**LAMPIRAN CODING DATA**

**Lampiran 1.** *Syntax Crawling* Data Menggunakan RStudio

library(twitteR)

*#ID Twitter API*

consumer\_key <- 'your consumer key'

consumer\_secret <- 'your consumer secret'

access\_token <- 'your acces token'

access\_secret <- 'your access secret'

*#Login Twitter API*

setup\_twitter\_oauth(consumer\_key,consumer\_secret,access\_token,access\_secret)

#Searching Tweet

gocar <- searchTwitter('gocar', lang="id", n=15000, resultType="recent")

write.csv(twListToDF(gocar), file="gocar.csv")

grabcar <- searchTwitter('grabcar', lang="id", n=15000, resultType="recent")

write.csv(twListToDF(grabcar), file="grabcar.csv")

bluebird <- searchTwitter('bluebird', lang="id", n=15000, resultType="recent")

write.csv(twListToDF(bluebird), file="bluebird.csv")

**Lampiran 2.** Data *Tweet* BlueBird Taxi dari Twitter API

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *Tweet* Ke- | Class | *Tweet* | ReplyToSN | *ScreenName* |
| 1 | 0 | Nawar bajay dr Tanah Abang ke Gambir. Si abang buka harga 35rb (kaget).. Tawar 20, dia turunin cuma sampe 25. Dalam… https://t.co/w6AJjyXooY | NA | firman23 |
| 2 | 1 | antrian taksi non bluebird 40sekian, taksi bluebird 80 sekian. semua taksi online ga bisa diakses saking padatnya permintaan.  Jakarta ;( | NA | let666be |
| 3 | 1 | Berkat bluebird gak ada yg pickup, dan gocar harganya jadi 2x lipat. Suda tida paham lagi.  Tp nggak apa, ngirit juga naik brt. | NA | riniiik |
| 4 | 1 | @Bluebirdgroup easyride lagi error ya.. beberapa kali saya naik bluebird dan booking pembayaran melalui kartu tidak bisa | Bluebirdgroup | teguhekas |
| . | . | . | . | . |
| . | . | . | . | . |
| 3109 | 1 | @Bluebirdgroup taxi blue bird no pintu RD 3699 Limo parkir dari shubuh tadi didepan rumah tetangga saya. Supir ga a… https://t.co/zITx2x3udy | Bluebirdgroup | Wina\_aswir |
| 3110 | 1 | @Bluebirdgroup sampai saat ini tidak ada follow up dr blue bird untuk saya sebagai customer | wishnuturangan | wishnuturangan |
| 3111 | 1 | @Bluebirdgroup Saya pelanggan setia blue bird tetapi hari ini saya order jam 5.30 sampai saat email ini di kirim sa… https://t.co/hiAM5Ocm91 | Bluebirdgroup | wishnuturangan |

Keterangan : 0 = Positif

1 = Negatif

**Lampiran 3.** Data *Tweet* GoCar dari Twitter API

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *Tweet* Ke- | Class | *Tweet* | ReplyToSN | *ScreenName* |
| 1 | 0 | terima kasih go car.......sudah antar jemput..........murah sekali https://t.co/PR9BpCFWOd | NA | Solichinzm |
| 2 | 1 | Halo @gojekindonesia saya naik gocar jarak pendek 8k #gopay tapi driver ga mau alasannya posisinya jauh. Dia minta… https://t.co/77ZCuDtsmY | NA | bundaPandu2013 |
| 3 | 1 | @gojekindonesia selamat sore min, mau lapor dong. td pagi saya pesen go car dengan pembayaran gopay. saldonya udh kepotong. tp… #Jakarta | gojekindonesia | BeritaIndonesa |
| 4 | 1 | @gojekindonesia selamat sore min, mau lapor dong. td pagi saya pesen go car dengan pembayaran gopay. saldonya udh k… https://t.co/8Oa7Pk7bLy | gojekindonesia | juliolim13 |
| . | . | . | . | . |
| . | . | . | . | . |
| 4332 | 1 | @gojekindonesia min, saya order gocar, saya blm naik tp udh on the way with you statusnya https://t.co/1Sb6D5dqja | gojekindonesia | rereniren |
| 4333 | 0 | @askmenfess Gatau seenaknya aja. Kalo malem2 naik motor gapapa. Cuma kalo siang2 mending naik gocar bersama drpd na… https://t.co/Tj6Hq5zA9t | askmenfess | 95pals |
| 4334 | 1 | @gojekindonesiaa Aplikasi gojek lagi error kah? Tadi saya 3x coba order gocar, dapat driver posisi di ancol, sem… https://t.co/IYHgUN9HH9 | gojekindonesia | tubiwityu |

Keterangan : 0 = Positif

1 = Negatif

**Lampiran 4.** Data *Tweet* GrabCar dari Twitter API

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| *Tweet* Ke- | Class | *Tweet* | ReplyToSN | *ScreenName* |
| 1 | 1 | Udah gupuh banget sampek cancel grab car, ngebut naik grabbike, eeeh kretanya telat. Harusnya jam 21.00. Sekarang 2… https://t.co/Ed4dDsa3xQ | NA | lalanuradhilla |
| 2 | 1 | Selalu ada pengalaman buruk naik grab car. Driver yang kasar, tak berbudi bahasa, suruh cancel trip. Lepas ni kalau… https://t.co/JCzhRfRy9d | NA | ainunlmuaiyanah |
| 3 | 1 | @GrabID latest update apps grab ada bugs gak bisa pesan grab car. Dari grab gak ada solusi dan gak niat benerin. Sa… https://t.co/ZWPXghETjP | GrabID | JF897 |
| 4 | 1 | Pesen grabcar dari apartemen buat balik kos. Sekalian bawa balik koper + tentengan duty free. Drivernya bantuin ang… https://t.co/edvxz4Nq9G | NA | Rerefransiska |
| . | . | . | . | . |
| . | . | . | . | . |
| 2284 | 1 | kecurigaanku terjawab. beberapa kali order grabcar dan selalu beda mobil &amp; platnya sama di app. ternyata doi beli akun bodong yawlaa | NA | sofiarinda |
| 2285 | 0 | Setelah sekian lama gak naik grabcar akhirnya hari ini naik grabcar karena lebih murah dr gocar. Dan yah dapet lagi… https://t.co/misQ3EEgXg | NA | Aderinaa |
| 2286 | 1 | @GrabID sy order grabcar, driver udh blg bntar lg dia jemput sy eh tiba2 muncul rating, kaya sy udh diantar. Gmn tuh? Pdhl ga ketemu sm skli | GrabID | dllazz |

