JURNAL CAHAYA MANDALIKA

P-ISSN: 2828-495X
E-ISSN: 2721-4796

PERBANDINGAN FEATURE EXTRACTION TF-IDF DAN BOW UNTUK ANALISIS SENTIMEN BERBASIS SVM

Kurniawan Tri Putra[1], Mokhamad Amin Hariyadi[2], Cahyo Crysdian[3]
[1,2,3]Jurusan Magister Informatika, Fakultas Sains dan Teknologi, Universitas Islam Malik Maulana Ibrahim, Malang

email: [1]200605210019@student.uin-malang.ac.id,
[2]adyt2002@uin-malang.ac.id, [3]cahyo@ti.uin-malang.ac.id

Kata kunci:
Big Data, Analisis sentimen, TF-IDF, BOW, SVM

ABSTRAK

Dengan adanya transformasi society 5.0 pegaruh paling besar yang bisa dirasakan saat ini adalah berkembang pesatnya jumlah data yang ada di seluruh dunia baik yang bermanfaat secara langsung maupun data yang tidak bermanfaat secara langsung atau dikenal dengan istilah big data, dengan adanya sumber big data tersebut banyak peneliti-peneliti yang memanfaatkanya menjadi suatu data yang berharga dan berguna jika diproses dan diolah dengan cara yang baik dan benar salah satunya adalah dengan tujuan analisis sentimen. Pada permasalahan yang ada penelitian ini bertujuan untuk mencari dan mendapatkan alur dan teknik yang benar serta memiliki hasil optimal pada pengolahan data teks dengan tujuan analisis sentimen dengan membandingakan penerapan TF-IDF dan BOW yang menggunakan metode SVM. Pada penelitian analisis sentimen menggunakan data teks bersumber dari aplikasi media social twitter hasil yang didapatkan adalah pada penerapan teknik TF-IDF yang dipadukan dengan metode SVM memiliki hasil yang lebih baik dengan nilai Accuracy 86%, Precission 85%, Recall 85% dan F1-Score 85% sedangkan penerapan teknik BOW yang dipadukan metode SVM hanya unggul pada nilai Recall sebesar 89%.

Keywords:
Big Data, Sentiment Analysis, TF-IDF, BOW, SVM

ABSTRACT

With the transformation of Society 5.0, the biggest impact that can be felt today is the rapid growth of data worldwide, both beneficial and non-beneficial, commonly known as big data. With the existence of big data sources, many researchers utilize it to become valuable and useful data if processed and analyzed properly, one of which is for sentiment analysis purposes. This research aims to find and obtain the correct flow and techniques that have optimal results in text data processing for sentiment analysis by comparing the application of TF-IDF and BOW using the SVM method. In sentiment analysis research using text data sourced from Twitter social media applications, the results show that the application of the TF-IDF technique combined with the SVM method has better results with an Accuracy value of 86%, Precision 85%, Recall 85%, and F1-Score 85%, while the application of the BOW technique combined with the SVM method only excels in the Recall value of 89%..

PENDAHULUAN

Seiring dengan pesatnya pengaruh perkembangan zaman hingga saat ini yang dikenal dengan istilah society 5.0 atau dengan pengertian semua umat manusia menggunakan ilmu pengetahuan berbasis teknologi modern untuk memenuhi kebutuhan dan mempermudah segala pekerjaan manusia. Dengan adanya transformasi tersebut salah satu pengaruh paling besar yang bisa dirasakan adalah dalam bidang big data, karena jumlah pengguna media sosial yang sudah terlalu banyak untuk memenuhi segala kebutuhan seperti sebagai sarana berkomunikasi, berjualan,

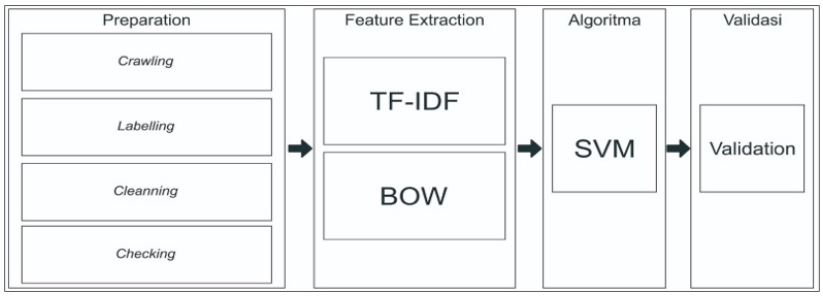
bertukar informasi bahkan sebagai tempat untuk bercerita dan meluapkan apa yang sedang dirasakannya. Dengan adanya sumber big data tersebut banyak penelitian-penelitian yang memanfaatkan data untuk diproses dan menghasilkan sesuatu yang lebih bermanfaat lagi, salah satunya dengan memanfaatkan machine learning untuk mengolahnya dengan hasil analisis sentimen.

