ANALISIS SENTIMEN TERHADAP KANG DEDI MULYADI PADA MEDIA SOSIAL X MENGGUNAKAN ALGORITMA NAÏVE BAYES DAN SUPPORT VECTOR MACHINE (SVM)

Anugrah Alfiansyah Husain, Dolly Indra, Sitti Rahmah Jabir
Sistem Informasi, Universitas Muslim Indonesia
Jalan Urip Sumoharjo Km. 5 Kota Makassar, Indonesia
13120220025@student.umi.ac.id

ABSTRAK

Platform X menjadi ruang utama bagi masyarakat dalam mengekspresikan opini terhadap tokoh publik, termasuk Kang Dedi Mulyadi. Penelitian ini bertujuan menganalisis sentimen publik sekaligus menjawab keterbatasan studi sebelumnya yang umumnya berfokus pada isu politik nasional atau hanya menggunakan satu algoritma klasifikasi, sehingga perbandingan kinerja model pada konteks figur politik daerah masih terbatas. Data penelitian terdiri dari 7.190 tweet yang dikumpulkan melalui *crawling* selama Februari–September 2025. Data diproses melalui tahapan *pre-processing*, kemudian dilabeli otomatis menggunakan *InSet Lexicon* ke dalam sentimen positif dan negatif. Ekstraksi fitur dilakukan dengan TF-IDF menggunakan konfigurasi *n-gram* (1,2), dengan pembagian data 80:20 untuk pelatihan dan pengujian. Hasil pengujian menunjukkan bahwa SVM Linear memberikan performa terbaik dengan akurasi 0,8982, presisi 0,8991, recall 0,8970, dan F1-score 0,8977. Sementara itu, Naïve Bayes menghasilkan akurasi 0,8287, presisi 0,8297, recall 0,8268, dan F1-score 0,8276. Temuan ini menunjukkan bahwa SVM lebih efektif dibandingkan Naïve Bayes dalam mengklasifikasikan sentimen publik pada data media sosial X, serta memberikan kontribusi empiris pada kajian analisis sentimen tokoh politik berbasis media sosial.

Kata kunci : *Analisis Sentimen, Naïve Bayes, Support Vector Machine, Media Sosial X,*

1. PENDAHULUAN

Kemajuan teknologi informasi membawa pergeseran fundamental pada cara khalayak berinteraksi dan mengekspresikan pendapat mereka di ruang publik. Media sosial X (yang dahulu dikenal dengan nama Twitter) menjadi salah satu platform utama bagi publik untuk mengungkapkan pendapat, kritik, maupun dukungan terhadap tokoh publik, termasuk politisi [1]. Media sosial X menyediakan arus opini masyarakat yang diperbarui secara waktu nyata, sehingga data yang diperoleh bersifat aktual dan terus berkembang. Sebagai platform mikroblog, X banyak dimanfaatkan pengguna karena kemudahan akses dan penggunaannya [2]. Salah satu tokoh yang aktif diperbincangkan di media sosial adalah Kang Dedi Mulyadi. Ia menjabat sebagai Gubernur Jawa Barat periode 2025–2030 dan sebelumnya memiliki pengalaman sebagai anggota DPR serta Bupati Purwakarta selama dua periode [3]. Gaya kepemimpinannya yang populis dan aktif di media sosial membuatnya memiliki banyak pendukung sekaligus menuai kritik, sehingga menarik untuk dikaji melalui analisis sentimen publik [4].

Masifnya arus diskursus terkait Kang Dedi Mulyadi di jagat digital menempatkan analisis sentimen sebagai instrumen krusial guna mengidentifikasi arah pandangan publik secara akurat. Sebagai figur politik yang aktif dalam berbagai kebijakan sosial serta menjadi salah satu aktor kunci dalam dinamika politik regional Jawa Barat, opini masyarakat terhadap Kang Dedi Mulyadi di platform digital berpotensi memengaruhi pembentukan citra publik serta dinamika komunikasi politik di ruang maya. Fluktuasi opini digital ini menjadi sangat penting untuk dipetakan mengingat platform X sering kali menjadi barometer awal untuk mengukur tingkat penerimaan (akseptabilitas) dan sentimen elektoral seorang tokoh di tingkat daerah. Analisis sentimen berfungsi sebagai instrumen untuk mendeteksi muatan subjektif dalam tulisan dan kemudian mengategorikannya berdasarkan sifat kecenderungan opini, baik itu pro (positif) maupun kontra (negatif) [5]. Dalam implementasinya, analisis sentiment memanfaatkan Kemampuan kecerdasan buatan melalui teknik *Natural Language Processing* (NLP) memungkinkan data teks dari media sosial yang cenderung singkat dan menggunakan bahasa sehari-hari dapat diproses serta dikelompokkan secara otomatis tanpa intervensi manual yang besar [6]. Oleh karena itu, penelitian ini penting dilakukan untuk memberikan gambaran empiris mengenai respons publik terhadap Kang Dedi Mulyadi, sekaligus menunjukkan bagaimana media sosial dapat merepresentasikan kecenderungan opini masyarakat terhadap figur politik secara digital.

