

## Implementasi Metode *Support Vector Machine* dalam Analisis Sentimen Opini Masyarakat Terhadap Pilkada 2020 pada Media Sosial Twitter

Martin Paramarta<sup>1</sup>, J.B. Budi Darmawan<sup>2</sup>

<sup>1,2</sup>Program Studi Informatika, Universitas Sanata Dharma Yogyakarta

Korespondensi: b.darmawan@usd.ac.id

### ABSTRAK

Pemilihan kepala daerah (Pilkada) tahun 2020 mengalami penundaan selama beberapa bulan karena efek pandemi COVID-19, hingga akhirnya ditetapkan untuk tetap digelar pada tanggal 9 Desember 2020. Banyak masyarakat beropini tentang pro dan kontra pelaksanaan Pilkada di tengah pandemi COVID-19 di media sosial, khususnya Twitter. Penelitian ini bertujuan untuk mengetahui jumlah sentimen positif dan negatif serta performansi dari metode *Support Vector Machine* dalam mengklasifikasikan sentimen *tweet* tersebut. Data penelitian bersumber dari *tweet* dengan kata kunci "Pilkada 2020" yang berjumlah 6037 *tweet*. Data akan diberi label polaritas sentimen positif dan negatif secara otomatis. Hasil pengujian menunjukkan sebanyak 4864 data bersentimen positif dan 1173 data lainnya bersentimen negatif. Selain itu, hasil pengujian dalam penelitian ini menunjukkan bahwa metode *Support Vector Machine* memiliki performansi yang cukup baik dalam mengklasifikasikan sentimen *tweet* dengan hasil rata-rata akurasi sebesar 87,94%.

**Kata kunci:** analisis sentimen, klasifikasi, pilkada, *support vector machine*, *tweet*.

### ABSTRACT

The 2020 regional elections were postponed for several months because of the COVID-19 pandemic, until finally set to be held on December 9, 2020. Many people have opinions about the pros and cons of holding regional elections during the COVID-19 pandemic on social media, especially Twitter. This research aims to determine the number of positive and negative sentiments and the performance of the *Support Vector Machine* method in classifying *tweet* sentiments. The research data was sourced from *tweets* with the keyword "Pilkada 2020" which amounted to 6037 *tweets*. The data will be labelled positive and negative sentiment polarity automatically. The test results showed as many as 4864 data with positive sentiment and 1173 other data with negative sentiment. In addition, the test results in this research show that the *Support Vector Machine* method has a fairly good performance in classifying *tweet* sentiment with an average accuracy result of 87.94%.

**Keywords:** classification, regional elections, sentiment analysis, *support vector machine*, *tweet*.

### PENDAHULUAN

Pemilihan umum (Pemilu) menjadi sarana agregasi aspirasi masyarakat dalam menyatakan kehendak menentukan individu yang mewakili mereka duduk dalam lembaga perwakilan politik dan pemerintahan baik sebagai presiden dan wakil presiden, anggota legislatif pusat dan daerah, gubernur dan wakil gubernur, bupati atau walikota dan wakil bupati atau wakil walikota [1]. Sebelumnya, pemilihan kepala daerah (Pilkada) serentak di Indonesia telah direncanakan akan berlangsung pada tanggal 23 September 2020, namun karena pandemi COVID-19 (*Coronavirus Disease-19*) yang melanda dunia berdampak ke berbagai sektor, salah satunya di sektor ketatanegaraan maka Pilkada mengalami penundaan pelaksanaan. Selanjutnya, Pemerintah, DPR RI, dan Penyelenggara Pemilu (Bawaslu, KPU, dan DKPP) menetapkan bahwa Pilkada tetap dilaksanakan walaupun di masa pandemi COVID-19, yaitu pada tanggal 9 Desember 2020 tentunya dengan menerapkan protokol kesehatan yang ketat.

Pasca ditetapkannya keputusan Pilkada tetap dilaksanakan di masa pandemi COVID-19, ternyata banyak memicu opini masyarakat di dunia maya, khususnya di media sosial Twitter. Twitter adalah salah satu media sosial yang cukup populer saat ini. Keberadaan Twitter memungkinkan penggunaannya untuk berbagi pendapat, opini, kritik atau apapun yang berbentuk teks singkat dengan batasan karakter maksimal 140 karakter. Teks singkat ini akrab disebut dengan istilah *tweet*.

