

Analisis Sentimen Masyarakat Terhadap Hasil Kerja Petahana Dalam Kaitan Dengan Pemilihan Presiden tahun 2019 Pada Sosial Media Twitter Menggunakan *Support Vector Machine* (SVM)

Ridea Valentini Peristiwari Siwabessy¹, Anisa Herdiani², Ade Romadhony³

^{1,2,3}Fakultas Informatika, Universitas Telkom, Bandung

¹rideavalentini@students.telkomuniversity.ac.id, ²anisaherdiani@telkomuniversity.ac.id,

³aderomadhony@telkomuniversity.ac.id

Abstrak

Calon presiden dan wakil presiden pada pemilihan umum tahun 2019, terdiri dari petahana dan salah satu calon presiden yang baru. Petahana telah bekerja selama 4 tahun untuk membangun Indonesia. Dalam masa kepemimpinannya banyak program kerja yang telah dikerjakannya. Berbagai penilaian terhadap hasil kerjanya bermunculan. Ada yang pro, tetapi juga ada yang kontra dengan hasil kerja petahana. Sosial media Twitter merupakan salah satu platform yang sering digunakan untuk menyampaikan berbagai penilaian terhadap hasil kerja petahana. Informasi yang terdapat pada Twitter berupa pertanyaan, opini atau komentar, baik yang bersifat positif maupun negatif. Setiap tweet yang menyatakan apresiasi maupun penolakan merupakan bentuk ekspresi dari masyarakat sebagai respon terhadap hasil kerja petahana. Dalam penelitian ini, dibangun sebuah sistem yang dapat mengklasifikasikan tweet berdasarkan sentiment masyarakat terhadap hasil kerja sang petahana berdasarkan tweet. Untuk mengklasifikasikan sentimen berdasarkan parameternya digunakan metode Support Vector Machine (SVM) sebagai classifiernya. Hasil yang didapatkan bahwa skenario 3 (kombinasi TF-IDF + *Stemming*) dan skenario 8 (kombinasi *Word Count* + *Stemming*) memiliki akurasi baik yaitu 81,58% dan 77,56%.

Kata kunci : *support vector machine*, sentimen, twitter, pilpres

Abstract

Presidential and vice presidential candidates in the 2019 general election, consisting of incumbent and one of the new presidential candidates. Incumbent has worked for 4 years to develop Indonesia. In his leadership program many work programs he has done. Various considerations on the results of discussions emerged. There are pros, but there are also cons with the work of incumbents. Social media Twitter is one of platform that is often used to present various assessments of incumbent's work. Information suggested on Twitter includes questions, opinions or comments, both positive and negative. Every tweet that expresses appreciation is also a form of response from the community in response to the incumbent's work. In this study, a system was built that could classify tweets based on community sentiment towards the incumbent's work based on tweets. To classify sentiments based on their parameters the Support Vector Machine (SVM) method is used as the classifier. The results obtained were scenario 3 (TF-IDF + *Stemming* combination) and scenario 6 (*Word Count* + *Stemming* combination) have good accuracy that is 81,58% and 77,56%.

Keywords: *support vector machine*, sentiment, twitter, general election

1. Pendahuluan

Pemilihan umum Presiden dan Wakil Presiden Republik Indonesia telah dilaksanakan pada bulan Rabu, 17 April 2019 [1]. Salah satu calon presiden yang maju dalam pemilihan umum 2019 adalah sang petahana, yaitu Bapak Joko Widodo. Selama 4 tahun petahana telah bekerja untuk memimpin dan membangun negara Indonesia. Petahana telah mengerjakan tugasnya sesuai dengan program kerja yang disampaikan di awal kepemimpinannya. Berbagai penilaian terhadap hasil kerjanya bermunculan dan semakin marak dibicarakan di dunia nyata maupun media nyata. Masyarakat memiliki hak dan kebebasan untuk berpendapat dan beropini mengenai hasil kerja petahana melalui berbagai media yang tersedia. Opini yang disampaikan pun beragam, baik positif, netral ataupun negatif.

Berdasarkan data dari We Are Social yang bekerjasama dengan Hootsuite [2], menyatakan bahwa setengah dari total populasi penduduk Indonesia yang mencapai 265,4 juta jiwa adalah pengguna internet yaitu sekitar 132,7 juta jiwa. 130 juta jiwa diantaranya adalah pengguna aktif media sosial. Hal tersebut menunjukkan bahwa penyebaran informasi di kalangan masyarakat Indonesia cukup pesat. Salah satu *platform* yang sering digunakan oleh masyarakat Indonesia adalah Twitter. Sekitar 27% pengguna internet menggunakan *platform* twitter.

Melihat *trend* penggunaan sosial media terkhusus twitter saat ini, masyarakat bukan hanya sebatas menunjukkan eksistensinya secara personal, tetapi juga mulai diarahkan untuk melakukan kritik-kritik sosial. Kritik-kritik sosial yang dilakukan salah satunya adalah menilai kinerja pemerintah. Melalui *hashtag* maupun *tweet*, masyarakat dengan bebas memberikan opini atau pendapat melalui twitter.

