JITET (Jurnal Informatika dan Teknik Elektro Terapan)
Vol. 13 No. 2, pISSN: 2303-0577 eISSN: 2830-7062 http://dx.doi.org/10.23960/jitet.v13i2.6280

ANALISIS SENTIMEN OPINI MASYARAKAT TERHADAP PILKADA 2024 DI MEDIA SOSIAL TWITTER MENGGUNAKAN ALGORITMA NAÏVE BAYES

Muhammad Jafar Siddiq[1*], Sandika Jayasri[2], Aldi Suhendi[3], Taufik Hidayat[4], Robby Rizky[5]

[1,2,3,4]Universitas Islam Syekh Yusuf; alamat; Jl. Maulana Yusuf Babakan, Tangerang, Banten
[5]Universitas Mathla'ul Anwar; alamat; Jalan Raya Labuan Cikaliung, Pandeglang, Banten

Received: 25 Februari 2025
Accepted: 27 Maret 2025
Published: 14 April 2025

Keywords:
Pilkada, Sentiment Analysis, Naïve Bayes, TF-IDF, Twitter.

Corespondent Email:
thidayat@unis.ac.id

Abstrak. Penelitian ini membahas analisis sentimen opini masyarakat terhadap Pilkada 2024 di media sosial Twitter menggunakan algoritma Naïve Bayes dan pembobotan TF-IDF. Data dikumpulkan melalui proses crawling menggunakan Python, menghasilkan 5.182 tweet yang kemudian diproses melalui tahap preprocessing, termasuk case folding, cleansing, stemming, dan labeling. Setelah preprocessing, 4.041 data digunakan untuk analisis sentimen dengan kategori sentimen Netral, Positif, dan Negatif. Hasil penelitian menunjukkan bahwa algoritma Naïve Bayes mampu memberikan akurasi sebesar 77%, dengan F1-score tertinggi pada kategori Netral sebesar 0,85. Tema dominan yang ditemukan melalui pembobotan TF-IDF meliputi keamanan, partisipasi masyarakat, dan keberhasilan Pilkada. Evaluasi menggunakan Confusion Matrix membuktikan bahwa metode Naïve Bayes efektif untuk memahami opini masyarakat, sehingga hasil analisis ini dapat memberikan wawasan berharga bagi pemangku kepentingan dalam meningkatkan strategi komunikasi dan keterlibatan publik.

Abstract. *This study examines public opinion sentiment analysis regarding the 2024 regional elections (Pilkada) on Twitter using the Naïve Bayes algorithm and TF-IDF weighting. Data was collected through crawling with Python, yielding 5,182 tweets processed through preprocessing stages, including case folding, cleansing, stemming, and labeling. After preprocessing, 4,041 data points were analyzed for sentiment classification into Neutral, Positive, and Negative categories. The results show that the Naïve Bayes algorithm achieved an accuracy of 77%, with the highest F1-score of 0.85 in the Neutral category. Key themes identified through TF-IDF weighting include security, public participation, and election success. Evaluation using the Confusion Matrix demonstrated that the Naïve Bayes method is effective in understanding public opinion, providing valuable insights for stakeholders to enhance communication strategies and public engagement.*

1. PENDAHULUAN

Pemilihan umum merupakan perwujudan kehendak rakyat dalam sistem demokrasi yang menjadi dasar kehidupan berbangsa dan bernegara. Demokrasi, seperti yang didefinisikan oleh Maurice Duverger, adalah sistem pemerintahan di mana kedudukan antara yang memerintah dan yang diperintah sejajar. Oleh karena itu, pemilu memiliki peran penting dalam

menentukan regenerasi kepemimpinan yang mewakili aspirasi rakyat [1].

Pilkada sebagai bagian dari rezim pemilu mulai diterapkan secara langsung oleh rakyat sejak berlakunya Undang-Undang Nomor 32 Tahun 2004 dan pertama kali dilaksanakan pada tahun 2005. Penyelenggaraan pilkada bertujuan untuk memastikan kedaulatan rakyat melalui partisipasi aktif masyarakat dalam memilih pemimpin daerah [2] [3].

Di era digital, media sosial seperti *Twitter* telah menjadi platform utama bagi masyarakat untuk menyampaikan opini terkait Pilkada. Analisis sentimen terhadap opini masyarakat di media sosial dapat memberikan wawasan penting tentang persepsi publik terhadap pelaksanaan Pilkada dan proses demokrasi secara umum [4].

Penelitian ini memanfaatkan teknologi *machine learning*, khususnya metode *supervised learning*, untuk menganalisis sentimen masyarakat. *Supervised learning* bekerja dengan melatih model menggunakan data berlabel, seperti kategori sentimen positif, negatif, dan netral. Salah satu algoritma yang digunakan adalah *Naïve Bayes*, yang dikenal karena kecepatan dan akurasinya dalam mengklasifikasikan teks [5] [6]. Dalam studi berjudul *"Analisis Sentimen Ulasan Pengguna Aplikasi MyPertamina di Google Playstore dengan Metode Naïve Bayes"*, menunjukkan bahwa algoritma *Naïve Bayes* mampu mencapai tingkat akurasi hingga 91% dalam analisis sentimen teks. Berdasarkan potensi ini, penelitian dilakukan untuk menerapkan algoritma *Naïve Bayes* dan model *TF-IDF* dalam menganalisis sentimen positif, negatif, dan netral terhadap Pilkada di media sosial *Twitter*.

Penelitian ini bertujuan untuk mengevaluasi akurasi algoritma *Naïve Bayes* dalam mengklasifikasikan sentimen masyarakat serta menentukan kecenderungan opini publik terhadap Pilkada. Hasil dari analisis ini diharapkan dapat memberikan informasi berharga bagi para pemangku kebijakan, penyelenggara Pilkada, dan masyarakat untuk meningkatkan kualitas demokrasi [7] [8].

2. TINJAUAN PUSTAKA

2.1. Pilkada

Pilkada adalah bentuk implementasi sistem demokrasi tidak langsung atau demokrasi perwakilan. Pemilihan Kepala Daerah dilakukan langsung oleh rakyat melalui mekanisme Pemilu untuk memastikan pemimpin daerah menjalankan tugas atas nama rakyat [9].

2.2. Naïve Bayes

Naïve Bayes adalah metode klasifikasi sederhana namun efektif, dengan tingkat akurasi tinggi dalam analisis teks. Metode ini memberikan hasil yang cepat dan andal [10].

2.3. TF-IDF

TF-IDF (Term Frequency-Inverse Document Frequency) adalah metode pembobotan kata untuk menghitung seberapa penting sebuah kata dalam dokumen. Proses ini mengalikan frekuensi kemunculan kata dengan inversi frekuensi dokumen [11].

2.4. Machine Learning

Machine learning adalah bidang *interdisipliner* yang menggabungkan konsep dari ilmu komputer, statistik, ilmu kognitif, teknik, teori optimasi, serta berbagai cabang matematika dan sains. Bidang ini berfokus pada pengembangan sistem yang mampu belajar dan membuat keputusan berdasarkan data [12][13].

