

## Analisis Sentimen Pengguna Twitter Terhadap Kinerja Walikota Medan Menggunakan Metode *Naive Bayes Classifier*

**Fajar Muharram**

Program Studi Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas  
Negeri Medan

Korespondensi penulis: [fajarmuharram06@mhs.unimed.ac.id](mailto:fajarmuharram06@mhs.unimed.ac.id)

**Kana Saputra S**

Program Studi Ilmu Komputer, Fakultas Matematika dan Ilmu Pengetahuan Alam, Universitas  
Negeri Medan

Email: [kanasaputras@unimed.ac.id](mailto:kanasaputras@unimed.ac.id)

Jl. William Iskandar Ps. V, Kenangan Baru, Kec. Percut Sei Tuan, Kabupaten Deli Serdang,  
Sumatera Utara 20221

**ABSTRACT.** *Technological developments today make it easy for people to use social media as a means of expressing opinions, including Twitter. The case study taken by the researcher is the sentiment towards the performance of the mayor of Medan. The case was taken because it was widely discussed by Indonesian people, especially the city of Medan on Twitter social media. One of the uses of this research is to find out the trend of Twitter user comments on the performance of the mayor of Medan by conducting a sentiment analysis. Sentiment will be classified as positive, negative and neutral. The algorithm used in sentiment analysis is Naïve Bayes. The stages in conducting sentiment analysis in this study are data preprocessing, data processing, classification, and evaluation. The results of this study are using the SMOTE method, the training and testing ratio is 80:20 because it has the highest accuracy, which is 78% compared to other ratios. The prediction results resulting from the classification turned out to be more dominant towards neutral labels. In addition to classifying for sentiment analysis, this study also measures the performance of the model created. The results showed that the Naïve Bayes algorithm has a precision value of 78%, a recall of 78%, and an f1-score of 77%.*

**Keywords:** *Sentiment Analysis, TextBlob, TF-IDF, Multinomial Naïve Bayes, Confusion Matrix.*

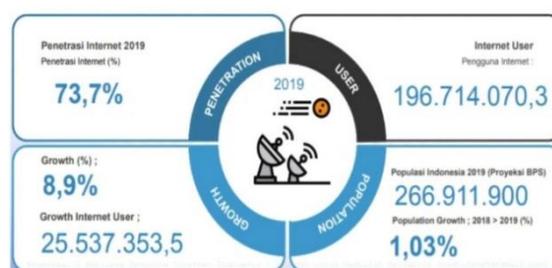
**ABSTRAK.** Perkembangan teknologi pada zaman sekarang memberikan kemudahan bagi masyarakat dalam menggunakan media sosial sebagai salah satu sarana untuk mengekspresikan pendapat, diantaranya *Twitter*. Studi kasus yang diambil oleh peneliti adalah sentimen terhadap kinerja walikota medan. Kasus tersebut diambil karena ramai diperbincangkan masyarakat Indonesia khususnya kota medan di media sosial *Twitter*. Salah satu pemanfaatan penelitian ini adalah untuk mengetahui kecenderungan komentar pengguna *Twitter* terhadap kinerja walikota medan dengan melakukan analisis sentimen. Sentimen akan diklasifikasikan dengan label positif, negatif dan netral. Algoritma yang digunakan dalam melakukan analisis sentimen adalah *Naïve Bayes*. Tahapan dalam melakukan analisis sentimen pada penelitian ini adalah *preprocessing* data, pengolahan data, klasifikasi, dan evaluasi. Hasil dari Penelitian ini yaitu menggunakan metode *SMOTE* rasio *training* dan *testing* yaitu 80:20 dikarenakan memiliki akurasi yang tertinggi yaitu 78% dibandingkan dengan rasio lainnya. Hasil prediksi yang dihasilkan dari klasifikasi ternyata lebih dominan kepada label netral. Selain melakukan klasifikasi untuk analisis sentimen, penelitian ini juga mengukur performa

dari model yang dibuat. Hasil penelitian menunjukkan bahwa algoritma Naive Bayes memiliki nilai *precision* 78%, *recall* 78%, dan *f1-score* 77%.