Melalui Twitter, pengguna dapat menyampaikan komentar atau pendapat yang seringkali disertai dengan emosi. Pengenalan emosi pada *tweet* dapat dilakukan dengan analisis sentimen untuk mengetahui *tweet*

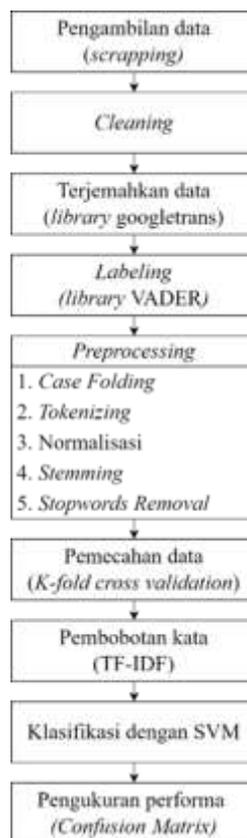
merupakan emosi yang bersifat positif atau negatif. Analisis sentimen dilakukan dengan cara mengklasifikasikan sejumlah *tweet* dengan metode *Support Vector Machine (SVM)*.

Adapun penelitian dengan topik serupa namun dengan algoritma yang berbeda tentang analisis sentimen masyarakat di Twitter terhadap Pilkada 2020 di tengah pandemi COVID-19 dengan menggunakan algoritma *Naïve Bayes*. Dari penelitian pada jurnal tersebut diperoleh hasil akurasi sebesar 92,2% [2]. Sementara itu, terdapat pula penelitian analisis sentimen dengan salah satu algoritma serupa yang membandingkan akurasi algoritma *Naïve Bayes* dan *Support Vector Machine* dengan topik tentang Presiden Jokowi 3 periode dengan sumber data yang berasal dari Twitter. Penelitian tersebut melakukan pelabelan sentimen secara manual, menghasilkan akurasi sebesar 94,07% untuk algoritma *Naïve Bayes* dan akurasi sebesar 95,42% untuk algoritma *Support Vector Machine* [3].

Oleh karena itu, peneliti tertarik mengambil topik tentang analisis sentimen menggunakan metode klasifikasi *Support Vector Machine* dengan data *tweet* tentang Pilkada 2020 atas dasar banyaknya opini masyarakat di Twitter tentang pro kontra pelaksanaan Pilkada 2020 dengan pelabelan secara otomatis menggunakan *library VADER*. Dari penelitian ini diharapkan mampu membantu pihak penyelenggara Pilkada dalam melakukan evaluasi Pilkada 2020 yang telah berlangsung agar tercapai pelaksanaan Pilkada yang lebih baik di masa mendatang.

### METODE PENELITIAN

Terdapat beberapa tahapan untuk memperoleh hasil dari klasifikasi analisis sentimen opini masyarakat terhadap Pilkada 2020. Tahapan berikut mengacu pada teori *text mining* yang telah dipaparkan dalam landasan teori bab sebelumnya. Berikut tahapan tersebut ditampilkan pada Gambar 1.



Gambar 1. Diagram Tahapan Penelitian



ISSN: 1907-5995

### Pengambilan Data

Data diambil secara bertahap selama bulan Desember 2020. Data berupa teks diambil menggunakan teknik *scraping* dari media sosial Twitter dengan kata kunci “pilkada 2020”. *Scraping* merupakan teknik mengumpulkan data atau informasi berupa yang dapat berupa teks, gambar, video, atau lainnya dari suatu *website* secara otomatis [4].

### Cleaning

Hasil pengambilan data akan dibersihkan dari atribut-atribut yang tidak diperlukan. Atribut selain teks dianggap sebagai *noise*, karena data yang dibutuhkan dari *tweet* hanya data teks saja. Contoh *noise* yang akan dibuang adalah *link* URL, tanda baca, angka, dan karakter lain yang tidak dibutuhkan.

### Terjemahkan Data

Data *tweet* yang telah dibersihkan seluruhnya akan diterjemahkan ke dalam bahasa Inggris agar nanti dapat diproses oleh modul VADER, menggunakan bantuan *library* *googletrans* yang merupakan modul API (*Application Programming Interface*) Google Translate.

### Labelling

Proses *labeling* atau pemberian label polaritas sentimen dilakukan dengan menggunakan modul VADER yang merupakan salah satu modul dari *library* *Natural Language Toolkit* yang dikembangkan dalam bahasa pemrograman Python. VADER merupakan model *rule-based* sederhana berbasis *lexicon* untuk menangani kasus analisis sentimen. Pemilihan modul VADER didasari oleh performa yang sangat baik untuk *domain* media sosial dibanding model analisis sentimen berbasis *lexicon* lainnya [5].