Keterangan : 0 = Positif

1 = Negatif

**Lampiran 5.** *Syntax Input* dan Praproses Data Menggunakan Python 3.6

import pandas as pd

import pandas as dataframe

import string

import nltk

from nltk.tokenize import word\_tokenize

import re

import sys

import os

from Sastrawi.Stemmer.StemmerFactory import StemmerFactory

from IPython.display import display

*#Input Data*

bluebird = pd.read\_csv('D:/bluebird.csv', engine = 'python')

print(bluebird)

bluebirdtweet = bluebird['text']

print (bluebirdtweet)

*#Cleansing*

*#Menghapus Link*

bluebirdclearlink = []

for line in bluebirdtweet:

result = re.sub(r"http\S+","",line)

bluebirdclearlink.append(result)

*#Menghapus Simbol RT*

bluebirdclearrt = []

for line in bluebirdclearlink:

result = re.sub(r"RT","",line)

bluebirdclearrt.append(result)

*#Menghapus Username*

bluebirdclearusername = []

for line in bluebirdclearrt:

result = re.sub(r"@\S+","",line)

bluebirdclearusername.append(result)

*#Menghapus Hastag*

bluebirdclearhastag = []

for line in bluebirdclearusername:

result = re.sub(r"3(\w+)","",line)

bluebirdclearhastag.append(result)

**Lampiran 5.** *Syntax Input* dan Praproses Data Menggunakan Python 3.6 (Lanjutan)

*#Menghapus Nama*

bluebirdclearname = []

for line in bluebirdclearhastag:

result = re.sub(r"bluebird","",line)

bluebirdclearname.append(result)

*#Clear Space Enter*

bluebirdclear = []

for line in bluebirdclearname:

result=re.sub(r"…"," ",line)

bluebirdclear.append(result)

*#Case Folding*

bluebirdlower = []

for line in bluebirdclear:

a = line.lower()

bluebirdlower.append(a)

*#Stemming*

factory = StemmerFactory()

stemmer = factory.create\_stemmer()

bluebirdstemmed = map(lambda x: stemmer.stem(x), bluebirdlower)

bluebird\_no\_punc = map(lambda x: x.translate(str.maketrans('','', string.punctuation)), bluebirdstemmed)

bluebird\_no\_punc = list(bluebird\_no\_punc)

*#Stopwords dan Tokenizing*

stopwords = open('D:/stoplist.txt', 'r').read()

bluebirdtrain = []

bluebirdfinal = []

df = []

for line in bluebird\_no\_punc:

word\_token = word\_tokenize(line)

word\_token = [word for word in word\_token if not word in stopwords and not word[0].isdigit()]

bluebirdfinal.append(word\_token)

df.append(" ".join(word\_token))

**Lampiran 5.** *Syntax Input* dan Praproses Data Menggunakan Python 3.6 (Lanjutan)

for l in bluebirdfinal:

bluebirdtrain+= l

final\_bluebird={v: bluebirdtrain.count(v) for v in set(bluebirdtrain)}

import csv

with open ('D:/final\_bluebird.csv','w',newline="") as csv\_file:

writer = csv.writer(csv\_file)

for key, value in final\_bluebird.items():

writer.writerow([key, value])

**Lampiran 6.** Frekuensi Kata Kunci

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| BlueBird Taxi | | GoCar | | GrabCar | |
| Kata Kunci | **Frek** | **Kata Kunci** | **Frek** | **Kata**  **Kunci** | **Frek** |
| Driver | 463 | Driver | 1392 | Mitra | 456 |
| App | 457 | Uber | 916 | Ngobrol | 450 |
| Tumpang | 327 | Selamat | 772 | Timdudukdepan | 438 |
| Supir | 319 | Rating | 757 | Inspirasi | 429 |
| Bandara | 263 | Bahas | 756 | Driver | 393 |
| Tarif | 249 | Aneh | 753 | Pake | 194 |
| Gojek | 248 | Simpel | 753 | Katague | 144 |
| Org | 210 | Gojek | 389 | Grabbike | 142 |
| Nyegat | 202 | Order | 346 | Mobil | 140 |
| Gada | 200 | Pake | 330 | Order | 139 |
| sumber | 195 | mobil | 201 | uber | 133 |
| jalan | 194 | banget | 179 | pergi | 129 |
| aplikasi | 172 | drivernya | 175 | aplikasi | 123 |
| online | 172 | jalan | 173 | bike | 111 |
| pake | 165 | abang | 146 | pilih | 106 |
| manado | 156 | pagi | 144 | tinggal | 105 |
| . | . | . | . | . | . |
| . | . | . | . | . | . |
| . | . | . | . | . | . |
| genang | 1 | bapaknyah | 1 | palsu | 1 |
| soeta3 | 1 | setdaahhhh | 1 | reflek | 1 |
| kpan | 1 | mangkal | 1 | ehhhh | 1 |

**Lampiran 7.** *Syntax Naïve Bayes Classifier* (NBC) Menggunakan Python 3.6

*#Count Vectorizer*

from pandas import DataFrame

from sklearn.feature\_extraction.text import CountVectorizer

vect = CountVectorizer(min\_df=0., max\_df=1.0)

X = vect.fit\_transform(df)

cv = DataFrame(X.A, columns=vect.get\_feature\_names())

print (cv)