Beberapa penelitian sebelumnya telah dilakukan mengenai *Sentiment Analysis*. Kharde dalam jurnalnya [3], menulis bahwa klasifikasi menggunakan SVM pada data Twitter lebih tinggi nilai akurasi sebesar 76,68% dibandingkan dengan *Maximum Entropy* dan *Naïve Bayes Classifier*. Selanjutnya perkembangan penelitian *Sentiment Analysis* pada teks bahasa Indonesia semakin berkembang. Taufik [4] telah melakukan penelitian analisis sentimen terhadap beberapa tokoh publik pada sosial media twitter dengan menggunakan SVM. Penelitian ini menggunakan TF-IDF sebagai ekstraksi fitur dan algoritma *Support Vector Machine* (SVM). Nurirwan [5] dalam penelitiannya dijelaskan bahwa akurasi yang terbaik dalam penelitian tersebut adalah dengan dilakukan normalisasi dan *stemming* pada data menggunakan metode SVM dibandingkan metode *Naïve Bayes* dalam menganalisis data presiden Joko Widodo.

Maka dalam penelitian ini dilakukan analisis sentimen terhadap hasil kerja petahana menggunakan TF-IDF dan algoritma *Support Vector Machine* (SVM). Metode tersebut digunakan karena berdasarkan penelitian-penelitian sebelumnya, SVM memiliki nilai akurasi yang cukup baik dibandingkan dengan metode yang lain.

Berdasarkan latar belakang yang telah diuraikan, adapun rumusan masalah pada penulisan ini yaitu:

1. Bagaimana cara mengklasifikasikan *tweet* berdasarkan sentimen yang berkaitan dengan hasil kerja petahana?
2. Bagaimana mendapatkan performansi yang baik pada sistem untuk mengklasifikasikan *tweet* berdasarkan sentimen yang berkaitan dengan hasil kerja petahana menggunakan metode *Support Vector Machine* (SVM)?

Batasan masalah pada penelitian ini yaitu:

1. Data yang digunakan adalah *tweet* bahasa Indonesia.
2. Konfigurasi kernel SVM tidak akan dikaji dalam penelitian ini.
3. Label yang digunakan hanya 2 yaitu, positif dan negatif.

Berdasarkan rumusan masalah yang ada, maka tujuan dari penelitian ini adalah sebagai berikut:

1. Untuk mengklasifikasikan *tweet* berdasarkan sentimen yang berkaitan dengan hasil kerja petahana menggunakan *Support Vector Machine* (SVM).
2. Untuk mendapatkan performansi dari hasil klasifikasi *tweet* berdasarkan sentimen yang berkaitan dengan hasil kerja petahana menggunakan *Support Vector Machine* (SVM).

Urutan penulisan laporan adalah sebagai berikut: bagian pertama menjelaskan mengenai latar belakang penelitian yang dilakukan Pada bagian kedua menjelaskan tentang studi terkait dari penelitian yang dilakukan sebelumnya. Pada bagian ketiga menjelaskan mengenai cara kerja sistem yang dibangun. Pada bagian keempat menjelaskan mengenai hasil dan analisis dari hasil pengujian yang dilakukan. Pada bagian kelima adalah kesimpulan dari penelitian yang telah dilakukan.

2. Studi Terkait

2.1 Analisis sentimen

Analisis sentimen dan *opinion mining* adalah bidang studi yang menganalisis pendapat seseorang, sentiment seseorang, evaluasi seseorang, sikap seseorang dan emosi seseorang ke dalam bahasa tertulis [5]. Perkembangan dunia maya yang semakin cepat, menyebabkan berbagi perusahaan, lembaga ataupun perorangan menggunakan konten sosial media sebagai salah satu parameter dalam pengambilan keputusan.

Secara general, analisis sentimen dibagi dalam tiga level [6], yaitu:

1. Level dokumen
Level ini mengklasifikasikan apakah seluruh dokumen mengekspresikan sentimen positif atau negatif. Biasanya, level ini dikenal sebagai klasifikasi tingkat dokumen.
2. Level kalimat
Level ini menentukan apakah setiap kalimat menyatakan sentimen positif, negatif atau netral. Level dokumen dan kalimat memiliki kemiripan yaitu tidak dapat dianalisis secara langsung apa yang sebenarnya disukai atau tidak disukai orang.
3. Level entitas atau aspek
Level aspek lebih spesifik dalam menentukan sentimen, karena difokuskan terhadap aspek atau entitas yang dinilai dari keseluruhan dokumen atau kalimat.