### Preprocessing

*Preprocessing* bertujuan untuk mempersiapkan data dan mengubah data menjadi lebih terstruktur sebelum diolah dalam proses klasifikasi. Tahap *preprocessing* meliputi analisis leksikal untuk memperlakukan angka, tanda hubung, tanda baca, dan huruf kapital, kemudian penghapusan *stopwords* untuk menyaring kata dengan nilai diskriminasi rendah, lalu *stemming* untuk menghapus imbuhan, setelah itu pemilihan indeks kata untuk menentukan kata sebagai elemen pengindeksan, dan yang terakhir adalah konstruksi struktur tesaurus [6].

### Pemecahan Data

Setelah melewati tahap *preprocessing* data kemudian akan dipecah menggunakan *k-fold cross validation*. Sampel data akan dibagi secara acak lalu dikelompokkan sebanyak nilai *k*. Kelompok *k-fold* dibagi menjadi dua bagian, salah satu menjadi *data testing* dan sisanya menjadi *data training* [7].

### Pembobotan Kata

Setiap kata dalam dokumen kemudian akan diberi bobot dengan TF-IDF yang meliputi pembobotan *term frequency* (*tf*) dan *inverse document frequency* (*idf*). Berikut persamaan untuk menghitung IDF yang ditunjukkan pada persamaan (1) [8].

$$idf_t = \log\left(\frac{n+1}{df_{a,t}+1}\right) + 1 \quad (1)$$

Keterangan:

$idf_t$  : *inverse document frequency* pada kata  $t$ .  
 $n$  : jumlah keseluruhan dokumen.  
 $df_{a,t}$  : banyak dokumen yang mengandung kata  $t$ .  
 $\log$  : 2,718281828459045 (*natural logarithm*)

Kemudian persamaan untuk mendapatkan nilai bobot kata ditunjukkan pada persamaan (2).

$$W_{a,t} = tf_{a,t} * idf_t \quad (2)$$

Keterangan:

$W_{a,t}$  : bobot kata  $t$  pada dokumen  $d$   
 $tf_{a,t}$  : frekuensi kata  $t$  pada dokumen  $d$   
 $IDF_t$  : *inverse document frequency*  $t$

**Klasifikasi dengan SVM**

Setelah proses pembobotan kata, data akan diklasifikasikan dengan metode SVM. Klasifikasi dengan metode SVM melibatkan beberapa *hyperparameter*, seperti nilai  $\gamma$ , nilai  $C$ , dan nilai *degree* (khusus *kernel polynomial*), dan fungsi *kernel*. Fungsi *kernel* memberikan banyak kemudahan, untuk menentukan *support vector* dalam proses *training* SVM hanya cukup mengetahui fungsi *kernel* yang dipakai, dan tanpa harus mengetahui bentuk dari fungsi *non-linear*  $\phi$  [9]. Berikut beberapa *fungsi* kernel beserta persamaannya yang umum dipakai dalam SVM yang ditunjukkan pada Tabel 1.

**Tabel 1.** Fungsi *Kernel* SVM dan Persamaannya

<b>Kernel</b>	<b>Persamaan</b>
<i>Linear</i>	$K(x_i, x_i) = (x_i, x_i)$
<i>Polynomial</i>	$K(x_i, x_i) = (\gamma(x_i, x_i) + r)^d$
<i>Sigmoid</i>	$K(x_i, x_i) = \tanh(\gamma(x_i, x_i) + r)$
<i>Radial Basis Functions (RBF)</i>	$K(x_i, x_i) = \exp(-\gamma\ x_i - x_i\ ^2 + C)$

Keterangan:

- $x$  : data
- $\gamma$  : *gamma*
- $C$  : *cost*
- $r$  : *coefficient*
- $d$  : *degree*.

**Pengukuran Performa**

Evaluasi performa dengan *confusion matrix* dilakukan dengan cara membandingkan hasil aktual dengan hasil klasifikasi dengan metode SVM. Terdapat empat rasio prediksi dalam *confusion matrix* untuk merepresentasikan nilai hasil klasifikasi, yaitu TP (*True Positive*), TN (*True Negative*), FP (*False Positive*), dan FN (*False Negative*) [10]. Berikut empat rasio prediksi *confusion matrix* yang ditunjukkan pada Tabel 2.