2.5. Supervised Machine Learning

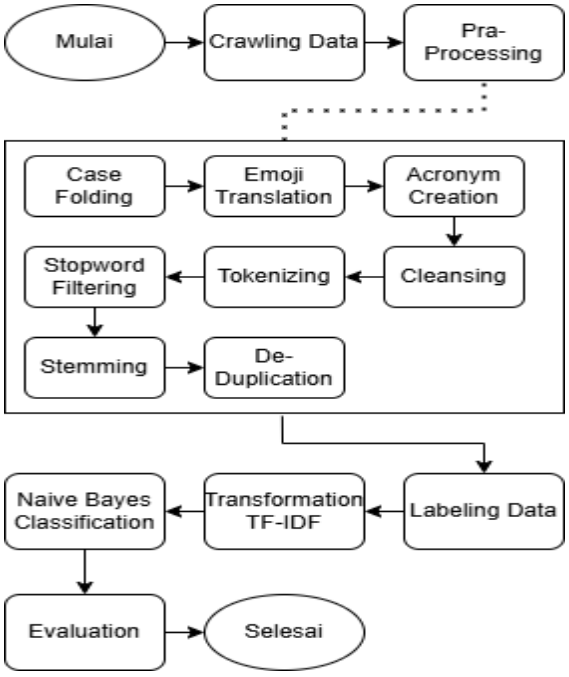
Supervised machine learning adalah teknik yang digunakan untuk mempelajari hubungan antara atribut input dan atribut target, sehingga memungkinkan model memprediksi nilai target berdasarkan data input yang diberikan [13].

2.6. Twitter

Twitter adalah platform media sosial yang populer di Indonesia dan digunakan masyarakat untuk menyampaikan sentimen publik tentang berbagai isu, termasuk Pilkada. *Twitter* menjadi sumber data penting dalam analisis sentimen karena beragam topik yang dibahas [14].

3. METODE PENELITIAN

Penelitian ini menggunakan metode klasifikasi dengan algoritma *Naïve Bayes*. Tahapan yang dilakukan meliputi: Pengambilan Data *(Crawling)*, Pemberian Label pada Data, Pra-pemrosesan, Transformasi Data, Klasifikasi dengan *Naïve Bayes*, Evaluasi, dan Visualisasi. Kerangka penelitian dapat dilihat pada Gambar 1 di bawah ini.



Gambar 1. Diagram Alur

3.1. Crawling

Crawling adalah tahap awal untuk mengumpulkan data dari *Twitter* menggunakan alat seperti *tweet-harvest*. Program ini memerlukan input berupa nama file, kata kunci pencarian, batas jumlah data, dan token autentikasi *Twitter*. Data diambil untuk mendapatkan *tweet* yang relevan dengan Pilkada [15].

3.2. Preprocessing

Preprocessing adalah langkah awal untuk membersihkan dan menyiapkan teks agar siap dianalisis. Proses ini melibatkan penghapusan elemen-elemen tidak informatif, seperti URL, tanda baca, atau karakter lain yang tidak relevan [16]. Sebelum dianalisis, data yang telah dikumpulkan diproses melalui tahapan berikut: *Case Folding, Cleansing, Tokenizing, Stemming, Stopword Filtering, Emoji Translation, Acronym Creation*, dan *Deduplication.* Data yang telah diproses akan lebih siap untuk digunakan dalam analisis sentimen.

3.3. Labelling Data

Labelling Data adalah proses pemberian label sentimen pada data yang telah dikumpulkan menggunakan metode seperti VADER *(Valence Aware Dictionary and Sentiment Reasoner)*. Kategori sentimen yang digunakan mencakup positif, negatif, dan netral untuk tweet yang tidak menunjukkan kecenderungan emosi tertentu [17].

3.4. Transformation TF-IDF

Transformation menggunakan metode *TF-IDF (Term Frequency-Inverse Document Frequency)* untuk menghitung bobot setiap kata berdasarkan relevansinya dalam suatu dokumen dibandingkan dengan keseluruhan dataset. Proses ini memungkinkan analisis kemiripan antar kata dalam dokumen yang berbeda [18].

3.5. Naïve Bayes

Naïve Bayes Classification adalah metode statistika yang digunakan untuk mengklasifikasikan sentimen *tweet* berdasarkan probabilitas kata-kata dalam teks. Algoritma ini mengasumsikan semua atribut bersifat independen dan terbukti efektif untuk *text mining* dengan tingkat akurasi dan kecepatan tinggi [19]. Dalam penelitian ini, data dibagi menjadi data pelatihan (80%) untuk membangun model dan data pengujian (20%) untuk mengukur akurasi klasifikasi sentimen menjadi positif, negatif, atau netral.

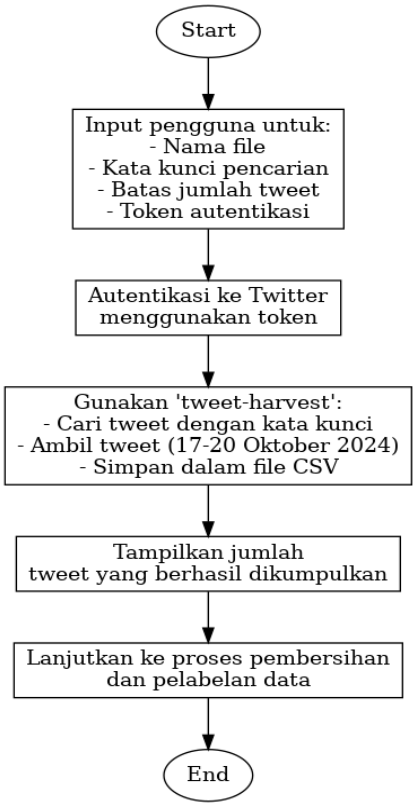
3.6. Evaluation

Evaluation dilakukan untuk menilai kinerja model setelah proses klasifikasi menggunakan *Naïve Bayes*. Hasilnya dievaluasi menggunakan *Confusion Matrix*, yang menunjukkan jumlah prediksi benar dan salah untuk setiap kelas. Pengukuran mencakup *Accuracy, Precision, dan Recall*, memberikan gambaran seberapa baik model memprediksi sentimen berdasarkan data uji [8].

4. HASIL DAN PEMBAHASAN

4.1. Crawling Data

Proses awal penelitian ini adalah mengumpulkan data dari platform media sosial *Twitter* menggunakan alat *tweet-harvest*. Program meminta input berupa nama file, kata kunci pencarian, batas jumlah data yang diambil, serta token autentikasi *Twitter*. Data yang diambil disimpan dalam format CSV pada folder tertentu untuk diolah lebih lanjut. Berikut adalah *Flowchart* untuk melakukan *Crawling*:



Data di bawah ini merupakan hasil crawling yang belum melalui proses pembersihan dan pelabelan. Data yang ditampilkan masih dalam bentuk aslinya, dan Tabel 1 ini hanya menampilkan dua data sebagai contoh hasil *crawling* yang diperoleh dari total 5182 data.