2.2 Term Frequency – Inverse Document Frequency (TF-IDF)

Term Frequency adalah perhitungan yang sederhana dalam proses pembobotan. Masing-masing *term* pada metode ini diasumsikan mempunyai proporsi kepentingan sesuai jumlah kemunculannya dalam teks/dokumen [7]. Berikut persamaan yang digunakan untuk menghitung nilai TF [8], yaitu:

$$TF_{(t)} = \frac{f_{t,d}}{\sum t, d} \quad (1)$$

Dimana:

$f_{t,d}$ = frekuensi setiap kata (t) muncul di dalam dokumen d

$\sum t, d$ = total keseluruhan kata yang terdapat di dalam dokumen d

Inverse Document Frequency adalah metode pembobotan *term* yang memperhatikan kemunculan *term* pada keseluruhan kumpulan teks. Pada IDF, *term* yang jarang muncul pada keseluruhan koleksi teks dinilai lebih berharga [7]. Berikut persamaan yang digunakan untuk menghitung nilai IDF [8], yaitu:

$$IDF_{(t)} = \log \frac{|D|}{f_{t,d}} \quad (2)$$

Dimana:

$|D|$ = jumlah dokumen yang ada dalam koleksi

$f_{t,d}$ = jumlah dokumen dimana muncul t di dalam dokumen D

Maka, formulasi dari TF-IDF adalah

$$TF - IDF = TF * IDF \quad (3)$$

2.3 Support Vector Machine (SVM)

Support vector machine (SVM) adalah suatu teknik untuk melakukan prediksi, baik dalam kasus klasifikasi maupun regresi [9]. SVM termasuk dalam kelas *supervised learning*. Teknik SVM digunakan untuk menemukan fungsi pemisah (*hyperplane*) dengan memaksimalkan jarak antar kelas (*margin*) agar mampu mengklasifikasikan data secara akurat. Misalkan data yang ada pada himpunan data *training* dinotasikan sebagai $x_i \in R^d$ sedangkan label kelas dinotasikan sebagai $y_i \in -1, +1$ untuk $i = 1, 2, \dots, l$, dimana l adalah jumlah data. Misalkan kedua kelas -1 dan +1 diasumsikan dapat dipisahkan secara sempurna oleh *hyperplane* berdimensi d yang didefinisikan pada persamaan (4), (5) dan (6) [10].

$$w \cdot x + b = 0 \quad (4)$$

Sebuah data x_i diklasifikasikan sebagai kelas -1 jika

$$w \cdot x_i + b \leq -1 \quad (5)$$

dan diklasifikasikan sebagai kelas +1 jika

$$w \cdot x_i + b > 1 \quad (6)$$

Dimana:

w = kemiringan baris

b = bias untuk mengatur letak garis

Untuk mencari *hyperplane* yang optimum, dilakukan dengan salah satu cara, yaitu *Lagrange multiplier* dapat dalam persamaan (7) [11].

$$L(w, b, \alpha) = \frac{1}{2} \|w\|^2 - \sum_{i=1}^l \alpha_i y_i (x_i \cdot w + b) - 1, i = 1, 2, \dots, l \quad (7)$$

Dimana $\alpha_i \geq 0$ adalah *Lagrange multipliers*

Untuk mendapatkan nilai optimal pada persamaan (7), maka dapat meminimalkan L terhadap w dan b sekaligus memaksimalkan L terhadap α_i . Persamaan (7) dapat dimodifikasi karena mengingat titik optimal gradien $L=0$, dengan memaksimalkannya pada persamaan (8) dengan batasan persamaan (9) [11].

$$\sum_{i=1}^l \alpha_i - \frac{1}{2} \sum_{i,j=1}^l \alpha_i \alpha_j y_i y_j x_i x_j \quad (8)$$

$$\alpha_i \geq 0 (i = 1, 2, \dots, l) \sum_{i=1}^l \alpha_i y_i = 0 \tag{9}$$

Hasil pemaksimalan ini menghasilkan sejumlah α_i yang bernilai positif. Data yang berhubungan dengan α_i positif inilah yang disebut sebagai *support vector*. Selanjutnya dapat diketahui *hyperplane* yang optimum dari klasifikasi SVM ini dengan mencari persamaan $w \cdot x + b = 0$ dengan nilai vektor w dan skalar b dari persamaan (10) dan (11) [11].

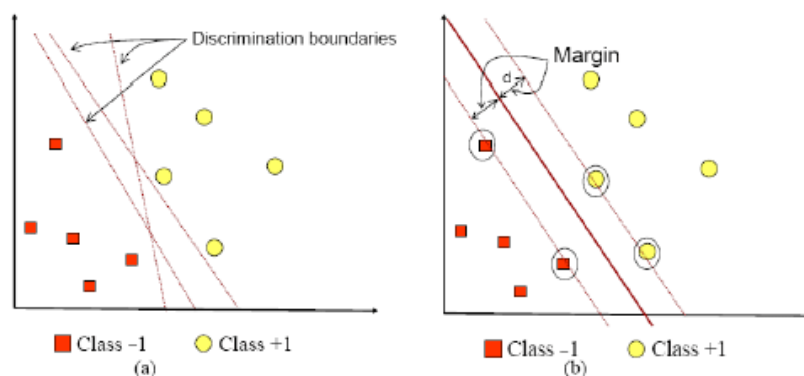
$$w = \sum_{i=1}^n \alpha_i y_i x_i \tag{10}$$

$$b = y_i - w^T x \tag{11}$$

Setelah ditemukan *hyperplane* yang optimum, maka selanjutnya adalah mencari vektor yang tegak lurus dengan margin yang diberi nama w yang memiliki panjang yang sama dengan margin. Vektor w berbanding terbalik dengan margin. Sehingga untuk mendapatkan *hyperplane* yang optimum dapat mencari margin maksimum dengan mencari nilai terkecil dari vektor w melalui persamaan (12) [11].

$$margin = \frac{2}{\|w\|} \tag{12}$$

Setelah mendapatkan margin yang maksimal dan nilai α_i konvergen (perubahan relatif kecil) maka algoritma pembelajaran suport vector machine telah menemukan *hyperplane* optimal.