Analisis sentimen pada pemanfataan big data ini dengan cara kerja mencari dan mengolah data dari pengguna aplikasi media sosial dengan data berbentuk teks yang dituangkan berdasarkan persaaan maupun ekpresi yang sedang atau telah dirasakan seseorang tersebut terhadap objek yang sedang dibicangkan. Menurut [1] salah satu media sosial yang paling banyak digunakan oleh masyarakat Indonesia saat ini adalah media sosial twitter dengan jumlah persentase 56% setelah youtube, whatapp, faceebook dan Instagram. Kemudian untuk penelitian analisis sentimen dengan data yang bersumber dari aplikasi media social twitter juga sedang ramai dilakukan dan sudah benyak diimplementasikan khususnya di Indonesia [2][3][4][5], penelitian-penelitian tersebut dibuat dengan berbagai tujuan untuk mengoptimalkan cara kerja dan kinerja dari program analisis sentimen menggunakan machine learning, dengan cara mencari dan membandingkan beberapa metode yang terbaik saja tanpa melihat aspek lain yang juga sangat mempengaruhi hasil kinerja dari machine leraning tersebut seperti pada proses feature extraxtion.

Berdasarkan pada permasalahan yang telah dibahas beberapa teknik yang paling umum dan sering digunakan dalam feature extraction yaitu TF-IDF dan BOW, dikarenakan kedua teknik tersebut sangat bersaing serta berperan baik dan sama-sama digunakan untuk merepresentasikan numerik dari data teks serta memiliki kekurangan dan kelebihan masing masing. Pada penelitian kali ini akan membandingkan kedua metode tersebut yan dipadukan dengan menggunakan metode SVM (Support Vector Machine) dengan tujuan untuk mendapatkan kinerja machine learning yang optimal dalam mengolah data teks utuk analisis sentiment dengan data bersumber dari media social twitter pada studi kasus pelayanan jne di masyarakat.

METODE

Pada penelitian analisis sentimen menggunakan machine learning ini mempunyai rancangan atau alur dari beberapa tahapan yaitu mulai dari tahapan pengumpulan data, pelabelan data, preprocessing data, feature extraction data, klasifikasi data, evaluasi dan pengujian data, perbandingan hasil pengujian dan kesimpulan.



Gambar 1. Alur penelitian

Dari diagram alir diatas menunjukan beberapa langkah untuk melakukan proses analisis sentiment menggunakan machine learning dengan penjelasan berikut:

A. Preparation

Dalam fase ini adalah proses pengumpulan data mentah dari twitter yang diolah menjadi data siap di proses oleh mesin dengan beberapa tahap yang harus dilalui:

1. Crawling

Proses penambangan data atau pengambilan data yang bersumber dati aplikasi social media bernama twitter yang dilakukan dengan bantuan google colab dari penggunaan bahasa pemrograman python, data yang diambil berupa teks dari unggahan tweet para pengguna dengan mengelompokan berdasarkan topik yang akan diangkat yaitu JNE yang dikelompokkan berdasarkan Hastag (#jne) dengan data berjumlah 300 tweet.

2. Labelling

Proses labeling ini merupakan suatu pelabelan data yang sudah didapatkan dari aplikasi twitter dengan data berjumlah 300 tweet, pada fase ini dilakukan secara manual oleh orang yang berkompeten bidang hukum dan perundang-undangan. Pelabelan ini menggunakan dua kategori saja yaitu jika komentar tweet pada satu kalimat mengandung makna atau konotasi yang jelek maka akan diberi label negatif, begitupun sebaliknya jika makna yang baik akan diberi label positif.

3. Cleaning

Proses pembersihan data dari segala bentuk karakter pada kata yang tidak berguna dan tidak memiliki manfaat untuk menghasilkan sentimen maka akan dibuang dan dibersihkan Langkah pada fase ini yaitu case folding atau menyetarakan menjadi huruf kecil, remove punctuation atau penghilangan url, karakter, angka dll pada setiap kalimat, stopword removal atau penghilangan kata hubung seperti di, ke, yang dll dan terakhir adalah stemming yaitu proses menghilangkan kata imbuhan seperti “memakan” jadi “makan”.

4. Data checking

Proses persiapan data terakhir adalah data checking yaitu pengecekan data setelah melalui tahap labeling dan cleaning dengan tujuan tidak terdapat data kosong atau blank data, tidak terdapat duplicate data atau data yang ganda serta imbalance data yaitu jumlah data yang seimbang antara kedua label positif maupun negatif.

B. Feature Extraction

Dalam fase ini adalah proses pengekstrakan kata menjadi sebuah angka berupa vektor, dikarenakan komputer hanya dapat mengenali dan memproses angka bukan sebuah kata dengan catatan masih mempunyai arti dari kata terbut.

