



# Analisis Sentimen Masyarakat Terhadap Pilpres 2019 Berdasarkan Opini Dari *Twitter* Menggunakan Metode *Naive Bayes Classifier*

Khoirul Zuhri<sup>1</sup>, Nurul Adha Oktarini Saputri<sup>2</sup>

<sup>1</sup>Information System Departement, Bina Darma University, Palembang, Indonesia

<sup>2</sup>Informatics Departement, Bina Darma University, Palembang, Indonesia

Email: <sup>1</sup>khoirulzuhri014@gmail.com, <sup>2</sup>nuruladhaos@binadarma.ac.id

## Abstract

Twitter is a social media that is currently popular, where the public is free to comment and write anything. It is not uncommon for the public to comment with harsh words and even hate speech. The 2019 presidential election drew many comments, some praised, criticized and insulted. To be able to dig up information and classify a text, sentiment analysis is needed. In this study, sentiment analysis is a process of classifying textual documents into two classes, namely negative and positive sentiment classes. Opinion data were obtained from the Twitter social network in the form of tweets. The data used was 3337 tweets consisting of 80% training data and 20% training data. Training data is data with known sentiment. This study aims to determine whether a tweet is a positive or negative tweet conveyed on Twitter in Indonesian. The classification of tweet data uses the naïve Bayes classifier algorithm. The classification results of the test data show that the Naïve Bayes Classifier algorithm provides an accuracy value of 71%. The accuracy value for each sentiment is 71% for positive sentiment and 70% for negative sentiment.

**Keywords:** Sentiment Analysis, Classification, Naïve Bayes, Twitter, Pilpres 2019

## 1. PENDAHULUAN

Seiring dengan semakin majunya teknologi telekomunikasi, pengguna internet di dunia juga mengalami kemajuan yang cukup pesat. Hal ini dikarenakan semakin mudahnya aktifitas sehari-hari yang bisa dilakukan dengan memanfaatkan internet, baik itu hanya sekedar berkomunikasi melalui media sosial, mencari informasi secara *up to date* dan bahkan melakukan transaksi jual beli. Menurut data dari *Internet World Status*, Indonesia merupakan salah satu Negara dengan pengguna internet



terbesar di Dunia yang menempati posisi ke-5 dengan 143.260.000 pengguna. Perilaku pengguna internet lebih didominasi untuk mengakses media sosial seperti *Facebook*, *Twitter*, *Instagram*, *YouTube* sebesar 97,4% dari total pengguna internet.

Media sosial didefinisikan sebagai alat (*tools*) teknologi informasi *online* yang memungkinkan setiap pengguna melakukan komunikasi dengan mudah melalui internet dalam berbagi informasi seperti pesan teks, audio, video, gambar dan lain sebagainya [1]. Media sosial saat ini bukan hanya dipergunakan untuk sarana pertemanan saja, tapi sudah banyak untuk kegiatan lain, seperti pengenalan suatu produk ataupun untuk sarana masyarakat dalam mengomentari hasil pilpres 2019.

Pemilihan Presiden 2019 sudah selesai dilakukan, dan banyak menghasilkan bermacam-macam opini dari masyarakat terutama melalui media sosial *Twitter*. Opini melalui *tweet* inilah yang dapat dimanfaatkan untuk melihat bagaimana sentimen yang dimunculkan oleh masyarakat mengenai Pilpres 2019. Penentuan sentimen positif, negatif sebenarnya bisa dilakukan secara manual, tetapi karena data yang sangat banyak maka menjadi semakin banyak tentunya waktu dan usaha yang dibutuhkan untuk mengklasifikasikan opini-opini tersebut. Oleh sebab itu peneliti tertarik dalam menerapkan metode pembelajaran mesin untuk mengklasifikasikan opini dari data yang sangat banyak tersebut. Untuk melakukan klasifikasi data tersebut peneliti menggunakan metode *Naive Bayes Classifier*.

Metode *Naive Bayes Classifier* merupakan salah satu metode *machine learning* yang menggunakan perhitungan probabilitas. "Cara kerja dari metode *Naive Bayes Classifier* menggunakan perhitungan probabilitas". Konsep dasar yang digunakan oleh *Naive Bayes* adalah *Theorema Bayes*, yaitu *theorema* yang dipakai dalam statistik untuk menghitung suatu peluang, "*Bayes Optimal Classifier* menghitung peluang dari satu kelas dari masing-masing kelompok atribut yang ada", dan menentukan kelas mana yang paling optimal.