Tabel 1. Hasil *Crawling* Data

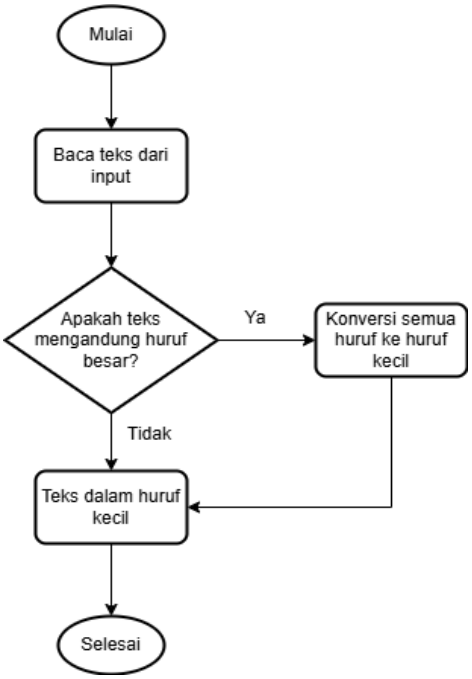
no	created_at	full_text	username
1	Sat Oct 19 06:55:09 +0000 2024	Pilkada 2024 harus berjalan dengan penuh sukacita. Seluruh elemen masyarakat perlu menciptakan suasana pemilihan yang damai dan harmonis. #sukseskanpilkada2024 #PilkadaAmanDamai #NetralitasPilkada2024 #Pilkada2024	IrihamYauna
		https://t.co/K0h7EXrCRj	
2	Thu Oct 17 05:11:39 +0000 2024	Calon Gubernur Jakarta nomor urut 3 @pramonoanung melakukan blusukan ke pasar Munjul Cipayung Jakarta Timur Kamis (17/10) di masa kampanye Pilkada 2024. Dalam kunjungannya Pramono mendatangi sejumlah los untuk menyapa para pedagang dan juga berdialog dengan Perkumpulan https://t.co/ctBK3IkwwH	Mata_Netizen62

4.2. Prepocessing Data

Sebelum dianalisis, data yang telah dikumpulkan perlu diproses agar lebih siap digunakan. Tahapan *preprocessing* meliputi beberapa langkah berikut:

4.2.1. Case Folding

Case Folding adalah proses dalam *text preprocessing* yang mengubah semua huruf dalam teks menjadi huruf kecil. Berikut adalah *Flowchart* untuk melakukan *Case Folding* :



Hasil dari proses *Case Folding* dapat dilihat pada Tabel 2 dibawah ini:

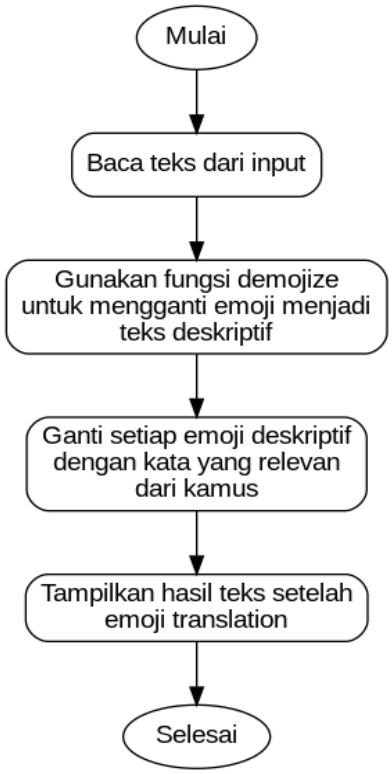
Tabel 2. Hasil Proses *Case Folding*

Sebelum
Pilkada 2024 harus berjalan dengan penuh sukacita. Seluruh elemen masyarakat perlu menciptakan suasana pemilihan yang damai dan harmonis.
Sesudah
pilkada 2024 harus berjalan dengan penuh sukacita. seluruh elemen masyarakat perlu menciptakan suasana pemilihan yang damai dan harmonis.

Pada Tabel 2, kolom "Sebelum" menunjukkan teks sebelum dilakukan proses *case folding*, di mana terdapat huruf kapital seperti "Pilkada" dan "Seluruh". Kolom "Sesudah" menunjukkan hasil setelah *case folding*, di mana semua huruf telah diubah menjadi huruf kecil.

4.2.2. Emoji Translation

Library emoji menyediakan fungsi untuk mengonversi emoji menjadi representasi teks deskriptif. Berikut adalah *Flowchart* untuk melakukan *Emoji Translation* :



Hasil dari proses *Emoji Translation* dapat dilihat pada Tabel 3 dibawah ini:

Tabel 3. Hasil Proses *Emoji Translation*

Sebelum
ayo dukung pilkada aman damai 😊, pakai hak pilih bijak dan baik!
Sesudah
Ayo dukung pilkada aman damai senang, pakai hak pilih bijak dan baik!

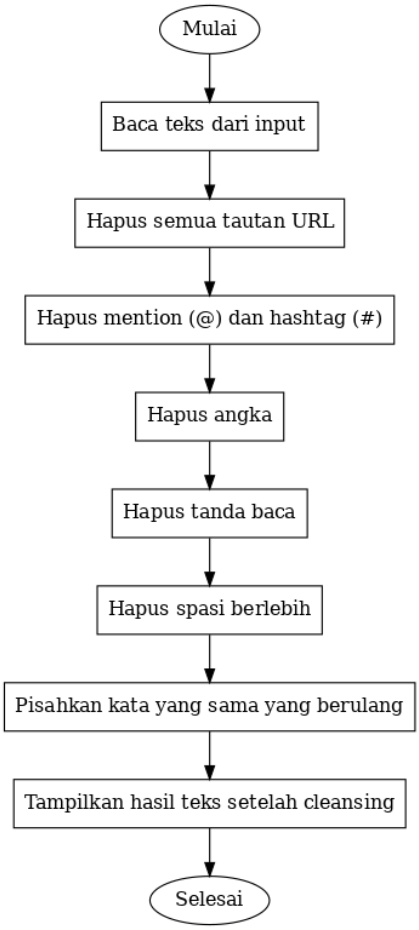
Pada Tabel 3, kolom "Sebelum" menunjukkan teks yang mengandung emoji, seperti 😊, yang merupakan representasi grafis dari ekspresi atau perasaan. Emoji ini digunakan untuk menyampaikan perasaan positif, dalam hal ini, kegembiraan atau kebahagiaan, terkait dengan dukungan terhadap pilkada yang aman dan damai.

Namun, dalam analisis teks atau pemrosesan data, emoji tersebut tidak dapat langsung dianalisis sebagai teks. Setelah melalui proses *Emoji Translation*, emoji tersebut pertama kali digantikan dengan kalimat deskriptif yang lebih jelas dan mudah dipahami. Fungsi *demojize* dalam library emoji mengonversi emoji 😊 menjadi bentuk teks deskriptif, yaitu *":grinning_face_with_smiling_eyes:"*, yang secara eksplisit menggambarkan emoji tersebut.

Selanjutnya di ubah menggunakan kamus yang sudah dibuat *":grinning_face_with_smiling_eyes: 'senang'"*, jadi output dari emoji 😊 akan menjadi kata 'senang' yang akan dilakukan analisis.