**Kata Kunci:** Analisis Sentimen, *TextBlob*, *TF-IDF*, *Multinomial Naive Bayes*, *Confusion Matrix*.

## PENDAHULUAN

Saat ini, *internet* di Indonesia berkembang cukup pesat. Hal tersebut membuat masyarakat semakin mudah untuk mengakses *internet* kapan pun dan dimana pun. Dengan kemudahan mengakses internet ini mengubah kebiasaan masyarakat untuk mencari informasi melalui *internet* dibanding media cetak. Terlihat pada (Gambar 1), dari survei yang dilakukan oleh Asosiasi Penyelenggaraan Jasa Internet Indonesia (APJII) mendapatkan hasil jika pengguna *internet* di Indonesia mencapai 196.7 juta pengguna [1].



Gambar 1. Pengguna *Internet* 2019-2020

Pesatnya perkembangan internet membuat media sosial semakin berkembang pesat. Media sosial merupakan media yang saat ini digunakan oleh masyarakat untuk mengakses informasi, sarana berkomunikasi, berbagi kegiatan yang sedang dilakukan, bertukar opini, dan masih banyak kegunaan lainnya yang bisa didapatkan dari media sosial. Dari berbagai media yang ada, *twitter* menjadi salah satu media sosial yang banyak digunakan oleh masyarakat Indonesia. *Twitter* memungkinkan pengguna untuk mengekspresikan tulisan mereka dalam 280 karakter dan dapat dilihat oleh pengguna lainnya dengan fitur *tweet*. Per-Oktober 2020, jumlah pengguna aktif *twitter* di Indonesia mencapai 13,2 juta pengguna [2].

*Twitter* ini juga menjadi media sosial yang kerap sekali digunakan oleh pemberi sentimen atau pendapat kepada orang lain (*netizen*). Masyarakat juga dapat memberikan pendapat dan tanggapannya akan postingan yang terdapat di *Twitter* dan tidak harus saling mem-*follow* terlebih dahulu. Berbagai instansi pemerintahan juga menggunakan *Twitter* ini sebagai media untuk mempublikasikan informasi mengenai instansi dan kegiatan instansi tersebut, yang didalamnya termasuk juga Pemerintah Wali Kota Medan. Dalam komentar dan tagar yang ditujukan kepada Wali Kota Medan

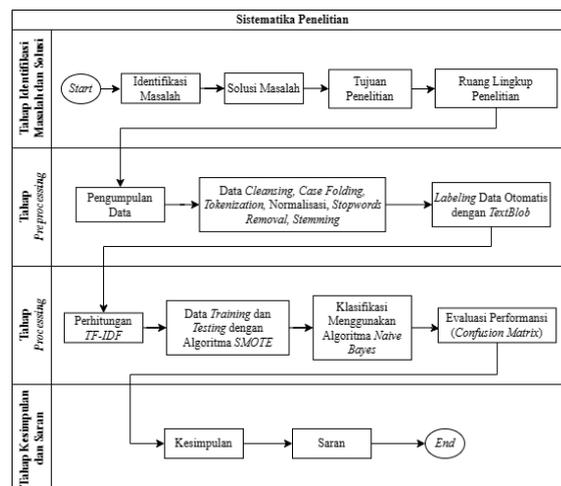
pastinya terdapat sentimen positif, netral, dan negatif. Hal itulah yang digunakan sebagai bahan data penelitian penambangan teks. Dari data *tweets* serta *hashtag* di akun *Twitter* Wali Kota Medan dapat diklasifikasikan sehingga didapat informasi baru.

Di dalam penelitian ini akan dilakukan sebuah analisis sentimen pengguna *Twitter* terhadap kinerja Pemerintah Kota Medan dengan menganalisa melalui akun *Twitter* Pemerintah Kota Medan. Berdasarkan penjelasan dari latar belakang yang telah diuraikan, maka dilakukan penelitian dengan data teks yang akan diambil melalui *Twitter* dengan menggunakan bantuan *Sns scrape Twitter*, dan data yang dimaksudkan adalah data tanggapan dari masyarakat akan postingan akun Pemerintah Kota Medan yang langsung diolah dan kemudian akan menghasilkan sentimen positif, negatif dan netral menggunakan *metode Naive Bayes Classifier*.