1. TF-IDF (*Term Frequency - Inverse Document Frequency*)

Merupakan proses perhitungan atau pengekstrakan kata menjadi sebuah angka berbentuk vektor yang digunakan untuk menentukan bobot dari sebuah kata dalam sebuah dokumen atau korpus. Bobot ini berguna untuk menentukan seberapa penting kata tersebut dalam sebuah dokumen [12]. Pada dasarnya untuk perhitungan atau rumus TF-IDF terbagi menjadi dua yaitu TF

(*Term Frequency*) dan IDF (*Inverse Document Frequency*) dengan rumus dan cara kerja yang berbeda dan akan digabungkan di akhir perhitungan antara TF dan IDF sebagai berikut:

a. TF (*Term Frequency*)

Dengan cara kerja menghitung frekuensi jumlah kemunculan kata pada sebuah dokumen. Dalam setiap dokumen memiliki panajang yang berbeda-beda maka nilai TF akan dibagi dengan panjang dokumen:

$$tf_{t,d} = \frac{n_{t,d}}{(Total\ number\ of\ term\ in\ document)} \quad (1)$$

Keterangan

Tf = frekuensi kemunculan kata pada sebuah dokumen

b. IDF (*Inverse Document Frequency*)

Setelah berhasil menghitung nilai TF kita akan menghitung nilai IDF yang merupakan nilai untuk mengukur seberapa penting sebuah kata, dengan beracuan semakin kecil nilai IDF maka akan dianggap semakin tidak penting kata tersebut, begitupun sebaliknya:

$$idf_d = log\frac{Number\ of\ document}{(Total\ number\ of\ term\ in\ document)} \quad (2)$$

Keterangan

Idf = mengukur penting/tidak sebuah kata dalam dokumen

c. TF-IDF (*Term Frequency - Inverse Document Frequency*)

Setelah mendapatkan nilai TF dan IDF selanjutnya akan menghitung nilai TF-IDF dengan mengalikannya:

$$tfidf_{t,d} = tf_{t,d}\ \text{X}\ idf_d \quad (3)$$

Keterangan

TF-IFD = hasil penggabungan antara TF dan IDF

2. BOW (*Bag Of Word*)

Metode ini adalah salah satu metode yang cukup sederhana dalam memproses suatu data teks yang diubah menjadi angka berbentuk vektor agar dapat diproses dan diolah oleh komputer. Metode ini sejatinya hanya menghitung jumlah frekuensi keunculan kata pada seluruh dokumen yang di proses [13].

a. Langkah pertama membuat sekumpulan kata atau *vocabulary* dari seluruh dokumen yang akan di proses
b. Dari setiap dokumen, akan dihitung frekuensi kemunculan setiap kata dari dokumen tersebut
c. Masukan hasil perhitu8ngan frekuensi kata untuk setiap dokumen dalam sebuah vektor.
d. Vector tersebut merepresentasikan dokumen BOW

Dalam notasi matematis BOW dapat direpresentasikan jika d adalah sebuah dokumen dan v adalah sekumpulan kata atau *vocabulary* dari keseluruhan dokumen yang akan direpresentasikansebagai berikut:

$$BoW(d) = [count(w1,d), count(w2,d\ldots count(wn,d)] \quad (4)$$

Dimana count(wi,d) merupakan jumlah kemunculan katawi dalam dokumen d. sedang n adalah jumlah kata dalam V.

B. Algoritma SVM (*Support Vector Machine*)

Merupakan metode pembelajaran yang digunakan mesin untuk proses klasifikasi dan regresi, dengan cara kerja memisahkan dua kelas dengan memaksimalkan margin atau juga disebut jarak antar kelas dengan persamaan atau rumus dasar dari metode SVM adalah sebagai berikut [14].

$$y = f(x) = sign(w^* x+b) \quad (c=1\ g=1\ karnel=linear) \quad (5)$$

Dimana x adalah vektor atau vitur dari sebuah data, w adalah vektor bobot yang mengatur arah dan jarak garis atau *hyperplane,* b adalah bias, sedangkan y adalah label kelas prediksi (-1 atau +1), dan sign adalah fungsi dari signum yang menghasilkan -1 tau +1 tergantung kepada apakah $f(x) > 0$ atau $f(x) < 0$.

C. Validation

Langkah terahir pada proses klasifikasi sentiment analisis menggunakan data teks dari twitter ini adalah *validation*, untuk menguji dan mengukur keberhasilan dari teknik dan metode yang telah diterapkan. Pada pengukuran evaluasi model ini menggunakan teknik *confusion matrix* yang menggunakan acuan *accuracy, precission, recall* dan *f1-Score* dengan ketetapan table berikut.[15]

Tabel I. *confusion matrix*

TP (*True Positive*)	FP (*False Positive*)
FN (*False Negative*)	TN (*True Negative*)

Keterangan:
a. TP yaitu jumlah data positif yang terklasifikasi benar
b. TN yaitu jumlah data negatif yang terklasifikasi benar
c. FN yaitu jumlah data negatif yang terklasifikasi salah

d. FP yaitu jumlah data positif yang terklasifikasi salah

Accuracy (A) prediksi benar dari *true positif* dan *true negatif*.

$$A = \frac{(TP+TN)}{(TP+FP+FN+TN)} = 100\% \quad (6)$$

Precission (P) nilai *true positif* dari seluruh nilai *positif*.

$$P = \frac{(TP)}{(TP+FP)} = 100\% \quad (7)$$

Recall (R) persentase prediksi *positif* dengan *true positif*.

$$R = \frac{(TP)}{(TP+FN)} = 100\% \quad (8)$$

F1-Score (F) perbandingan rata-rata *precission* dan *recall*.