**Tabel 2.** *Confusion Matrix*

	<b>Actual Values</b>	
	<b>Positive (+1)</b>	<b>Negative (-1)</b>
<b>Predicted Values</b>	<i>Positive (+1)</i>	TP      FP
	<i>Negative (-1)</i>	FN      TN

Keterangan:

1. TP (*True Positive*) merupakan jumlah data positif yang terklasifikasi benar oleh sistem.
2. TN (*True Negative*) merupakan jumlah data negatif yang terklasifikasi benar oleh sistem.
3. FP (*False Positive*) merupakan jumlah data positif yang terklasifikasi salah oleh sistem.
4. FN (*False Negative*) merupakan jumlah data negatif yang terklasifikasi salah oleh sistem.

Hasil dari *confusion matrix* dapat digunakan untuk mengukur performansi sebuah algoritma *machine learning*. Berikut adalah persamaan untuk mengukur performansi algoritma *machine learning* [11]:

1. *Accuracy*

*Accuracy* adalah rasio data benar (*true positive* dan *true negative*) dari keseluruhan data ditunjukkan pada persamaan (3).

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \tag{3}$$

## 2. Precision

*Precision* adalah rasio data benar positif (*true positive*) dibandingkan dengan keseluruhan data yang diprediksi positif (*true positive* dan *false positive*) ditunjukkan pada persamaan (4).

$$Precision = \frac{TP}{TP+FP} \quad (4)$$

## 3. Recall

*Recall* adalah rasio data benar positif (*true positive*) dibandingkan dengan data positif aktual (*true positive* dan *false negative*) ditunjukkan pada persamaan (5).

$$Recall = \frac{TP}{TP+FN} \quad (5)$$

## 4. F1-score

*F1-score* adalah perbandingan nilai rata-rata *precision* dan *recall* ditunjukkan pada persamaan (6).

$$F1 - score = \frac{2(Recall \times Precision)}{Recall + Precision} \quad (6)$$

## HASIL DAN ANALISIS

Pengujian dilakukan dengan mengubah variasi parameter nilai k untuk *k-fold cross validation*, variasi *hyperparameter* fungsi *kernel*, variasi *hyperparameter* nilai *gamma*, variasi *hyperparameter* nilai C, dan variasi *hyperparameter* nilai *degree* (khusus untuk *kernel polynomial*). Pengujian terhadap sistem klasifikasi yang bertujuan untuk menemukan performa *kernel SVM* terbaik dalam menangani kasus analisis sentimen. Dengan variasi nilai k (3, 5, 7, 9, dan 10), variasi fungsi *kernel* (*linear*, *polynomial*, *sigmoid*, dan *RBF*), variasi nilai *gamma* (3, 5, 10), variasi nilai C (0,1, 1, dan 10), dan variasi nilai *degree* (1, 3, dan 5) diperoleh akurasi dan variasi nilai parameter dan *hyperparameter* terbaik untuk masing-masing fungsi *kernel* yang ditampilkan pada Tabel 2.

**Tabel 2.** Parameter dan *Hyperparameter* Hasil Akurasi Terbaik

<i>Kernel</i>	Nilai <i>k-fold</i>	<i>gamma</i>	C	<i>degree</i>	Akurasi
RBF	10	3	10	<i>default</i>	87,94%
Sigmoid	7	10	1	<i>default</i>	87,49%
<i>Linear</i>	9	3	10	<i>default</i>	87,23%
Polynomial	9	10	10	1	86,50%

Untuk mengukur kinerja sistem dibutuhkan pula nilai rasio data lain seperti *precision*, *recall*, dan *f1-score*. Berikut nilai rata-rata *precision*, *recall*, dan *f1-score* yang didapat dari *confusion matrix* untuk masing-masing *kernel* berdasarkan nilai rata-rata akurasi terbaik yang ditampilkan pada Tabel 3.

**Tabel 3.** *Recall*, *Precision*, dan *F1-Score* dari Akurasi Terbaik

<i>Kernel</i>	Akurasi	<i>Recall</i>	<i>Precision</i>	<i>F1-Score</i>
RBF	87,94%	90,34%	95,11%	92,64%
Sigmoid	87,49%	88,52%	96,97%	92,54%
<i>Linear</i>	87,23%	88,44%	96,67%	92,36%
Polynomial	86,50%	90,59%	92,70%	91,62%

Dari Tabel 2 dan Tabel 3 nilai akurasi terbaik sebesar 87,94% diperoleh dari pengujian dengan fungsi *kernel RBF*, dengan parameter k sebesar 10, *hyperparameter gamma* sebesar 3, dan *hyperparameter C* sebesar 10 dengan *recall* sebesar 90,34%, *precision* sebesar 95,10%, dan *f1-score* sebesar 92,65%.