Gambar 1. Ilustrasi Metode Support Vector Machine [12]

Dalam pengolahan teks pada proses klasifikasi menggunakan SVM dimulai dengan mengubah text menjadi data vektor. Vektor dalam penelitian ini memiliki dua komponen yaitu dimensi (*word id*) dan bobot. Bobot ini sering dikombinasikan ke dalam sebuah nilai tf-idf, secara sederhana dengan mengalikan mereka bersama-sama. Penilaian kemudian dibuat dengan menilai *score* apakah positif atau negatif yang merepresentasikan di sisi mana dari garis pemisah dokumen berada [13].

2.4 Performansi

Mengukur performansi sangat penting untuk mengetahui apakah model yang dibangun sudah sesuai dengan yang diharapkan atau tidak. Dengan menggunakan tabel *confusion matrix* kita dapat menentukan nilai *accuracy*, *precision*, *recall* dan *F1-measure*.

Tabel 1. Confusion Matrix [12]

Actual	Prediction	
	Positive	Negative
Positive	True Positive	False Negative
Negative	False Positive	True Negative

TP menunjukkan nilai yang diperoleh dari data jika hasil klasifikasi pada sistem dengan hasil aktualnya sama-sama bernilai *true*. Sedangkan TN menunjukkan nilai yang diperoleh dari data jika hasil klasifikasi pada sistem

dengan hasil aktualnya sama-sama bernilai *false*. FP menunjukkan nilai yang diperoleh dari data jika hasil klasifikasi pada sistem bernilai *true*, tetapi hasil aktualnya bernilai *false*. FN menunjukkan nilai yang diperoleh dari data jika hasil klasifikasi pada sistem bernilai *false*, tetapi hasil aktualnya bernilai *true*.

a. *Accuracy*

Akurasi merupakan tingkat kedekatan antara nilai prediksi dengan nilai sebenarnya. Rumus untuk mencari akurasi adalah sebagai berikut:

$$Accuracy = \frac{TP + TN}{TP + FP + TN + FN} \quad (13)$$

b. *Precision*

Presisi merupakan tingkat ketepatan antara informasi yang diminta oleh pengguna dengan jawaban yang diberikan oleh sistem. Rumus untuk mencari *precision* adalah sebagai berikut:

$$Precision = \frac{TP}{TP + FP} \quad (14)$$

c. *Recall*

Recall adalah tingkat keberhasilan sistem dalam menemukan kembali sebuah informasi. Rumus untuk mencari *recall* adalah sebagai berikut:

$$Recall = \frac{TP}{TP + FN} \quad (15)$$

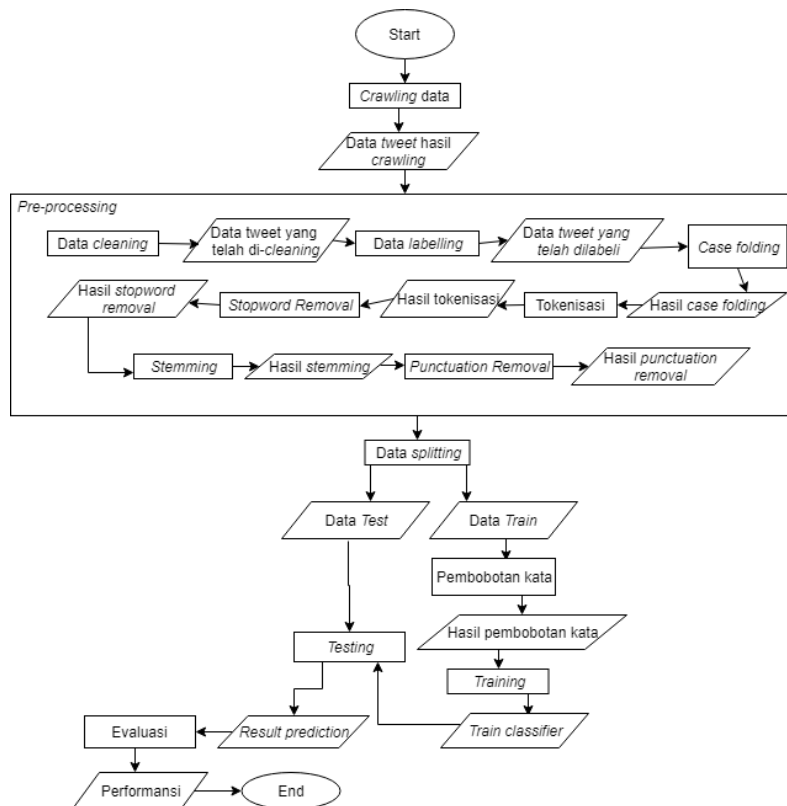
d. *F1-Measure*

Pengukuran F1 merupakan ukuran yang lebih baik untuk digunakan jika ingin mencari keseimbangan antara *precision* dan *recall*. Dikarenakan jika hanya menggunakan nilai *precision*, *recall* dan *accuracy* saja dapat menimbulkan bias yang cukup fatal. Rumus untuk mencari *F1-Measure* adalah sebagai berikut:

$$F1 = \frac{2 (Precision \times Recall)}{Precision + Recall} \quad (16)$$

3. Sistem yang Dibangun

Sistem dibangun menggunakan bahasa pemrograman python. Alur kerja dari sistem yang telah dibangun dapat dilihat pada gambar 2.