Pengelompokan opini dalam penelitian ini dijalankan dengan mengandalkan dua algoritma standar *text mining*, yaitu SVM dan Naïve Bayes, untuk menguji akurasi klasifikasi sentimen yang dihasilkan. Efektivitas kedua algoritma tersebut dalam mengorganisir informasi teks telah teruji, sehingga sering menjadi pilihan utama untuk kategorisasi data tekstual dan sering dibandingkan untuk menilai efektivitasnya pada berbagai konteks [5]. Kajian literatur menunjukkan efektivitas Naïve Bayes yang sangat tinggi dalam mendeteksi sentimen pengguna

YouTube, dengan perolehan akurasi mencapai persentase 96%. [7], sedangkan SVM dengan kernel linear menghasilkan *F1-score* sebesar 92,3% pada analisis sentimen ulasan publik [8]. Sejauh ini, literatur yang secara spesifik menguji performa algoritma Naïve Bayes dan SVM dalam memotret respons masyarakat terhadap Kang Dedi Mulyadi pada periode 2025 dengan memanfaatkan sumber data dari platform X masih sangat terbatas atau bahkan belum tersedia. Berdasarkan latar belakang tersebut, Identifikasi orientasi pendapat masyarakat terhadap Kang Dedi Mulyadi di media sosial X dilakukan dalam studi ini, yang dibarengi dengan pengukuran performa klasifikasi antara Naïve Bayes dan SVM menggunakan indikator *F1-score*, *recall*, presisi, dan akurasi.

2. TINJAUAN PUSTAKA

2.1. Penelitian Terkait

Penelitian [9] menganalisis sentimen mahasiswa terhadap penerapan *hybrid* learning menggunakan algoritma Naïve Bayes pada tiga kelas sentimen (positif, negatif, netral). Skema perbandingan 80:20 diimplementasikan pada total 112 data responden untuk memisahkan antara subset *training* dan subset *testing*. Hasil menunjukkan akurasi sebesar 60,87% yang meningkat menjadi 80,65% pasca implementasi metode SMOTE yang ditujukan untuk menyelaraskan distribusi antar kelas yang tidak proporsional.

Sebanyak 3.360 opini terkait layanan MyPertamina dianalisis dalam riset [10] dengan mengadu performa tiga algoritma klasifikasi populer, yaitu SVM, Random Forest, serta Naïve Bayes. Hasil menunjukkan bahwa Random Forest memperoleh performa terbaik dengan akurasi 99,77%, sementara SVM dan Naïve Bayes juga menunjukkan kinerja yang tinggi dengan akurasi masing-masing 99,31% dan 90,24%, sehingga ketiga algoritma dinilai efektif dalam klasifikasi sentimen.

Sementara itu, penelitian [8] menganalisis ulasan publik terkait renovasi Bandara Sultan Hasanuddin menggunakan Naïve Bayes Multinomial, SVM, dan Random Forest dengan fitur TF-IDF serta penerapan SMOTE. Secara empiris, penggunaan kernel linear pada SVM menunjukkan keunggulan yang signifikan, di mana model ini mampu mencatatkan parameter *F1-score* sebesar 92,3%, diikuti Naïve Bayes dengan *F1-score* 83,7%, dan Random Forest dengan *F1-score* 81,9%. Penerapan SMOTE meningkatkan performa Naïve Bayes, namun tidak memberikan pengaruh signifikan terhadap SVM.

2.2. Analisis Sentimen

Analisis sentimen merepresentasikan sebuah metodologi dalam ranah komputasi yang berfungsi untuk mengidentifikasi sekaligus menguraikan data teks tidak terstruktur guna memperoleh informasi yang terkandung dalam opini atau perilaku pengguna. Melalui teknik ini, respons atau persepsi seseorang terhadap suatu produk, layanan, atau isu tertentu dapat diklasifikasikan secara otomatis berdasarkan kecenderungan sentimennya [11].

2.3. InSet Lexicon

InSet Lexicon merupakan sebuah sumber daya leksikal yang memuat daftar istilah dengan atribusi bobot nilai yang setiap entrinya memiliki label orientasi positif ataupun negatif. Dalam proses pelabelan, setiap kata di dalam teks diperiksa sesuai dengan daftar di kamus untuk mengetahui arah sentimennya. Jika total polaritas suatu kalimat lebih dari 0, kalimat itu dianggap positif; jika kurang dari 0, dianggap negatif; sementara jika nilainya 0, kalimat tersebut diberi label netral [12].

2.4. TF-IDF

Term Frequency–Inverse Document Frequency (TF-IDF) mengonversi data tekstual ke dalam representasi vektor numerik dengan mengkalkulasi signifikansi setiap kata terhadap dokumen maupun keseluruhan korpus [13]. TF merepresentasikan kuantitas kemunculan sebuah leksikal di dalam suatu unit dokumen yang sedang dianalisis secara spesifik. IDF berfungsi mengurangi dominasi kata yang memiliki frekuensi kemunculan tinggi pada banyak dokumen. Kata yang muncul pada sebagian besar dokumen akan memiliki nilai IDF yang rendah [14]. Kombinasi keduanya menghasilkan nilai yang mencerminkan relevansi suatu kata terhadap dokumen dalam keseluruhan korpus [13]. Adapun rumus TF-IDF ditunjukkan pada persamaan (1):

$$W_{i,j} = tf_{i,j} \log\left(\frac{N+1}{df_i + 1}\right) + 1 \quad (1)$$

Keterangan:

- $W_{i,j}$: Nilai bobot TF-IDF untuk term ke-i pada dokumen ke-j.
- $tf_{i,j}$: Frekuensi kemunculan term ke-i di dalam dokumen ke-j.
- df_i : Banyaknya dokumen dalam korpus yang memuat term ke-i.
- N : Total keseluruhan dokumen yang terdapat dalam korpus data.

2.5. Naïve Bayes

Mengadopsi prinsip Teorema Bayes, algoritma Naïve Bayes bekerja melalui pendekatan probabilistik dengan menerapkan asumsi dasar bahwa setiap atribut data tidak memiliki keterikatan satu sama lain. Metode ini memanfaatkan perhitungan peluang untuk memprediksi kemungkinan suatu data termasuk ke dalam kelas tertentu berdasarkan pola dari data sebelumnya [15]. Berikut adalah rumus Naïve Bayes ditunjukkan pada persamaan (2):

$$P(H|X) = \frac{P(X|H)\,P(H)}{P(X)} \quad (2)$$

Keterangan:

- X : Data atau sampel yang belum diketahui kelasnya.
- H : Representasi dari asumsi awal yang memprediksi bahwa observasi X merupakan bagian dari kategori atau label spesifik.