$F = 2 \text{ x } (\text{R x P}) : (R + P)$ (9)

D. Komparasi Metode

Dari penerapan Teknik TF-IDF dan menggunakan algoritma SVM (*Support Vector Machine*) akan menghasilkan nilai-nilai yang ada pada *confusion matrix* dengan dibandingan dengan teknik BOW dengan menggunakan algoritma masih sama dengan sebelumnya yaitu algoritma *Support Vector Machine* SVM, tidak hanya dengan membandingkan nilai ahir saja tetapi pada kesimpulannya juga akan mendapatkan kekurangan dan kelebihan pada masing-masing komparasi metode yang telah dilakukan dengan dua jumlah penelitian.

Tabel II. Komparasi hasil perhitungan *confusion matrix*

Metode	*Accuracy*	*Precission*	*Recall*	*F1-Score*
TF-IDF + SVM	-%	-%	-%	-%
BOW + SVM	-%	-%	-%	-%

HASIL DAN PEMBAHASAN

A. Persiapan Data

Proses mempersipkan data mulai dari penambangan data twitter hingga mengolahnya, agar siap untuk dilakukan pemrosesan menggunakan *machine learning*.

1. Crawling twitter

Penambangan data twitter yang telah dilakukan didapat hasil 300 tweet dengan pengelompokan hastag #jne pada priode rentan waktu mulai 01 Januari 2023 sampai dengan 31 Januari 2023, yang berisikan variabel nama akun *account name,* isi teks unggahan *tweet* dan waktu unggah *date* dengan hasil pada tabel berikut:

Tabel III. Hasil penambangan teks tweet

No	Account Name	Tweet	Date

1	@alsawalsa	Paketan sy lama ya datangnya!!!	01/01/2023 13:45
2	@femmoy	Alhamdulilah sampai @jne389 >>>> 😔	01/01/2023 13:58
…	…	…	…
300	@mcflouury	Paket sy rusak bru sampai pengirimannya buruk 😰😰	31/01/2023 07:15

2. Labelling

Dari hasil penambangan data twitter yang telah dilakukan dengan jumlah data 300 tweet makan akan dilakukan pelabelan secara manual oleh orang yang berkompeten bidang hukum dengan hasil sebagai berikut:

Tabel IV. Hasil labelling teks tweet

No	Tweet	L1	L2	L3	Keputusan
1	Paketan sy lama ya datangnya!!!	-	-	-	Negatif
2	Alhamdulilah sampai @jne389 >>>> 😔	+	+	-	Positif
…	…	…	…	…	…
300	Paket sy rusak bru sampai pengirimannya buruk 😰😰	-	-	-	Negatif

Keterangan:

Pelabelan dilakukan oleh 3 orang dengan kode L1,L2 dan L3 dan keputusan label didapatkan jika jumlah label idak kurang dari 2.

3. Cleanning

Noise dan fitur tidak perlu yang terdapat pada data teks dapat mempengaruhi kinerja pemrosesan. Pada tahap ini data akan dilakukan proses pembersihan atau penyederhanaan kata mendaji baku, mulai dari *case folding* atau menyetarakan semua kata menjadi huruf kecil, *remove punctuation* atau penghilangan url, karakter, angka dll pada setiap kalimat, *stopword*

removal atau penghilangan kata hubung dan *stemming* yaitu proses menghilangkan kata imbuhan dengan hasil pada tabel berikut:

Tabel V. Hasil cleannig teks tweet

No	Tweet	Cleanning
1	Paketan sy lama ya datangnya!!!	paket lama datang
2	Alhamdulilah sampai @jne389 >>>> 😔	alhamdulilah sampai
…	…	…
300	Paket sy rusak bru sampai pengirimannya buruk 😨😨	Paket rusak pengiriman buruk

4. Data Checking

Data yang sudah siap untuk diproses akan dilakukan pengecekan lagi terlebih dahulu, agar data benar-benar vailid. Pengecekan ini memiliki hasil yang baik dengan tidak ada data kosong dan duplikat serta jumlah antar kedua label positif dan negative yang seimbang, sebagai berikut:

