

Comparative Study of SVM, KNN, and Naïve Bayes for Sentiment Analysis of Religious Application Reviews

Heti Aprilianti ^{1*}, Hery Mustofa^{2*}, Khothibul Umam ^{3*}, Maya Rini Handayani ^{4*}

* Teknologi Informasi, Fakultas Sains dan Teknologi, UIN Walisongo Semarang

2208096077_heti@student.walisongo.ac.id¹, herymustofa@walisongo.ac.id², khothibul_umam@walisongo.ac.id³, maya@walisongo.ac.id⁴

Article Info

Article history:

Received 2025-04-23

Revised 2025-05-07

Accepted 2025-05-09

Keyword:

*Sentiment Analysis,
Support Vector Machine (SVM),
NU Online,
User Reviews,
Classification Algorithms.*

ABSTRACT

This study aims to evaluate and compare the performance of three machine learning algorithms—Support Vector Machine (SVM), K-Nearest Neighbors (K-NN), and Naïve Bayes—for sentiment classification of user reviews on the NU Online application in the Google Play Store. NU Online is a religious digital platform providing Islamic content such as articles, prayers, and worship schedules. A total of 1,500 user reviews were collected using web scraping, and 1,491 were retained after data cleaning. Preprocessing steps included punctuation removal, case folding, normalization, stopword removal, stemming, and tokenization. Sentiment labels (positive or negative) were automatically assigned using a lexicon-based approach. The performance of the models was assessed using accuracy, precision, recall, and F1-score, calculated via confusion matrix with a training-testing data split. The results show that the SVM with a linear kernel achieved the best accuracy (81.6%), followed by Naïve Bayes (73.2%) and K-NN (66.9%). These findings indicate that SVM is the most effective algorithm in this context, providing practical contributions for developers of the NU Online digital religious platform and contributing to research in Indonesian natural language processing.



This is an open access article under the [CC-BY-SA](#) license.

I. PENDAHULUAN

Teknologi informasi yang berkembang secara pesat telah menunjukkan dampak secara besar untuk beberapa bidang kehidupan, mencakup untuk ranah keagamaan. Media digital kini menjadi sarana utama dalam penyebaran informasi, komunikasi, dan edukasi. Menurut [1] dalam Jurnal Manajemen Dakwah, pemanfaatan media sosial memungkinkan pesan-pesan dakwah disampaikan secara global dan efektif. Dengan demikian, lembaga keagamaan dapat menyampaikan ajaran agama secara lebih luas, cepat, dan adaptif terhadap kebutuhan masyarakat modern.

Salah satu wujud dari pemanfaatan teknologi dalam dakwah adalah hadirnya aplikasi keagamaan berbasis digital, seperti NU Online. Aplikasi ini memiliki banyak pengguna di Indonesia dan digunakan sebagai sarana untuk memperoleh informasi, akses berita, layanan tanya jawab, serta konten keagamaan lainnya. Pengguna sering memberikan ulasan

mengenai pengalaman mereka dalam menggunakan aplikasi ini, baik dari aspek fungsionalitas, kemudahan penggunaan, maupun manfaat yang diperoleh. Ulasan-ulasan tersebut menggambarkan persepsi dan kepuasan pengguna terhadap aplikasi, sehingga analisis terhadap data ini dapat memberikan wawasan penting bagi pengembang. Oleh karena itu, diperlukan pendekatan sistematis untuk mengklasifikasikan sentimen yang terkandung di dalamnya.

Sentiment analysis termasuk sistem analisis bahasa alami (*Natural Language Processing/NLP*) dengan memiliki peran untuk mendeteksi serta mengategorikan opini atau representasi perasaan dalam tulisan ke dalam klasifikasi seperti sentimen positif, sentimen negatif, maupun sentimen netral [2]. Metode ini telah banyak digunakan dalam berbagai sektor dan bidang penerapan yang beragam, termasuk dalam penilaian produk, pelayanan konsumen, serta dalam menganalisis konten pada platform media sosial. Beberapa metode populer yang diterapkan dalam pengolahan sentimen

meliputi *Support Vector Machine* (SVM), *K-Nearest Neighbors* (KNN), dan *Naïve Bayes* (NB), yang masing-masing memiliki karakteristik serta keunggulan tersendiri dalam pengolahan data teks.

