebelum melakukan scrapping data.

```
**Access twitter API**
```{r}
Key auth Twitter API
consumer.api_key <- "bYPNq9oD4EQAS635Of1QwPoxj"
consumer.api_secret_key <-
"JKFnRsof4m02H4fHoaU1I9I1qo8ssF9LQeNNgDDhkVJpcgzix8"
access.token <- "1168785141507842048-
cEacBdvnfW6qwpvTx29PKH0AygxmZf"
access.token_secret <-
"q8hBnzgwL1rc24l7z1idDHdAyzb2tep8K3kGb5YuBRIX1"

Start authentication with OAuth
setup_twitter_oauth(consumer.api_key,
consumer.api_secret_key, access.token, access.token_secret)
```
```

c. Mengambil data dari twitter

Disini kami mengambil data menggunakan keyword “bansos”. Data yang kami ambil dibatasi hanya 1000 data, dan data yang kami gunakan yaitu data yang menggunakan Bahasa Indonesia.

```
**Twitter Scrapping**
```{r}
tweets = searchTwitter('Bansos',
 n = 1000,
 lang = "id",
 retryOnRateLimit = 10e5)
text <- do.call("rbind", lapply(tweets, as.data.frame))
write.csv(text, file = 'dataMentah.csv')
```
```

d. Data Cleaning

Pada data cleansing ini kami membersihkan data dengan membuang karakter-karakter atau simbol yang tidak diperlukan.

```

**Cleaning Data**
```{r}
d <- read.csv("dataMentah.csv")
kata <- d$text
reviewC <- Corpus(VectorSource(kata))
#remove URL
removeURL <- function(x) gsub("http[^[:space:]]*", "", x)
reviewclean <- tm_map(reviewC, removeURL)
#remove New Line
removeNL <- function(y) gsub("\n", " ", y)
reviewclean <- tm_map(reviewclean, removeNL)
#remove koma
replacecomma <- function(y) gsub(",", "", y)
reviewclean <- tm_map(reviewclean, replacecomma)
#remove retweet
removeRT <- function(y) gsub("RT ", "", y)
reviewclean <- tm_map(reviewclean, removeRT)
#remove titik dua
removetitik2 <- function(y) gsub(":", "", y)
reviewclean <- tm_map(reviewclean, removetitik2)
#remove titik koma
removetitikkoma <- function(y) gsub(";", " ", y)
reviewclean <- tm_map(reviewclean, removetitikkoma)
#remove titik3
removetitik3 <- function(y) gsub("p..", "", y)
reviewclean <- tm_map(reviewclean, removetitik3)
#remove &
removeamp <- function(y) gsub("&", "", y)
reviewclean <- tm_map(reviewclean, removeamp)
#remove Mention
removeUN <- function(z) gsub("@\\w+", "", z)
reviewclean <- tm_map(reviewclean, removeUN)

removesym <- function(y) gsub("ð", "", y)
reviewclean <- tm_map(reviewclean, removesym)

#remove Emoji
removeEmoji <- function(z) gsub("[^\\x01-\\x7F]", "", z)
reviewclean <- tm_map(reviewclean, removeEmoji)
#remove Number
removeNum <- function(z) gsub("[0-9]+", "", z)

```

```

reviewclean <- tm_map(reviewclean, removeNum)
#remove space dll
remove.all <- function(xy) gsub("[^[:alpha:][:space:]]*", "",
xy)

reviewclean <- tm_map(reviewclean,remove.all)
reviewclean <- tm_map(reviewclean, removePunctuation) #tanda
baca
reviewclean <- tm_map(reviewclean, tolower) #mengubah huruf
kecil

#hapus data yang kosong
try.error = function(x)
{
 # create missing value
 y = NA
 # tryCatch error
 try_error = tryCatch(tolower(x), error=function(e) e)
 # if not an error
 if (!inherits(try_error, "error"))
 y = tolower(x)
 # result
 return(y)
}

lower case using try and error with sapply
reviewclean = sapply(reviewclean, try.error)

remove NAs in some_txt
reviewclean = reviewclean[!is.na(reviewclean)]
names(reviewclean) = NULL

dataframe<-data.frame(text=unlist(sapply(reviewclean, `[`)),
stringsAsFactors=F)
write.csv(dataframe,file = "data_clean.csv")
View(dataframe)
` ``

```

e. Skoring data dan konversi ke dalam sentiment

Berdasarkan tiga komponen yaitu, positif, negative, dan netral. jika terdapat data pada tweets yang memiliki kata negative atau hasil\$score <0, maka data tersebut akan digolongkan pada sentimen tweet negatif. Namun pada kasus lain, jika kedua kata ini bernilai sama, maka digolongkan dalam tweet netral.