Gambar 2. Ilustrasi flowchart sistem yang dibangun

3.1 Crawling data

Data yang digunakan adalah data twitter. Data tersebut didapatkan dengan cara di crawling menggunakan twitter API. Proses *crawling* data dilakukan selama bulan Maret-April 2019. Proses *crawling* dilakukan dengan cara mengambil tweet yang berisi kata kunci yang diinginkan. Kata kunci berkaitan dengan fokus kerja petahana selama 4 tahun untuk membangun Indonesia. List kata kunci yang digunakan dalam proses *crawling* dapat dilihat pada lampiran 1. Data yang diambil adalah data yang berisi opini terkait dengan kinerja petahana.

3.2 Preprocessing

a. Data Cleaning

Data hasil *crawling* kemudian dibersihkan. Pada proses ini semua *emoticon* diubah menjadi kata-kata yang menggambarkan *emoticon* tersebut. Selain itu, *tweet* yang berisi URL juga diakses satu per satu agar dapat diambil isi *tweet* secara lengkap. Selanjutnya pada tahap ini juga data tersebut di-filter dan yang diambil adalah data membicarakan mengenai hasil kerja petahana.

b. Data Labelling

Dataset tersebut akan diberikan kelas sesuai dengan sentimen dari data tersebut. Tiap kelas direpresentasikan dalam bentuk numerik, yaitu kelas nol untuk sentiment negatif dan kelas satu untuk sentiment positif. Contoh pelabelan data dapat dilihat pada tabel 2.

Tabel 2. Contoh Pelabelan Data

No.	Teks	Kelas
1.	Ini sekedar perbandingan aja ya infrastruktur jalan di Papua eranya Jokowi lebih bagus dari eranya SBY #jokowilagirakyathepi #infrastrukturuntukrakyat #jokowi2periode #01Indonesiamaju	1
2.	Indonesia negara kepulauan, tapi lautnya kotor banget. Heran sama yang nanti milih Jokowi lagi, ckckck mikir donggg #2019gantipresiden	0
3.	Terima kasih pak Jokowi sudah mau memberikan kartu Indonesia pintar untuk kepada kami yang kurang mampu	1

c. *Case Folding*

Pada tahap ini, dilakukan proses *case folding* yang bertujuan untuk mengubah semua huruf menjadi lower case. Contoh *case folding* dapat dilihat pada tabel 3.

Tabel 3. Hasil Case Folding

Input	Ini sekedar perbandingan aja ya infrastruktur jalan di Papua eranya Jokowi lebih bagus dari eranya SBY #jokowilagirakypat #infrastrukturuntukakyat #jokowi2periode #01Indonesiamaju
Output	ini sekedar perbandingan aja ya infrastruktur jalan di papua eranya jokowi lebih bagus dari eranya sby #jokowilagirakypat #infrastrukturuntukakyat #jokowi2periode #01indonesiamaju

d. Tokenisasi

Tokenisasi adalah proses mengubah kalimat menjadi potongan-potongan kata yang berdiri sendiri. Contoh tokenisasi dapat dilihat pada tabel 4.

Tabel 4. Hasil Tokenisasi

Input	Ini sekedar perbandingan aja ya infrastruktur jalan di Papua eranya Jokowi lebih bagus dari eranya SBY #jokowilagirakypat #infrastrukturuntukakyat #jokowi2periode #01Indonesiamaju
Output	ini sekedar perbandingan aja ya infrastruktur jalan di papua eranya jokowi lebih bagus dari eranya sby #jokowilagirakypat #infrastrukturuntukakyat #jokowi2periode #01indonesiamaju

e. *Stopword Removal*

Tahap ini adalah menghilangkan kata-kata tidak memiliki makna penting, kata hubung maupun kata ganti. Contoh: [ini, hasil, kerja, jokowi, untuk, masyarakat, indonesia] menjadi [hasil, kerja, jokowi, untuk, masyarakat, indonesia]. Pada penelitian ini menggunakan 2 jenis *stopword* yaitu, menggunakan *stopword* bahasa Indonesia yang diimpor dari NLTK dan kumpulan *stopword* yang diambil dari kata-kata terbanyak (TF-High) yang muncul pada label positif dan negatif [14]. Contoh *stopword removal* dapat dilihat pada tabel 5.