Berdasarkan penjelasan dari latar belakang yang telah diuraikan, maka dilakukan penelitian dengan data teks yang akan diambil melalui *Twitter* dengan menggunakan bantuan *Sns scrape Twitter*, dan data yang dimaksudkan adalah data tanggapan dari masyarakat akan postingan akun Pemerintah Kota Medan yang langsung diolah dan kemudian akan menghasilkan sentimen positif, negatif dan netral menggunakan *metode Naive Bayes Classifier*.

**METODE PENELITIAN**

Metode penelitian dibagi ke dalam 4 tahap, seperti pada (Gambar 2).



Gambar 2. Sistematika Penyelesaian Masalah.

**A. Pengumpulan Data**

Data yang digunakan merupakan data dari opini para pengguna *Twitter* mengenai kinerja pemerintahan Wali Kota Medan. Pengumpulan data dilakukan dengan bantuan *sns scrape*. *Sns scrape* adalah sebuah pustaka *Python* yang dapat digunakan untuk mengambil *Tweet* melalui *Sns scrape*

Twitter tanpa adanya batasan atau pembatasan permintaan [3]. Pengumpulan data *tweet* yang diambil tersebut berdasarkan *tweet* periode 26 Februari 2021 sampai 06 Januari 2023 yang dikirim sebagai balasan ke akun Twitter @bobbynasution\_ dan @pemko\_medan serta *tweet* yang menyebutkan akun-akun tersebut. Data yang terkumpul sebanyak 12.806 data.

## B. Tahap *Preprocessing*

Setelah data berhasil dikumpulkan, dilakukan *preprocessing* yang terdiri dari beberapa tahapan seperti:

- *Data cleansing*, yaitu proses penghapusan komponen yang tidak berpengaruh pada analisis seperti *retweet*, *hashtag*, *username*, dan *URL*
- *Case folding*, yaitu proses mengubah semua ukuran huruf menjadi huruf kecil
- *Tokenization*, yaitu proses memecah kalimat menjadi beberapa kata
- Normalisasi, yang mengubah kata tidak baku menjadi baku
- *Stopword removal*, yaitu proses penghapusan kata yang sering muncul tetapi tidak memiliki arti
- *Stemming*, yang merupakan proses mengubah kata ke bentuk dasarnya.

Setelah melakukan data *preprocessing*, pelabelan dilakukan secara otomatis menggunakan *Python TextBlob* untuk seluruh data *tweet*. Untuk saat ini *TextBlob* belum mendukung teks dalam bahasa Indonesia, maka dari itu teks atau data dari hasil proses *stemming* diubah terlebih dahulu kedalam bahasa *english*. *TextBlob* akan memberikan nilai polaritas pada skala dari -1 hingga 1, dimana nilai -1 menunjukkan sentimen negatif yang sangat kuat, 0 menunjukkan sentimen netral, dan 1 menunjukkan sentimen positif yang sangat kuat [4].

*TextBlob* akan menghitung nilai polaritas keseluruhan untuk seluruh teks dengan menggunakan rumus berikut:

$$\begin{aligned} & \text{Polaritas} \\ & = (\text{jumlah polaritas kalimat}) \quad (1) \\ & / (\text{jumlah kalimat}) \end{aligned}$$

## C. Tahap *Processing*

Data yang sudah melewati tahap *preprocessing*, data dibagi menjadi data *training* sebanyak 80% dan data *testing* sebanyak 20% [5]. Selanjutnya, dilakukan proses pemberian nilai atau bobot terhadap setiap *term* menggunakan metode *TF-IDF*.