4.2.3. Cleansing

Cleansing adalah proses dalam text preprocessing yang digunakan untuk membersihkan teks dari elemen-elemen yang tidak relevan seperti tautan URL, angka, tanda baca, dan simbol khusus. Berikut adalah *Flowchart* untuk melakukan *Cleansing* :



Hasil dari proses *Cleansing* dapat dilihat pada Tabel 4 dibawah ini:

Tabel 4. Hasil Proses *Cleansing*

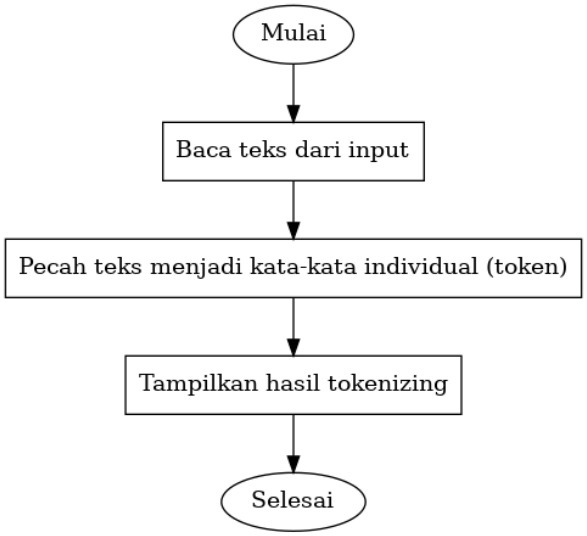
Sebelum
yuk kita dukung Pilkada 2024 aman damai! Pakai hak pilih dgn bijak pilih yg terbaik! https://t.co/qSuvtm4w2R
Sesudah
yuk kita dukung pilkada aman damai pakai hak pilih dengan bijak pilih yang terbaik

Pada Tabel 4, kolom "Sebelum" menunjukkan teks yang mengandung elemen-elemen yang tidak relevan untuk analisis, seperti tautan URL (https://t.co/qSuvtm4w2R), angka (2024). Semua elemen ini dapat mengganggu proses analisis teks karena tidak memberikan informasi yang berguna.

Setelah melalui proses *Cleansing*, elemen-elemen tersebut dihapus atau diganti. Tautan URL dihapus, angka 2024 dihapus. Hasilnya adalah teks yang lebih bersih dan terstruktur, yang hanya berfokus pada informasi relevan untuk analisis, seperti dukungan terhadap pilkada yang aman dan damai.

4.2.4. Tokenizing

Tokenizing adalah proses dalam text preprocessing yang digunakan untuk memecah teks menjadi unit-unit yang lebih kecil, biasanya berupa kata-kata yang disebut token. Berikut adalah *Flowchart* untuk melakukan *Tokenizing* :



Hasil dari proses *Tokenizing* dapat dilihat pada Tabel 5 dibawah ini:

Tabel 5. Hasil Proses *Tokenizing*

Sebelum
ayo gaes kita semua bijak milih biar pilkada aman lancar Indonesia
Sesudah
['ayo', 'gaes', 'kita', 'semua', 'bijak', 'milih', 'biar', 'pilkada', 'aman', 'lancar', 'Indonesia']

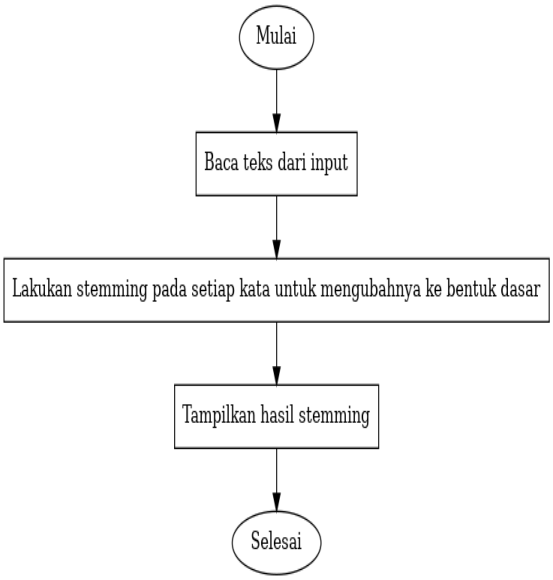
Pada Tabel 5, kolom "Sebelum" menunjukkan teks asli yang masih dalam bentuk kalimat utuh. Proses *Tokenizing* memecah kalimat tersebut menjadi token-token individual, yaitu kata-kata yang lebih kecil, yang ditampilkan di kolom "Sesudah". Token-token ini adalah unit dasar

yang akan dianalisis lebih lanjut dalam tahap-tahap berikutnya dalam text preprocessing.

Proses *Tokenizing* sangat berguna karena memudahkan analisis per kata. Setiap kata yang terpisah (token) dapat dianalisis secara terpisah, misalnya untuk identifikasi pola atau analisis sentimen. Dengan menggunakan *Tokenizing*, kita dapat mendapatkan representasi kata-per-kata dari teks, yang memungkinkan kita untuk melakukan pemrosesan lebih lanjut, seperti penghapusan stopwords, stemming, dan lain-lain.

4.2.5. Stemming

Stemming adalah proses dalam text preprocessing yang digunakan untuk mengubah kata ke bentuk dasarnya (*root word*). Berikut adalah *Flowchart* untuk melakukan *Stemming* :



Hasil dari proses *Stemming* dapat dilihat pada Tabel 6 dibawah ini:

Tabel 6. Hasil Proses *Stemming*

Sebelum
bersemangat pilkada pilih pemimpin yang jalankan visi misi ayo partisipasi
Sesudah
['semangat', 'pilkada', 'pilih', 'pemimpin', 'yang', 'jalankan', 'visi', 'misi', 'ayo', 'partisipasi']

Pada Tabel 6, kolom "Sebelum" menunjukkan teks asli yang mengandung kata-kata yang telah terinfleksi (berbeda bentuk, misalnya "bersemangat" atau "pemilihan"). Proses Stemming mengubah kata-kata tersebut menjadi bentuk dasarnya yang lebih seragam, seperti "bersemangat" menjadi "semangat" dan "pemilihan" menjadi "pilih". Hasil stemming ini ditampilkan pada kolom "Sesudah" dalam bentuk daftar kata dasar (*root words*).

Proses *Stemming* sangat penting dalam analisis teks karena memungkinkan untuk menyederhanakan variasi kata dan menyamakan kata-kata yang memiliki makna serupa tetapi bentuk yang berbeda. Hal ini membantu dalam meningkatkan efisiensi analisis teks, terutama untuk tugas seperti klasifikasi teks atau analisis sentimen.

Dengan menggunakan *Stemming*, kita dapat mengelompokkan kata-kata yang memiliki arti yang sama, meskipun memiliki bentuk yang berbeda. Ini memungkinkan analisis yang lebih tepat dan mengurangi kompleksitas data.

4.2.6. Stopword Filtering

Stopword Filtering adalah proses dalam text preprocessing untuk menghapus kata-kata yang sering muncul dalam bahasa tetapi tidak memiliki makna signifikan dalam analisis teks. Berikut adalah *Flowchart* untuk melakukan *Stopword Filtering* :

Hasil dari proses *Stopword Filtering* dapat dilihat pada Tabel 7 dibawah ini:

Tabel 7. Hasil *Stopword Filtering*

Sebelum
semoga pilkada di wilayah bojonegoro aman damai dan kondusif
Sesudah

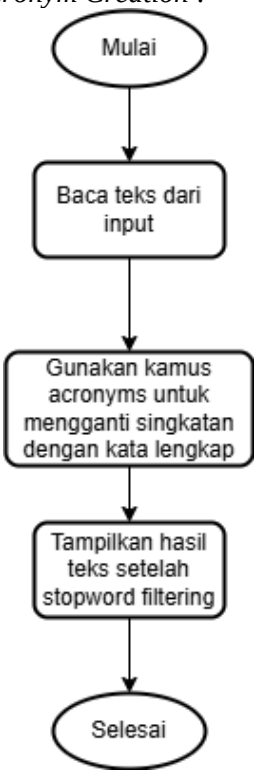
['semoga', 'pilkada', 'wilayah', 'bojonegoro', 'aman', 'damai', 'kondusif']

Pada Tabel 7, kolom "Sebelum" menunjukkan teks asli yang mengandung kata-kata umum seperti "di", "dan", dan "yang", yang biasanya tidak memiliki makna penting dalam konteks analisis teks. Proses *Stopword Filtering* menghapus kata-kata ini, sehingga hanya kata-kata yang lebih relevan yang tersisa untuk analisis lebih lanjut, seperti "pilkada", "wilayah", "bojonegoro", dan lainnya. Hasilnya ditampilkan pada kolom "Sesudah".