## 2. METODOLOGI PENELITIAN

### 2.1 Analisis Sentimen

Analisis sentimen adalah deteksi sikap-sikap (*attitudes*) terhadap objek atau orang [2]. Dari miliaran data cuitan di *twitter*, anda dapat melakukan analisis sentimen untuk menemukan berapa persentase

sentimen positif dan berapa persentase sentimen negatif terhadap seseorang, perusahaan, institusi, kelompok, atau sebuah situasi tertentu.

## 2.2 Metode Penelitian

Dalam penelitian ini peneliti menggunakan metode deskriptif (*Deskriptive Research*). Metode deskriptif adalah metode yang berfungsi untuk mendeskripsikan atau menggambarkan terhadap objek yang diteliti melalui data atau sampel yang telah terkumpul tanpa bermaksud membuat kesimpulan yang berlaku umum atau generalisasi [3].

## 2.3 Metode Pengumpulan Data

Peneliti menggunakan metode *crawling* untuk mengumpulkan data. *Crawling* adalah suatu teknik yang digunakan untuk mengumpulkan informasi yang ada dalam web. *Crawling* berkerja secara otomatis, dimana informasi yang dikumpulkan berdasarkan atas kata kunci yang diberikan oleh pengguna. Alat yang digunakan untuk *crawling* disebut *crawler*. *Crawler* berbentuk program yang diprogram dengan algoritma tertentu sehingga dapat melakukan pemindaian ke halaman-halaman *web*, sesuai dengan alamat *web* ataupun kata kunci yang diberikan pengguna. Peneliti menggunakan *Python* untuk *crawling* data *twitter* ini.

## 2.4 Metode Analisis

Dalam penelitian ini, peneliti menggunakan metode *Naive Bayes Classifier* untuk menganalisis data yang telah diambil dari *twitter*. *Naive Bayes* adalah metode terbaru yang digunakan untuk mengklasifikasikan sekumpulan dokumen. "Algoritma ini memanfaatkan metode probabilitas dan statistik yang dikemukakan oleh ilmuwan Inggris Thomas Bayes, yaitu memprediksi probabilitas di masa depan berdasarkan pengalaman di masa sebelumnya" [4]. Keuntungan penggunaan metode *Naive Bayes* adalah bahwa metode ini hanya membutuhkan jumlah data pelatihan (*Training Data*) yang kecil untuk menentukan estimasi parameter yang diperlukan dalam proses pengklasifikasian.

## 3. HASIL DAN PEMBAHASAN

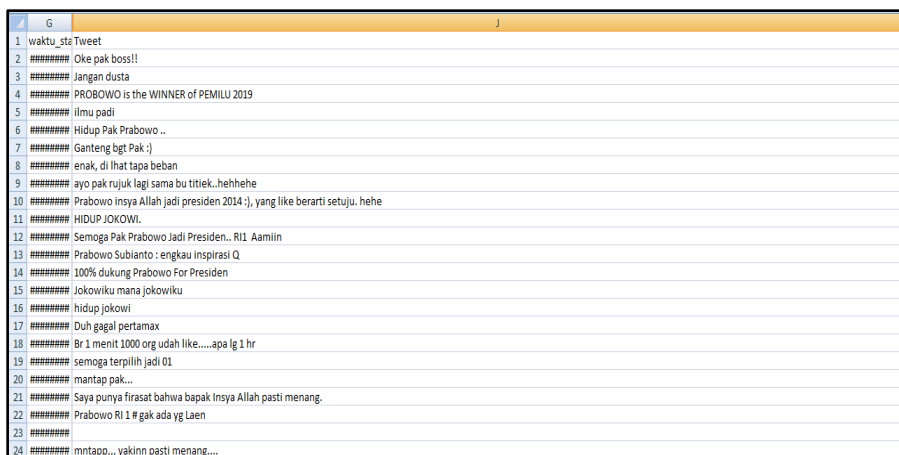
### 3.1 Perangkat

Adapun perangkat yang digunakan dalam melakukan penelitian ini adalah sebagai berikut

- a. Perangkat keras (*hardware*) yang digunakan adalah 1 buah *notebook* dengan spesifikasi *processor Intel(R) Celeron(R) CPU N2840 @ 2.16GHz*, dengan *RAM 2 GB* dan *HarDisk 500 GB*
- b. Perangkat lunak (*Software*) yang digunakan yaitu *Operating System Microsoft Windows 8.1* sebagai sistem operasi, *Microsoft Excel 2013*, *Microsoft Word 2013*, dan *Anaconda*.