$$TF(d, t) = \frac{\text{jumlah kata } t \text{ pada dokumen } d}{\text{total kata pada dokumen } d} \quad (2)$$

$$IDF(t) = \log \frac{D}{df} \quad (3)$$

$$TF - IDF = TF(d, t) * IDF(t) \quad (4)$$

Keterangan:

$t$  = kata

$d$  = dokumen

$df$  = jumlah dokumen yang mengandung kata  $t$

$D$  = jumlah dokumen

Setelah itu, dilakukan klasifikasi menggunakan algoritma *Multinomial Naïve Bayes* [6].

$$P(c) = \frac{N_c}{N} \quad (5)$$

Keterangan:

$P(c)$  = *prior probability* kelas  $c$

$N_c$  = jumlah kelas  $c$  pada seluruh dokumen

$N$  = jumlah seluruh dokumen

$$P(t_n|c) = \frac{W_{ct} + 1}{(\sum W' \in V W'_{ct}) + B'} \quad (6)$$

Keterangan:

$W_{ct}$  = nilai pembobotan *TF-IDF* pada *term*  $t$  di kelas  $c$

$\sum W' \in V W'_{ct}$  = jumlah total bobot dari keseluruhan *term* yang berada di kelas  $c$

$B'$  = jumlah bobot dari *term* yang unik pada seluruh dokumen

$$P(c|term \text{ dokumen } d) = P(c) * P(t_1|c) * P(t_2|c) * \dots * P(t_n|c) \quad (7)$$

Keterangan:

$P(c|term\ dokumen\ d)$  = probabilitas suatu dokumen d berada di kelas c

$P(c)$  = probabilitas *prior* dari kelas c

$P(t_n|c)$  = probabilitas kata ke-n pada kelas c

$t_n$  = *Term* ke-n dokumen d

#### D. Evaluasi Performansi

Proses terakhir yaitu mengevaluasi performa suatu model, digunakan Confusion Matrix. *Confusion Matrix* berisi informasi asli dan yang diprediksi. Dari matriks tersebut dapat diketahui *accuracy*, *precision*, *recall*, dan *f1-score*. Perhitungan *Confusion Matrix* dengan 3 kelas yaitu positif, netral dan negatif dapat dilihat pada (Tabel 1) berikut [7].

**Tabel 1. Confusion Matrix**

<i>Confusion Matrix</i>		<i>Actual Sentimen</i>		
		<b>Positif</b>	<b>Netral</b>	<b>Negatif</b>
<b><i>Predicted Sentiment</i></b>	<b>Positif</b>	<i>True Positive (TP)</i>	<i>False Positive (FP)</i>	<i>False Positive (FP)</i>
	<b>Netral</b>	<i>False Negative (FN)</i>	<i>True Neutral (TN)</i>	<i>False Neutral (FN)</i>
	<b>Negatif</b>	<i>False Negative (FN)</i>	<i>False Neutral (FN)</i>	<i>True Negative (TN)</i>

Keterangan:

- *True Positive (TP)* adalah jumlah data yang sebenarnya positif dan di-prediksi positif oleh model.
- *False Positive (FP)* adalah jumlah data yang sebenarnya negatif atau netral, tetapi di-prediksi positif oleh model.
- *False Negative (FN)* adalah jumlah data yang sebenarnya positif, tetapi di-prediksi negatif atau netral oleh model.
- *True Neutral (TN)* adalah jumlah data yang sebenarnya netral dan di-prediksi netral oleh model.
- *False Neutral (FN)* adalah jumlah data yang sebenarnya negatif, tetapi di-prediksi netral atau jumlah data yang sebenarnya netral, tetapi di-prediksi negatif oleh model.
- *True Negative (TN)* adalah jumlah data yang sebenarnya negatif dan di-prediksi negatif oleh model.

Nilai *accuracy*, *precesion*, *recall*, dan *f1-score* dapat diketahui dengan rumus berikut:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (8)$$

$$Precision = \frac{TP}{TP + FP} \quad (9)$$

$$Recall = \frac{TP}{TP + FN} \quad (10)$$

$$F1 - Score = \frac{2 * Precision * Recall}{Precision + Recall} \quad (11)$$

## HASIL DAN PEMBAHASAN

Bagian ini membahas mengenai hasil penelitian yang dilakukan dalam penelitian.