*#tf-idf*

from sklearn.feature\_extraction.text import TfidfTransformer

tfidf = TfidfTransformer(use\_idf=True).fit\_transform(cv)

tfidf\_train = (tfidf.toarray())

print (tfidf\_train)

print (tfidf\_train.shape)

tf = DataFrame(tfidf.A, columns=vect.get\_feature\_names())

print (tf)

*#Analisis Naïve Bayes Classifier*

data\_sentimen = grabcar['class']

sentiment = pd.DataFrame(data\_sentimen)

from sklearn.model\_selection import train\_test\_split,cross\_val\_score,StratifiedKFold,KFold,ShuffleSplit,StratifiedShuffleSplit

from sklearn.feature\_selection import SelectPercentile, f\_classif

from sklearn.model\_selection import StratifiedKFold

from sklearn.naive\_bayes import BernoulliNB

from sklearn.model\_selection import GridSearchCV

import numpy as np

X\_train,X\_test,Y\_train,Y\_test=train\_test\_split(x\_new,sentiment,test\_size=0.1,random\_state=43)

probasbaru = [BernoulliNB().fit(X\_train, Y\_train).predict\_proba(X\_train)]

results.append(probasbaru)

**Lampiran 7.** *Syntax Naïve Bayes Classifier* (NBC) Menggunakan Python 3.6 (Lanjutan)

probas = BernoulliNB()

probas.fit(X\_train, Y\_train)

y\_pred = probas.predict(X\_test)

print(confusion\_matrix(Y\_test,y\_pred))

print(classification\_report(Y\_test,y\_pred))

# get class probabilities for the first sample in the dataset

class1\_1 = [pr[0, 0] for pr in probasbaru]

class2\_1 = [pr[0, 1] for pr in probasbaru]

print ('--------------------------------------------------------------------')

print (probasbaru)

print ('hasil klasifikasi positif', class1\_1)

print ('hasil klasifikasi negatif', class2\_1)

YB = DataFrame.as\_matrix(sentiment)

print (YB)

X\_baru, Y = tfidf\_train, YB

kf = StratifiedKFold (n\_splits=10)

kf.get\_n\_splits(X\_baru)

cl = BernoulliNB()

for train, test in kf.split(X\_baru, YB):

(X\_baru[train],X\_baru[test])

(Y[train], Y[test])

a = cl.fit(X\_baru[train], Y[train])

b = cl.predict(X\_baru[test])

fpr, tpr, \_ = metrics.roc\_curve(Y[test], b)

auc = metrics.auc(fpr,tpr)

print (b)

print(confusion\_matrix(Y[test], b))

print(classification\_report(Y[test],b))

print("Area Under Curve ROC = {:.2f}% ".format(auc\*100))

**Lampiran 8.** *Confusion Matrix* Metode NBC pada BlueBird Taxi

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Data | *Fold* ke- | *True Positive* | *False Positive* | *True Negative* | *False Negative* | Jumlah *Tweet* |
| *Training* | 1 | 933 | 39 | 1671 | 156 | 2799 |
| 2 | 926 | 34 | 1676 | 164 | 2800 |
| 3 | 932 | 33 | 1677 | 158 | 2800 |
| 4 | 920 | 36 | 1674 | 170 | 2800 |
| 5 | 924 | 31 | 1679 | 166 | 2800 |
| 6 | 928 | 29 | 1681 | 162 | 2800 |
| 7 | 939 | 33 | 1677 | 151 | 2800 |
| 8 | 909 | 36 | 1674 | 181 | 2800 |
| 9 | 934 | 41 | 1669 | 156 | 2800 |
| 10 | 926 | 34 | 1676 | 164 | 2800 |
| *Testing* | 1 | 58 | 14 | 176 | 64 | 312 |
| 2 | 57 | 29 | 161 | 64 | 311 |
| 3 | 76 | 20 | 170 | 45 | 311 |
| 4 | 99 | 18 | 172 | 22 | 311 |
| 5 | 73 | 22 | 168 | 48 | 311 |
| 6 | 76 | 45 | 145 | 45 | 311 |
| 7 | 120 | 20 | 170 | 1 | 311 |
| 8 | 97 | 1 | 189 | 24 | 311 |
| 9 | 102 | 13 | 177 | 19 | 311 |
| 10 | 55 | 13 | 177 | 66 | 311 |

**Lampiran 9.** *Confusion Matrix* Metode NBC pada GoCar

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Data | *Fold* ke- | *True Positive* | *False Positive* | *True Negative* | *False Negative* | Jumlah *Tweet* |
| *Training* | 1 | 1907 | 77 | 1595 | 321 | 3900 |
| 2 | 1977 | 126 | 1546 | 251 | 3900 |
| 3 | 2012 | 147 | 1525 | 216 | 3900 |
| 4 | 2011 | 150 | 1522 | 217 | 3900 |
| 5 | 1948 | 111 | 1561 | 280 | 3900 |
| 6 | 1916 | 92 | 1580 | 312 | 3900 |
| 7 | 1904 | 84 | 1588 | 325 | 3901 |
| 8 | 1874 | 72 | 1600 | 355 | 3901 |
| 9 | 1904 | 79 | 1594 | 325 | 3902 |
| 10 | 1901 | 75 | 1598 | 328 | 3902 |

**Lampiran 9.** *Confusion Matrix* Metode NBC pada GoCar (Lanjutan)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Data | *Fold* ke- | *True Positive* | *False Positive* | *True Negative* | *False Negative* | Jumlah *Tweet* |
| *Testing* | 1 | 109 | 14 | 172 | 139 | 434 |
| 2 | 194 | 32 | 154 | 54 | 434 |
| 3 | 247 | 40 | 146 | 1 | 434 |
| 4 | 232 | 36 | 150 | 16 | 434 |
| 5 | 199 | 24 | 162 | 49 | 434 |
| 6 | 137 | 22 | 164 | 111 | 434 |
| 7 | 106 | 25 | 161 | 141 | 433 |
| 8 | 104 | 20 | 166 | 143 | 433 |
| 9 | 154 | 24 | 161 | 93 | 432 |
| 10 | 130 | 19 | 166 | 117 | 432 |

