

**LAPORAN PROYEK AKHIR
PRAKTIKUM DATA SCIENCE**

**Analisis Sentimen Tweet Film Sri Asih Menggunakan
Metode Naive Bayes**



NUR ROSYDATUN NAFIAH 123200001

NINDYA PUTRI MAHARANI 123200019

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

HALAMAN PENGESAHAN PEMBIMBING

**Analisis Sentimen Tweet Film Sri Asih Menggunakan
Metode Naive Bayes**

Disusun Oleh:

Nur Rosydatun Nafiah
Nindya Putri Maharani

123200001
123200019

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

Menyetujui,

Asiten Praktikum 1

Asisten Praktikum 2

Dio Cahyo Saputra, S.Kom.

Vincentius Willy Ardiyanto
123190100

1. PENDAHULUAN

Pertumbuhan industri film Indonesia semakin meningkat dengan semakin banyaknya produksi film dalam negeri dan jumlah penontonnya. Semakin tingginya jumlah penonton yang menyaksikan film lokal tentu merupakan hal yang sangat menggembirakan bagi industri film nasional. Pasalnya dengan semakin banyaknya jumlah penonton, diharapkan makin banyak investor yang melirik industri film lokal.

Saat ini sedang ramai diperbincangkan di sosial media film dalam negeri, seperti film Sri Asih yang diproduksi Joko Anwar dan disutradarai Upi Avianto. Film tersebut merupakan bagian dari Jagat Sinema Bumilangit yang telah dibuka dengan superhero pertamanya, Gundala(2019). Film Sri Asih dibintangi sederet bintang papan atas seperti Pevita Pearce, Reza Rahardian, Jefri Nichol, Randy Pangalila, Surya Saputra, Christine Hakim, Dimas Anggara, Revaldo, Ario Bayu, Jenny Zhang, dan lainnya.

Ketika film Sri Asih tayang banyak perbincangan bermunculan di media sosial, contohnya pada aplikasi twitter para penggemar film Indonesia juga memberi respon komentar dan pendapat yang menunjukkan respon berbeda-beda seperti gembira, sedih, marah dan lain nya lewat tweet yang mereka upload, dari beragam tweet yang ada dapat menjadi data yang bisa dianalisis untuk mengetahui ekspresi atau respon bagaimana penonton menilai film Sri Asih.

Rating menjadi penilaian terhadap suatu film dengan skala tertentu. Selain itu, data ulasan yang didapatkan lewat twitter menjadi penjabaran penilaian penggemar terhadap film. Aspek penilaian yang terdapat dalam ulasan antara lain penyampaian cerita, teknik pengambilan gambar, lakon pemerannya, efek visual dan lain sebagainya. Dalam ulasan sendiri terdapat kritik atau komentar yang berisikan sentimen terhadap film tersebut. Analisis sentimen dapat membantu penggemar film untuk melihat sentimen yang dimiliki suatu film tergolong ke dalam kelompok sentimen positif atau negatif. Dalam proses analisis sentimen dapat menggunakan metode klasifikasi Naïve Bayes. Pada proses klasifikasinya akan diperhitungkan kemunculan kata-kata yang memiliki muatan sentimen dalam ulasan serta memperhitungkan probabilitas nilai rating terhadap kelas sentimen positif atau negatif. Dengan adanya sentimen analisis terhadap ulasan dan rating film diharapkan memudahkan para penonton untuk mendapatkan penilaian terhadap film Sri Asih sehingga dapat mempertimbangkan apakah Sri Asih merupakan film yang akan ditonton.

2. METODE

2.1 Crawling Data

```
library(twitter)
api_key<- "cvE55mBpAA4ryBCO2QFvkkra3"
api_secret<-
"lyCrqoYeGj1zmk125Nn9DBgNFjXpUzEbRksT3LpdxP0jnxsz8q"
access_token<-
"14629975088591872049AiHfSDm17ogce6xs3VlNkrSv9utHK"
access_token_secret<-
"CkuYYSOajxV7fbh7pTZYFHlrLEz8kWo5q3hkggcVZkws3"
setup_twitter_oauth(api_key,api_secret,access_token,access_token_secret)

tw = searchTwitter('sri asih -filter:retweets',
                    n = 500,
                    retryOnRateLimit = 500,
                    lang = "id" ) #retryOnRateLimit untuk looping
saveRDS(tw,file = 'tweetSriAsih.rds')

datatori <- do.call("rbind", lapply(tw, as.data.frame))

write.csv(datatori,'OriginalTweetSriAsih.csv')
```