P(H | X) : Besaran peluang untuk hipotesis H setelah yang dihitung berdasarkan keberadaan data atau informasi X.
P(H) : Besaran nilai probabilitas apriori bagi asumsi H yang ditentukan tanpa melibatkan pengaruh dari observasi data apa pun.
P(X | H) : Besaran nilai probabilitas kondisional bagi observasi X dengan asumsi bahwa hipotesis H telah terbukti valid.
P(X) : Probabilitas kumulatif dari keberadaan fitur X yang dihitung secara menyeluruh pada seluruh ruang sampel penelitian.

2.6. *Support Vector Machine* (SVM)

Support Vector Machine (SVM) merupakan algoritma klasifikasi berbasis *supervised learning* yang menggunakan data berlabel dalam proses pelatihannya. Metode ini bekerja dengan menentukan *hyperplane* terbaik yang memisahkan data menjadi dua kategori yang berbeda dengan jarak maksimum, yang menghasilkan batas klasifikasi yang lebih tepat dan konsisten [16]. Rumus menentukan hyperplane pada SVM [17] dapat dilihat pada persamaan (3):

$$w.x + b = 0 \rightarrow w_0 + w_1 x_1 + w_2 x_2 \ldots + w_n x_n \quad (3)$$

Keterangan:
W : {w1 , w2 ,…, wn} → bobot
n : Jumlah atribut
b : Skala yang digunakan sebagai bias
x : nilai atribut

2.7. Evaluasi model

Evaluasi model dilakukan guna menakar kualitas klasifikasi dengan menerapka berbagai metrik performa seperti akurasi, presisi, *recall*, dan *F1-score*. Sebagai instrumen pendukung, *confusion matrix* dipergunakan untuk memetakan korelasi antara label hasil prediksi sistem dengan label yang sebenarnya (*ground truth*) pada dataset evaluasi [18]. Metrik akurasi memberikan gambaran mengenai rasio ketepatan prediksi total, sedangkan presisi menitikberatkan pada akurasi hasil dalam cakupan kelas spesifik. Melengkapi evaluasi tersebut, *recall* menilai sejauh mana kemampuan algoritma dalam meminimalisir data yang luput dari pendeteksian [19]. Kinerja model dapat dievaluasi menggunakan *F1-score*, yaitu ukuran yang mengintegrasikan precision dan recall dalam satu nilai [20]. Nilai akurasi dihitung menggunakan Persamaan (4).

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (4)$$

Nilai presisi dihitung menggunakan Persamaan (5).

$$Precision = \frac{TP}{TP+FP} \quad (5)$$

Nilai *recall* dihitung menggunakan Persamaan (6).

$$Recall = \frac{TP}{TP+FN} \quad (6)$$

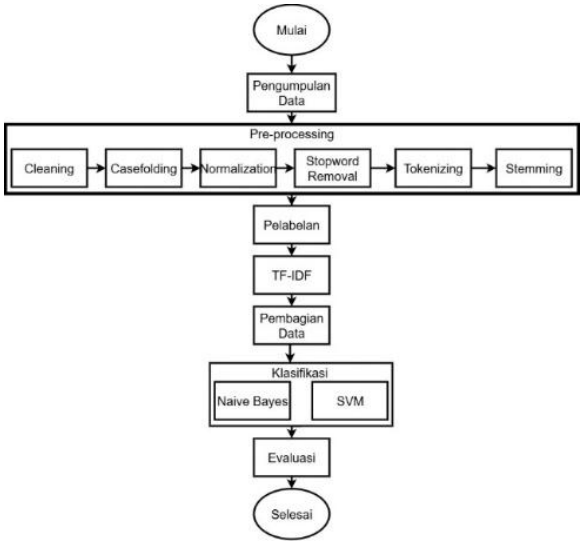
Nilai *F1-score* dihitung menggunakan Persamaan (7).

$$F1 - score = 2\,\frac{Precision \times Recall}{Precision + Recall} \quad (7)$$

Keterangan:
TP : Data positif yang terklasifikasi dengan tepat.
TN : Data negatif yang berhasil diidentifikasi secara akurat.
FP : Kondisi ketika model salah menetapkan label positif.
FN : Situasi di mana data aktual positif justru diprediksi negatif.

3. METODE PENELITIAN

Desain penelitian yang menggambarkan alur tahapan penelitian ini ditunjukkan pada Gambar 1.



Gambar 1. Desain penelitian

3.1. Pengumpulan Data

Data dikumpulkan melalui proses *crawling* menggunakan Tweet Harvest yang dijalankan pada Google Colab dengan kata kunci “Dedi Mulyadi” dan filter bahasa Indonesia (*lang:id*) pada periode Februari–September 2025. Dari proses tersebut diperoleh sebanyak 7.190 postingan sebagai *dataset* penelitian.