**Lampiran 10.** *Confusion Matrix* Metode NBC pada GrabCar

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Data | *Fold* ke- | *True Positive* | *False Positive* | *True Negative* | *False Negative* | Jumlah *Tweet* |
| *Training* | 1 | 1163 | 88 | 760 | 45 | 2056 |
| 2 | 1167 | 93 | 755 | 41 | 2056 |
| 3 | 1165 | 98 | 750 | 43 | 2056 |
| 4 | 1165 | 96 | 753 | 44 | 2058 |
| 5 | 1147 | 80 | 769 | 62 | 2058 |
| 6 | 1152 | 78 | 771 | 57 | 2058 |
| 7 | 1152 | 76 | 773 | 57 | 2058 |
| 8 | 1165 | 83 | 766 | 44 | 2058 |
| 9 | 1155 | 85 | 764 | 54 | 2058 |
| 10 | 1154 | 84 | 765 | 55 | 2058 |
| *Testing* | 1 | 108 | 24 | 71 | 27 | 230 |
| 2 | 132 | 21 | 74 | 3 | 230 |
| 3 | 128 | 28 | 67 | 7 | 230 |
| 4 | 129 | 24 | 70 | 5 | 228 |
| 5 | 94 | 21 | 73 | 40 | 228 |
| 6 | 71 | 14 | 80 | 63 | 228 |
| 7 | 80 | 19 | 75 | 54 | 228 |
| 8 | 129 | 18 | 76 | 5 | 228 |
| 9 | 104 | 25 | 69 | 30 | 228 |
| 10 | 85 | 20 | 74 | 49 | 228 |

**Lampiran 11.** *Syntax Support Vector Machine* (SVM) Menggunakan Python 3.6

data\_sentimen = grabcar['class']

sentiment = pd.DataFrame(data\_sentimen)

from \_\_future\_\_ import print\_function

from sklearn.model\_selection import train\_test\_split,cross\_val\_score,StratifiedKFold,KFold,ShuffleSplit,StratifiedShuffleSplit

from sklearn.feature\_selection import SelectPercentile, f\_classif

from sklearn.model\_selection import StratifiedKFold

from sklearn.svm import LinearSVC, SVC

from sklearn import svm, datasets

from sklearn.model\_selection import GridSearchCV

from sklearn.metrics import classification\_report

from sklearn.utils import shuffle

import numpy as np

X\_train,X\_test,Y\_train,Y\_test=train\_test\_split(tfidf\_train,sentiment,test\_size=0.1,random\_state=43)

# Spot Check Algorithms

tuned\_parameters = [{'kernel':['linear'], 'C':[0.01, 0.1, 1, 10, 100, 1000, 10000]}, {'kernel':['rbf'], 'C':[0.01, 0.1, 1, 10, 100, 1000, 10000], 'gamma':[1000, 100, 10, 1, 0.1, 0.01, 0.001]}]

scores = ['precision', 'recall']

for score in scores:

print ("# Turning hyper-parameters for %s" % score)

print ()

clf = GridSearchCV(SVC(), tuned\_parameters, cv=KFold(n\_splits=10, random\_state=0))

clf.fit(X\_train, Y\_train)

print("Best parameters set found on development set:")

print()

print(clf.best\_params\_)

print()

print("Grid scores on training set:")

print()

**Lampiran 11.** *Syntax Support Vector Machine* (SVM) Menggunakan Python 3.6 (Lanjutan)

means = clf.cv\_results\_['mean\_test\_score']

stds = clf.cv\_results\_['std\_test\_score']

for mean, std, params in zip(means, stds, clf.cv\_results\_['params']):

print("%0.3f (+/-%0.3f) for %r"

% (mean, std \* 2, params))

print()

print("Detailed classification report:")

print()

print("The model is trained on the full development set.")

print("The scores are computed on the full evaluation set.")

print()

y\_true, y\_pred = Y\_test, clf.predict(X\_test)

print(confusion\_matrix(y\_true, y\_pred))

print(classification\_report(y\_true, y\_pred))

print()

YB = DataFrame.as\_matrix(sentiment)

print (YB)

X\_baru, Y = tfidf\_train, YB

kf = StratifiedKFold (n\_splits=10)

kf.get\_n\_splits(X\_baru)

cl = SVC(kernel='linear', C=1)

print (kf)

for train, test in kf.split(X\_baru, YB):

(X\_baru[train],X\_baru[test])

(Y[train], Y[test])

a = cl.fit(X\_baru[train], Y[train])

b = cl.predict(X\_baru[test])

fpr, tpr, \_ = metrics.roc\_curve(Y[test], b)

auc = metrics.auc(fpr,tpr)

print (b)

print(confusion\_matrix(Y[test], b))

print(classification\_report(Y[test],b))

print("Area Under Curve ROC = {:.2f}% ".format(auc\*100))

**Lampiran 11.** *Syntax Support Vector Machine* (SVM) Menggunakan Python 3.6 (Lanjutan)

for train, test in kf.split(X\_baru, YB):

(X\_baru[train],X\_baru[test])

(Y[train], Y[test])

a = cl.fit(X\_baru[train], Y[train])

b = cl.predict(X\_baru[train])

fpr, tpr, \_ = metrics.roc\_curve(Y[train], b)

auc = metrics.auc(fpr,tpr)

print (b)

print(confusion\_matrix(Y[train], b))

print(classification\_report(Y[train],b))

print("Area Under Curve ROC = {:.2f}% ".format(auc\*100))