Pada tahapan ini dilakukan crawling data (proses pengambilan data) dari TwitterAPI. Library yang digunakan adalah twitterR. Pada tahapan awal dimulai dengan membuat akun developer twitter kemudian mendapatkan 4 key yaitu *api key*, *api secret key*, *access token key*, *access token secret key* yang akan digunakan untuk mengakses data pada twitter dan mencari apa yang akan dibutuhkan (film apa yang dibutuhkan). Setelah data berhasil dikumpulkan, data akan disimpan ke dalam file dengan format .csv, kemudian akan dibaca dan dilakukan proses pada tahap selanjutnya.

2.2 Preprocessing Data

```
library(vroom) #membaca data
library(tm) #untuk cleaning text data
library(tidyverse)

#CLEANING DATA
tw <- readRDS('tweetSriAsih.rds')
DataKotor = twListToDF(tw) #convert twitter list to data

#menampilkan semua tweet yang kita mining
DataKotor2 <- DataKotor$text

DataKotorCorpus <- Corpus(VectorSource(DataKotor2))

##hapus URL
removeURL <- function(x) gsub("http[^\s:]*", "", x)
twitclean <- tm_map(DataKotorCorpus, removeURL)

##hapus New Line
```

```

removeNL <- function(y) gsub("\n", "", y)
twitclean <- tm_map(twitclean, removeNL)

##hapus koma
replacecomma <- function(y) gsub(",", "", y)
twitclean <- tm_map(twitclean, replacecomma)

##hapus retweet
removeRT <- function(y) gsub("RT ", "", y)
twitclean <- tm_map(twitclean, removeRT)

##hapus titik
removetitik2 <- function(y) gsub(":", "", y)
twitclean <- tm_map(twitclean, removetitik2)

##hapus titik koma
removetitikkoma <- function(y) gsub(";", " ", y)
twitclean <- tm_map(twitclean, removetitikkoma)

#hapus titik3
removetitik3 <- function(y) gsub("p.", "", y)
twitclean <- tm_map(twitclean, removetitik3)

#hapus &
removeamp <- function(y) gsub("&", "", y)
twitclean <- tm_map(twitclean, removeamp)

#hapus Mention
removeUN <- function(z) gsub("@\\w+", "", z)
twitclean <- tm_map(twitclean, removeUN)

#hapus space dll
remove.all <- function(xy) gsub("[^[:alpha:][:space:]]*", "", xy)
twitclean <-tm_map(twitclean,stripWhitespace)
inspect(twitclean[1:10])
twitclean <- tm_map(twitclean,remove.all)
twitclean <- tm_map(twitclean, removePunctuation) #tanda baca
twitclean <- tm_map(twitclean, tolower) #mengubah huruf kecil
myStopwords <- readLines("stopword.txt", warn = FALSE)
twitclean <- tm_map(twitclean,removeWords,myStopwords)
twitclean <- tm_map(twitclean , removeWords,

c('kalo','akun','ada','sini','langsung','','gak','org','saya','nonton',
'udah','cok','babi','tinggal','njir','itu','tiktok','anjrit','aja',
'kan','dah','kayak','dari','gue','kemarin','bahan','kyk','tinggal',
'yg','game','tae','nya','gk','kakaa','anj','sudah','saka',
'dahal','jadi','kkalau','sebab','mesti','bgt','nak','gw','yg',
'sama','bisa','date', ' taiikk','ngen'))

#HAPUS DATA 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.error with sapply
twitclean = sapply(twitclean, try.error)

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

# dataframe data yg sudah bersih
dataframe<-data.frame(text=unlist(sapply(twitclean, `[`)),
stringsAsFactors=F)
View(dataframe)
write.csv(dataframe, 'TweetCleanSriAsih.csv')

```

Setelah data dari twitter didapatkan kemudian data disiapkan terlebih dahulu sebelum masuk ke tahap analisis sentimen. Preprocessing data ini bertujuan untuk membersihkan data dari format atau kata yang tidak diperlukan, sehingga program dapat berjalan dengan baik dan hasil akurat. Proses pembersihan data adalah seperti kode diatas.