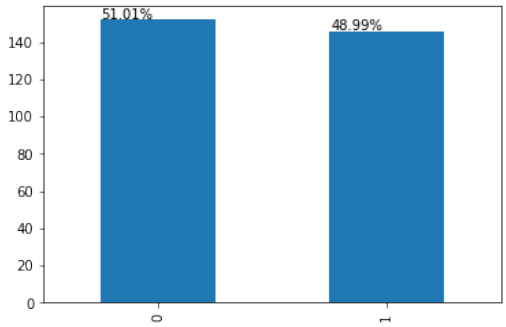
```
df.dropna(inplace=True)
df['tweet'].isna()
```

```
0      False
1      False
2      False
3      False
4      False
       ...
295    False
296    False
297    False
298    False
299    False
Name: tweet, Length: 300, dtype: bool
```

```
df.drop_duplicates(inplace=True)
df.duplicated()
```

```
0      False
1      False
2      False
3      False
4      False
       ...
295    False
296    False
297    False
298    False
299    False
Length: 298, dtype: bool
```

Gambar 2. Hasil cek *blank data* dan *duplicated data*



Gambar 3. Hasil cek *imbalance data*

B. Feature Extraction

Proses perubahan dari data berbentuk teks menjadi sebuah vektor berupa angka untuk dapat diproses oleh computer dengan python dan dibuktikan perhitungan manual.

1. TF-IDF (*Term Frequency- Inverse Document Frequency*)

Perhitungan atau rumus TF-IDF terbagi menjadi dua yaitu TF (*Term Frequency*) dan IDF (*Inverse Document Frequency*) dengan rumus dan cara kerja yang berbeda dan akan digabungkan di akhir perhitungan antara TF dan IDF menjadi TF-IDF (*Term Frequency- Inverse Document Frequency*) sebagai berikut:

DATA UTAMA			
Dokumen 1 (D1)	paket lama datang	Panjang Dokumen	3
Dokumen 2 (D2)	alhamdulilah sampai	Panjang Dokumen	2
Dokumen 3 (D3)	paket rusak pengiriman buruk	Panjang Dokumen	4
Total Dokumen			3

Gambar 4. Sempel data perhitungan manual

- TF (*Term Frequency*)

Hasil perhitungan TF dengan menetapkan rumus yang sudah ada pada sub bab sebelumnya:

Token	Frekuensi kemunculan kata			df	TF		
	D1	D2	D3		TF1	TF2	TF3
alhamdulilah	0	1	0	1	0	0,5	0
buruk	0	0	1	1	0	0	0,25
datang	1	0	0	1	0,333333	0	0
lama	1	0	0	1	0,333333	0	0
paket	1	0	1	2	0,333333	0	0,25
pengiriman	0	0	1	1	0	0	0,25
rusak	0	0	1	1	0	0	0,25
sampai	0	1	0	1	0	0,5	0

Gambar 5. Hasil perhitungan *Term Frequency*

Keterangan:

D1,D2 dan D3 merupakan jumlah frekuensi kemunculan kata pada setiap dokumen dengan hasil df, sedangkan TF1,TF2 dan TF3 merupakan notasi untuk setiap riview TF

- IDF (*Inverse Document Frequency*)

Setelah berhasil menghitung nilai TF selanjutnya kita menghitung nilai IDF dari hasil nilai TF:

Token	Frekuensi kemunculan kata			df	TF			IDF
	D1	D2	D3		TF1	TF2	TF3	
alhamdulilah	0	1	0	1	0	0,5	0	0,477121
buruk	0	0	1	1	0	0	0,25	0,477121
datang	1	0	0	1	0,333333	0	0	0,477121
lama	1	0	0	1	0,333333	0	0	0,477121
paket	1	0	1	2	0,333333	0	0,25	0,176091
pengiriman	0	0	1	1	0	0	0,25	0,477121
rusak	0	0	1	1	0	0	0,25	0,477121
sampai	0	1	0	1	0	0,5	0	0,477121

Gambar 6. Hasil perhitungan *Inverse Document Frequency*

Keterangan:

Nilai IDF adalah logaritma dari pembagian jumlah dokumen dengan nilai dari TF.

- TF-IDF (*Term Frequency- Inverse Document Frequency*)

Setelah kedua nilai TF dan IDF didapatkan kemudian akan digabungkan menjani nilai TF-IDF dengan menyusuk angka -angka dengan susunan berupa vektor berikut:

Token	Frekuensi kemunculan kata			df	TF			IDF	TF-IDF		
	D1	D2	D3		TF1	TF2	TF3		TF-IDF1	TF-IDF2	TF-IDF3
alhamdulilah	0	1	0	1	0	0,5	0	0,477121	0	0,238561	0
buruk	0	0	1	1	0	0	0,25	0,477121	0	0	0,11928
datang	1	0	0	1	0,333333	0	0	0,477121	0,15904	0	0
lama	1	0	0	1	0,333333	0	0	0,477121	0,15904	0	0
paket	1	0	1	2	0,333333	0	0,25	0,176091	0,058697	0	0,044023
pengiriman	0	0	1	1	0	0	0,25	0,477121	0	0	0,11928
rusak	0	0	1	1	0	0	0,25	0,477121	0	0	0,11928
sampai	0	1	0	1	0	0,5	0	0,477121	0	0,238561	0

Gambar 7. Hasil perhitungan*TF-IDF*

Keterangan:

Hasil ahir untuk tf idf didapatkan dari penggabungan antara nilai TF dan nilai IDF.