**Lampiran 12.** *Confusion Matrix* Metode SVM Kernel Linear pada BlueBird Taxi

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Data | *Fold* ke- | *True Positive* | *False Positive* | *True Negative* | *False Negative* | Jumlah *Tweet* |
| *Training* | 1 | 1019 | 27 | 1683 | 70 | 2799 |
| 2 | 1023 | 24 | 1686 | 67 | 2800 |
| 3 | 1015 | 22 | 1688 | 75 | 2800 |
| 4 | 1017 | 22 | 1688 | 73 | 2800 |
| 5 | 1020 | 21 | 1689 | 70 | 2800 |
| 6 | 1015 | 25 | 1685 | 75 | 2800 |
| 7 | 1018 | 28 | 1682 | 72 | 2800 |
| 8 | 1010 | 28 | 1682 | 80 | 2800 |
| 9 | 1018 | 28 | 1682 | 72 | 2800 |
| 10 | 1023 | 21 | 1689 | 67 | 2800 |
| *Testing* | 1 | 46 | 8 | 182 | 76 | 312 |
| 2 | 67 | 18 | 172 | 54 | 311 |
| 3 | 79 | 15 | 175 | 42 | 311 |
| 4 | 105 | 15 | 175 | 16 | 311 |
| 5 | 59 | 28 | 162 | 62 | 311 |
| 6 | 71 | 35 | 155 | 50 | 311 |
| 7 | 118 | 18 | 172 | 3 | 311 |
| 8 | 96 | 0 | 190 | 25 | 311 |
| 9 | 99 | 17 | 173 | 22 | 311 |
| 10 | 61 | 17 | 173 | 60 | 311 |

**Lampiran 13.** *Confusion Matrix* Metode SVM Kernel Linear pada GoCar

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Data | *Fold* ke- | *True Positive* | *False Positive* | *True Negative* | *False Negative* | Jumlah *Tweet* |
| *Training* | 1 | 2125 | 68 | 1604 | 103 | 3900 |
| 2 | 2124 | 93 | 1579 | 104 | 3900 |
| 3 | 2129 | 88 | 1584 | 99 | 3900 |
| 4 | 2131 | 98 | 1574 | 97 | 3900 |
| 5 | 2135 | 90 | 1582 | 93 | 3900 |
| 6 | 2139 | 81 | 1591 | 89 | 3900 |
| 7 | 2125 | 70 | 1602 | 104 | 3901 |
| 8 | 2129 | 72 | 1600 | 100 | 3901 |
| 9 | 2128 | 75 | 1598 | 101 | 3902 |
| 10 | 2137 | 70 | 1603 | 92 | 3902 |
| *Testing* | 1 | 175 | 43 | 143 | 73 | 434 |
| 2 | 213 | 51 | 135 | 35 | 434 |
| 3 | 247 | 57 | 129 | 1 | 434 |
| 4 | 240 | 60 | 126 | 8 | 434 |
| 5 | 218 | 39 | 147 | 30 | 434 |
| 6 | 185 | 39 | 147 | 63 | 434 |
| 7 | 146 | 45 | 141 | 101 | 433 |
| 8 | 179 | 35 | 151 | 68 | 433 |
| 9 | 198 | 51 | 134 | 49 | 432 |
| 10 | 165 | 34 | 151 | 82 | 432 |

**Lampiran 14.** *Confusion Matrix* Metode SVM Kernel Linear pada GrabCar

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Data | *Fold* ke- | *True Positive* | *False Positive* | *True Negative* | *False Negative* | Jumlah *Tweet* |
| *Training* | 1 | 1176 | 19 | 829 | 32 | 2056 |
| 2 | 1179 | 25 | 823 | 29 | 2056 |
| 3 | 1173 | 22 | 826 | 35 | 2056 |
| 4 | 1177 | 21 | 828 | 32 | 2058 |
| 5 | 1167 | 16 | 833 | 42 | 2058 |
| 6 | 1159 | 14 | 835 | 50 | 2058 |
| 7 | 1138 | 9 | 840 | 71 | 2058 |
| 8 | 1177 | 18 | 831 | 32 | 2058 |
| 9 | 1176 | 16 | 833 | 33 | 2058 |
| 10 | 1149 | 11 | 838 | 60 | 2058 |

**Lampiran 14.** *Confusion Matrix* Metode SVM Kernel Linear pada GrabCar (Lanjutan)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Data | *Fold* ke- | *True Positive* | *False Positive* | *True Negative* | *False Negative* | Jumlah *Tweet* |
| *Testing* | 1 | 110 | 23 | 72 | 25 | 230 |
| 2 | 133 | 24 | 71 | 2 | 230 |
| 3 | 128 | 32 | 63 | 7 | 230 |
| 4 | 128 | 27 | 67 | 6 | 228 |
| 5 | 89 | 18 | 76 | 45 | 228 |
| 6 | 67 | 8 | 86 | 67 | 228 |
| 7 | 52 | 13 | 81 | 82 | 228 |
| 8 | 129 | 17 | 77 | 5 | 228 |
| 9 | 85 | 23 | 71 | 49 | 228 |
| 10 | 66 | 12 | 82 | 68 | 228 |

**Lampiran 15.** *Confusion Matrix* Metode SVM Kernel RBF pada BlueBird Taxi

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Data | *Fold* ke- | *True Positive* | *False Positive* | *True Negative* | *False Negative* | Jumlah *Tweet* |
| *Training* | 1 | 1086 | 11 | 1699 | 3 | 2799 |
| 2 | 1082 | 4 | 1706 | 8 | 2800 |
| 3 | 1088 | 12 | 1698 | 2 | 2800 |
| 4 | 1079 | 3 | 1707 | 11 | 2800 |
| 5 | 1080 | 3 | 1707 | 10 | 2800 |
| 6 | 1087 | 9 | 1701 | 3 | 2800 |
| 7 | 1080 | 4 | 1706 | 10 | 2800 |
| 8 | 1080 | 5 | 1705 | 10 | 2800 |
| 9 | 1081 | 5 | 1705 | 9 | 2800 |
| 10 | 1083 | 5 | 1705 | 7 | 2800 |
| *Testing* | 1 | 40 | 6 | 184 | 82 | 312 |
| 2 | 51 | 18 | 172 | 70 | 311 |
| 3 | 74 | 11 | 179 | 47 | 311 |
| 4 | 95 | 13 | 177 | 26 | 311 |
| 5 | 58 | 21 | 169 | 63 | 311 |
| 6 | 65 | 10 | 180 | 56 | 311 |
| 7 | 119 | 10 | 180 | 2 | 311 |
| 8 | 96 | 0 | 190 | 25 | 311 |
| 96 | 8 | 182 | 25 | 311 | 96 |
| 53 | 11 | 179 | 68 | 311 | 53 |