2.3 Pembuatan Shiny, Barplot Sentimen Analisis, dan Implementasi Algoritma Naive Bayes dan Wordcloud

```

library(tm) #library untuk penggunaan corpus dalam cleaning data
library(RTextTools) #mengkalifikasi text secara otomatis dengan
supervised learning
#library yang terdapat sebuah algoritma naivebayes
library(e1071)
library(dplyr)
library(caret)
library(syuzhet) #untuk membaca fungsi get_nrc
library(shiny) #package shiny
library(wordcloud)

kalimat2<-read.csv("TweetCleanSriAsih.csv",header=TRUE)
#skoring
kata.positif <- scan("positive-
words.txt",what="character",comment.char=";")
kata.negatif <- scan("negative-
words.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")

dataLabel <- read.csv("datalabel.csv")
# bagian yang mengatur tampilan web, baik input maupun output yang
akan ditampilkan dalam web app.
ui <- fluidPage(
  titlePanel("Sentimen Analisis Review Film Sri Asih"), #halaman
judul dr fluid page
  mainPanel( #tab pada fluidpage
    #plot output : untuk scatterplot
    tabsetPanel(type = "tabs",
                 tabPanel("Barplot", plotOutput("barplot")), #tab
berupa scatterplot/grafik

```

```

        tabPanel("Data",
                  DT::dataTableOutput('tbl')), #tab berupa data clening twitter
        tabPanel("Wordcloud", plotOutput("Wordcloud")) #tab
        berupa worldcloud
    )
)

# SERVER
# Disinialah tempat dimana data akan dianalisis dan diproses lalu
# hasilnya akan ditampilkan atau diplotkan pada bagian mainpanel() ui
# yang telah dijelaskan sebelumnya.
server <- function(input, output) {

#Output Data
  output$tbl = DT::renderDataTable({
    DT::datatable(dataLabel, options = list(lengthChange = FALSE)) #
    data akan ditampilkan dalam beberapa halaman.
  })

#Pembuatan Barplot Sentimen Analisis
  output$barplot <- renderPlot({sriasih_dataset<-
read.csv("TweetCleanSriAsih.csv",stringsAsFactors = FALSE)
  review <-as.character(sriasih_dataset$text)
  s<-get_nrc_sentiment(review)
  review_combine<-cbind(sriasih_dataset$text,s)
  par(mar=rep(3,4))
  barplot(colSums(s),col=rainbow(10),ylab='count',main='Sentimen
Analisis')
  }, height=400)

#Implementasi Algoritma Naive Bayes dan Wordcloud
  output$Wordcloud <- renderPlot({
    set.seed(20)
    df<-df[sample(nrow(df)),]
    df<-df[sample(nrow(df)),]
    glimpse(df)
    df$X=as.factor(df$X)
    corpus<-Corpus(VectorSource(df$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, stopwords(kind="en"))%>%
  tm_map(stripWhitespace)
dtm<-DocumentTermMatrix(corpus.clean)
inspect(dtm[1:10,1:20])
df.train<-df[1:100,]
df.test<-df[101:200,]
dtm.train<-dtm[1:100,]
dtm.test<-dtm[101:200,]
corpus.clean.train<-corpus.clean[1:100]
corpus.clean.test<-corpus.clean[101:200]

dim(dtm.train)
fivefreq<-findFreqTerms(dtm.train,5)
length(fivefreq)
dtm.train.nb<-DocumentTermMatrix(corpus.clean.train,control =
list(dictionary=fivefreq))
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)
classifier<-naiveBayes(trainNB,df.train$X,laplace = 1)
wordcloud(corpus.clean,min.freq =
4,max.words=100,random.order=F,colors=brewer.pal(8,"Dark2"))
})

}
shinyApp(ui = ui, server = server, options = list(height = "1080px"))

```

Data yang bersih kemudian dianalisis sentimennya dengan `get_nrc_sentimen`, hasilnya akan dikemas ke dalam bentuk barplot yang memperlihatkan bagaimana analisis sentimen secara keseluruhan untuk setiap emosi. Dengan dibentuk ke dalam sebuah barplot maka dominansi dari setiap emosi akan terlihat secara jelas. Kemudian data di setting dan dibagi sebagai data training dan data testing untuk penerapan algoritma naive bayes. Setelah itu, data ditampilkan dalam bentuk wordcloud.