Support Vector Machine (SVM) sebagai salah satu algoritma pemrosesan data pada teknik machine learning bertujuan menentukan *hyperplane* terbaik untuk mengklasifikasikan data berdasarkan kategori tertentu, terutama pada data berdimensi tinggi [3]. Algoritma ini juga dikenal memiliki performa yang tinggi dalam klasifikasi teks. Sementara itu, *K-Nearest Neighbors* (KNN) mengklasifikasikan data berdasarkan kedekatan jarak dengan sejumlah tetangga terdekatnya, dan cenderung memberikan hasil yang baik pada dataset dengan distribusi yang jelas [4]. Adapun *Naïve Bayes* merupakan metode klasifikasi berbasis probabilistik yang memprediksi kelas suatu data dengan mengandalkan informasi historis, dalam asumsi mengenai setiap fitur pada data tidak memiliki keterkaitan dengan data yang lain [5]. Metode ini dikenal karena efisiensinya dan kemampuannya dalam mengolah data teks berskala besar.

Beberapa penelitian sebelumnya telah melakukan perbandingan terhadap ketiga algoritma tersebut dalam konteks analisis sentimen. Sebagai contoh, studi [6] membandingkan performa SVM, *Naïve Bayes*, serta KNN dalam menganalisis ulasan gadget, algoritma SVM menunjukkan akurasi tertinggi mencapai 82,5%, disusul oleh *Naïve Bayes* 79,5% dan KNN 75%. Studi lain [7] membandingkan algoritma *Naïve Bayes* serta KNN pada analisis sentimen pengguna e-wallet aplikasi DANA, di mana *Naïve Bayes* memiliki keunggulan dari segi tingkat akurasi 79,41%, sementara KNN mencapai 75,63%.

Penelitian lainnya [8] melakukan studi terkait kajian sentimen terhadap kebijakan sistem ganjil-genap di ruas Tol Bekasi dilakukan dengan menerapkan metode SVM dan K-NN. Data yang dipergunakan dalam penelitian berasal dari platform digital seperti Instagram, YouTube, Twitter, dan Facebook. Penelitian ini memperoleh kesimpulan mengenai metode SVM mengindikasikan tingkat ketepatan lebih tinggi sebesar 78,18%, berbanding dengan K-NN yang hanya memperoleh 57,05%.

Penelitian [9] mengevaluasi kinerja algoritma klasifikasi menggunakan *Naïve Bayes* dan *Support Vector Machine* dalam analisis sentimen terhadap ulasan teks di Google Play. Berdasarkan hasil analisis, *Naïve Bayes* memperoleh akurasi sebesar 87,82%, sementara SVM memperoleh akurasi 90,01%.

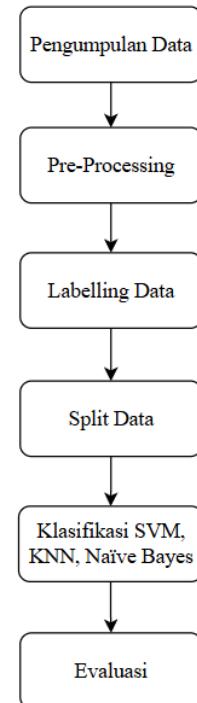
Hasil penelitian [10] menunjukkan bahwa model CRNN-SVM memiliki tingkat akurasi tertinggi dalam mengenali kecenderungan sentimen, yaitu 93,5%, dengan rata-rata akurasi sistem sebesar 91% dan stabilitas yang tinggi. Penelitian [11] menunjukkan bahwa sebagian besar respons publik terhadap vaksin Covid-19 bersifat positif (56,80%) dan model SVM dengan kernel RBF menghasilkan akurasi tertinggi sebesar 92%.