**Lampiran 16.** *Confusion Matrix* Metode SVM Kernel RBF pada GoCar

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Data | *Fold* ke- | *True Positive* | *False Positive* | *True Negative* | *False Negative* | Jumlah *Tweet* |
| *Training* | 1 | 2225 | 7 | 1665 | 3 | 3900 |
| 2 | 2224 | 6 | 1666 | 4 | 3900 |
| 3 | 2224 | 7 | 1665 | 4 | 3900 |
| 4 | 2225 | 7 | 1665 | 3 | 3900 |
| 5 | 2225 | 7 | 1665 | 3 | 3900 |
| 6 | 2224 | 7 | 1665 | 4 | 3900 |
| 7 | 2225 | 6 | 1666 | 4 | 3901 |
| 8 | 2226 | 8 | 1664 | 3 | 3901 |
| 9 | 2225 | 7 | 1666 | 4 | 3902 |
| 10 | 2227 | 7 | 1666 | 2 | 3902 |
| *Testing* | 1 | 167 | 39 | 147 | 81 | 434 |
| 2 | 211 | 46 | 140 | 37 | 434 |
| 3 | 247 | 53 | 133 | 1 | 434 |
| 4 | 241 | 44 | 142 | 7 | 434 |
| 5 | 218 | 36 | 150 | 30 | 434 |
| 6 | 151 | 34 | 152 | 97 | 434 |
| 7 | 135 | 40 | 146 | 112 | 433 |
| 8 | 179 | 29 | 157 | 68 | 433 |
| 9 | 191 | 38 | 147 | 56 | 432 |
| 10 | 157 | 34 | 151 | 90 | 432 |

**Lampiran 17.** *Confusion Matrix* Metode SVM Kernel RBF pada GrabCar

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Data | *Fold* ke- | *True Positive* | *False Positive* | *True Negative* | *False Negative* | Jumlah *Tweet* |
| *Training* | 1 | 1207 | 3 | 845 | 1 | 2056 |
| 2 | 1207 | 4 | 844 | 1 | 2056 |
| 3 | 1208 | 4 | 844 | 0 | 2056 |
| 4 | 1208 | 4 | 845 | 1 | 2058 |
| 5 | 1208 | 3 | 846 | 1 | 2058 |
| 6 | 1209 | 3 | 846 | 0 | 2058 |
| 7 | 1208 | 4 | 845 | 1 | 2058 |
| 8 | 1208 | 4 | 845 | 1 | 2058 |
| 9 | 1208 | 3 | 846 | 1 | 2058 |
| 10 | 1208 | 4 | 845 | 1 | 2058 |

**Lampiran 17.** *Confusion Matrix* Metode SVM Kernel RBF pada GrabCar (Lanjutan)

|  |  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- | --- |
| Data | *Fold* ke- | *True Positive* | *False Positive* | *True Negative* | *False Negative* | Jumlah *Tweet* |
| *Testing* | 1 | 105 | 16 | 79 | 30 | 230 |
| 2 | 133 | 22 | 73 | 2 | 230 |
| 3 | 128 | 27 | 68 | 7 | 230 |
| 4 | 129 | 21 | 73 | 5 | 228 |
| 5 | 84 | 15 | 79 | 50 | 228 |
| 6 | 70 | 10 | 84 | 64 | 228 |
| 7 | 65 | 11 | 83 | 69 | 228 |
| 8 | 129 | 19 | 75 | 5 | 228 |
| 9 | 103 | 17 | 77 | 31 | 228 |
| 10 | 73 | 9 | 85 | 61 | 228 |

**Lampiran 18.** *Output* Persamaan *Hyperplane* BlueBird Taxi

Kernel used:

RBF Kernel: K(x,y) = exp(-1.0\*(x-y)^2)

Classifier for classes: 0, 1

BinarySMO

- 1.2321 \* < 0 0 0 0 0 0 0 0 0 0 ... 0 0 0> \* X]

- 1.3167 \* < 0 0 0 0 0 0 0 0 0 0 ... 0 0 0> \* X]

+ 10 \* < 0 0 0 0 0 0 0 0 0 0 ... 0 0 0> \* X]

+ 0.525 \* < 0 0 0 0 0 0 0 0 0 0 ... 0 0 0> \* X]

- 1.2677 \* < 0 0 0 0 0 0 0 0 0 0 ... 0 0 0> \* X]

- 10 \* < 0 0 0 0 0 0 0 0 0 0 ... 0 0 0> \* X]

- 10 \* < 0 0 0 0 0 0 0 0 0 0 ... 0 0 0> \* X]

.

.

.