Untuk dapat menampilkan hasil analisis secara interaktif maka digunakan shiny untuk membuat website sederhana dalam menampilkan hasil analisis. Prosesnya dimulai dengan melakukan penyesuaian untuk UI yang ingin ditampilkan, seperti pembentukan tab, output, dan lain lain. Kemudian dilakukan penyesuaian server untuk menampilkan output dari analisis ke dalam website. Setelah itu, website sudah dapat ditampilkan. Proses pembentukan shiny, barplot sentimen analisis dan implementasi algoritma naive bayes dan wordcloud dapat dilihat pada kode diatas.

2.4 Perhitungan Probabilitas Algoritma Naive Bayes

No	Class	Data Testing
1.	Positif	166
2.	Negatif	166
3.	Netral	166
Total		500

Tabel 1. Pembagian Data Testing

Menghitung probabilitas kata dilakukan untuk bertujuan mendapatkan term dengan nilai yang lebih penting dan dianggap relevan untuk dijadikan kata kunci. Proses pembobotan menggunakan algoritma Naive Bayes Classifier dalam proses perhitungan persamaan dapat dilihat pada Tabel 2.

Kata	Frekuensi kemunculan kata (W_k)		
	166 Tweet - Positif	166 Tweet - Negatif	166 Tweet - Netral
Bagus	9	10	22
Keren	7	3	4
Gundala	7	12	40
Banget	20	1	10
Film	23	21	57

Tabel 2. Contoh Kemunculan Term Frekuensi

Selanjutnya mencari probabilitas kata pagi, cuaca, baik, oke, hoak, jangan, salah, bantu, mungkin, tumben:

Diketahui :

nTweet Positif : 166

nTweet Negatif : 166

nTweet Netral : 166

a. Kata Bagus

$$P(\text{Bagus}|\text{Positif}) = \frac{9 + 1}{166 + 334} = 0,020$$

$$P(\text{Bagus}|\text{Negatif}) = \frac{10 + 1}{166 + 334} = 0,022$$

$$P(\text{Bagus}|\text{Netral}) = \frac{22 + 1}{166 + 334} = 0,046$$

b. Kata Keren

$$P(\text{Keren}|\text{Positif}) = \frac{7 + 1}{166 + 334} = 0,016$$

$$P(\text{Keren}|\text{Negatif}) = \frac{3 + 1}{166 + 334} = 0,008$$

$$P(\text{Keren}|\text{Netral}) = \frac{4 + 1}{166 + 334} = 0,010$$

c. Kata Gundala

$$P(\text{Gundala}|\text{Positif}) = \frac{7 + 1}{166 + 334} = 0,016$$

$$P(\text{Gundala}|\text{Negatif}) = \frac{12 + 1}{166 + 334} = 0,026$$

$$P(\text{Gundala}|\text{Netral}) = \frac{40 + 1}{166 + 334} = 0,082$$

d. Kata Banget

$$P(\text{Banget}|\text{Positif}) = \frac{20 + 1}{166 + 334} = 0,042$$

$$P(\text{Banget}|\text{Negatif}) = \frac{1 + 1}{166 + 334} = 0,004$$

$$P(\text{Banget}|\text{Netral}) = \frac{10 + 1}{166 + 334} = 0,022$$

e. Kata Film

$$P(\text{Film}|\text{Positif}) = \frac{23 + 1}{166 + 334} = 0,048$$

$$P(\text{Film}|\text{Negatif}) = \frac{21 + 1}{166 + 334} = 0,044$$

$$P(\text{Film}|\text{Netral}) = \frac{57 + 1}{166 + 334} = 0,116$$

Setelah mendapatkan hasil dari probabilitas kata, kemudian akan menghitung probabilitas dari dokumen (Tweet) sampel atau contoh. Diasumsikan $P(V_j)$ (probabilitas kategori dokumen) sama dengan $\frac{|doc_j|}{|Contoh|}$ (jumlah dokumen setiap kategori) dibagi dengan $|Contoh|$ (jumlah dokumen yang digunakan sebagai data training dari seluruh kategori). Diperoleh persamaan

$$P(v_j) = \frac{|doc_j|}{|Contoh|}$$

Perhitungan probabilitas Tweet dimana terdapat 500 Tweet yang terbagi 166 Tweet kelas positif, 166 Tweet kelas negatif dan 166 Tweet kelas netral :

$$P(\text{Positif}) = \frac{166}{500} = 0,3$$

$$P(\text{Negatif}) = \frac{166}{500} = 0,3$$

$$P(\text{Netral}) = \frac{166}{500} = 0,3$$

Data kata dari Tweet nilai probabilitas dapat dilihat pada Tabel 3.