3.2. *Pre-processing*

Kesiapan dataset untuk tahap klasifikasi diatur melalui fase *pre-processing*, di mana data mentah dikurasi dan diformat sedemikian rupa agar hasil prediksi yang dihasilkan lebih akurat. Proses ini meliputi *cleaning* untuk menghapus hashtag, username, URL, angka, dan tanda baca yang tidak relevan; *case folding* untuk menjamin konsistensi data dengan mentransformasikan seluruh karakter alfabet ke dalam format huruf kecil; *normalization* untuk mengonversi kata tidak baku atau slang ke bentuk baku menggunakan kamus slang; *stopword removal* dengan memanfaatkan pustaka NLTK untuk menjalankan prosedur eliminasi kata-kata fungsional yang tidak memiliki kontribusi semantik signifikan; *tokenizing* untuk memecah teks menjadi token kata; serta Implementasi *stemming* melalui pustaka Sastrawi diintegrasikan ke dalam sistem untuk menyeragamkan variasi kata turunan menjadi bentuk leksikal yang lebih sederhana.

3.3. Pelabelan

Proses pelabelan pada penelitian ini dilakukan menggunakan InSet Lexicon, dengan membandingkan setiap kata hasil *stemming* terhadap kamus sentimen untuk menentukan label positif, negatif, atau netral secara otomatis.

3.4. TF-IDF

Pembobotan TF-IDF digunakan untuk mengubah teks yang telah melalui proses *pre-processing* menjadi representasi vektor numerik supaya dapat digunakan sebagai input dalam algoritma klasifikasi.

3.5. Pembagian Data

Dataset yang telah terbobot kemudian dipisahkan secara sistematis menggunakan rasio 80% dan 20%. Komposisi ini menempatkan mayoritas data sebagai basis pelatihan model, sementara bagian minoritas berfungsi sebagai instrumen pengujian efikasi klasifikasi.

3.6. Klasifikasi

Dua arsitektur algoritma, yaitu SVM dan Multinomial Naïve Bayes, dikerahkan untuk melakukan klasifikasi. Karakteristik Multinomial Naïve Bayes yang bekerja melalui perhitungan distribusi peluang kelas menjadikannya instrumen yang ideal untuk mengolah data teks yang telah ditransformasikan ke dalam bobot TF-IDF. Sementara itu, SVM dengan kernel linear digunakan karena efektif dalam menangani fitur berdimensi tinggi serta mampu menentukan *hyperplane* optimal untuk memisahkan kelas sentimen. Kedua model diuji pada skenario dua kelas (positif dan negatif) dan dievaluasi menggunakan data pengujian untuk mengukur kinerja klasifikasi.

3.7. Evaluasi

Penilaian terhadap efikasi model klasifikasi dijalankan dengan mengadopsi *confusion matrix* sebagai basis perhitungan parameter statistik, yang meliputi nilai akurasi, presisi, *recall*, serta *F1-score*, guna mengukur ketajaman prediksi pada kategori positif maupun negatif. Nilai-nilai tersebut digunakan sebagai dasar perbandingan performa antara algoritma Naïve Bayes dan SVM guna menentukan metode yang memberikan hasil paling optimal.

4. HASIL DAN PEMBAHASAN

4.1. Pengumpulan Data

Pengumpulan data dilakukan melalui proses *crawling* menggunakan *tools* tweet-harvest berbasis Node.js untuk mengambil tweet dari media sosial X dengan kata kunci “dedi mulyadi” pada periode Februari–September 2025. Karena keterbatasan jumlah data dalam satu kali pengambilan, proses *crawling* dilakukan secara bertahap agar mencakup seluruh periode penelitian. Hasil *crawling* menghasilkan 12 file CSV yang kemudian digabungkan menjadi satu *dataset* utama dengan total 7.190 tweet untuk tahap preprocessing dan analisis sentimen.

4.2. *Pre-processing*

a. *Cleaning*

Pada tahap *cleaning*, hanya kolom *full_text* yang digunakan sebagai sumber data karena memuat isi opini pengguna. Proses ini dilakukan dengan menghapus elemen yang tidak relevan seperti karakter HTML, URL, mention (@user), hashtag, emoji, angka, serta simbol atau karakter khusus. Selain itu, spasi berlebih juga dihilangkan agar teks lebih rapi dan siap diproses pada tahap selanjutnya. Hasil proses ini dapat dilihat pada tabel 1.

Tabel 1. Hasil proses *cleaning*

Sebelum *Cleaning*	**Sesudah *Cleaning***
@JanissaryD_Last Mau ngadain acara sebesar apapun gak akan ada hebat² ny Selama angka pengangguran di Jawa barat masih tinggi & kesejahteraan masyarakat nya belum terwujud Emang perut masyarakat bs kenyang Emang angka pengangguran bs turun Hanya dgn melihat video ocehan Dedi Mulyadi di sosmed? https://t.co/9pl3jMXGlD	Mau ngadain acara sebesar apapun gak akan ada hebat ny Selama angka pengangguran di Jawa barat masih tinggi kesejahteraan masyarakat nya belum terwujud Emang perut masyarakat bs kenyang Emang angka pengangguran bs turun Hanya dgn melihat video ocehan Dedi Mulyadi di sosmed

b. *Case Folding*

Pasca prosedur pembersihan, diterapkan standardisasi karakter melalui mekanisme *case folding*. Tujuan utama dari prosedur ini adalah menyeragamkan penulisan karakter agar perbedaan huruf kapital tidak dianggap sebagai entitas yang berbeda, yang pada akhirnya meningkatkan koherensi data sebelum memasuki algoritma klasifikasi. Hasil proses *case folding* dapat dilihat pada tabel 2.