## 3.2 Mengumpulkan Data Tweet

Pengumpulan data dilakukan dengan teknik *crawling*, Tahap awal untuk melakukan proses pengambilan data dari *Twitter*, peneliti harus memiliki *key number* dan *secret number* dari *API Key Twitter*””Mendapatkan *API Key Twitter* peneliti harus mendaftarkan atau melakukan pengajuan terhadap pihak *developer* atau pengembang *Twitter* untuk mendapatkan *API Key Twitter* tersebut. Hasil proses *crawling* data, menggunakan *API Key Twitter* maka didapatkan data dengan jumlah *3.337 record Tweet*, dengan ketentuan data latih sebesar **70%** dan data training sebesar **30%**.”



G	J
1	waktu_ste Tweet
2	##### Oke pak boss!!
3	##### Jangan dusta
4	##### PROBOWO is the WINNER of PEMILU 2019
5	##### Ilmu padi
6	##### Hidup Pak Prabowo ...
7	##### Ganteng bgt Pak :)
8	##### enak, di lihat tapa beban
9	##### ayo pak rujuk lagi sama bu titiek..hehhehe
10	##### Prabowo insya Allah jadi presiden 2014 :), yang like berarti setuju. hehe
11	##### HIDUP JOKOWI.
12	##### Semoga Pak Prabowo Jadi Presiden... RI1 Aamin
13	##### Prabowo Subianto : engkau inspirasi Q
14	##### 100% dukung Prabowo For Presiden
15	##### Jokowiku mana jokowiku
16	##### hidup jokowi
17	##### Duh gagal pertamax
18	##### Br 1 menit 1000 org udah like.....apa lg 1 hr
19	##### semoga terpilih jadi 01
20	##### mantap pak...
21	##### Saya punya firasat bahwa bapak Insya Allah pasti menang.
22	##### Prabowo RI 1 # gak ada yg Laen
23	#####
24	##### mntapp,,, yakinn pasti menang....

**Gambar 1** Hasil *Crawling*

Hasil dari proses *crawling* data di atas kemudian akan dilakukan proses *labeling* data, proses ini dilakukan untuk menklasifikasikan data dari hasil *tweet* yang telah *dicrawling* sebelumnya. Pada proses *labeling* data ini, data dibedakan menjadi 2 kelas, yaitu kelas positif, dan kelas negatif. berikut merupakan contoh proses *labeling* data yang dapat dilihat di bawah ini.

Tabel 1. Data Hasil *labeling*

<i>Tweet</i>	<i>Pembersihan Text</i>	<i>Label</i>
b'RT @hd_hijau : jokowi yesssssss #pilpres#jokowi#prabowo #pemilu#indonesia#prabowosandi #pileg	Jokowi yes	0
B'Prabowo insya Allah jadi presiden 2014 :), yang like berarti setuju. Hehe #pilpres2014#pemilu	prabowo insya allah jadi presiden 2019, yang like berarti setuju. hehe	0
B'@detikcom : Tolak prabowo jd presiden,masa lalunya yg penuh dosa,sampai2 mantan mertuanya bilang prabowo itu penghianat,jd para pendukung prabowo ga usah banyak omong lah #jokowi#prabowo#pilpres #jokowipresidenri	tolak prabowo jd presiden,masa lalunya yg penuh dosa,sampai mantan mertuanya bilang prabowo itu penghianat,jd para pendukung prabowo ga usah banyak omong lah	1
B'Moga Bapak Jokowi Menjadi Presiden NKRI,And Tentunya Tegas, Ramah, Dan Bertanggung jawab, :)#pilpres#jokowi	moga bapak jokowi menjadi presiden nkri, tentunya tegas, ramah, dan bertanggung jawab	0
b'@detikcom Kampret Sesumbar. HIDUP JOKOWI kampret\n#JokowiPresidenRI'	kampret sesumbar, pembunuh mau jadi presiden. hidup jokowi	1

Dalam kasus ini kelas negatif berlabelkan 1 menyatakan bahwa *tweet* tersebut mengandung unsur *hatespeech* atau kata-kata yang mengandung ujaran kebencian, sedangkan kelas positif berlabelkan 0 adalah kata-kata yang netral yang tidak mengandung unsur ujaran kebencian di dalamnya.