No.	Kata	Prob positif V1	Prob Negatif V2	Prob Netral V3
1.	Bagus	0,020	0,022	0,046
2.	Keren	0,016	0,008	0,010
3.	Gundala	0,016	0,026	0,082
4.	Banget	0,042	0,004	0,022
5.	Film	0,048	0,044	0,116

Tabel 3. Daftar Uji Kata

Untuk mendapatkan nilai probabilitas nilai tertinggi langkah pertama yakni menghitung $(P(W_{k1}|V_j)P(V_j))$. $P(V_j)$ didapat probabilitas dokumen:

$$(\text{Bagus}|\text{Positif}) : 0,020 * 0,3 = 0,0060$$

$$(\text{Bagus}|\text{Negatif}) : 0,022 * 0,3 = 0,0066$$

(Bagus Netral)	: $0,046 * 0,3$	= 0,0138
(Keren Positif)	: $0,016 * 0,3$	= 0,0048
(Keren Negatif)	: $0,008 * 0,3$	= 0,0024
(Keren Netral)	: $0,010 * 0,3$	= 0,0030
(Gundala Positif)	: $0,016 * 0,3$	= 0,0048
(Gundala Negatif)	: $0,026 * 0,3$	= 0,0078
(Gundala Netral)	: $0,082 * 0,3$	= 0,0246
(Banget Positif)	: $0,042 * 0,3$	= 0,0126
(Banget Negatif)	: $0,004 * 0,3$	= 0,0012
(Banget Netral)	: $0,022 * 0,3$	= 0,0066
(Film Positif)	: $0,048 * 0,3$	= 0,0144
(Film Negatif)	: $0,044 * 0,3$	= 0,0132
(Film Netral)	: $0,116 * 0,3$	= 0,0348

Setelah itu menghitung nilai probabilitas tertinggi dari masing-masing kategori.

Probabilitas Positif Tertinggi	= $0,0060 * 0,0048 * 0,0048 * 0,0126 * 0,0144 = 2,50822656E-11$
Probabilitas Negatif Tertinggi	= $0,0066 * 0,0024 * 0,0078 * 0,0012 * 0,0132 = 1,95706368E-11$
Probabilitas Netral Tertinggi	= $0,0138 * 0,0030 * 0,0246 * 0,0066 * 0,0348 = 2,339152992E-10$

Probabilitas Positif Tertinggi	Probabilitas Positif Tertinggi	Probabilitas Positif Tertinggi
2,50822656E-11	1,95706368E-11	2,339152992E-10

Tabel 4. Nilai Probabilitas

Dari perhitungan algoritma *naive bayes* yang telah dilakukan, kemudian dicari perbandingan nilai probabilitas tertinggi dari masing-masing *Tweet* sehingga *Tweet* yang sudah ditesing dapat di kategorikan kedalam dokumen *Tweet* opini yang sesuai dengan isi teksnya. Pada tabel 4 dapat dilihat hasil dari keseluruhan proses perhitungan probabilitas tertinggi dengan *naive bayes*. Pada tabel tersebut probabilitas positif memiliki nilai tertinggi. Sehingga dapat dipastikan *Tweet* yang dipilih merupakan dokumen *Tweet* positif.

3. HASIL DAN PEMBAHASAN

3.1 Data Original

Crawling data dari TwitterAPI menghasilkan data sebagai berikut :