HASIL VEKTOR TF-IDF	
Dokumen 1	[0, 0, 0.15904, 0.15904, 0.15904, 0, 0, 0]
Dokumen 2	[0.238561, 0, 0, 0, 0, 0, 0, 0.238561]
Dokumen 3	[0, 0.11928, 0, 0, 0.044023, 0.11928, 0.11928, 0]

Gambar 8. Hasil Vektorisasi TF-IDF

Untuk penerapan dengan menggunakan machine learning berbasis *python* adalah sebagai berikut:

```
[3] from sklearn.feature_extraction.text import TfidfVectorizer

[4] tfidf = TfidfVectorizer()
    respons = tfidf.fit_transform(corpus)
    print (respons)

[5] tfidf.get_feature_names()

[6] respons.todense()

matrix([[0.        , 0.        , 0.62276601, 0.62276601, 0.4736296 ,
         0.        , 0.        , 0.        ],
        [0.70710678, 0.        , 0.        , 0.        , 0.        ,
         0.        , 0.        , 0.70710678],
        [0.        , 0.52863461, 0.        , 0.        , 0.40204024,
         0.52863461, 0.52863461, 0.        ]])
```

Gambar 9. Hasil vektorisasi menggunakan python

2. BOW (*Bag of Word*)

Dalam implememntasinya dari korpus yang ada hanya mengambil kata yang unik saja, setiap kata yang berulang akan ditulis sekali:

Review	alhamdulilah	buruk	datang	lama	paket	pengiriman	rusak	sampai
Dokumen 1	0	0	1	1	1	0	0	0
Dokumen 2	1	0	0	0	0	0	0	1
Dokumen 3	0	1	0	0	1	1	1	0

Gambar 10. Hasil perhitungan *Bag of Word*

Keterangan:

Menghitung frekuensi setiap kemunculan kata pada korpus tersebut pada tiga riview sebelumnya. Jika kata tersebut muncul maka diberi nilai satu sebaliknya jika tidak muncul maka diberi nilai 0 dengan hasil vector berikut:

HASIL VEKTOR BAG OF WORD	
Dokumen 1	[0, 0, 1, 1, 1, 0, 0, 0]
Dokumen 2	[1, 0, 0, 0, 0, 0, 0, 1]
Dokumen 3	[0, 1, 0, 0, 1, 1, 1, 0]

Gambar 11. Hasil Vektorisasi BOW

Untuk hasil penerapan dengan menggunakan machine learning berbasis *python* adalah sebagai berikut:

```
[3] from sklearn.feature_extraction.text import CountVectorizer

[5] bow = CountVectorizer()
    responsbow = bow.fit_transform(corpus)
    print (responsbow)

[8] bow.get_feature_names()

[7] responsbow.todense()

    matrix([[0, 0, 1, 1, 1, 0, 0, 0],
            [1, 0, 0, 0, 0, 0, 0, 1],
            [0, 1, 0, 0, 1, 1, 1, 0]])
```

Gambar 12. Hasil vektorisasi menggunakan python

C. Algorithm SVM (*Support Vector Machine*)

Setelah mendapatkan hasil vektor dari data yang sudah melewati beberapa tahapan, pada proses SVM dengan cara kerja memisahkan dua kelas dengan memaksimalkan margin. Pada proses metode SVM ini dilakukan untuk setiap percobaan feature extraction TF-IDF maupun BOW menggunakan konfigurasi standart sebagai berikut:

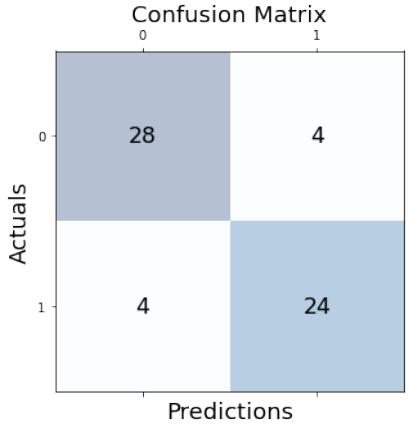
```
[ ] from sklearn.svm import SVC

[ ] model1 = SVC(C=1, gamma=1, kernel='linear')
    model1.fit(X_train, y_train)

    SVC(C=1, gamma=1, kernel='linear')
```