Tabel 5. Hasil Stopword Removal

Input	Ini sekedar perbandingan aja ya infrastruktur jalan di Papua eranya Jokowi lebih bagus dari eranya SBY #jokowilagirakypat #infrastrukturuntukakyat #jokowi2periode #01Indonesiamaju
Output	sekedar perbandingan infrastruktur jalan papua eranya jokowi

	lebih bagus dari eranya sby #jokowilagirakyathepi #infrastrukturuntukrakyat #jokowi2periode #01indonesiamaju
--	--

f. *Stemming*

Tahap ini merupakan proses pemetaan dan penguraian bentuk dari suatu kata menjadi bentuk kata dasarnya. Contoh *stemming* dapat dilihat pada tabel 6.

Tabel 6. Hasil Stemming

Input	Ini sekedar perbandingan aja ya infrastruktur jalan di Papua eranya Jokowi lebih bagus dari eranya SBY #jokowilagirakyathepi #infrastrukturuntukrakyat #jokowi2periode #01Indonesiamaju
Output	sekedar banding infrastruktur jalan papua era jokowi lebih bagus dari era sby #jokowilagirakyathepi #infrastrukturuntukrakyat #jokowi2periode #01indonesiamaju

g. *Punctuation Removal*

Tahap ini merupakan tahap menghilangkan tanda baca.

Tabel 7. Hasil Punctuation Removal

Input	Ini sekedar perbandingan aja ya infrastruktur jalan di Papua eranya Jokowi lebih bagus dari eranya SBY #jokowilagirakyathepi #infrastrukturuntukrakyat #jokowi2periode #01Indonesiamaju
Output	sekedar banding infrastruktur jalan papua era jokowi lebih bagus dari era sby jokowilagirakyathepi infrastrukturuntukrakyat jokowi2periode 01indonesiamaju

3.3 Data splitting

Pada tahap ini, dataset dibagi menjadi 2 bagian yaitu data *train* dan data *test*. Pembagian dataset menggunakan split *k-fold cross-validation* dengan jumlah *k* yang beragam yaitu 2, 3, 5, 6, 9 dan 10. Hal ini dilakukan untuk mengukur kualitas dari model klasifikasi yang dibangun dengan cara mempartisi data secara acak menjadi *k fold* yang saling bebas, sehingga masing-masing *fold* berisi $1/k$ bagian data.

3.4 Pembobotan Kata

Dalam penelitian ini, pembobotan kata dilakukan dengan 2 cara, yaitu:

a. *Word Count*

Menghitung kemunculan suatu kata dalam dokumen.

b. TF-IDF

Pada proses ini dilakukan untuk menghitung kemunculan kata (TF), menghitung banyaknya *tweet* yang mengandung kata tersebut (DF), menghitung IDF dan TF-IDF. Hasil pembobotan dapat dilihat pada tabel 8.

Tabel 8. Hasil Pembobotan TF-IDF

Term	tf		df	$\frac{n}{df}$	idf	tf*idf	
	t1	t2				t1	t2
sekedar	1	0	1	2	0.3	0.3	0
banding	1	0	1	2	0.3	0.3	0
infrastruktur	1	0	1	2	0.3	0.3	0
jalan	1	0	1	2	0.3	0.3	0
papua	1	0	1	2	0.3	0.3	0
era	1	0	1	2	0.3	0.3	0
jokowi	1	1	2	1	0	0	0
lebih	1	0	1	2	0.3	0.3	0
bagus	1	0	1	2	0.3	0.3	0
dari	1	0	1	2	0.3	0.3	0
eranya	1	0	1	2	0.3	0.3	0
sby	1	0	1	2	0.3	0.3	0
jokowilagirakypathepi	1	0	1	2	0.3	0.3	0
infrastrukturuntukrakyat	1	0	1	2	0.3	0.3	0
jokowi2periode	1	0	1	2	0.3	0.3	0
01indonesiamaju	1	0	1	2	0.3	0.3	0
terima	0	1	1	2	0.3	0	0.3
kasih	0	1	1	2	0.3	0	0.3
pak	0	1	1	2	0.3	0	0.3
sudah	0	1	1	2	0.3	0	0.3
mau	0	1	1	2	0.3	0	0.3
memberikan	0	1	1	2	0.3	0	0.3
kartu	0	1	1	2	0.3	0	0.3
indonesia	0	1	1	2	0.3	0	0.3
pintar	0	1	1	2	0.3	0	0.3
untuk	0	1	1	2	0.3	0	0.3
kepada	0	1	1	2	0.3	0	0.3
kami	0	1	1	2	0.3	0	0.3
yang	0	1	1	2	0.3	0	0.3
kurang	0	1	1	2	0.3	0	0.3
mampu	0	1	1	2	0.3	0	0.3

3.5 Klasifikasi SVM

Data yang telah dilakukan *preprocessing* dan pembobotan kata kemudian dilakukan *training* menggunakan metode SVM. Proses training dilakukan dengan beberapa skenario yang diinginkan. Proses ini dilakukan dengan cara mencari *hyperplane* yang optimum untuk memisahkan data yang ada.