```
Scoring
```{r}
try_clean <- read.csv('data_clean.csv')
View(try_clean)

#skoring
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)
  #preprocessing
  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(try_clean$text, kata.positif,
kata.negatif)

#konversi score ke sentiment
hasil$klasifikasi<- ifelse(hasil$score<0,
"Negatif",ifelse(hasil$score==0,"Netral","Positif"))
hasil$klasifikasi
View(hasil)

#menukar urtan baris
data <- hasil[c(3,1,2)] #ubah urutan kolom
View(data)
write.csv(data, file = "dataLabel.csv")
```

```

#### f. Barplot Lexicon

Pembuatan barplot dari hasil analisa sentimen yang menggunakan sentimen directory happy dan excitement.

```

Barplot Lexicon
```{r}
library(e1071) #library yang terdapat sebuah algoritma
naivebayes
library(caret) #library yang terdapat sebuah algoritma
naivebayes

d<-read.csv("data_clean.csv",stringsAsFactors = FALSE)
#membaca file csv yang sudah di cleaning data

review <-as.character(d$text) #set variabel cloumn text
menjadi char
get_nrc_sentiment('happy')
get_nrc_sentiment('excitement')
#deklarasi var s utnuk memanggil sentimen dictionary untuk
menghitung presentasi dari beberapa emotion dan mengubahnya
ke dalam text file
s <- get_nrc_sentiment(review, cl = NULL, language =
"english", lowercase = TRUE)

```

```

review_combine<-cbind(d$text,s) #klasifikasi data
par(mar=rep(3,4))
a<- barplot(colSums(s),col=rainbow(10), xlab ='emotion',
ylab='count',main='Sentiment Analysis')
barplt <- a
...

```

g. Naïve Bayes

Pada konsep naive bayes ini kita akan mengambil dari probabilitas byes untuk membuat satu model ketidakpastian dari suatu kejadian dengan menggabungkan pengetahuan umum dengan fakta hasil pengamatan dari data yang diambil.

```

*Naive Bayes*
```{r}
require(corpus)
data.frame <- read.csv("dataLabel.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])

#fungsinya untuk membersihkan data data yang tidak dibutuhkan

corpus.clean<-corpus %>%
 tm_map(content_transformer(tolower)) %>% #digunakan untuk
mengubah huruf besar dari string 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])

df.train<-data.frame[1:800,,]

```

```

df.test<-data.frame[801:1000,]

dtm.train<-dtm[1:800,]
dtm.test<-dtm[801:1000,]

corpus.clean.train<-corpus.clean[1:800]
corpus.clean.test<-corpus.clean[801:1000]

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)

#Boolean Naive Bayes
convert_count <- function(x) {
 y<-ifelse(x>0,1,0)
 y<-factor(y,levels=c(0,1),labels=c("no","yes"))
 y
}

#Naive Bayes Model
trainNB<-apply(dtm.train.nb,2,convert_count)
testNB<-apply(dtm.test.nb,2,convert_count)
#Training
classifier <- naiveBayes(trainNB, df.train$klasifikasi,
laplace = 1)

#Use the NB classifier we built to make predictions on the
test set
pred <- predict(classifier, testNB)

```

```

#Create a truth table by tabulating the predicted class
labels with the actual predicted class labels with the actual
class labels
NB_table=table("Prediction"= pred, "Actual" =
df.test$klasifikasi)
NB_table

#confussion Matrix
conf.matNB <- confusionMatrix(pred, df.test$klasifikasi)
conf.matNB

```