- 10 \* < 0 0 0 0 0 0 0 0 0 0 ... 0 0 0> \* X]

+ 0.7883 \* < 0 0 0 0 0 0 0 0 0 0 ... 0 0 0> \* X]

+ 0.5974 \* < 0 0 0 0 0 0 0 0 0 0 ... 0 0 0> \* X]

+ 0.2219

Number of support vectors: 2766

**Lampiran 19.** *Output* Persamaan *Hyperplane* GoCar

Kernel used:

RBF Kernel: K(x,y) = exp(-1.0\*(x-y)^2)

Classifier for classes: 0, 1

BinarySMO

- 0.7201 \* < 0 0 0 0 0 0 0 0 0 0 ... 0 0 0> \* X]

- 0.9657 \* < 0 0 0 0 0 0 0 0 0 0 ... 0 0 0> \* X]

+ 1.1104 \* < 0 0 0 0 0 0 0 0.166626 0 ... 0 0 0> \* X]

- 0.8803 \* < 0 0 0 0 0 0 0 0 0 0 ... 0 0 0> \* X]

- 0.655 \* < 0 0 0 0 0 0 0 0 0 0 ... 0 0 0> \* X]

- 3.0518 \* < 0 0 0 0 0 0 0 0 0 0 ... 0 0 0> \* X]

+ 1.6109 \* < 0 0 0 0 0 0 0 0 0 0 ... 0 0 0> \* X]

.

.

.

- 0.8129 \* < 0 0 0 0 0 0 0 0 0 0 ... 0 0 0> \* X]

- 0.9078 \* < 0 0 0 0 0 0 0 0 0 0 ... 0 0 0> \* X]

- 0.8828 \* < 0 0 0 0 0 0 0 0 0 0 ... 0 0 0> \* X]

- 0.1029

Number of support vectors: 3883

**Lampiran 20.** *Output* Persamaan *Hyperplane* GrabCar

Kernel used:

RBF Kernel: K(x,y) = exp(-1.0\*(x-y)^2)

Classifier for classes: 0, 1

BinarySMO

1.1245 \* < 0 0 0 0 0 0 0 0 0 0 ... 0 0 0> \* X]

- 0.8394 \* < 0 0 0 0 0 0 0 0 0 0 ... 0 0 0> \* X]

+ 1.1816 \* < 0 0 0 0 0 0 0 0 0 0 ... 0 0 0> \* X]

- 9.7768 \* < 0 0 0 0 0 0 0 0 0 0 ... 0 0 0> \* X]

- 0.8789 \* < 0 0.459174 0 0 0 0 0 0 0 ... 0 0 0> \* X]

+ 1.1012 \* < 0 0 0 0 0 0 0 0 0 0 ... 0 0 0> \* X]

- 0.8851 \* < 0 0 0 0 0 0 0 0 0 0 ... 0 0 0> \* X]

.

.

.

**Lampiran 20.** *Output* Persamaan *Hyperplane* GrabCar (Lanjutan)

- 0.8646 \* < 0 0 0 0 0 0 0 0 0 0 ... 0 0 0> \* X]

+ 10 \* < 0 0 0 0 0 0 0 0 0 0 ... 0 0 0> \* X]

- 10 \* < 0 0 0 0 0 0 0 0 0 0 ... 0 0 0> \* X]

- 0.1097

Number of support vectors: 2044

**Lampiran 21.** *Syntax Word Cloud* Menggunakan Python 3.6

import numpy as np

import matplotlib as mpl

import matplotlib.pyplot as plt

%matplotlib inline

from subprocess import check\_output

from wordcloud import WordCloud, STOPWORDS

#mpl.rcParams['figure.figsize']=(8.0,6.0) #(6.0,4.0)

mpl.rcParams['font.size']=12 #10

mpl.rcParams['savefig.dpi']=100 #72

mpl.rcParams['figure.subplot.bottom']=.1

wordcloud = WordCloud(collocations = False,

background\_color='white',

stopwords=stopwords,

max\_words=1000,

max\_font\_size=200,

random\_state=43

).generate(str(df))

print(wordcloud)

fig = plt.figure(1)

plt.imshow(wordcloud)

plt.axis('off')

plt.show()

fig.savefig("D:/bluebird.png", dpi=900)

**Lampiran 22.** *Output Social Network Analysis* BlueBird Taxi

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| *ID* | *Username* | *Degree Centrality* | *Closeness Centrality* | *Betweenness Centrality* | *Eigenvector Centrality* |
| 1 | \_\_\_\_\_\_ems | 1 | 1 | 0 | 0 |
| 2 | \_\_aih | 0 | 0 | 0 | 0 |
| 3 | \_\_hyee13 | 0 | 0 | 0 | 0 |
| 4 | \_\_lattepapi | 1 | 0 | 0 | 0.001533 |
| 5 | \_\_lisna | 0 | 0 | 0 | 0 |
| . | . | . | . | . | . |
| . | . | . | . | . | . |
| 4619 | ZoneWarPuji | 4 | 0.8 | 4 | 0.001533 |
| 4620 | Zsinarf | 1 | 1 | 0 | 0 |
| 4621 | Zuhaald | 0 | 0 | 0 | 0 |
| 4622 | Zulhidayat\_AY | 0 | 0 | 0 | 0 |
| 4623 | zzy\_my | 0 | 0 | 0 | 0 |

**Lampiran 23.** *Output Social Network Analysis* GoCar

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| *ID* | *Username* | *Degree Centrality* | *Closeness Centrality* | *Betweenness Centrality* | *Eigenvector Centrality* |
| 1 | \_\_\_anka | 0 | 0 | 0 | 0 |
| 2 | \_\_\_rahayu | 1 | 1 | 0 | 0 |
| 3 | \_\_aih | 1 | 0 | 0 | 0.001079 |
| 4 | \_\_danar | 0 | 0 | 0 | 0 |
| 5 | \_\_gpp | 1 | 0 | 0 | 0.001079 |
| . | . | . | . | . | . |
| . | . | . | . | . | . |
| 9640 | zvlfq | 0 | 0 | 0 | 0 |
| 9641 | zwolf\_me | 1 | 1 | 0 | 0 |
| 9642 | zwolfuhr | 0 | 0 | 0 | 0 |
| 9643 | zxxxxaad | 0 | 0 | 0 | 0 |
| 9644 | ZyaUlhuda | 1 | 0 | 0 | 0.001079 |