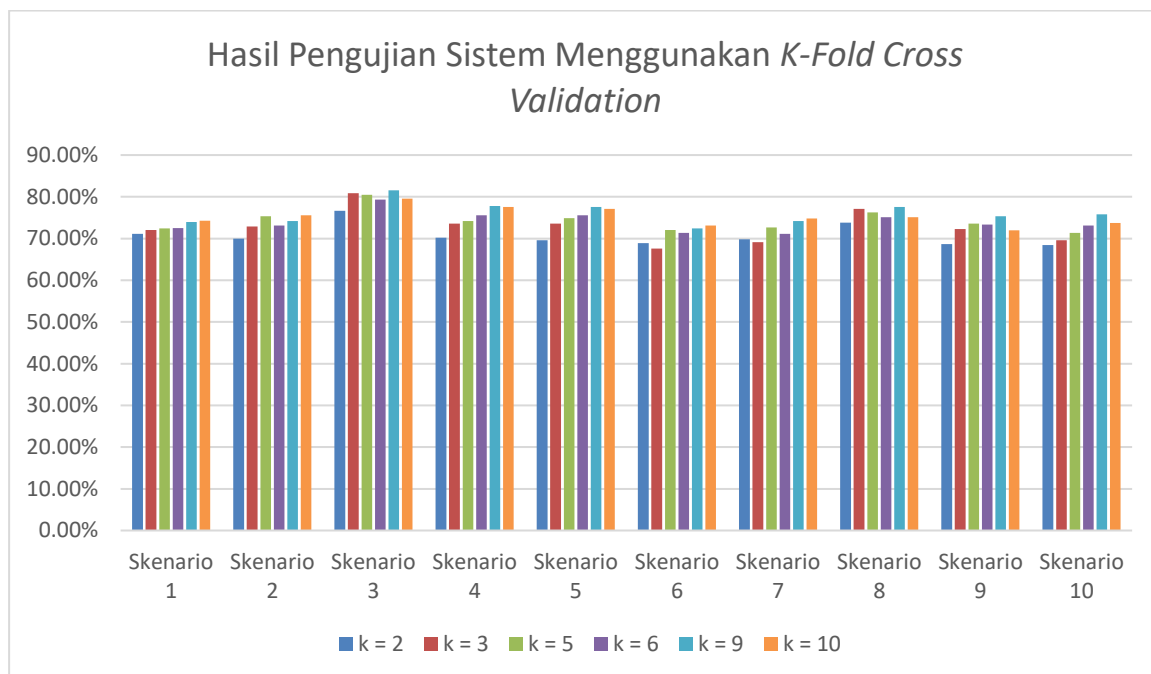
4. Evaluasi

Dalam penelitian ini dilakukan beberapa kombinasi skenario pengujian untuk mendapatkan model yang memiliki performa paling baik. Terdapat 10 kombinasi skenario pengujian dalam penelitian ini, antara lain:

1. TF-IDF + *Stopword* NLTK + Stemming
2. TF-IDF + *Stopword* NLTK + *Stopword* TF-High + Stemming
3. TF-IDF + Stemming
4. TF-IDF + *Stopword* NLTK
5. TF-IDF + *Stopword* NLTK + *Stopword* TF-High
6. *Word Count* + *Stopword* NLTK + Stemming
7. *Word Count* + *Stopword* NLTK + *Stopword* TF-High + Stemming
8. *Word Count* + Stemming
9. *Word Count* + *Stopword* NLTK
10. *Word Count* + *Stopword* NLTK + *Stopword* TF-High

4.1 Hasil Pengujian

Pengujian terhadap 10 kombinasi skenario dilakukan menggunakan sistem yang telah dibangun. Maka didapatkan hasil akurasi dengan menggunakan *K-Fold Cross Validation* dengan $k=2, 3, 5, 6, 9$ dan 10 . Hasil pengujian sistem untuk setiap skenario dengan masing-masing k dapat dilihat pada gambar 3.



Gambar 3. Hasil Pengujian Sistem

4.2 Analisis Hasil Pengujian

Berdasarkan sistem yang dibuat, dapat disimpulkan bahwa performansi terbaik didapat dari skenario 3 dan skenario 8. Kedua skenario tersebut memiliki jumlah fitur yang sama yaitu 1768 fitur dan juga tidak menggunakan metode *stopword removal* pada proses *pre-processing*. Skenario 3 dan skenario 8 sama-sama tidak menggunakan *stopword removal* pada saat *pre-processing*. Sehingga berdasarkan penelitian pada proses *preprocessing* dapat disimpulkan bahwa metode *stopword removal* dapat menurunkan akurasi sistem.

Berdasarkan hasil penelitian yang dilakukan untuk pembagian dataset menjadi data *train* dan data *test* menggunakan *K-Fold Cross Validation* didapati bahwa hasil akurasi yang terbaik adalah dengan menggunakan jumlah $k=9/10$. Hal ini disebabkan karena akurasi yang dihasilkan dari nilai k tersebut memiliki nilai bias dan variasi yang rendah.