Tabel 2. Hasil proses *case folding*

Sebelum *Case Folding*	**Sesudah *Case Folding***
Mau ngadain acara sebesar apapun gak akan ada hebat ny Selama angka pengangguran di Jawa barat masih tinggi kesejahteraan masyarakat nya belum terwujud Emang perut masyarakat bs kenyang Emang angka pengangguran bs turun Hanya dgn melihat video ocehan Dedi Mulyadi di sosmed	mau ngadain acara sebesar apapun gak akan ada hebat ny selama angka pengangguran di jawa barat masih tinggi kesejahteraan masyarakat nya belum terwujud emang perut masyarakat bs kenyang emang angka pengangguran bs turun hanya dgn melihat video ocehan dedi mulyadi di sosmed

c. *Normalization*

Tahap *normalization* dilakukan untuk mengubah kata tidak baku atau slang menjadi bentuk baku agar teks lebih konsisten dan mudah diproses. Proses ini

memanfaatkan kamus slang dari *dataset kamus-slang* (Kaggle) untuk mengganti kata informal secara otomatis dengan padanan yang sesuai dalam Bahasa Indonesia. Hasil proses *normalization* dapat dilihat pada tabel 3.

Tabel 3. Hasil proses *normalization*

Sebelum *normalization*	**Sesudah *normalization***
mau ngadain acara sebesar apapun gak akan ada hebat ny selama angka pengangguran di jawa barat masih tinggi kesejahteraan masyarakat nya belum terwujud emang perut masyarakat bs kenyang emang angka pengangguran bs turun hanya dgn melihat video ocehan dedi mulyadi di sosmed	mau mengadakan acara sebesar apapun tidak akan ada hebat nya selama angka pengangguran di jawa barat masih tinggi kesejahteraan masyarakat ya belum terwujud memang perut masyarakat bisa kenyang memang angka pengangguran bisa turun hanya dengan melihat video ocehan dedi mulyadi di sosmed

d. *Stopword Removal*

Tahap *stopword removal* dilakukan untuk menghapus kata-kata umum yang tidak berkontribusi terhadap penentuan sentimen guna mengurangi noise pada data. Proses ini menggunakan daftar stopword Bahasa Indonesia dari library NLTK yang kemudian ditambahkan dengan kata-kata umum di media sosial seperti "yah", "dong", "nih", serta ekspresi tawa. Selain itu, pola kata tawa berulang juga dihapus menggunakan *regular expression*. Hasil proses *stopword removal* dapat dilihat pada tabel 4.

Tabel 4. Hasil proses *stopword removal*

Sebelum *Stopword Removal*	**Sesudah *Stopword Removal***
mau mengadakan acara sebesar apapun tidak akan ada hebat nya selama angka pengangguran di jawa barat masih tinggi kesejahteraan masyarakat ya belum terwujud memang perut masyarakat bisa kenyang memang angka pengangguran bisa turun hanya dengan melihat video ocehan dedi mulyadi di sosmed	mengadakan acara apapun hebat angka pengangguran jawa barat kesejahteraan masyarakat terwujud perut masyarakat kenyang angka pengangguran turun video ocehan dedi mulyadi sosmed

e. *Tokenizing*

Tahap *tokenizing* dilakukan dengan memecah teks yang telah dibersihkan menjadi unit kata berdasarkan pemisahan spasi, sehingga setiap kalimat diubah menjadi daftar token sebagai dasar proses selanjutnya. Hasil proses *tokenizing* dapat dilihat pada tabel 5.

Tabel 5. Hasil proses *tokenizing*

Sebelum *Tokenizing*	**Sesudah *Tokenizing***
mengadakan acara apapun hebat angka pengangguran jawa barat kesejahteraan masyarakat terwujud perut masyarakat kenyang angka pengangguran turun video ocehan dedi mulyadi sosmed	mengadakan, acara, apapun, hebat, angka, pengangguran, jawa, barat, kesejahteraan, masyarakat, terwujud, perut, masyarakat, kenyang, angka, pengangguran, turun, video, ocehan, dedi, mulyadi, sosmed

f. ***Stemming***

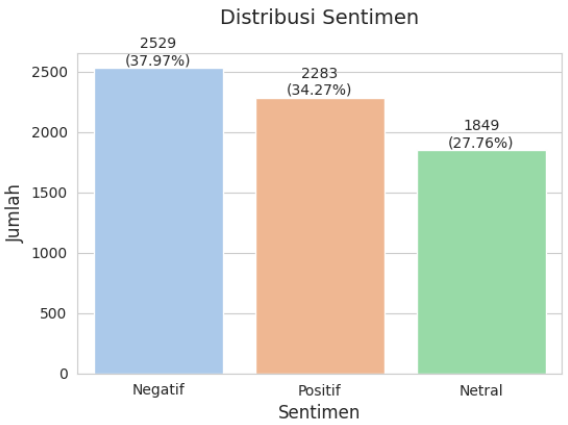
Tahap stemming dilakukan untuk mengubah kata berimbuhan menjadi bentuk dasar guna menyeragamkan representasi makna. Proses ini menggunakan *library* Sastrawi sebagai *stemmer* Bahasa Indonesia, sehingga kata seperti "menjalankan" menjadi "jalan" dan "perubahan" menjadi "ubah". Hasil proses *stemming* dapat dilihat pada tabel 6.

Tabel 6. Hasil proses *stemming*

Sebelum *Stemming*	**Sesudah *Stemming***
mengadakan, acara, apapun, hebat, angka, pengangguran, jawa, barat, kesejahteraan, masyarakat, terwujud, perut, masyarakat, kenyang, angka, pengangguran, turun, video, ocehan, dedi, mulyadi, sosmed	ada acara apa hebat angka anggur jawa barat sejahtera masyarakat wujud perut masyarakat kenyang angka anggur turun video oceh dedi mulyadi sosmed

4.3. Pelabelan

Dari 7.190 data awal hasil crawling, setelah melalui tahap *pre-processing* jumlah data berkurang menjadi 6.661 tweet akibat penghapusan data duplikat. Proses pelabelan sentimen dilakukan menggunakan InSet Lexicon dengan membandingkan setiap kata hasil *stemming* terhadap kamus sentimen yang memuat daftar kata bernilai positif dan negatif. Setiap kemunculan kata akan diberi skor sesuai polaritasnya, kemudian seluruh skor dalam satu *tweet* dijumlahkan untuk menentukan label akhir. *Tweet* diberi label positif apabila total skor lebih dari 0, negatif apabila kurang dari 0, dan netral apabila bernilai 0. Proses ini dilakukan secara otomatis untuk menghasilkan label sentimen awal pada dataset. Gambar 2 menunjukkan distribusi sentimen dari pelabelan InSet Lexicon.