### A. Pengolahan Data *Preprocessing*

Setelah melakukan pengumpulan data, selanjutnya melakukan pengolahan data dengan proses data *preprocessing* yang terdiri dari beberapa tahapan yaitu data *cleansing*, *case folding*, *tokenization*, *stopwords removal*, dan *stemming*. Kemudian, dilanjutkan dengan *labeling* data otomatis dengan *TextBlob*

**Tabel 2. Data Preprocessing**

Tahapan	Hasil
<i>crawling</i>	Kerja keras @bobbynasution_ walikota Medan terbayar lunas, kota @pemko_medan ini jdi lebih indah dan bersih dari sebelumnya
<i>cleansing</i>	Kerja keras walikota Medan terbayar lunas, kota ini jdi lebih indah dan bersih dari sebelumnya
<i>case Folding</i>	kerja keras walikota medan terbayar lunas, kota ini jdi lebih indah dan bersih dari sebelumnya
<i>tokenization</i>	['kerja', 'keras', 'walikota', 'medan', 'terbayar', 'lunas', '.', 'kota', 'ini', 'jdi', 'lebih', 'indah', 'dan', 'bersih', 'dari', 'sebelumnya']
normalisasi	['kerja', 'keras', 'walikota', 'medan', 'terbayar', 'lunas', '.', 'kota', 'ini', 'jadi', 'lebih', 'indah', 'dan', 'bersih', 'dari', 'sebelumnya']
<i>stopwords removal</i>	['kerja', 'keras', 'walikota', 'medan', 'terbayar', 'lunas', '.', 'kota', 'jadi', 'lebih', 'indah', 'bersih', 'dari', 'sebelumnya']
<i>stemming</i>	kerja keras walikota medan bayar lunas. kota jadi lebih indah bersih dari sebelum
<i>translate</i>	hard work of the mayor of medan pays off. the city becomes more beautiful and cleaner than before
polaritas	0.06225490196
sentimen	positif

### B. Pembagian Rasio Data

*Split* data adalah langkah pembagian data menjadi 2 bagian, data *training* dan data *testing*. Pada proses klasifikasi, dilakukan optimasi dengan menerapkan *SMOTE* dalam menangani data yang tidak seimbang [8].

**Tabel 3. Pembagian Data Sebelum dan Sesudah *SMOTE***

Rasio	Proses Sebelum <i>SMOTE</i>		Proses Sesudah <i>SMOTE</i>	
	<i>Training</i>	<i>Testing</i>	<i>Training</i>	<i>Testing</i>
<b>80:20</b>	10244	2562	19414	4853
	<b>Total</b>		<b>Total</b>	
	12806		24267	

### C. Pembobotan *TF-IDF*

Menghitung frekuensi kemunculan suatu kata dalam dokumen serta *inverse* frekuensi dokumen yang memiliki kata tersebut.

**Tabel 4. Perhitungan *TF-IDF***

D1	kerja keras walikota medan bayar lunas kota jadi lebih indah bersih dari sebelum								
D2	kinerja walikota medan lambat masyarakat banyak kecewa merasa tidak puas hasil kerja								
D3	walikota medan hadir acara resmi gedung bangun kota medan								
Token	<i>TF</i>			<i>df</i>	$\frac{D}{df}$	<i>IDF</i> $= \log \frac{D}{df}$	<i>TF * IDF</i>		
	D1	D2	D3				D1	D2	D3
kerja	$\frac{1}{13}$ = 0,076	$\frac{1}{12}$ = 0,83	$\frac{0}{9}$ = 0	2	$\frac{3}{2}$ = 1,5	$\log 1,5$ = 0,176	0,013	0,146	0
walikota	$\frac{1}{13}$ = 0,076	$\frac{1}{12}$ = 0,83	$\frac{1}{9}$ = 0,1	3	$\frac{3}{3}$ = 1	$\log 1$ = 0	0	0	0
medan	$\frac{1}{13}$ = 0,076	$\frac{1}{12}$ = 0,83	$\frac{1}{9}$ = 0,1	3	$\frac{3}{3}$ = 1	$\log 1$ = 0	0	0	0

D. Klasifikasi Model *Multinomial Naïve Bayes*

Memprediksi kelas suatu anggota dengan probabilitas yang difokuskan untuk klasifikasi teks.