```

h. Wordcloud

Pembuatan wordcloud untuk mengetahui kata kata yang sering muncul.

```

**Wordclouds**
```{r}
library(dplyr)
library(wordcloud2)

data1 <- read.csv('dataLabel.csv')
text <- data1$text
docs <- Corpus(VectorSource(text))

docs <- tm_map(docs, removeWords,"nya")
docs <- tm_map(docs, removeWords,"untuk")
docs <- tm_map(docs, removeWords,"gak")
docs <- tm_map(docs, removeWords,"õý")
docs <- tm_map(docs, removeWords,"amp")
docs <- tm_map(docs, removeWords,"dan")
docs <- tm_map(docs, removeWords,"gerai")
docs <- tm_map(docs, removeWords,"yang")
docs <- tm_map(docs, removeWords,"dari")

dtm <- TermDocumentMatrix(docs)
matrix <- as.matrix(dtm)
words <- sort(rowSums(matrix),decreasing=TRUE)
df <- data.frame(word = names(words),freq=words)
df

tweets_words <- data1 %>%
 select(text) %>%

```



```

unnest_tokens(word, text)

words <- tweets_words %>% dplyr::count(word, sort=TRUE)

p <- wordcloud2(data=df, size=1.5, color='random-dark')
p
```

```

i. Histogram

Pembuatan histogram untuk mengetahui frekuensi data dari masing masing sentimen.

```

**Histogram Freq**
```{r Freq}
data1 = read.csv("dataLabel.csv")
corpus = Corpus(VectorSource(data1$text))
corpus <- tm_map(corpus, removeWords,"nya")
corpus <- tm_map(corpus, removeWords,"untuk")
corpus <- tm_map(corpus, removeWords,"gak")
corpus <- tm_map(corpus, removeWords,"ðŸ")
corpus <- tm_map(corpus, removeWords,"amp")
corpus <- tm_map(corpus, removeWords,"dan")
corpus <- tm_map(corpus, removeWords,"gerai")
corpus <- tm_map(corpus, removeWords,"yang")
corpus <- tm_map(corpus, removeWords,"dari")
dtm <- TermDocumentMatrix(corpus)
m <- as.matrix(dtm)
v <- sort(rowSums(m),decreasing=TRUE)
d <- data.frame(word = names(v),freq=v)
barplot(d[1:20,]$freq, las = 2, names.arg = d[1:20,]$word,
col=rainbow(5),
 main = "Kata Paling Sering Muncul", ylab = "Frekuensi")
```

```

j. User Interface

Pembuatan user interface agar memudahkan pengguna dalam melihat analisis yang telah dibuat.

```
**User Interface**
```${r}
#shiny
#membuka file csv
dataLabel<- read.csv("datalabel.csv")
dataKotor <- read.csv("dataMentah.csv")

#mengatur tampilan web
ui <- fluidPage(
 titlePanel("Analisis sentimen dari dampak kebijakan
pemberian bantuan sosial di Indonesia menggunakan scripping
twitter dengan metode Naive Bayes"), #judul
 # Show a plot of the generated distribution
 mainPanel(#tab
 #plot output : untuk scatterplot
 tabsetPanel(type = "tabs",
 tabPanel("Term Document Matrix and
Statistic", verbatimTextOutput("result")),
 #tab data kotor dan hasil sentiment
 tabPanel("Data Kotor",
DT::dataTableOutput('tbl1')),
 tabPanel("Data sentiment",
DT::dataTableOutput('tbl2')),
 #tab scatterplot/grafik
 tabPanel("Histogram",
plotOutput("scatterplot")),
 tabPanel("Frequency",
plotOutput("freqplot")),
 # tab wordcloud
 tabPanel("Wordcloud",
wordcloud2Output("Wordcloud2")),
)
)
)
````
```

k. Server

Adalah proses untuk menampilkan desain UI dari proyek, dimana nantinya akan menampilkan Term document matrix dan statistic, list kotor, list sentiment, histogram, frequency, dan wordcloud.

```
**Server**
```{r global}
#tempat data akan dianalisis dan diproses, hasilnya
ditampilkan/diplotkan pada bagian mainpanel() ui
server <- function(input, output) {
 #output Data
 output$result <-renderPrint({
 conf.matNB
 })
 #data ditampilkan dalam beberapa halaman
 output$tbl1 = DT::renderDataTable({
 DT::datatable(dataKotor, options = list(lengthChange
= FALSE))
 })

 output$tbl2 = DT::renderDataTable({
 DT::datatable(dataLabel, options = list(lengthChange
= FALSE))
 })

 #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)

 #wordcloud2
 output$Wordcloud2 <- renderWordcloud2({
```

```

p
})
}
```

```

l. Run Shiny

Proses untuk memunculkan hasil analisa menggunakan shiny.

```

**Run Shiny**
```{r}
shinyApp(ui = ui, server = server)
```