Dengan melakukan *Stopword Filtering*, kita mengurangi "noise" dalam data, sehingga analisis dapat lebih fokus pada kata-kata yang membawa informasi penting, yang meningkatkan efisiensi dan akurasi dalam proses analisis teks.

4.2.7. Acronym Creation

Acronym Creation adalah proses dalam text preprocessing yang mengubah singkatan atau istilah gaul yang sering digunakan, seperti "ttg" (tentang) atau "ttp" (tetap), menjadi kata lengkap agar teks lebih mudah dipahami dan diproses oleh algoritma. Berikut adalah *Flowchart* untuk melakukan *Acronym Creation* :



Hasil dari proses *Acronym Creation* dapat dilihat pada Tabel 8 dibawah ini:

Tabel 8. Hasil *Acronym Creation*

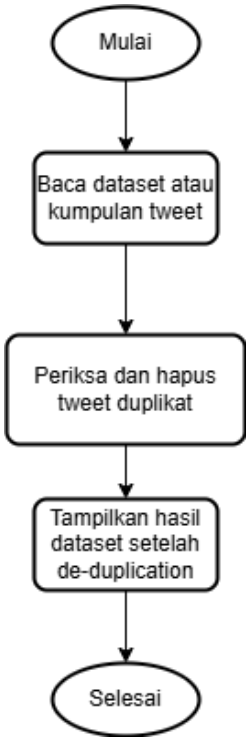
Sebelum
ttp jaga kedamaian di pilkada serentak ayo buktiin kalau kita dewasa dan bijaksana
Sesudah
tetap jaga kedamaian di pilkada serentak ayo buktiin kalau kita dewasa dan bijaksana

Pada Tabel 8, kolom "Sebelum" menunjukkan teks yang mengandung singkatan seperti "ttp", yang merujuk pada "tetap". Proses *Acronym Creation* menggantikan singkatan tersebut dengan kata lengkap yang lebih jelas, dalam hal ini "tetap". Kolom "Sesudah" menunjukkan hasil setelah proses acronym creation, di mana semua singkatan telah digantikan dengan kata-kata yang lebih lengkap dan mudah dipahami.

Dengan melakukan *Acronym Creation*, teks yang berisi singkatan dapat dipahami lebih baik oleh algoritma analisis teks, karena singkatan-singkatan tersebut digantikan dengan istilah yang lebih formal dan lengkap. Proses ini membantu meningkatkan kualitas analisis dan pemrosesan data, sehingga hasil analisis lebih akurat.

4.2.8. De-Duplication

De-Duplication adalah proses dalam text preprocessing yang menghapus tweet atau data yang sama atau duplikat dari dataset. Berikut adalah *Flowchart* untuk melakukan *De-Duplication* :



Hasil dari proses *De-Duplication* dapat dilihat pada Tabel 9 dibawah ini:

Tabel 9. Hasil *De-Duplication*

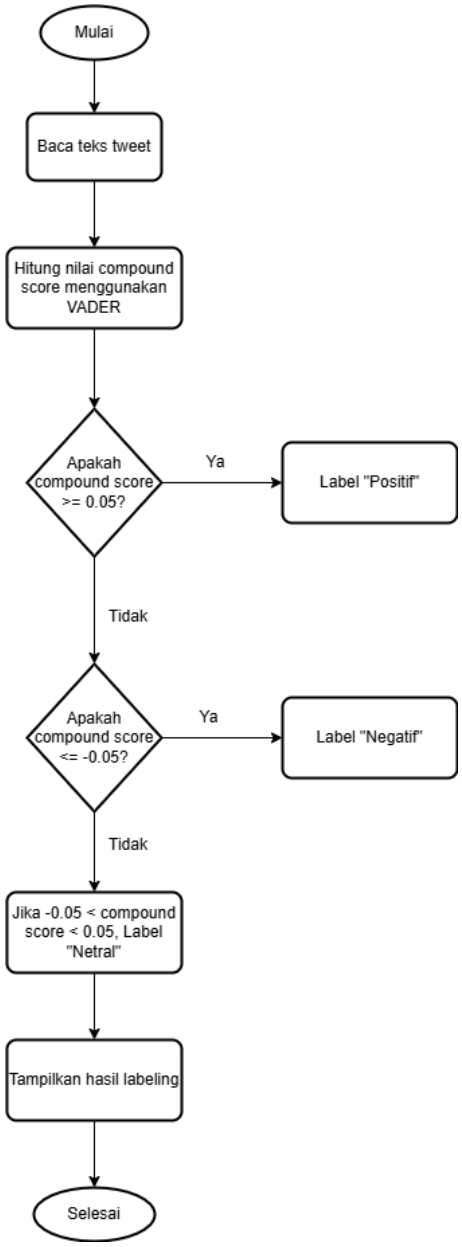
Sebelum
semoga tahapan pilkada serentak berjalan dalam keadaan aman lancar dan kondusif
semoga tahapan pilkada serentak berjalan dalam keadaan aman lancar dan kondusif
Sesudah
semoga tahapan pilkada serentak berjalan dalam keadaan aman lancar dan kondusif

Pada Tabel 9, kolom "Sebelum" menunjukkan dua tweet yang sama persis, yang dapat terjadi dalam pengumpulan data dari media sosial. Proses *De-Duplication* menghapus tweet duplikat tersebut sehingga hanya ada satu tweet yang tersisa di dataset. Kolom "Sesudah" menunjukkan hasil setelah proses *de-duplication*, di mana tweet yang berulang telah dihapus dan hanya satu tweet yang tersisa.

Dengan melakukan *De-Duplication*, kita memastikan bahwa dataset yang digunakan untuk analisis tidak mengandung data yang duplikat, yang bisa mengganggu hasil analisis atau memberikan bobot yang berlebihan pada data yang sama. Proses ini membantu menjaga kualitas dan keakuratan dataset yang digunakan dalam penelitian atau analisis.