Berdasarkan hasil klasifikasi pada saat pengujian, terdapat beberapa kesalahan klasifikasi pada data *test*. Pada skenario 3, dari 90 data *test* yang digunakan untuk pengujian didapati bahwa 13 data *test* terjadi *misclassified*. Aktualnya 5 dari 13 data *test* tersebut memiliki label negatif, namun prediksi label pada sistem adalah positif. Sedangkan 8 data *test* yang seharusnya diklasifikasikan positif, namun label yang dihasilkan dari prediksi sistem adalah negatif. Hal yang sama juga terjadi pada skenario 8, terdapat 18 data *test* terjadi *misclassified*. Kesalahan klasifikasi tersebut disebabkan karena tidak adanya penanganan terhadap kata-kata negasi. Penangan terhadap kata

negasi sangat penting agar perbedaan antara kalimat yang memiliki label positif dan negatif semakin nyata, sehingga tidak terjadi lagi kesalahan dalam proses klasifikasi.

5. Kesimpulan

Kesimpulan yang didapatkan dari hasil penelitian ini adalah

1. Cara mengklasifikasikan data tweet pada kasus ini adalah dengan melakukan proses *crawling data*, *pre-processing* (data *cleaning*, data *labelling*, *case folding*, tokenisasi, *stopword removal*, *stemming* dan *punctuation removal*), pembobotan kata dengan menggunakan TF-IDF atau *word count* dan proses klasifikasi dengan menggunakan metode SVM.
2. Berdasarkan hasil pengujian didapatkan bahwa skenario 3 dan skenario 8 memiliki performansi sistem yang baik untuk mengklasifikasikan *tweet* dibandingkan dengan skenario yang lain. Skenario 3 dengan kombinasi TF-IDF dan *stemming* memiliki akurasi 81,58% sedangkan skenario 8 dengan kombinasi *word count* dan *stemming* memiliki akurasi 77,56%. Namun, nilai akurasi yang tinggi tidak menjamin sistem yang dibangun adalah yang paling baik. Didapati bahwa dalam proses klasifikasi masih terdapat beberapa kesalahan klasifikasi yang disebabkan oleh data bernegasi, sehingga harapan berikutnya sistem dapat dikembangkan dengan penanganan data bernegasi.

Daftar Pustaka

- [1] P. R. Widyastuti, "Pilpres 2019," 19 April 2019. [Online]. Available: <https://perbedaan.budisma.net/perbedaan-webpage-dan-website.html>.
- [2] A. T. Haryanto, "130 Juta Orang Indonesia Tercatat Aktif di Medsos," [Online]. Available: <https://inet.detik.com/cyberlife/d-3912429/130-juta-orang-indonesia-tercatat-aktif-di-medsos>. [Accessed 13 Oktober 2018].
- [3] V. A. Kharde and S. S. Sonawane, "Sentiment Analysis of Twitter Data: A Survey of Techniques," *International Journal of Computer Application*, pp. 5-15, 2016.
- [4] I. Taufik and S. Pamungkas, "Analisis Sentimen Terhadap Tokoh Publik Menggunakan Algoritma Support Vector Machine (SVM)," *LOG!K@*, pp. 69-79, 2018.
- [5] N. Saputra, T. B. Adji and A. E. Permasari, "ANALISIS SENTIMEN DATA PRESIDEN JOKOWI DENGAN PREPROCESSING NORMALISASI DAN STEMMING MENGGUNAKAN METODE NAIVE BAYES DAN SVM," *Dinamika Informatika*, vol. 5, November 2015.
- [6] B. Liu, *Sentiment Analysis and Opinion Mining*, Morgan & Claypool Publishers, 2012, p. 2012.
- [7] R. Y. Yanis and A. Iriani, "Analisis Sentimen terhadap Pelayanan BPJS Kesehatan pada Guru-Guru SMK Eklesia Dan Bina Insani Jailolo," *JUTEI*, vol. 2, pp. 113-122, 2008.
- [8] F. Meisya, "Perancangan Sistem Temu Balik Informasi dengan Metode Pembobotan Kombinasi TF-IDF untuk Pencarian Dokumen Berbahasa Indonesia," in *Universitas Tanjung Puras*, Semarang, 2013.
- [9] B. Santosa, *Data Mining Teknik Pemanfaatan Data untuk Keperluan Bisnis*, Yogyakarta: Graha Ilmu, 2007.
- [10] Suyanto, *Data Mining Untuk Klasifikasi Dan Klusterisasi Data*, Bandung: Informatika Bandung, 2017.
- [11] Suyanto, *Machine Learning Tingkat Dasar dan Lanjut*, Bandung: Informatika Bandung, 2018.
- [12] G. A. Buntoro, "Analisis Sentimen Calon Gubernur DKI Jakarta 2017 Di Twitter," *Integer Journal*, vol. 2, pp. 32-41, 2017.
- [13] N. W. Sumartini, "NAÏVE BAYES CLASSIFIER DAN SUPPORT VECTOR MACHINES UNTUK SENTIMENT ANALYSIS," in *Seminar Nasional Sistem Informasi Indonesia*, 2013.
- [14] H. Saif, M. Fernandez, Y. He and H. Alani, "On stopwords, filtering and data sparsity for sentiment analysis of Twitter," in *Ninth International Conference on Language Resources and Evaluation*, 2014.