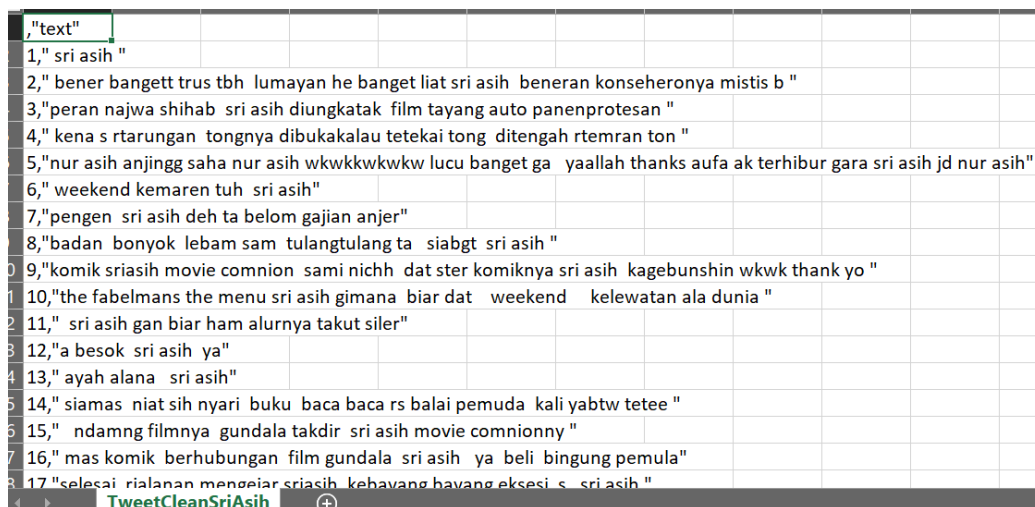


The image shows a screenshot of a JSON array containing multiple tweet objects. Each object includes fields like 'text', 'favoriteCount', 'replyToSN', 'created', 'truncated', 'replyToSID', 'id', 'replyToUID', 'statusSource', 'screenName', 'retweetCount', 'isRetweet', 'retweeted', 'longitude', and 'latitude'. The tweets are in Indonesian and mention 'Sri Asih' and 'Jagobumi'.

Gambar 3.1 Hasil Crawling Data dari TwitterAPI

3.2 Data Bersih

Data yang telah diambil dari twitter kemudian melalui proses preprocessing data. Dari proses tersebut menghasilkan data yang sudah bersih dan siap untuk dilakukan sentimen analisis. Berikut adalah potongan data yang sudah bersih :



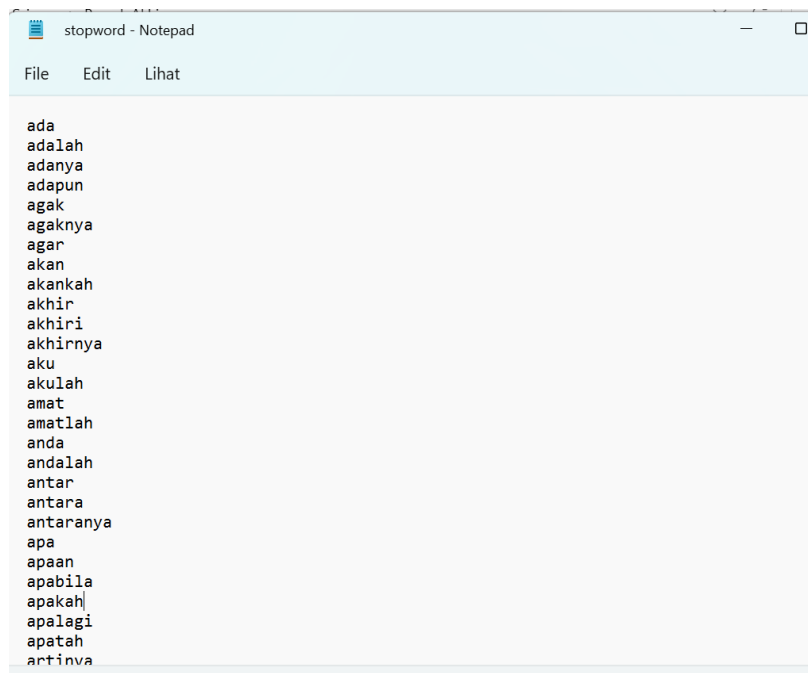
The image shows a screenshot of a cleaned dataset of tweets. The first column is labeled 'text' and contains 17 rows of cleaned Indonesian text. The second column is labeled 'TweetCleanSriAsih' and contains a plus sign icon. The text in the first column is as follows:

text	TweetCleanSriAsih
1, " sri asih "	+
2," bener banget trus tbh lumayan he banget liat sri asih beneran konseheronya mistis b "	
3,"peran najwa shihab sri asih diungkatak film tayang auto panenprotesan "	
4," kena s rtarungan tongnya dibukakalau tetekai tong ditengah rtemran ton "	
5,"nur asih anjingg saha nur asih kwkwkwkwkw lucu banget ga yaallah thanks aufa ak terhibur gara sri asih jd nur asih"	
6," weekend kemaren tuh sri asih"	
7,"pengen sri asih deh ta belum gajian anjer"	
8,"badan bonyok lebam sam tulangtulang ta siabgt sri asih "	
9,"komik sriasih movie comnion sami nichh dat ster komiknya sri asih kagebunshin kwkw thank yo "	
10,"the fabelmans the menu sri asih gimana biar dat weekend kelewatan ala dunia "	
11," sri asih gan biar ham alurnya takut siler"	
12,"a besok sri asih ya"	
13," ayah alana sri asih"	
14," siamas niat sih nyari buku baca baca rs balai pemuda kali yabtw tetee "	
15," ndamng filmnya gundala takdir sri asih movie comnionny "	
16," mas komik berhubungan film gundala sri asih ya beli bingung pemula"	
17,"selesai rialanan mengelar sriasih kebavang havang ekseksi s sri asih "	