4.3. Labelling Data

Labelling Data adalah tahap di mana setiap tweet diberikan label sentimen menggunakan metode analisis sentimen seperti *VADER (Valence Aware Dictionary and sEntiment Reasoner).* Berikut adalah *Flowchart* untuk melakukan *Labeling Data* :



Hasil dari proses Pelabelan dapat dilihat pada Tabel 10 dibawah ini:

Tabel 10. Hasil *Labeling Data*

full_text	**sentiment**
kalau kelompok kalian yang menang baru pemilu dan pilkada jujur ya otak unta ya begini jadinya orang luar kau agung agungkan tapi pribumi asli kau kesampingkan makanya belajar menghormati leluhur sendiri seringin datang	Negatif

ke kuburnya kirim doa dan ziarah jangan junjung leluhur orang	
setuju banget ayo gunakan hak pilih kita dengan bijak demi pilkada damai dan ceria	Positif
pemilu sudah selesai tinggal pilkadanya sekarang	Netral

Pada Tabel 10, kolom "full_text" berisi tweet asli, dan kolom "sentiment" menunjukkan label sentimen yang diberikan berdasarkan nilai compound score yang dihitung menggunakan *VADER*. Tweet pertama yang berisi pernyataan negatif tentang kelompok tertentu diberi label "Negatif", karena skor compound yang dihitung *VADER* berada di bawah -0,05. Tweet kedua, yang berisi ajakan untuk menggunakan hak pilih dengan bijak untuk pilkada damai, diberi label "Positif", karena skor compound lebih besar dari 0,05. Tweet ketiga yang menyatakan bahwa pemilu sudah selesai dan sekarang tinggal pilkada, diberi label "Netral", karena skor compound berada di antara -0,05 dan 0,05.

Dengan melakukan Labeling Data menggunakan *VADER*, kita dapat dengan mudah mengelompokkan tweet berdasarkan sentimen yang terkandung dalam teks. Proses ini mempermudah analisis sentimen secara otomatis tanpa memerlukan penilaian manual, yang sangat efisien terutama untuk dataset yang besar.

Hasil dari Jumlah Tweet dapat dilihat pada Tabel 11 dibawah ini:

Tabel 11. Jumlah Tweet Netral, Positif dan Negatif

sentiment	jumlah_data
Netral	3493
Positif	353
Negatif	195
Total	**4041**

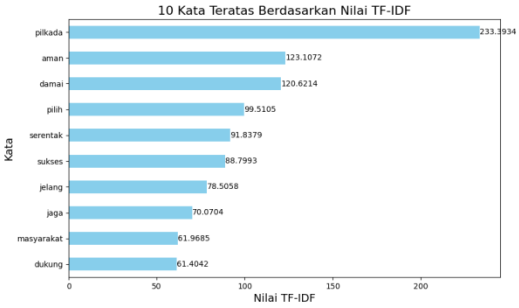
Pada Tabel 11, jumlah data yang dilabeli tidak sama dengan data awal karena proses *de-duplication* yang dilakukan sebelumnya, di mana tweet yang duplikat telah dihapus. Sebelumnya, data terdiri dari 5.182 tweet, namun setelah melalui proses *de-duplication*, jumlah data yang tersisa adalah 4041 tweet, yang sudah dibagi ke dalam tiga kategori sentimen: Netral (3493 tweet), Positif (353 tweet), dan Negatif (195 tweet). Proses ini penting untuk memastikan bahwa hasil analisis sentimen tidak terganggu oleh data duplikat yang dapat memberikan bias pada hasil akhir.

4.4. TF-IDF

TF-IDF (Term Frequency-Inverse Document Frequency) adalah metode yang digunakan untuk mengevaluasi seberapa penting sebuah kata dalam suatu dokumen relatif terhadap keseluruhan dataset. berlangsung. Berikut adalah *Pseudocode* untuk melakukan *TF-IDF* :

1. Mulai
2. Baca dataset
3. Inisialisasi objek *TF-IDF* menggunakan *TfidfVectorizer*
4. Fitting dan transformasi dataset menggunakan *TF-IDF*
5. Dapatkan nilai *TF-IDF* untuk setiap kata dalam dokumen
6. Tampilkan hasil *TF-IDF*
7. Selesai

Hasil dari *TF-IDF* dapat dilihat pada Gambar 2 dan 3 dibawah ini:



Gambar 2. Hasil Frekuensi *TF-IDF*

Gambar 3. Hasil Word Cloud *TF-IDF*

Kata "pilkada" memiliki nilai *TF-IDF* yang tinggi (233,3934), menunjukkan bahwa kata ini sangat penting dalam konteks pilkada yang dibahas dalam dataset. Kata "aman" (123,1072)

dan "damai" (120,6214) juga memiliki bobot yang tinggi, menandakan bahwa tema utama dari dataset berfokus pada aspek keamanan dan damainya pelaksanaan pilkada. Kata-kata lain seperti "pilih" (99,5105), "serentak" (91,8379), dan "sukses" (88,7993) menonjolkan pentingnya partisipasi masyarakat dan keberhasilan pilkada. Kata "masyarakat" (61,9685) dan "dukung" (61,4042) mengindikasikan fokus pada peran serta masyarakat dan dorongan untuk mendukung pelaksanaan pilkada.

Proses *TF-IDF* ini memberikan gambaran yang jelas tentang kata-kata yang paling relevan dan penting dalam dataset, yang berguna untuk analisis lebih lanjut mengenai sentimen atau topik yang dibahas dalam teks.

4.5. Naïve Bayes Classification

Algoritma *Naïve Bayes* digunakan untuk mengklasifikasikan sentimen tweet berdasarkan probabilitas kata-kata yang ada dalam tweet tersebut. Algoritma ini menghitung kemungkinan suatu tweet termasuk dalam kategori sentimen tertentu (positif, negatif, atau netral) berdasarkan kata-kata yang muncul di dalamnya. Berikut adalah *Pseudocode* untuk melakukan *Naïve Bayes Classification* :

1. Mulai
2. Baca dataset tweet yang telah melalui tahap pra-processing
3. Pisahkan dataset menjadi data pelatihan dan data pengujian
4. Inisialisasi model *Naïve Bayes*
5. Latih model menggunakan data pelatihan
6. Prediksi sentimen untuk data pengujian
7. Hitung metrik evaluasi (*precision, recall, F1-score*, dan *accuracy*)
8. Tampilkan hasil evaluasi model
9. Selesai

Hasil dari *Naïve Bayes Classification* dapat dilihat pada Tabel 12 dibawah ini:

Tabel 12. Hasil *Naïve Bayes Classification*

	precision	**recall**	**F1-score**	**support**	**Accuracy**
negatif	0.44	0.72	0.54	39.00	0.77
netral	0.96	0.76	0.85	699.00	
positif	0.31	0.83	0.45	71.00	

Pada Tabel 12, *Precision* mengukur ketepatan model dalam mengklasifikasikan tweet ke dalam kategori yang benar. Nilai *precision* yang lebih tinggi menunjukkan bahwa model lebih jarang salah dalam memprediksi kelas sentimen. Negatif: 0.44 (model cukup sering salah mengklasifikasikan tweet negatif). Netral: 0.96 (model sangat tepat dalam mengklasifikasikan tweet netral). Positif: 0.31 (model kurang tepat dalam mengklasifikasikan tweet positif)

Recall mengukur kemampuan model untuk menangkap semua tweet yang termasuk dalam kategori sentimen tersebut. *Recall* yang tinggi menunjukkan bahwa model tidak melewatkan banyak tweet yang relevan untuk kelas tersebut. Negatif: 0.72 (model mampu menangkap sebagian besar tweet negatif). Netral: 0.76 (model mampu menangkap sebagian besar tweet netral). Positif: 0.83 (model menangkap sebagian besar tweet positif)

F1-Score adalah rata-rata harmonis dari *precision* dan *recall*. Nilai yang lebih tinggi menunjukkan keseimbangan yang baik antara *precision* dan *recall*. Negatif: 0.54. Netral: 0.85. Positif: 0.45

Accuracy adalah ukuran seberapa banyak prediksi yang benar dibandingkan dengan total data. Dalam hal ini, model memiliki *accuracy* 0.77, yang berarti 77% dari prediksi model benar.