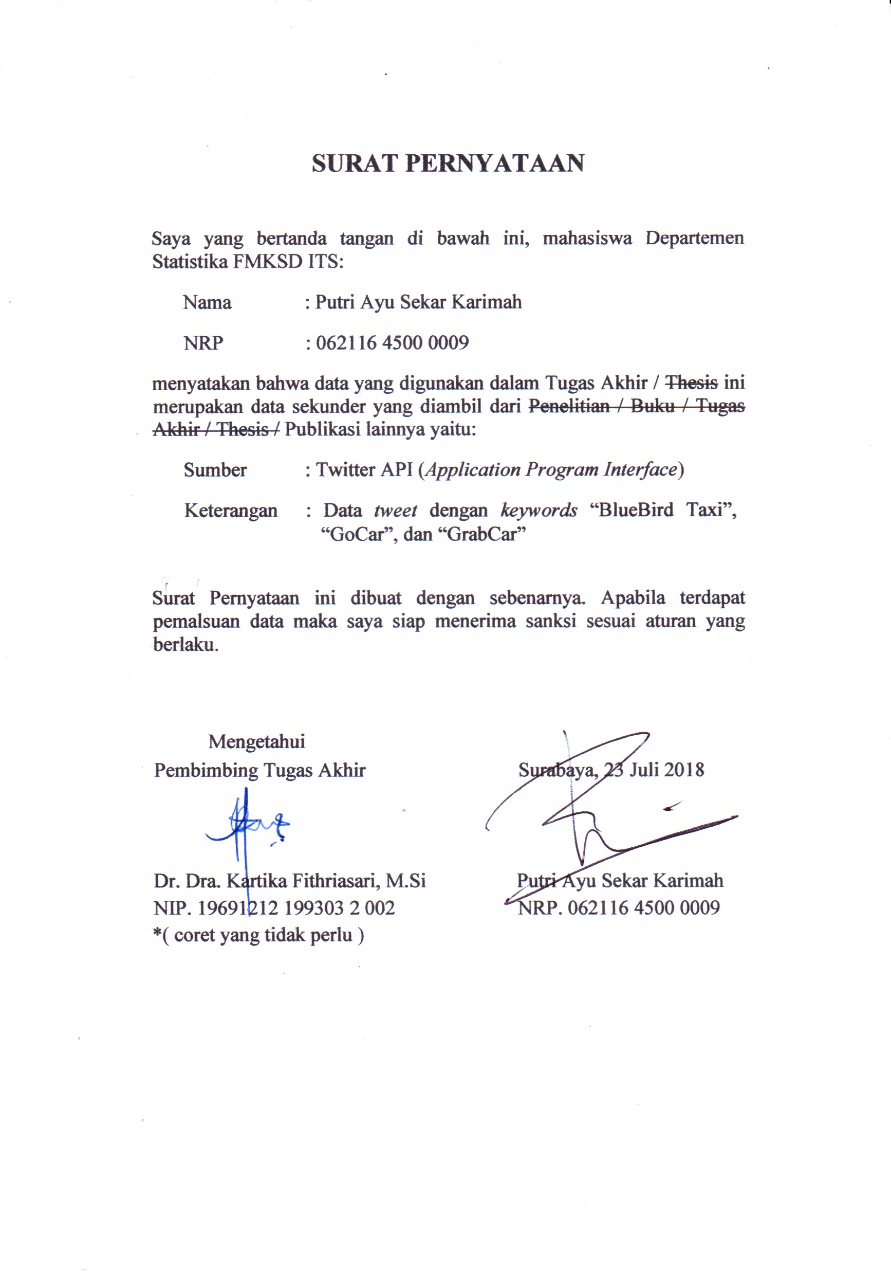
**Lampiran 24.** *Output Social Network Analysis* GrabCar

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| *ID* | *Username* | *Degree Centrality* | *Closeness Centrality* | *Betweenness Centrality* | *Eigenvector Centrality* |
| 1 | \_\_\_\_\_honeybuhni | 0 | 0 | 0 | 0 |
| 2 | \_\_\_Maaryam | 0 | 0 | 0 | 0 |
| 3 | \_\_\_rahayu | 1 | 1 | 0 | 0 |
| 4 | \_\_\_ridwnn | 1 | 0 | 0 | 0.001579 |
| 5 | \_\_aih | 1 | 0.666667 | 0 | 0 |
| . | . | . | . | . | . |
| . | . | . | . | . | . |
| 5824 | zyd\_zbr | 1 | 1 | 0 | 0 |
| 5825 | zyjmeanie | 1 | 0.490066 | 0 | 0 |
| 5826 | zz\_aa\_uu | 0 | 0 | 0 | 0 |
| 5827 | zzellano | 0 | 0 | 0 | 0 |
| 5828 | zzfrx | 0 | 0 | 0 | 0 |

**Lampiran 25.** *Output Social Network Analysis* Taksi Konvensional dan Taksi Berbasis *Online*

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| *ID* | *Username* | *Degree Centrality* | *Closeness Centrality* | *Betweenness Centrality* | *Eigenvector Centrality* |
| 1 | \_\_\_\_\_\_ems | 2 | 0.409174 | 0 | 0.038638 |
| 2 | \_\_\_\_\_honeybuhni | 0 | 0 | 0 | 0 |
| 3 | \_\_\_anka | 0 | 0 | 0 | 0 |
| 4 | \_\_\_Maaryam | 0 | 0 | 0 | 0 |
| 5 | \_\_\_rahayu | 1 | 1 | 0 | 0 |
| . | . | . | . | . | . |
| . | . | . | . | . | . |
| 18197 | zyjmeanie | 1 | 0.309942 | 0 | 0 |
| 18198 | zz\_aa\_uu | 0 | 0 | 0 | 0 |
| 18199 | zzellano | 0 | 0 | 0 | 0 |
| 18200 | zzfrx | 0 | 0 | 0 | 0 |
| 18201 | zzy\_my | 0 | 0 | 0 | 0 |

**Lampiran 26.** Surat Pernyataan Data



*(Halaman ini sengaja dikosongkan)*