Gambar 2. Distribusi sentimen

Berdasarkan Gambar 2, hasil pelabelan awal menggunakan InSet Lexicon menunjukkan bahwa kelas negatif memiliki jumlah tertinggi sebanyak 2.529 data (37,97%), diikuti oleh kelas positif sebanyak 2.283 data (34,27%), serta kelas netral sebanyak 1.849 data (27,76%). Namun, pada tahap selanjutnya penelitian ini difokuskan pada skenario dua kelas, sehingga data berlabel netral dihapus. Setelah penghapusan kelas netral, dataset yang digunakan dalam proses klasifikasi terdiri dari sentimen negatif dan positif dengan distribusi yang relatif seimbang.

4.4. TF-IDF

Setelah tahap pelabelan dan penghapusan kelas netral, data teks dikonversi ke dalam bentuk numerik menggunakan metode TF-IDF dengan konfigurasi *n-gram* (1,2), *min_df* = 2, dan normalisasi L2. Proses ini menghasilkan matriks fitur yang merepresentasikan bobot setiap kata terhadap dokumen. Gambar 3 menampilkan contoh sebagian matriks TF-IDF pada lima dokumen pertama. Nilai pada setiap sel menunjukkan bobot kepentingan suatu term dalam dokumen tertentu, di mana nilai 0 menunjukkan bahwa term tersebut tidak muncul pada dokumen tersebut. Representasi ini selanjutnya digunakan sebagai input pada tahap klasifikasi menggunakan algoritma Naïve Bayes dan SVM.

Contoh matriks TF-IDF (5 dokumen pertama):

	aa	abah	abai	abai risiko	abang	abang anak	abar	abdi	abdi negri	\
0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
1	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
2	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
3	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	
4	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0	

	abdul
0	0.0
1	0.0
2	0.0
3	0.0
4	0.0

Gambar 3. Contoh matriks TF-IDF

4.5. Pembagian Data

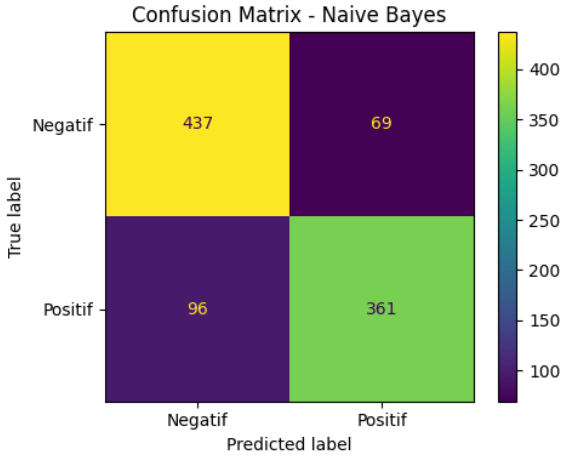
```
Split 80:20
Train: 3849 | Test: 963
Distribusi TRAIN:
Sentiment
Negatif    2023
Positif    1826

Distribusi TEST:
Sentiment
Negatif    506
Positif    457
```

Gambar 4. Hasil pembagian data dengan rasio 80:20

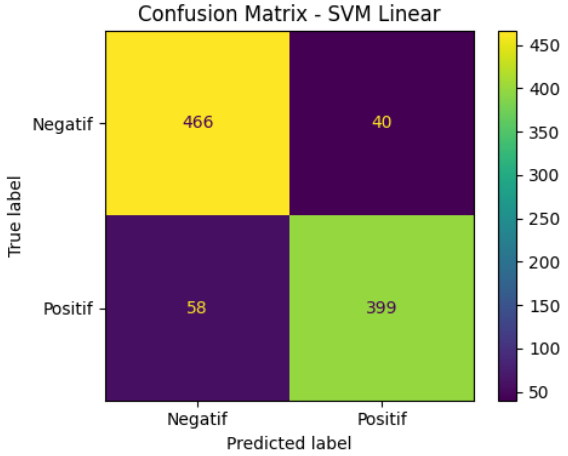
Data kemudian dibagi menjadi data pelatihan dan data pengujian dengan rasio 80:20 menggunakan metode *stratified split*. Dari total 4.812 data, sebanyak 3.849 data digunakan sebagai data pelatihan dan 963 data sebagai data pengujian. Distribusi kelas pada data pelatihan terdiri dari 2.023 sentimen negatif dan 1.826 sentimen positif, sedangkan pada data pengujian terdapat 506 sentimen negatif dan 457 sentimen positif. Penggunaan *stratified split* bertujuan untuk menjaga proporsi distribusi kelas tetap seimbang pada kedua subset data. Gambar 4 memperlihatkan hasil proses pembagian data pelatihan dan data pengujian dengan rasio 80:20.

4.6. Klasifikasi dan Evaluasi



Gambar 5. *Confusion matrix* Naïve Bayes

Berdasarkan Gambar 5, model berhasil mengklasifikasikan 437 data sentimen negatif dan 361 data sentimen positif dengan benar. Namun, Kesalahan dalam penentuan label masih ditemukan, dengan distribusi sebesar 69 data negatif yang salah diprediksi dan 96 data positif yang luput dari klasifikasi kelas yang seharusnya. Hasil ini menunjukkan bahwa Naïve Bayes memiliki kemampuan klasifikasi yang cukup baik, meskipun masih terdapat kesalahan prediksi terutama pada kelas positif yang lebih sering tertukar menjadi negatif.