4.6. Evaluation

Setelah proses klasifikasi *menggunakan Naïve Bayes* selesai, tahap evaluasi bertujuan untuk menilai seberapa baik model dalam memprediksi sentimen berdasarkan data pengujian. Evaluasi ini dapat dilakukan dengan menggunakan *Confusion Matrix*, yang memberikan gambaran lebih mendalam tentang kinerja model dengan menampilkan jumlah prediksi yang benar dan salah untuk masing-masing kelas. Berikut adalah *Pseudocode* untuk melakukan *Confusion Matrix* :

1. Mulai
2. Baca hasil prediksi model (y_pred) dan label sebenarnya (y_test)
3. Hitung *Confusion Matrix* menggunakan fungsi *confusion_matrix* dari *scikit-learn*
4. Tampilkan *Confusion Matrix*
5. Selesai

Adapun *Confussion Matrix* tersebut dapat dilihat pada Tabel 13 dibawah ini:

Tabel 13. Hasil *Confussion Matrix*

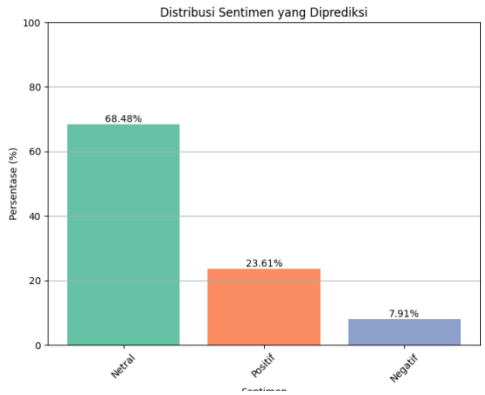
Aktual	**Prediksi**

	Positif	Negatif	Netral
Positif	28	10	1
Negatif	25	533	131
Netral	1	11	59

Pada Tabel 13, *Confusion Matrix* diatas, setiap nilai merepresentasikan hasil prediksi model dibandingkan dengan nilai aktual. Sebagai contoh, sebanyak 28 prediksi pada baris Positif dan kolom Positif menunjukkan bahwa model memprediksi Positif, dan nilai aktual juga Positif (benar/true positive). Namun, ada 10 kasus pada baris Positif dan kolom Negatif di mana model memprediksi Positif, tetapi nilai aktual adalah Negatif (salah/false positive). Selain itu, terdapat 1 kasus pada baris Positif dan kolom Netral di mana model memprediksi Positif, tetapi nilai aktual adalah Netral (salah/false positive).

Pada baris Negatif, terdapat 35 kasus di mana model memprediksi Negatif, tetapi nilai aktual adalah Positif (salah/false negative), dan sebanyak 533 kasus di mana model memprediksi Negatif dengan nilai aktual juga Negatif (benar/true negative). Selain itu, ada 131 kasus di mana model memprediksi Negatif, tetapi nilai aktual adalah Netral (salah/false negative).

Pada baris Netral, terdapat 1 kasus di mana model memprediksi Netral, tetapi nilai aktual adalah Positif (salah/false positive). Selain itu, sebanyak 11 kasus menunjukkan bahwa model memprediksi Netral, tetapi nilai aktual adalah Negatif (salah/false positive). Akhirnya, terdapat 59 kasus di mana model memprediksi Netral, dan nilai aktual juga Netral (benar/true positive). Analisis ini membantu mengidentifikasi kekuatan dan kelemahan model dalam mengklasifikasikan data.



Gambar 4. Hasil *Naïve Bayes Classification*

Pada Gambar 4, menunjukkan hasil analisis sentimen yang dilakukan menggunakan model *Naïve Bayes*, dengan distribusi sentimen yang diprediksi ke dalam tiga kategori: netral, positif, dan negatif. Sentimen netral mendominasi hasil prediksi dengan persentase sebesar 68,48%, menunjukkan bahwa mayoritas teks yang dianalisis memiliki konten yang bersifat netral, tanpa emosi positif atau negatif yang kuat. Sentimen positif berada di posisi kedua dengan persentase 23,61%, yang mencerminkan sebagian teks menunjukkan emosi positif. Sementara itu, sentimen negatif memiliki proporsi paling kecil, yaitu 7,91%, menunjukkan bahwa data cenderung jarang mengandung emosi negatif. Dari distribusi ini, dapat disimpulkan bahwa data yang dianalisis lebih banyak bersifat netral, yang kemungkinan disebabkan oleh sifat dataset yang digunakan, seperti jenis teks atau topik yang tidak terlalu emosional. Jika diperlukan keseimbangan antara kategori sentimen, metode seperti oversampling atau undersampling dapat dipertimbangkan.

KESIMPULAN

a. Hasil akurasi Analisis Sentimen, penelitian ini menggunakan algoritma *Naïve Bayes* yang dipadukan dengan pembobotan *TF-IDF* untuk menganalisis sentimen opini masyarakat terkait Pilkada 2024 di media sosial Twitter. Hasil analisis menunjukkan bahwa model *Naïve Bayes* mampu menghasilkan akurasi sebesar 77%, yang menunjukkan bahwa algoritma ini dapat diterapkan dengan baik dalam mengklasifikasikan sentimen masyarakat.
b. Pengumpulan dan Pemrosesan Data, data opini masyarakat dikumpulkan melalui *crawling* di Twitter menggunakan tool *tweet-harvest*, yang menghasilkan total 5.182 tweet. Setelah melalui proses *preprocessing* yang meliputi *case folding, cleansing, stemming, dan labeling*, jumlah data yang diproses berkurang menjadi 4.041 tweet yang terbagi dalam tiga kategori sentimen: Netral (3.493 tweet), Positif (353 tweet), dan Negatif (195 tweet). Proses de-duplication telah dilakukan untuk memastikan data yang digunakan bebas dari duplikat, sehingga hasil analisis lebih akurat.
c. Efektivitas Algoritma *Naïve Bayes*, Algoritma *Naïve Bayes* terbukti efektif dalam mengidentifikasi sentimen masyarakat. Hasil

evaluasi menunjukkan bahwa kategori sentimen netral memiliki *F1-score* tertinggi (0.85), menunjukkan kemampuan model yang baik dalam mengklasifikasikan *tweet-netral*. Sementara itu, kategori positif dan negatif memiliki F1-score yang lebih rendah, yaitu 0.45 dan 0.54, menunjukkan bahwa model memiliki tantangan dalam mengklasifikasikan tweet positif dan negatif dengan presisi yang lebih rendah.

d. Evaluasi dengan *Confusion Matrix*, Evaluasi model menggunakan *Confusion Matrix* menunjukkan bahwa *Naïve Bayes* dapat mengenali sentimen dengan cukup baik, meskipun terdapat beberapa kekeliruan dalam klasifikasi sentimen positif dan negatif. Model cenderung mengklasifikasikan tweet sebagai netral lebih sering daripada sebagai positif atau negatif, yang dapat disebabkan oleh sifat dataset yang lebih banyak berisi opini yang tidak terlalu emosional.
e. Untuk penelitian selanjutnya, dapat dipertimbangkan penggunaan teknik pengolahan data yang lebih optimal, seperti metode *balancing* data agar distribusi sentimen lebih merata. Selain itu, eksplorasi algoritma lain seperti *SVM (Support Vector Machine)* atau *Random Forest* mungkin memberikan hasil yang lebih baik, khususnya dalam meningkatkan akurasi pada sentimen positif dan negatif. Memperluas sumber data dengan melibatkan platform lain juga dapat menjadi langkah yang menarik untuk mendapatkan gambaran opini masyarakat yang lebih komprehensif dan representatif.