### 3.3 Preprocessing Data

*Preprocessing* merupakan proses untuk menghilangkan kata-kata yang tidak diperlukan dalam analisis data ataupun kata-kata yang tidak memiliki arti.

*a. Cleaning*

*Cleaning* merupakan proses untuk menghapus tanda baca, simbol, huruf kapital maupun bilangan angka yang muncul pada *tweet* pengguna *Twitter*, yang menyebabkan data tersebut menjadi data yang kurang efektif untuk dilakukan analisa. Penerapan proses *cleanning* dapat dilihat pada Tabel 2 berikut ini

**Tabel 2.** Data Hasil *Cleaning*.

<b><i>Tweet Sebelum Cleaning</i></b>	<b><i>Tweet Sesudah Cleaning</i></b>
b'RT @hd_hijau : jokowi yesssssss #pilpres#jokowi#prabowo #pemilu#indonesia#prabowosandi #pileg	jokowi yes
B'Prabowo insya Allah jadi presiden 2014 :), yang like brarti setuju. Hehe #pilpres2014#pemilu	prabowo insya allah jadi presiden 2019, yang like brarti setuju. hehe
B'@detikcom : Tolak prabowo jd presiden,masa lalunya yg penuh dosa,sampai2 mantan mertuanya bilang prabowo itu penghianat,jd para pendukung prabowo ga usah banyak omong lah #jokowi#prabowo#pilpres #jokowipresidenri	tolak prabowo jd presiden,masa lalunya yg penuh dosa,sampai mantan mertuanya bilang prabowo itu penghianat,jd para pendukung prabowo ga usah banyak omong lah
B'Moga Bapak Jokowi Menjadi Presiden NKRI,And Tentunya Tegas, Ramah, Dan Bertanggung jawab, :) #pilpres#jokowi	moga bapak jokowi menjadi presiden nkri, tentunya tegas, ramah, dan bertanggung jawab
b'@detikcom Kampret Sesumbar. HIDUP JOKOWI kampret\n#JokowiPresidenRI'	kampret sesumbar, pembunuh mau jadi presiden. hidup jokowi

*b. Remove Stop word*

*Remove Stopword* merupakan proses untuk menghapus kata - kata yang tidak memiliki makna ataupun arti seperti kata dan, atau, dengan, dan lain-lain. Penerapan proses *Remove Stopword* dapat dilihat pada Tabel 3 berikut ini.

**Tabel 3.** Data Hasil *Remove Stop word*

<b><i>Teweeet Se belum Remove Stop</i></b>	<b><i>Tweet Sesudah Remove Stopword</i></b>
--------------------------------------------	---------------------------------------------

**word**

jokowi yes	jokowi yes
prabowo insya allah jadi presiden 2019, yang like brarti setuju. hehe	prabowo insya allah presiden 2019, brarti setuju.
tolak prabowo jd presiden, masa lalunya yg penuh dosa, sampai mantan mertuanya bilang prabowo itu penghianat, jd para pendukung prabowo ga usah banyak omong lah	tolak prabowo presiden, masa penuh dosa, sampai mantan mertuanya bilang prabowo penghianat, para pendukung prabowo ga usah banyak omong
moga bapak jokowi menjadi presiden nkri, tentunya tegas, ramah, dan bertanggung jawab	bapak jokowi presiden nkri, tegas, ramah, bertanggung jawab
kampret sesumbar, pembunuh mau jadi presiden. hidup jokowi	kampret sesumbar, pembunuh presiden. hidup jokowi

*c. Tokenization*

*Tokenization* merupakan proses untuk memisahkan kalimat kedalam bagian-bagian kecil yang disebut token. Suatu *token* bisa dianggap menjadi satu bentuk kata, frasa, dan atau suatu element yang mengandung arti. Proses *tokenization* dapat dilihat pada Tabel 4 berikut ini.