```

m. Confusion matrix dan Statistik

Analisis sentimen dari dampak kebijakan pemberian bantuan sosial di Indonesia menggunakan scripping twitter dengan metode Naive Bayes

| Term Document Matrix and Statistics | Data Kotor | Data sentiment | Histogram | Frequency | Wordcloud |
|---|------------|----------------|-----------|-----------|-----------|
| Confusion Matrix and Statistics | | | | | |
| Reference | | | | | |
| Prediction | Negatif | Netral | Positif | | |
| Negatif | 57 | 1 | 0 | | |
| Netral | 37 | 83 | 18 | | |
| Positif | 2 | 3 | 21 | | |
| Overall Statistics | | | | | |
| Accuracy : 0.705 | | | | | |
| ROC AUC : (0.4366, 0.7672) | | | | | |
| No Information Rate : 0.48 | | | | | |
| P-Value [Acc > NIR] : 9.395e-11 | | | | | |
| Kappa : 0.5888 | | | | | |
| McNemar's Test P-Value : 9.253e-10 | | | | | |
| Statistics by Class: | | | | | |
| Class: Negatif Class: Netral Class: Positif | | | | | |
| Sensitivity | 0.5050 | 0.9402 | 0.5676 | | |
| Specificity | 0.9984 | 0.6815 | 0.9693 | | |
| Pos Pred Value | 0.9828 | 0.5431 | 0.8077 | | |
| Neg Pred Value | 0.7354 | 0.9523 | 0.2066 | | |
| Prevalence | 0.4880 | 0.3359 | 0.1850 | | |
| Detection Rate | 0.2810 | 0.1159 | 0.1050 | | |
| Detection Prevalence | 0.2388 | 0.1889 | 0.1385 | | |
| Balanced Accuracy | 0.7921 | 0.7709 | 0.7684 | | |

Berisi akurasi mengenai proyek, yaitu sebesar 0,705.

n. Tabel List Kotor

Analisis sentimen dari dampak kebijakan pemberian bantuan sosial di Indonesia menggunakan scripping twitter dengan metode Naive Bayes

| Term Document Matrix and Statistics | | | | | | | | | | | | Data Kotor | Data sentiment | Histogram | Frequency | Wordcloud |
|-------------------------------------|--|-----------|---------------|--------------|---------------------|-----------|---------------------|---------------------|--------------------|--|-----------------|------------|----------------|-----------|-----------|-----------|
| | | | | | | | | | | | | Search: | | | | |
| X | text | favorited | favoriteCount | replyToSN | created | truncated | replyToSID | id | replyToUID | statusSource | screenName | | | | | |
| 1 | Perempuan Lelaki Meninggal Dunia Saat Antre Bantuan di Cigugur Bogor
https://t.co/5S49ZXM8od | false | 0 | | 2022-12-01 18:19:02 | false | 1598381173545310000 | | | https://t.co/5S49ZXM8od | kangandibdy | | | | | |
| 2 | @destralme
@onix_indonesia
@destralme dapat bantuan dari dt | false | 0 | Anul_Santana | 2022-12-01 18:18:55 | false | 1598247697168810000 | 1598381143594000000 | 810467652740239000 | Twitter for Android | Anul_Santana | | | | | |
| 3 | RT @gabulunt
@somederas Lagi keadaan kaya gini bukannya sibuk cari logistik buat survive malah bikin tenda buat ngemil... PKORANANVA PAD... | false | 0 | | 2022-12-01 18:04:13 | false | 159837744892510000 | | | Twitter for Android | bocor_aluah | | | | | |
| 4 | RT @gabulunt
@somederas Lagi keadaan kaya gini bukannya sibuk cari logistik buat survive malah bikin tenda buat ngemil... PKORANANVA PAD... | false | 0 | | 2022-12-01 17:32:40 | false | 159836950250840000 | | | Twitter for Android | yogap88 | | | | | |
| 5 | Pemerintah Kota (Pemkot) Bandung menyatakan bantuan sosial dalam rangka pengendalian inflasi tahun 2023 sudah di... | false | 0 | | 2022-12-01 17:27:01 | true | 159836808089550000 | | | Twitter Web App | Warta_Pajajaran | | | | | |

Berisi tabel data mentah sebelum melakukan cleaning data.

o. Tabel List Sentimen



Berisi ulasan positif, dimana dari 1000 data yang dicari, terdapat 240 opini yang bersifat positif.