UCAPAN TERIMA KASIH

Terima kasih kepada semua pihak yang telah berkontribusi dalam penyelesaian jurnal ini.

DAFTAR PUSTAKA

[1] A. M. A. Mooduto and U. N. Huda, "Urgensi Keberadaan Lembaga Pemantau Pemilihan Sebagai Pengawal Suara Kolom Kosong," *ADLIYA J. Huk. dan Kemanus.*, vol. 15, no. 1, pp. 19–36, 2021, doi: 10.15575/adliya.v15i1.9409.

[2] E. Y. Ekowati, "Pragmatisme Politik: Antara Koalisi, Pencalonan, dan Calon Tunggal Dalam Pilkada," *J. Transform.*, vol. 5, no. 1, pp. 16–37, 2019.

[3] Syarifuddin and S. Hasanah, "Analisis Dampak Penyelenggaraan Pilkada Serentak Tahun 2024," *J. Gov. Polit.*, vol. 4, no. 2, pp. 252–269, 2020.

[4] N. Sucahyo, I. Kurniati, and K. Harvit, "Analisis Sentimen Masyarakat Terhadap Uu Cipta Kerja Pada Media Sosial Twitter," *Jris J. Rekayasa Inf. Swadharma*, vol. 2, no. 1, pp. 63–70, 2022, doi: 10.56486/jris.vol2no1.167.

[5] E. Undamayanti *et al.*, "Analisis Sentimen Menggunakan Metode Naive Bayes Berbasis Particle Swarm Optimization Terhadap Pelaksanaan Program Merdeka Belajar Kampus Merdeka," *J. Sains Komput. Inform. (J-SAKTI*, vol. 6, no. 2, pp. 916–930, 2022.

[6] G. Darmawan, S. Alam, and M. I. Sulistyo, "Analisis Sentimen Berdasarkan Ulasan Pengguna Aplikasi Mypertamina Pada Google Playstore Menggunakan Metode Naïve Bayes," *STORAGE – J. Ilm. Tek. dan Ilmu Komput.*, vol. 2, no. 3, pp. 100–108, 2023.

[7] R. Rahmadani, A. Rahim, and R. Rudiman, "Analisis Sentimen Ulasan 'Ojol the Game' Di Google Play Store Menggunakan Algoritma Naive Bayes Dan Model Ekstraksi Fitur Tf-Idf Untuk Meningkatkan Kualitas Game," *J. Inform. dan Tek. Elektro Terap.*, vol. 12, no. 3, 2024, doi: 10.23960/jitet.v12i3.4988.

[8] M. Ilmar Rifaldi, Y. Raymond Ramadhan, and I. Jaelani, "Analisis Sentimen Terhadap Aplikasi Chatgpt Pada Twitter Menggunakan Algoritma Naïve Bayes," *J. Sains Komput. Inform. (J-SAKTI*, vol. 7, no. 2, pp. 802–814, 2023.

[9] cucu sutrisno, "Partisipasi Warga Negara Dalam Pilkada," *J. Pancasila dan Kewarganegaraan*, vol. 2, no. 2, pp. 36–48, 2017, doi: 10.24269/v2.n2.2017.36-48.

[10] L. B. Ilmawan and M. A. Mude, "Perbandingan Metode Klasifikasi Support Vector Machine dan Naïve Bayes untuk Analisis Sentimen pada Ulasan Tekstual di Google Play Store," *Ilk. J. Ilm.*, vol. 12, no. 2, pp. 154–161, 2020, doi: 10.33096/ilkom.v12i2.597.154-161.

[11] F. A. Larasati, D. E. Ratnawati, and B. T. Hanggara, "Analisis Sentimen Ulasan Aplikasi Dana dengan Metode Random Forest," *J. Pengemb. Teknol. Inf. dan Ilmu Komput.*, vol. 6, no. 9, pp. 4305–4313, 2022.

[12] T. Hidayat *et al.*, "Performance Prediction Using Cross Validation (GridSearchCV) for Stunting Prevalence," in *2024 IEEE International Conference on Artificial Intelligence and Mechatronics Systems (AIMS)*, 2024, pp. 1–6. doi: https://doi.org/10.1109/AIMS61812.2024.10512657.

[13] N. L. P. C. Savitri, R. A. Rahman, R. Venyutzky, and N. A. Rakhmawati, "Analisis Klasifikasi Sentimen Terhadap Sekolah Daring pada Twitter Menggunakan Supervised Machine Learning," *J. Tek. Inform. dan Sist. Inf.*, vol. 7, no. 1, pp. 47–58, 2021, doi:

10.28932/jutisi.v7i1.3216.
[14] T. N. Wijaya, R. Indriati, and M. N. Muzaki, "Analisis Sentimen Opini Publik Tentang Undang-Undang Cipta Kerja Pada Twitter," *Jambura J. Electr. Electron. Eng.*, vol. 3, no. 2, pp. 78–83, 2021, doi: 10.37905/jjeee.v3i2.10885.
[15] A. Anugrah, T. I. Hermanto, and I. Kaniawulan, "Sentiment Analysis of Internet Service Providers Using Naïve Bayes Based on Particle Swarm Optimization," *J. Ris. Inform.*, vol. 4, no. 4, pp. 371–378, 2022, doi: 10.34288/jri.v4i4.408.
[16] A. E. Augustia, R. Taufan, Y. Alkhalifi, and W. Gata, "Analisis Sentimen Omnibus Law Pada Twitter Dengan Algoritma Klasifikasi Berbasis Particle Swarm Optimization," *Paradig. - J. Komput. dan Inform.*, vol. 23, no. 2, 2021, doi: 10.31294/p.v23i2.10430.
[17] A. Nabillah *et al.*, "Twitter User Sentiment Analysis Of TIX ID Applications Using Support Vector Machine Algorithm," *RISTEC Res. Inf. Syst. Technol.*, vol. 3, no. 1, pp. 14–27, 2022, doi: 10.31980/ristec.v3i1.1898.
[18] D. A. Wulandari, R. Rohmat Saedudin, and R. Andreswari, "Analisis Sentimen Media Sosial Twitter Terhadap Reaksi Masyarakat Pada Ruu Cipta Kerja Menggunakan Metode Klasifikasi Algoritma Naive Bayes," *e-Proceeding Eng.*, vol. 8, no. 5, pp. 9007–9016, 2021.
[19] N. Helmiah *et al.*, "Penerapan Metode Naïve Bayes dalam Analisis Persepsi Masyarakat mengenai Rencana Pengesahan RUU Omnibus Law di Bidang Investasi dan Ketenagakerjaan Tahun 2020 di Indonesia," *J. MSA (Mat. dan Stat. serta Apl.)*, vol. 8, no. 2, p. 48, 2020, doi: 10.24252/msa.v8i2.16743.