Gambar 13. Metode SVM di python

D. Validation

Proses menguji dan mengukur keberhasilan dari teknik dan metode yang telah diterapkan untuk masing-masing percobaan yaitu antara TF-IDF dengan metode SVM (*Support Vector Machine*) dan BoW juga dengan menggunakan metode SVM (*Support Vector Machine*):

- Berikut hasil validasi TF-IDF dengan metode SVM:



Gambar 14. *Confusion matrix* TF-IDF

Keterangan hasil:

a. Berjumlah 28 data positif yang terklasifikasi benar
b. Berjumlah 24 data negatif yang terklasifikasi benar
c. Berjumlah 4 data negatif yang terklasifikasi salah
d. Berjumlah 4 data positif yang terklasifikasi salah

Hasil dari *confusion matrix* akan digunakan sebagai acuan dalam mencari *Accuacy*, *Precission*, *Recall* dan *F1-score*:

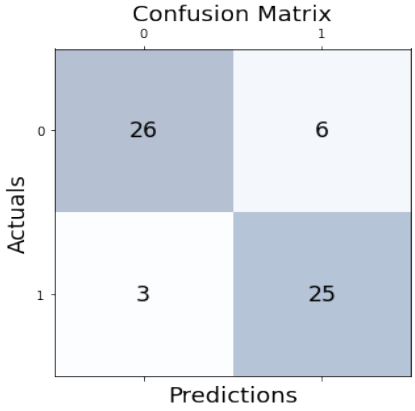
Accuracy (0.86) prediksi benar dari *TP* dan *TN*.

Precission (0.85) nilai *true positif* dari seluruh nilai *positif*.

Recall (0.85) persentase prediksi *positif* dengan *true positif*.

F1-Score (0.85) perbandingan rata-rata *precission* dan *recall*.

- Berikut hasil validasi BOW dengan metode SVM:



Gambar 15. *Confusion matrix* BOW

Keterangan hasil:

a. Berjumlah 26 data positif yang terklasifikasi benar
b. Berjumlah 25 data negatif yang terklasifikasi benar
c. Berjumlah 3 data negatif yang terklasifikasi salah
d. Berjumlah 6 data positif yang terklasifikasi salah

Hasil dari *confusion matrix* akan digunakan sebagai acuan dalam mencari *Accuacy, Precission, Recall* dan *F1-score*:

Accuracy (0.85) prediksi benar dari *TP* dan *TN*.

Precission (0.80) nilai *true positif* dari seluruh nilai *positif*.

Recall (0.89) persentase prediksi *positif* dengan *true positif*.

F1-Score (0.84) perbandingan rata-rata *precission* dan *recall*.

E. Komparasi Metode

Membandingkan hasil uji validasi guna untuk mendapatkan nilai hasil yang paling optimal antara kedua teknik yaitu TF-IDF dan BOW untuk klasifikasi analisis sentimen menggunakan metode SVM dengan hasil komparasi sebagai berikut.

Tabel VI. Komparasi hasil *confusion matrix*

Metode	*TP*	*TN*	*FN*	*FP*
TF-IDF + SVM	28	24	4	4
BOW + SVM	26	25	3	6

Didapatkan hasil komentar positif yang diprediksi benar lebih unggul pada TF-IDF + SVM daripada komentar negative yang di prediksi salah lebih besar pada BOW +SVM sedangkan untuk tipe eror FN dan FP pada kasus ini lebih berpihak kepada dengan komentar positif yang terdeteksi sebagai komentar negative.

Tabel VI. Komparasi hasil validasi

Metode	*Accuracy*	*Precission*	*Recall*	*F1-Score*
TF-IDF + SVM	86%	85%	85%	85%
BOW + SVM	85%	80%	89%	84%

Didapatkan hasil rasio prediksi benar baik komentar negative dan positif dengan selisih 1 angka dan prediksi benar positif diabandingan dengan keseluruhan hasil yang diprediksi positif dengan selisih 5 angka dan perbandingan rata-rata dengan presisi yang dibobotkan selisih 1 angka yang semuanya unggul pada TF-IDF+BOW, Tetapi benar positif diabandingan dengan keseluruhan data yang benar positif unggul pada BOW+SVM.

KESIMPULAN

Kesimpulan dari hasil perbandingan kedua teknik dan metode yang diterapkan yaitu teknik feature extraction TF-IDF dengan metode SVM dan teknik feature extraction BOW juga dengan metode SVM sebagai berikut:

- Untuk model penelitian TF-IDF dengan SVM mendapatkan hasil Accurasi sebesar 86%, hasil nilai precission sebesar 85%, hasil Recall sebesar 85% dan untuk hasil F1-Score sebesar 85%. Dengan nilai yang didapatkan pada komparasi ini sangat bagus yang menggunakan system mempertimbankan frekuensi kemunculan kata dalam suatu dokumen.
- Untuk model penelitian BOW dengan SVM mendapatkan hasil Accurasi sebesar 85%, hasil nilai precission sebesar 80%, hasil Recall sebesar 89% dan untuk hasil F1-Score sebesar 84%. Dengan nilai yang didapatkan pada nilai Recall untuk komparasi teknik ini dapat mengungguli TF-IDF+BOW dikarenakan memiliki perhitungan dengan mempertahankan urutan kata pada setiap dokumen.