**Tabel 4.** Data Hasil *Tokenization*

<b>Tweet Sebelum Tokenization</b>	<b>Tweet Sesudah Tokenization</b>
jokowi yes	['jokowi', 'yes']
prabowo insya allah presiden 2019, like berarti setuju.	['prabowo', 'insya', 'allah', 'presiden', '2019', 'berarti', 'setuju']
tolak prabowo presiden, masa penuh dosa, sampai mantan mertuanya bilang prabowo penghianat, para pendukung prabowo ga usah banyak omong	['tolak', 'prabowo', 'presiden', 'masa', 'penuh', 'dosa', 'sampai', 'mantan', 'mertuanya', 'bilang', 'prabowo', 'penghianat', 'para', 'pendukung', 'prabowo', 'banyak', 'omong']
bapak jokowi presiden nkri, tegas, ramah, bertanggung jawab	['bapak', 'jokowi', 'presiden', 'nkri', 'tegas', 'ramah', 'bertanggung', 'jawab']
kampret sesumbar, pembunuh presiden. hidup jokowi	['kampret', 'sesumbar', 'pembunuh', 'presiden', 'hidup', 'jokowi']

#### d. Stemming

*Stemming* merupakan proses untuk merubah suatu kata kedalam bentuk dasar dari kata tersebut dengan menghapus imbuhan katanya baik di depan ataupun di belakang kata. Proses pada tahap *stemming* dapat dilihat pada Tabel 5 di bawah ini.

**Tabel 5.** Tabel Hasil *Stemming*

<b>Tweet Sebelum Stemming</b>	<b>Tweet Sesudah Stemming</b>
['Jokowi', 'yes']	jokowi yes
['prabowo', 'insya', 'allah', 'presiden', '2019', 'like', 'berarti', 'setuju'].	prabowo insya allah presiden 2019 berarti setuju.
['tolak', 'prabowo', 'presiden', 'masa', 'penuh', 'dosa', 'sampai', 'mantan', 'mertuanya', 'bilang', 'prabowo', 'penghianat', 'para', 'pendukung', 'prabowo', 'banyak', 'omong']	tolak prabowo presiden masa penuh dosa sampai mantan mertua bilang prabowo khianat para dukung prabowo banyak omong
['bapak', 'jokowi', 'presiden', 'nkri', 'tegas', 'ramah', 'bertanggung', 'jawab']	bapak jokowi presiden nkri tegas ramah tanggung jawab
['kampret', 'sesumbar', 'pembunuh', 'presiden', 'hidup', 'jokowi']	kampret sesumbar bunuh presiden hidup jokowi

### 3.4 Ekstraksi Fitur

Pada proses ekstraksi fitur, proses pertama yang dilakukan sistem setelah *tokenization* adalah merubah dataset menjadi suatu representasi *vektor*.

(DTweet1) "Jokowi Yes"

(DTweet2) "Prabowo Insya Allah Presiden 2019 Berarti Setuju"

(DTweet3) "Kampret Sesumbar Bunuh Presiden Hidup Jokowi"

Setelah sistem melakukan proses *preprocessing*, terdapat 41 kata baku dari 3 kalimat di atas yaitu "Presiden", "Setuju", "Hidup", dan "Bunuh".

Setelah tahapan di atas dari setiap dokumen ditampilkan menjadi sebuah *vektor* dengan elemen, ketika kata tersebut terdapat di dalam dokumen maka diberikan nilai 1, jika tidak ada maka diberikan nilai 0.

**Tabel 6.** Pembuatan *Word Vector*

	Presiden	Setuju	Hidup	Bunuh
<i>DTweet1</i>	1	1	0	0
<i>DTweet2</i>	0	0	2	1
<i>DTweet3</i>	1	0	0	1

Dokumen yang telah dirubah menjadi *word vector* maka selanjutnya akan dilakukan perhitungan menggunakan rumus *TF-IDF*, menggunakan rumus ini maka akan menghasilkan *word vector* yang memiliki nilai sudah terbobot. Berikut merupakan proses perhitungan *TF-IDF*

**Tabel 7.** Perhitungan *TF (Term Frequency)*

	<i>DTweet1</i>	<i>DTweet2</i>	<i>DTweet3</i>
Presiden	1	0	1
Setuju	1	0	0
Hidup	0	2	0
Bunuh	0	1	1

Setelah proses perhitungan menentukan bobot *TF* selesai, maka dilanjutkan dengan proses menentukan *DF (Document Frequency)* dengan cara menghitung banyaknya *term (t)* yang muncul dalam suatu dokumen. Berikut adalah hasil dari proses tersebut.