**Tabel 5. Source Code Python Multinomial Naïve Bayes**

```

modelMNB = MultinomialNB()
modelMNB.fit(X_train, y_train)
modelMNB.fit(X_trainSmote, y_trainSmote)
y_pred = modelMNB.predict(X_test)
y_predSmote = modelMNB.predict(X_testSmote)
print(classification_report(y_test, y_pred))
print(classification_report(y_testSmote, y_predSmote))
    
```

	precision	recall	f1-score	support
-1	0.89	0.20	0.32	405
0	0.68	0.92	0.78	1348
1	0.75	0.60	0.67	809
accuracy			0.70	2562
macro avg	0.77	0.57	0.59	2562
weighted avg	0.73	0.70	0.67	2562

**Gambar 3. Output Source Code Multinomial Naïve Bayes Sebelum SMOTE**

	precision	recall	f1-score	support
-1	0.76	0.91	0.83	1349
0	0.81	0.61	0.69	1348
1	0.78	0.81	0.79	1348
accuracy			0.78	4045
macro avg	0.78	0.78	0.77	4045
weighted avg	0.78	0.78	0.77	4045

**Gambar 4. Output Source Code Multinomial Naïve Bayes Sesudah SMOTE**

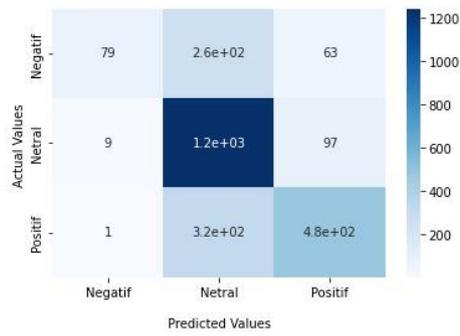
E. Hasil Klasifikasi dan Evaluasi Performansi

Proses klasifikasi dilakukan menggunakan *Python* dengan *library scikit learn*.

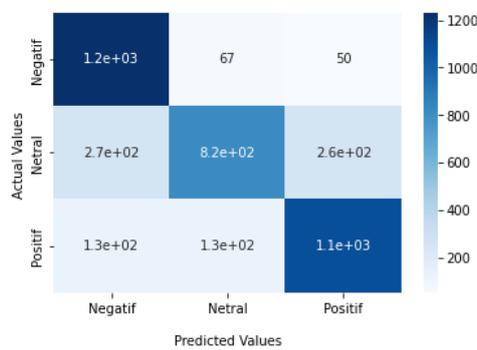
**Tabel 6. Source Code Confusion Matrix**

```

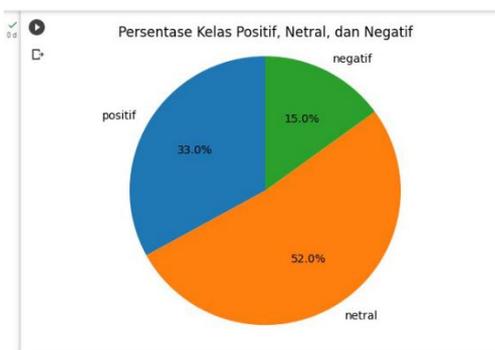
confusion_matrix = confusion_matrix(y_test, y_pred)
confusion_matrix_smote = confusion_matrix(y_testSmote, y_predSmote)
print(confusion_matrix)
print(confusion_matrix_smote)
    
```