Dengan perolehan nilai-nilai tersebut untuk hasil yang maksimal dalam melakukan proses analisis sentimen menggunakan data teks twitter yaitu dengan menerapkan teknik feature extraction TF-IDF dan SVM yang memiliki keunggulan lebih baik pada setiap pengukuranya baik Accuracy, Precission maupun F1-Score sedangkan pada teknik BOW dan SVM hanya unggul untuk nilai Recall saja.

DAFTAR PUSTAKA

1] M. Fadilah Arfat et al., “Analisis Sentimen Masyarakat Indonesia Terkait Vaksin Covid-19 Pada Media Sosial Twitter Menggunakan Metode Support Vector Machine (SVM),” vol. 7, no. 2, 2022.

[2] Y. Romadhoni, K. Fahmi, and H. Holle, “Analisis Sentimen Terhadap PERMENDIKBUD No.30 pada Media Sosial Twitter Menggunakan Metode Naive Bayes dan LSTM,” vol. 7, no. 2, 2022.

[3] C. Ayunda et al., “Analisis Komparasi Algoritma Machine Learning untuk Sentiment Analysis (Studi Kasus: Komentar YouTube ‘Kekerasan Seksual’),” vol. 7, no. 2, 2022.

[4] R. Aprillya, P. : Perbandingan, M. Klasifikasi, R. A. Putri, and N. S. Fatonah, “Perbandingan Metode Klasifikasi serta Analisis Faktor Akademis Pola Kelulusan Mahasiswa di Perguruan Tinggi,” vol. 7, no. 2, 2022.

[5] A. Setyawinda, B. Setiyadi, and A. D. Hartanto, “Perbandingan Algoritma Word Matching dan Naive Bayes untuk Klasifikasi Sentimen Analisis Komentar Instagram,” vol. 5, no. 1, 2020.

[6] O. I. Gifari, M. Adha, I. Rifky Hendrawan, F. Freddy, and S. Durrand, “Analisis Sentimen Review Film Menggunakan TF-IDF dan Support Vector Machine,” JIFOTECH (JOURNAL OF INFORMATION TECHNOLOGY, vol. 2, no. 1, 2022.

[7] “JOURNAL OF INTELLIGENT SYSTEMS AND COMPUTATION 43.” [Online]. Available: https://t.co/9WloaWpfD5

[8] C. H. Yutika, A. Adiwijaya, and S. al Faraby, “Analisis Sentimen Berbasis Aspek pada Review Female Daily Menggunakan TF-IDF dan Naïve Bayes,” JURNAL MEDIA INFORMATIKA BUDIDARMA, vol. 5, no. 2, p. 422, Apr. 2021, doi:

10.30865/mib.v5i2.2845.
[9] M. M. Munir, M. A. Fauzi and R. S. Perdana, "Implementasi Metode Backpropagation Neural Network berbasis lexion Based Features dan Bag of Words Untuk Identifikasi Ujaran Kebencian Pada Twitter," vol. 2, no. 10, 2018.
[10] Hartanto, "TEXT MINING DAN SENTIMEN ANALISIS TWITTER PADA GERAKAN LGBT," vol. 9, no. 1, 2017.
[11] A. P. Wibawa, M. Guntur, A. Purnama, M. Fathony Akbar, and F. A. Dwiyanto, "Metode-metode Klasifikasi," Prosiding Seminar Ilmu Komputer dan Teknologi Informasi, vol. 3, no. 1, 2018.
[12] D. Farah Zhafira, B. Rahayudi, and P. Korespondensi, "ANALISIS SENTIMEN KEBIJAKAN KAMPUS MERDEKA MENGGUNAKAN NAIVE BAYES DAN PEMBOBOTAN TF-IDF BERDASARKAN KOMENTAR PADA YOUTUBE," 2021.
[13] J. Muara Sains, dan Ilmu Kesehatan, W. Trisari Harsanti Putri, and R. Hendrowati, "PENGGALIAN TEKS DENGAN MODEL BAG OF WORDS TERHADAP DATA TWITTER," vol. 2, no. 1, pp. 129–138, 2018.
[14] N. Hendrastuty, A. Rahman Isnain, and A. Yanti Rahmadhani, "Analisis Sentimen Masyarakat Terhadap Program Kartu Prakerja Pada Twitter Dengan Metode Support Vector Machine," vol. 6, no. 3, 2021, [Online]. Available: http://situs.com
[15] D. Normawati and S. A. Prayogi, "Implementasi Naïve Bayes Classifier Dan Confusion Matrix Pada Analisis Sentimen Berbasis Teks Pada Twitter," 2021.