**Table 8.** Perhitungan *DF (Document Frequency)*

<i>T (Term)</i>	<i>DF (Documeent Frequency)</i>
Presiden	2
Setuju	1
Hidup	2
Bunuh	2

Selanjutnya dilakukan perhitungan nilai *IDF (Inverse Document Frequency)* yaitu dengan cara melakukan perhitungan terhadap nilai dari log hasil D yang merupakan jumlah *document*. Contoh dalam kasus ini terdapat 3 *tweet*, dari 3 *tweet* tersebut dibagi dengan hasil nilai *DF*. Berikut adalah hasil dari perhitungan *IDF*

**Tabel 9.** Proses *IDF (Inverse Document Frequency)*

<b>T (Term)</b>	<b><i>DF (Document Frequency)</i></b>	<b><i>D/DF (Inverse Document Frequency)</i></b>
---------------------	-------------------------------------------	---------------------------------------------------------

Presiden	2	1.5	$\log 1,5 = 0,176$
Setuju	1	3	$\log 3 = 0,477$
Hidup	2	1.5	$\log 1,5 = 0,176$
Bunuh	2	1.5	$\log 1,5 = 0,176$

Setelah mendapatkan nilai *IDF*, maka selanjutnya dilakukan perhitungan *TF-IDF* seperti dibawah ini

**Tabel 10.** Proses Perhitungan *TF-IDF*

Q	TF			DF	D/DF	IDF	IDF+1	Dtweet1	Dtweet2	Dtweet3
	Dtweet1	Dtweet2	Dtweet3							
Presiden	1	0	1	2	1.5	0.477	1.176	1.176	0	1.176
Setuju	1	0	0	1	3	0.176	1.477	1.477	0	0
Hidup	0	2	0	2	1.5	1.176	0	0	2.35	0
Bunuh	0	1	1	2	1.5	0.176	1.176	0	1.176	1.176
								2.653	3.528	2.352

Hasil yang didapat dari *word vector* yang telah dilakukan pembobotan dapat dilihat pada Tabel 11 di bawah ini.

**Tabel 11.** Word Vector yang Sudah Dibobotkan

	Presiden	Setuju	Hidup	Bunuh
DTweet1	1.176	1.477	0	0
DTweet2	0	0	2.352	1.176
DTweet3	1.176	0	0	1.176

### 3.5 Klasifikasi Naive Bayes Pada Python

Pada proses ekstraksi fitur dan proses pengklasifikasian *Naive Bayes* yang nantinya akan di kompres menjadi satu kelas *pipeline vectorizer kemudian transformer kemudian classifier*. Untuk melakukan proses klasifikasi tersebut dijalankan menggunakan bantuan *library* pada bahasa pemrograman *Python* yaitu *library scikit-learn* untuk proses pengklasifikasian, selain itu digunakan juga *library numpy* dan *library pandas* untuk pembacaan data. Untuk *library scikit-learn* disini yang digunakan yaitu *Pipeline, Naive Bay es, CountVectorizer, Multinomia INB, Tfidf Transformer, f1 Score, dan Confusion Matrix*.

Tahap awal yang perlu dilakukan untuk proses ekstraksi fitur dan klasifikasi yaitu melakukan proses pengistalan *library* yang dibutuhkan. Selanjutnya setelah semua *library terinstall*, maka dilanjutkan dengan pendeklarasian semua *library* yang akan digunakan di dalam program

*python*. Berikut adalah kode program untuk deklarasi library di *python* yang ditunjukkan pada Gambar 2 di bawah ini.

```
import pandas as pd
import numpy as np
from sklearn.naive_bayes import MultinomialNB
from sklearn.svm import LinearSVC, SVC
from sklearn.feature_extraction.text import CountVectorizer
from sklearn.feature_extraction.text import TfidfTransformer
from sklearn.pipeline import Pipeline
from sklearn.model_selection import train_test_split
from sklearn.model_selection import cross_val_score
from sklearn.metrics import classification_report, confusion_matrix, accuracy_score, f1_score, precision_score, recall_score
```

**Gambar 2.** Fungsi pendeklarasi an *library*

Setelah *library* selesai dideklarasikan maka dilanjutkan dengan proses pengambilan *dataset* yang akan digunakan sebagai data *training* yang dilakukan menggunakan *library pandas*. Berikut adalah kode program dapat dilihat pada Gambar 3 di bawah ini.

```
data = pd.read_excel('analissentimenpilpres2019.CSV', encoding='Latin-1')
len(data)
```