Gambar 5. Output Source Code Confusion Matrix Sebelum SMOTE



Gambar 6. Output Source Code Confusion Matrix Sesudah SMOTE



Gambar 7. Output Source Code Persentase Hasil Sentimen

**Tabel 7. Hasil Klasifikasi Data Sebelum SMOTE**

<i>Confusion Matrix</i>		<i>Actual</i>		
		Positif	Netral	Negatif
<i>Predicted</i>	Positif	483	97	63
	Netral	325	1242	263
	Negatif	1	9	79
<i>Accuracy</i>		70%		
<i>Recall</i>		57%		
<i>Precision</i>		77%		
<i>F1 Score</i>		59%		

**Tabel 7. Hasil Klasifikasi Data Sesudah SMOTE**

<i>Confusion Matrix</i>		<i>Actual</i>		
		Positif	Netral	Negatif
<i>Predicted</i>	Positif	1093	265	56
	Netral	126	817	67
	Negatif	129	266	1232
<i>Accuracy</i>		78%		
<i>Recall</i>		78%		
<i>Precision</i>		78%		
<i>F1 Score</i>		77%		

## **KESIMPULAN**

Berdasarkan analisis, hasil performansi evaluasi dengan tingkat akurasi algoritma *Naïve Bayes* dalam melakukan analisis sentimen terhadap kinerja walikota medan adalah sebesar 78% serta dengan nilai *recall* sebesar 78%, nilai *precision* sebesar 78%, dan nilai *f1-score* sebesar 77%. Serta perbandingan hasil sentimen penelitian ini adalah sentimen positif 33%, sentimen negatif 15% dan sentimen netral 52%. Dengan hasil sentimen netral yang lebih besar, maka pengguna *Twitter* menandakan sikap yang tidak berpihak ke sentimen apapun terhadap kinerja Walikota Medan.

## **REFERENSI**

- [1] APJII. (2020). "Penetrasi Internet di Indonesia Capai 73,7 Persen." Dailysocial.Id.
- [2] Paisal, P. (2020). "Analisis Sentimen Masyarakat Berdasarkan Opini dari Sosial Media Menggunakan Metode Naive Bayes Classifier (Study Kasus: Universitas Sjakhyakirti)." *Jurnal Ilmiah Informatika Global*, 11(1), 41–46. doi:10.36982/jig.v11i1.1071
- [3] Utsu, K., Saito, J., & Uchida, O. (2018). "Sentiment Polarity Estimation of Emoticons by Polarity Scoring of Character Components." 2018 IEEE Region 10 Symposium, Tensymp 2018, 237–242. doi:10.1109/TENCONSpring.2018.8691984
- [4] Kevin Perdana, Titania Pricillia, & Zulfachmi. (2021). "Optimasi Textblob Menggunakan Support Vector Machine Untuk Analisis Sentimen (Studi Kasus Layanan Telkomsel)." *Jurnal Bangkit Indonesia*, 10(1), 13–15. doi:10.52771/bangkitindonesia.v10i1.120
- [5] Vashishtha, S., & Susan, S. (2020). "Fuzzy Interpretation of Word Polarity Scores for Unsupervised Sentiment Analysis." 2020 11th International Conference on Computing, Communication and Networking Technologies, ICCCNT 2020, 1–6. doi:10.1109/ICCCNT49239.2020.9225646
- [6] Afdhaluzzikri, A., Mawengkang, H., & Sitompul, O. S. (2022). "Performance analysis of Naive Bayes method with data weighting." *Sinkron*, 7(3), 817–821. doi:10.33395/sinkron.v7i3.11516
- [7] Singla, S., & Kumar, V. (2020). "Multi-Class Sentiment Classification using Machine Learning and Deep International Journal of Computer Sciences and Engineering Open Access Multi-Class Sentiment Classification using Machine Learning and Deep Learning Techniques." November. doi:10.26438/ijcse/v8i11.1420
- [8] Sulistiyowati, N., & Jajuli, M. (2020). "Integrasi Naive Bayes Dengan Teknik Sampling Smote Untuk Menangani Data Tidak Seimbang." *Nuansa Informatika*, 14(1), 34. doi:10.25134/nuansa.v14i1.2411