## KESIMPULAN

Berdasarkan hasil dan analisis yang diperoleh dari implementasi metode SVM dalam analisis sentimen opini masyarakat terhadap Pilkada 2020 pada media sosial Twitter menggunakan sebanyak 6037 data dengan pelabelan sentimen secara otomatis dengan *library VADER* yang menghasilkan sebanyak 4864 data bersentimen positif dan 1173 data lainnya bersentimen negatif, dapat ditarik kesimpulan bahwa nilai rata-rata

akurasi tertinggi diperoleh dari pengujian dengan fungsi *kernel* RBF, dengan parameter *k* sebesar 10, *hyperparameter* *gamma* sebesar 3, dan *hyperparameter* *C* sebesar 10 menghasilkan nilai rata-rata akurasi sebesar 87,94%, nilai rata-rata *recall* sebesar 90,34%, nilai rata-rata *precision* sebesar 95,10%, dan nilai rata-rata *f1-score* sebesar 92,65%.

#### DAFTAR PUSTAKA

- [1] J. R. A. Sandi And . S., “Fenomena Pengawasan Pemilihan Kepala Daerah Di Kalimantan Tengah Masa Pandemi Covid-19,” *Jurnal Politik Pemerintahan Dharma Praja*, Vol. 13, No. 1, Pp. 1–13, Jun. 2020, Doi: 10.33701/Jppdp.V13i1.1072.
- [2] A. Muzaki And A. Witanti, “Sentimen Analisis Masyarakat Di Twitter Terhadap Pilkada 2020 Ditengah Pandemic Covid-19 Dengan Metode Naïve Bayes Classifier,” *Jurnal Teknik Informatika (Jutif)*, Vol. 2, No. 2, Pp. 101–107, Mar. 2021, Doi: 10.20884/1.Jutif.2021.2.2.51.
- [3] Luthfanida, “Analisis Sentimen Data Twitter Menggunakan Metode Naive Bayes Dan Support Vector Machine (Svm) Tentang Presiden Jokowi 3 Periode,” *Journal Of Information Technology Research*, Vol. 3, Jun. 2022.
- [4] V. A. Flores, P. A. Permatasari, And L. Jasa, “Penerapan Web Scraping Sebagai Media Pencarian Dan Menyimpan Artikel Ilmiah Secara Otomatis Berdasarkan Keyword,” *Majalah Ilmiah Teknologi Elektro*, Vol. 19, No. 2, P. 157, Dec. 2020, Doi: 10.24843/Mite.2020.V19i02.P06.
- [5] C. Hutto And E. Gilbert, “Vader: A Parsimonious Rule-Based Model For Sentiment Analysis Of Social Media Text,” *Proceedings Of The International Aaai Conference On Web And Social Media*, Vol. 8, No. 1, Pp. 216–225, May 2014, Doi: 10.1609/Icwsml.V8i1.14550.
- [6] A. Shiri, *Introduction To Modern Information Retrieval (2nd Edition)*. Emerald Group Publishing Limited, 2004. Doi: 10.1108/00242530410565256.
- [7] A. Hutapea And M. Tanzil Furqon, “Penerapan Algoritme Modified K-Nearest Neighbour Pada Pengklasifikasian Penyakit Kejiwaan Skizofrenia,” *Jurnal Pengembangan Teknologi Informasi Dan Ilmu Komputer*, Vol. 2, No. 10, Pp. 3957–3961, 2018, [Online]. Available: [Http://J-Ptiik.Ub.Ac.Id](http://J-Ptiik.Ub.Ac.Id)
- [8] S. Jansen, *Hands-On Machine Learning For Algorithmic Trading : Design And Implement Investment Strategies Based On Smart Algorithms That Learn From Data Using Python*. Packt Publishing, 2018.
- [9] J. Asian, H. E. Williams, And S. M. M. Tahaghoghi, “Stemming Indonesian,” In *Conferences In Research And Practice In Information Technology Series*, 2005, Pp. 307–314. Doi: 10.1145/1316457.1316459.
- [10] J. Han, M. Kamber, And J. Pei, *Data Mining. Concepts And Techniques*, 3rd Edition (The Morgan Kaufmann Series In Data Management Systems). 2011.
- [11] S. Ghoneim, “Accuracy, Recall, Precision, F-Score & Specificity, Which To Optimize On?,” *Towards Data Science*, Apr. 02, 2018. <https://Towardsdatascience.Com/Accuracy-Recall-Precision-F-Score-Specificity-Which-To-Optimize-On-867d3f11124>