**Gambar 3.** Fungsi Memang gil *Data Set*

Selanjutnya yaitu proses dalam membuat *class pipeline* yang didalamnya terdapat 3 tahapan yaitu mengubah *dataset* hasil dari *crawling data Twitter* menjadi *representasi vector* yaitu mengubah huruf menjadi angka dengan bantuan *library Count Vectorizer* serta dengan pembobotan menggunakan *word vector* didalam *library Tfidf Transformer*, kemudian dilakukan pengklasifikasian dengan menggunakan bantuan *library Multinomial NaiveBayes*. Berikut adalah proses dari pengimplementasian ketiga proses pembuatan *class pipeline* dapat dilihat pada Gambar 4 di bawah ini.

```
pipeline_mnb = Pipeline([
    ('vect', CountVectorizer()),
    ('tfidf', TfidfTransformer(use_idf=True, smooth_idf=True)),
    ('clf', MultinomialNB(alpha=1))
])

txt = data['cleantext'].values.astype('U')
#X_train, X_test, y_train, y_test = train_test_split(data['cleantext'], data['label'], test_size=0.33, random_state = 0)
X_train, X_test, y_train, y_test = train_test_split(txt, data['label'], test_size=0.33, random_state = 0)
pipeline_mnb.fit(X_train, y_train)
```

**Gambar 4.** Proses Pengimplementasian *Class Pipeline*

### 3.6 Uji Model

Peneliti melakukan uji model supaya dapat mengetahui tingkat dari performa *Algoritma Naive Bayess*. Nantinya hasil dari kasifikasi akan ditampilkan kedalam bentuk *coun fusion matrix*". Tabel yang ditampilkan dalam *con fusion matrix* tersebut terdiri dari class *predicted* dan juga class *actual*. Berikut adalah Model *con fusion matrix* seperti terlihat pada Ta bel 12 dibawah ini.

**Tabel 12.** Model *Confusion Matrix*

		Pre dict Class	
		C lass A	C lass B
Actual Class	Positif	TP	FP
	Negatif	FN	TN

Untuk mendapatkan nilai dari akurasi model diperoleh dengan hasil perhitungan banyaknya data dari hasil klarifikasi dibagi dengan total data, berikut adalah rumus untuk mencari akurasi.

$$Akurasi = \frac{TP + TN}{TP + FP + TN + FN} \quad (1)$$

Pada saat proses pengujian model dilakukan, maka hasilnya adalah nilai akurasi dan *confusion matrix* 2x2 sebesar 71.0% yang proses perhitungannya didasarkan pada jumlah nilai dari diagonal *con fusion matrix* dibagi dengan keseluruhan jumlah data. Karena menyangkut dari jumlah data yang ada di setiap kelas data *training* tidak seimbang, maka besarnya dari nilai akurasi tidak menjadi hal yang paling penting.

### 3.7. Evaluasi Model

Proses dari evaluasi model di lakukan setelah selesai melakukan uji model. Evaluasi model berfungsi untuk mengetahui atau menghitung performasi dari metode yang digunakan. Pada proses ini akan menghasilkan *confusion matrix* dengan ukuran 2x2.

**Tabel 13.** Hasil *Confusion Matrix*

		Predict Class	
		Positif	Negatif
Actual Class	Positif	558	105
	Negatif	261	334

Seperti yang terlihat pada Table 13 diatas, matriks yang ber ukuran 2x2 disetiap kolomnya mewakili nilai dari setiap dari kelas yaitu kelas positif, dan juga kelas negatif. Nilai presisi pada keseluruhan sistem adalah sebesar **0.704** sementara itu untuk nilai *recall* keseluruhan sistem sebesar **0.615** dan selanjutnya untuk nilai *f-1 Score* untuk pengevaluasian yang dihitung dengan mengombinasikan nilai dari *precision recall* yaitu didapatkan nilai sebesar **0.65.6**.

Dengan diketahuinya nilai kinerja dari *recall*, *precision*, dan juga *f-1 Score* dikeseluruhan *sistem*, maka peneliti dapat mengetahui dari kemampuan sistem tersebut untuk mencari kebenaran atau ketepatan dari setiap informassi yang diminta oleh pengguna dengan hasil dari jawaban yang dikeluarkan oleh sistem dengan tingkat keberhasilan sistem untuk menentukan kembali informasi yang dibeikan dengan *accuracy* nilai sebesar **71 %**.