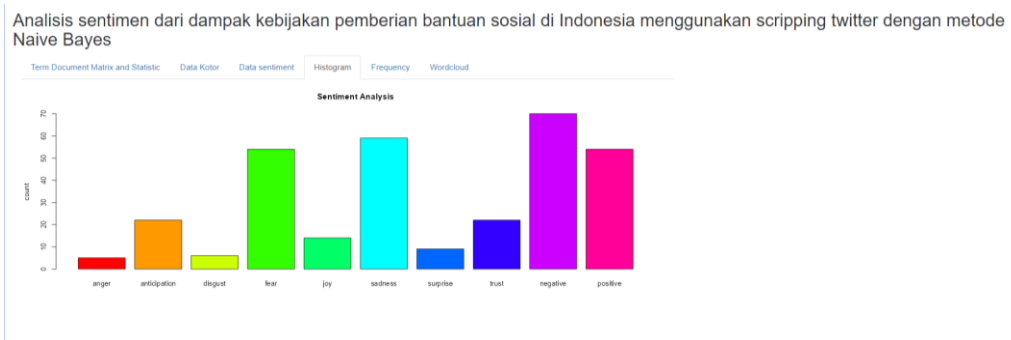


Berisi ulasan netral, dimana dari 1000 data yang dicari, terdapat 305 opini yang bersifat netral.



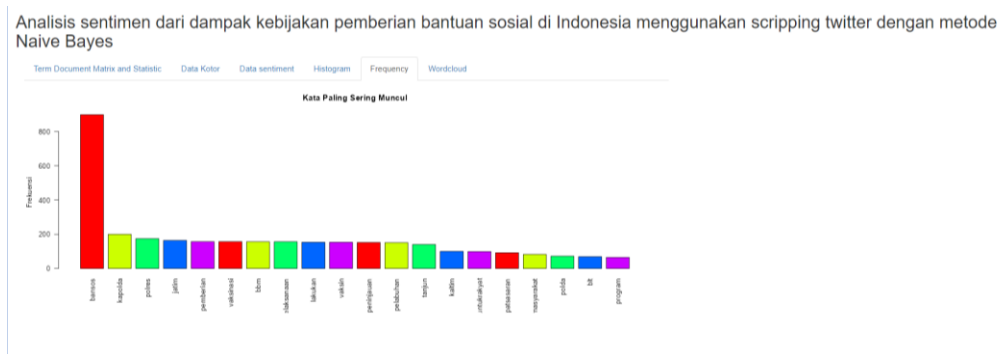
Berisi ulasan negatif, dimana dari 1000 data yang dicari, terdapat 456 opini yang bersifat negatif.

p. Histogram



Disini, ada histogram atau grafik emosi atau gambaran dari opini-opini para pengguna twitter mengenai bansos.

q. Frekuensi kata yang sering muncul



Disini ada kata yang paling sering muncul, dimana pada saat kita melakukan pencarian pada data twitter, maka kata-kata ini yang sering digunakan.

r. Wordcloud



Disini, dimunculkan dan ditampilkan pada wordcloud beberapa kata yang ditemukan pada ulasan twitter.

4. KESIMPULAN

Kesimpulan yang kami dapat pada project ini yaitu respon dan opini-opini dari para pengguna twitter mengenai bansos di Indonesia, dimana setelah melakukan analisa kami mendapatkan hasil, jika respon negatif lebih dominan dibandingkan dengan netral dan positif. Dapat dilihat dengan tingginya grafik dari analisa sentimen negative, sadness, dan fear. Dan juga pada list sentimen dapat kita lihat bahwa berdasarkan klasifikasinya, data dengan sentimen negatif terdapat 456 data, lalu netral dengan 305 data, sedangkan data dengan sentimen positif memiliki 240 data. Dari hasil wordcloud dan frekuensi kata yang sering muncul juga dapat dilihat bahwa kata yang pertama yang sering muncul yaitu kata bansos yang merupakan kata kunci kami dalam mengambil data. Kemudian diikuti dengan BBM, Kapolda, Kapolres dan lain lain. Data tersebut sudah kami bersihkan dengan cara menghapus kata dengan frekuensi kemunculan tinggi yang tidak digunakan dalam analisis kami.

Kekurangan dari project kami ini yaitu masih banyak data tweet dengan isi yang sama dalam proses pengolahan data yang kami lakukan, sumber data yang kami gunakan juga hanya dari satu sumber. Hasil yang didapatkan mungkin kurang optimal, namun kami berharap project kami ini dapat memberikan gambaran tentang sentimen masyarakat mengenai bantuan sosial yang diberikan oleh pemerintah dengan data yang didapat dari pengguna twitter.