Gambar 6. *Confusion matrix* SVM

Berdasarkan Gambar 6, model berhasil mengklasifikasikan 466 data sentimen negatif dan 399 data sentimen positif dengan benar. Dibandingkan dengan performa Naïve Bayes, model ini memperlihatkan jumlah miskalsifikasi yang lebih sedikit, yakni distribusi *error* sebesar 40 unit pada

label negatif dan 58 unit pada label positif. Hasil ini menunjukkan bahwa SVM memiliki kemampuan pemisahan kelas yang lebih baik dan lebih konsisten dalam mengklasifikasikan sentimen dibandingkan Naïve Bayes.

Tabel 7. Perbandingan kinerja Naïve Bayes dan SVM

Model	Accuracy	Precision	Recall	F1-score
Naïve Bayes	0.8287	0.8297	0.8268	0.8276
SVM Linear	0.8982	0.8991	0.8970	0.8977

Berdasarkan Tabel 7, SVM Linear memperlihatkan keunggulan komprehensif dibandingkan Naïve Bayes pada seluruh metrik evaluasi. SVM memperoleh akurasi sebesar 0,8982, *precision* 0,8991, *recall* 0,8970, dan *F1-score* 0,8977, sedangkan *Naïve Bayes* memperoleh akurasi sebesar 0,8287, *precision* 0,8297, *recall* 0,8268, dan *F1-score* 0,8276. Keunggulan ini dapat dijelaskan secara teknis melalui karakteristik representasi fitur TF-IDF yang menghasilkan data berdimensi tinggi dan bersifat *sparse*, di mana sebagian besar elemen matriks bernilai nol karena tidak semua *term* muncul pada setiap dokumen. Kondisi ini cenderung lebih sesuai untuk SVM, khususnya kernel linear, karena algoritma ini dirancang untuk bekerja optimal pada ruang fitur berdimensi tinggi dengan mencari *hyperplane* pemisah terbaik antar kelas berdasarkan *margin* maksimum. Sebaliknya, Naïve Bayes mengandalkan distribusi probabilitas tiap fitur dengan asumsi independensi antar kata, sehingga performanya dapat menurun ketika menghadapi fitur teks yang jarang muncul atau memiliki korelasi kontekstual antar *term*, terutama pada konfigurasi *n-gram* (1,2) yang meningkatkan jumlah fitur secara signifikan. Hal ini menjelaskan mengapa Naïve Bayes menghasilkan tingkat misklasifikasi yang lebih tinggi, khususnya pada kelas positif, dibandingkan SVM. Dengan demikian, hasil penelitian menunjukkan bahwa SVM lebih *robust* dalam menangani kompleksitas representasi teks berbasis TF-IDF pada analisis sentimen media sosial X.

5. KESIMPULAN DAN SARAN

Hasil pengujian menunjukkan bahwa SVM Linear memberikan performa yang lebih unggul dibandingkan Naïve Bayes, dengan akurasi sebesar 0,8982 berbanding 0,8287. Keunggulan SVM pada seluruh metrik evaluasi menunjukkan bahwa pendekatan *margin-based* lebih efektif dalam menangani data teks berdimensi tinggi dan bersifat *sparse* yang dihasilkan oleh representasi TF-IDF. Sebaliknya, performa Naïve Bayes yang lebih rendah mengindikasikan bahwa asumsi independensi antar fitur belum sepenuhnya mampu merepresentasikan kompleksitas hubungan antar kata pada data media sosial X. Berdasarkan hasil pengujian tersebut, penelitian selanjutnya disarankan untuk melakukan optimasi parameter model, khususnya pada SVM melalui penyesuaian nilai parameter C atau eksplorasi kernel lain, guna memperoleh performa yang lebih optimal. Selain itu, adanya potensi misklasifikasi pada data media sosial yang mengandung bahasa informal, singkatan, sarkasme, dan konteks ambigu menunjukkan perlunya penyempurnaan tahap *pre-processing*, seperti perluasan kamus normalisasi *slang* dan penanganan ekspresi kontekstual. Kualitas pelabelan juga perlu ditingkatkan melalui validasi manual oleh anotator atau pendekatan *semi-supervised*, mengingat pelabelan otomatis berbasis leksikon berpotensi menghasilkan bias klasifikasi.

DAFTAR PUSTAKA

[1] C. Cholifah, H. H. Handayani, and A. R. Juwita, "Analisis Sentimen Twitter Terhadap Uu Omnibus Law Menggunakan Metode Support Vector Machine (Svm) Dan Naïve Bayes Classifier (Nbc)," *JINTEKS (Jurnal Inform. Teknol. dan Sains)*, vol. 4, no. 4, pp. 483 – 488, 2022.

[2] M. F. A. Shidiq and D. Alita, "Analisis Sentimen Masyarakat Terhadap Kasus Judi Online Menggunakan Data Dari Media Sosial X Pendekatan Naive Bayes dan SVM," *J. Sist. Inf. dan Inform.*, vol. 8, no. 1, pp. 24–35, 2025.