Setelah semua proses diatas selesai dilakukan, untuk performa dari metode pengklasifikasian dari seitiap kelas bisa di ketahui dengan *recall*, *precision*, dan juga *f-1 Score* disetiap kelasnya. Hasil dari *recall*, *precision*, dan juga *f-1 Score* memiliki penilaian sebesar 0 sampai 1. Semakin tinggi dari nilai yang dihasilkan maka itu berarti semakin baik sistem bekerja, dalam arti semakin nilai mendekati angka 1 maka itu artinya sistem semakin baik. Untuk hasil yang didapat dari proses pengevaluasian model dapat dilihat pada Gambar 5 di bawah ini.

```
Accuracy: 0.7103
Confusion matrix:
[[358 96]
 [143 228]]
```

	precision	recall	f1-score	support
0	0.71	0.79	0.75	454
1	0.70	0.61	0.66	371
avg / total	0.71	0.71	0.71	825

**Gambar 5.** Hasil dari Proses Pengevaluasian Model

Hasil yang didapat dari nilai *recall*, *precision*, dan juga *f-1 Score* di setiap kelas bisa dilihat dari Tabel 14 berikut ini.

**Tabel 14.** Hasil dari Nilai *Recall*, *Precision*, dan *F-1 Score*

Klasifikasi	<i>Precision</i>	<i>Recall</i>	<i>f-1 Score</i>
Positif	0.71	0.79	0.75
Negatif	0.71	0.61	0.66

Dari hasil evaluasi model pada Tabel 14 dapat diketahui nilai precision dan recall dari setiap kelas dapat dilihat tingkat kemampuan pemrosesan sistem dalam mencari tingkat ketepatan antara informasi yang diinginkan oleh pengguna sebagai kelas positif adalah “71%”, dan untuk kelas negatif adalah “70%”. Tingkat keberhasilan dari pemrosesan sistem dalam memperoleh kembali informasi kelas positif adalah “79%”, untuk kelas negatif adalah “61%”. Berdasarkan hasil *Tweet* dengan data 3.337, dengan ketentuan data latih sebesar 70% dan data *training* sebesar 30%. Dengan hasil yang didapat peneliti menyimpulkan dalam analisis sentiment pemilihan presiden 2019 nilai positif lebih tinggi dengan selisih sebesar 18% dengan tingkat akurasi sangat tinggi yaitu sebesar 71%. Dengan hasil nilai tersebut dapat diartikan bahwa keberhasilan sistem untuk menentukan kembali sebuah informasi yang mengandung nilai positif dan negatif dalam suatu dokumen sangat tinggi.

#### 4. KESIMPULAN

Berdasarkan penelitian yang dilakukan dan hasil pembahasan yang diuraikan di atas, maka dapat ditarik kesimpulan sebagai berikut:

1. Metode *Naive Bayes* bisa digunakan untuk mengklasifikasikan data berupa teks, terutama teks yang berasal dari *tweet Twitter*.
2. Jumlah kata dalam setiap kelas *training* sangat mempengaruhi hasil dari klasifikasi pada data *testing*, oleh karena itu keseimbangan dari data perlu dijaga.
3. Kosakata tidak baku dapat mempengaruhi hasil klasifikasi terhadap suatu kelas *testing* apabila suatu kelas *training* memiliki data dari jumlah kata tidak baku lebih banyak dibanding kelas *training* yang lain.

#### DAFTAR PUSTAKA

- [1] Odewole. (2017) The Role of Librarian in Using Social Media Tools to Promote the Research Output of HIS/HER Clienteles. *Journal of Education and Practice*, 8 (27), 109-113.
- [2] Beineke, P., Hastie, T., Manning, C., & Vaithyanathan, S. 2004. Exploring Sentiment Summarization. In Y. Qu, J. Shanahan, & J.

- Wiebe (eds) Proceedings of the {AAAI} Spring Symposium on Exploring Attitude and Affect in Text: Theories and Applications, AAAI Press.
- [3] Sugiono (2016). Metode Penelitian Kuantitatif, Kualitatif, dan R&D. Alfabeta. Bandung.
- [4] Dr. Ghayda A. Al-Talib<sup>1</sup>, Hind S. Hassan, A . (2013). Study on Analysis of SMS Classification Using TF-IDF Weighting