Gambar 3.2 Data yang Sudah Melalui Tahap Preprocessing

3.3 Stopword

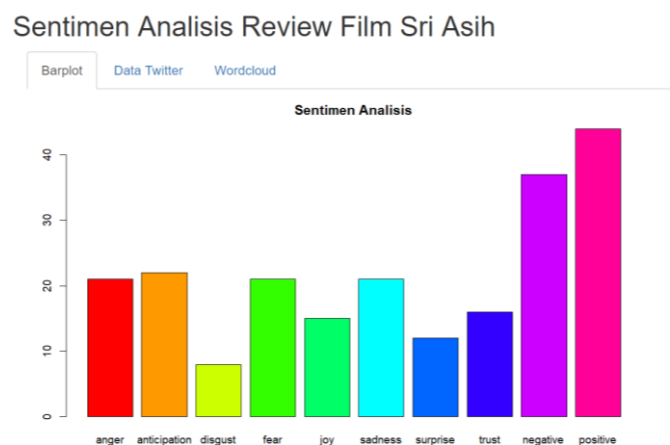
Dalam Tahapan preprocessing data, stopwords atau bisa dikatakan sebagai kata kata yang dianggap tidak diperlukan akan dibuang. Untuk melakukan pembersihan data dari stopwords diperlukan file stopwords yang berisi seperti berikut :



Gambar 3.3 List Stopword yang Digunakan dalam Data Preprocessing

3.4 Tampilan Shiny untuk Tab Scatterplot

Hasil dari proses analisis sentimen dibentuk dalam bentuk barplot. Barplot memperlihatkan emosi apa saja yang ada dalam tweet yang dianalisis dan juga bagaimana kemunculan masing masing emosi tersebut dalam tweet terhadap Film Sri Asih.



Gambar 3.4 Tampilan Shiny Scatterplot

3.5 Tampilan Shiny Tabel Data

Selain daripada Barplot, shiny juga dapat menampilkan tabel dari data yang sudah diolah. Data yang sudah melalui tahap preprocessing pada tampilan dalam bentuk tabel dan dimuat dalam beberapa halaman karena terdapat banyak data.

Barplot		Data Twitter		Wordcloud	
					Search: <input type="text"/>
X	klasifikasi	score	text		
1	1	Netral	0	karakter film surhero sri asih	
2	2	Negatif	-1	penyesalan sri asih the menu kelur bioskojadi kesel gini yak dibalik	
3	3	Netral	0	nasaran the menu ta review sri asih bagus sosmed yh	
4	4	Netral	0	jarang film indonesia cgi sri asih	
5	5	Netral	0	the menusri asih	
6	6	Positif	1	teruntut hak terlibatfilm film adisatria terbaik adimin tonton an kemungkinnya	
7	7	Negatif	-1	gatot kaca no gundala sri asih yes	
8	8	Negatif	-1	ah na kentang scane ending berantem sri asih	
9	9	Netral	0	sri asih	
10	10	Positif	1	ta emang mas film sri asih cgi keren	



4. KESIMPULAN

Berdasarkan hasil analisis sentimen menggunakan metode naive bayes pada data respon dan review penonton yang beragam pada film Sri Asih lewat tweet di aplikasi twitter program yang praktikan buat dan divisualisasikan dengan tampilan Shiny Scatterplot disimpulkan bahwa, pada visualisasi diagram menunjukkan hasil tertinggi adalah respon positif dan respon terendah adalah joy (sukacita), pada wordcloud menunjukkan hasil kata yang paling banyak disebutkan adalah kata sri asih, film, gundala, dan bagus. Sehingga dengan adanya sentimen analisis terhadap ulasan dan rating film diharapkan memudahkan para penonton untuk mendapatkan penilaian terhadap film Sri Asih sehingga dapat mempertimbangkan apakah Sri Asih merupakan pilihan film yang akan ditonton.