[3] N. Abdurrahman, "Profil Gubernur Jabar Dedi Mulyadi, Riwayat Pendidikan dan Karier Politik," TribunJabar. Accessed: May 20, 2025. [Online]. Available: https://jabar.tribunnews.com/2025/02/21/profil-gubernur-jabar-dedi-mulyadi-riwayat-pendidikan-dan-karier-politik

[4] V. Rossa, "Mengenal Gaya Populisme Dedi Mulyadi: Dekat dengan Rakyat Lewat Cara Tak Biasa," Suara.Com. Accessed: Nov. 12, 2025. [Online]. Available: https://www.suara.com/lifestyle/2025/06/11/201206/mengenal-gaya-populisme-dedi-mulyadi-dekat-dengan-rakyat-lewat-cara-tak-biasa?page=1

[5] S. Berliani and S. Lestari, "Analisis Sentimen Masyarakat Terhadap Isu Pecat Sri Mulyani Pada Twitter Menggunakan Metode Naive Bayes Dan Support Vector Machine," *J. Sains dan Teknol.*, vol. 5, no. 3, pp. 951–960, 2024, doi: 10.55338/saintek.v5i3.2746.

[6] M. Fazri and A. Voutama, "ANALISIS SENTIMEN PUBLIK TERHADAP DANANTARA DI MEDIA SOSIAL X MENGGUNAKAN NLP DAN PEMBELAJARAN MESIN," *JOISIE (Journal Inf. Syst. Informatics Eng.*, vol. 9, no. 1, pp. 197–206, 2025.

[7] M. D. Taufiqurahman, S. Anraeni, and H. Darwis, "Analisis Sentimen Komentar Konten Kreator Gaming Menggunakan Metode Naive Bayes dan KNN," vol. 1, no. 4, pp. 317–327, 2024.

[8] M. F. Hermansyah and D. Indra, "Comparison Of Sentiment Classification Models At Sultan Hasanuddin Airport In Makassar," pp. 54–68.

[9] W. S. Dolly Indra, Ramdaniah, "Analysis of hybrid learning sentiment among information systems students using the naïve bayes classifier," vol. 8, no. 2, pp. 91–99, 2024.

[10] A. Amelia, L. Nur, and H. Darwis, "Analisis Sentimen Masyarakat Terhadap Sistem Pembayaran Mypertamina dengan Metode Random Forest , SVM , dan Naïve Bayes," vol. 1, no. 1, pp. 28–44, 2024.

[11] M. I. S, L. Nur, and D. Widyawati, "Analisis Sentimen Masyarakat Terhadap System Perbelanjaan Di Alfagift Dengan Metode Naive Bayes," vol. 2, no. 2, pp. 255–263, 2025.

[12] E. R. S. M. I Wayan Suardi, Irene R.H.T. Tangkawarow, "Perbandingan Nilai Akurasi Analisa Sentiment Pada Kata Kunci Pemilu 2024," vol. 14, no. 2, pp. 3105–3118, 2025.

[13] V. Alviani, S. Alam, and I. Kurniawan, "Analisis Sentimen Review Aplikasi Wetv Pada Platform Twitter Menggunakan Support Vector Machine," *STORAGE J. Ilm. Tek. dan Ilmu Komput.*, vol. 2, no. 3, pp. 143–149, 2023, doi: 10.55123/storage.v2i3.2351.

[14] A. R. Putra and D. E. Ratnawati, "ANALISIS SENTIMEN BERBASIS ASPEK PADA APLIKASI MOBILE MENGGUNAKAN NAÏVE BAYES BERDASARKAN ULASAN PENGGUNA PLAYSTORE (STUDI KASUS : JCONNECT MOBILE)," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 12, no. 2, pp. 293–300, 2025, doi: 10.25126/jtiik.2025127556.

[15] A. Sitanggang, Y. Umaidah, and R. I. Adam, "Analisis Sentimen Masyarakat Terhadap Program Makan Siang Gratis Pada Media Sosial X Menggunakan Algoritma Naïve Bayes," *JITET (Jurnal Inform. dan Tek. Elektro Ter.*, vol. 12, no. 3, 2024.

[16] H. C. Husada and A. S. Paramita, "Analisis Sentimen Pada Maskapai Penerbangan di Platform Twitter Menggunakan Algoritma Support Vector Machine (SVM)," *Teknika*, vol. 10, no. 1, pp. 18–26, 2021, doi: 10.34148/teknika.v10i1.311.

[17] N. Khoirunnisaa, K. Nabila Nastiti Kesuma, S. Setiawan, and A. Yunizar Pratama Yusuf, "Klasifikasi Teks Ulasan Aplikasi Netflix Pada Google Play Store Menggunakan Algoritma Naïve Bayes Dan Svm," *SKANIKA Sist. Komput. dan Tek. Inform.*, vol. 7, no. 1, pp. 64–73, 2024, doi: 10.36080/skanika.v7i1.3138.

[18] D. S. Nurrochmah, N. Rahaningsih, R. D. Dana, and C. L. Rohmat, "PENERAPAN ALGORITMA NAIVE BAYES DALAM ANALISIS SENTIMEN ULASAN APLIKASI KITALULUS DI GOOGLE PLAY STORE," *J. Inform. Terpadu*, vol. 11, no. 1, pp. 1–11, 2025.

[19] S. N. Nugraha, R. Pebrianto, A. Latif, and M. R. Firdaus, "Analisis Sentimen Twitter Terhadap Menteri Indonesia Dengan Algoritma Support Vector Machine Dan Naive Bayes," *E-Link J. Tek. Elektro dan Inform.*, vol. 17, no. 1, p. 1, 2022, doi: 10.30587/e-link.v17i1.3965.

[20] V. H. Leonardi, A. Ibrahim, R. D. Kurnia, and M. Afrina, "Analisis Sentimen Ulasan Aplikasi Pembelajaran Bahasa Menggunakan Metode VADER," *J. Algoritm.*, vol. 22, no. 1, pp. 767–776, 2025, doi: 10.33364/algoritma/v.22-1.2285.