Penelitian lainnya [12] menunjukkan bahwa SVM lebih efektif untuk klasifikasi SARA negatif dan netral, dengan F1-

Score tertinggi masing-masing 0,72 dan 0,65, sedangkan LIWC lebih baik untuk klasifikasi SARA positif dengan F1-Score sebesar 0,69. Penelitian [13] tentang analisis sentimen pada Twitter menggunakan model CA-SVM, yang menghasilkan akurasi 92,48% dan skor sentimen 92,05%, lebih baik dibandingkan metode lain.

Berdasarkan hasil dari penelitian sebelumnya mengindikasikan tingkat efektifitas algoritma klasifikasi sangat dipengaruhi oleh karakteristik dari dataset yang digunakan. Penelitian ini berkontribusi pada pengembangan studi Islam digital dengan menerapkan pendekatan analisis sentimen berbasis machine learning terhadap ulasan pengguna aplikasi dakwah. Sebagai salah satu studi awal yang secara khusus menganalisis aplikasi NU Online, hasil penelitian ini diharapkan memberikan wawasan baru tentang persepsi pengguna terhadap platform dakwah digital dan membantu pengembang meningkatkan kualitas konten serta pengalaman pengguna.

II. METODOLOGI PENELITIAN



Gambar 1. Alur Penelitian

A. Pengumpulan Data

Untuk penelitian ini, data dikumpulkan dengan menggunakan pustaka *Google Play Scraper*. Teknik *data scraping* dilakukan dengan memanfaatkan bahasa pemrograman Python yang dijalankan melalui platform Google Collaboratory [14]. Sebanyak 1.500 ulasan yang relevan diambil dari Play Store, mencakup berbagai tingkat penilaian dari 1 hingga 5. Data yang diperoleh melalui proses *scraping* mencakup rating, ulasan, dan tanggal yang menunjukkan waktu pengguna memberikan review di

Google. Seluruh data hasil *scraping* disimpan dalam format CSV.

B. Pre-Processing

Setelah pengumpulan data, langkah selanjutnya adalah *pre-processing*, yang mencakup *removing duplicate* untuk penghapusan data duplikat; *data cleaning* untuk menghapus URL, emoji, angka, dan simbol; *case folding* bertujuan menyamakan setiap huruf dalam teks dengan mengkonversinya ke huruf kecil; normalisasi kata dalam mengubah kata tidak baku ke dalam baku; *tokenization* dalam membagi teks ke dalam unit kata atau token secara terpisah; *filtering* untuk menghapus istilah yang tidak relevan atau tidak bermakna; serta *stemming* digunakan untuk merubah kata ke dalam bentuk dasar. Setelah dilakukan *pre-processing*, diperoleh sebanyak 1.491 data yang sesuai dan dapat digunakan untuk penelitian.

C. Labelling Data

Label sentimen ditentukan secara otomatis menggunakan pendekatan *lexicon-based*. Kamus sentimen yang digunakan terdiri dari daftar kata berbahasa Indonesia yang telah diberi bobot polaritas (positif atau negatif). Tiap ulasan dianalisis dan diberi label berdasarkan kata yang paling dominan dalam teks. Hanya dua label yang digunakan dalam klasifikasi, yaitu positif dan negatif, sementara ulasan dengan polaritas netral atau ambigu dikeluarkan dari dataset.

D. Split Data

Pada tahap pembagian data (*split data*), dataset dibagi ke dalam perbandingan 80:20, dimana 80% pada pelatihan (training) dengan 20% pada pengujian (testing). Proses pengelompokan data ini digunakan untuk pelatihan model melalui penggunaan data *training* dan mengevaluasi performa menggunakan data *testing*. Pemilihan rasio 80:20 bertujuan untuk melihat performa masing-masing algoritma dalam mengklasifikasikan sentimen ulasan pengguna.

E. Klasifikasi

Klasifikasi merupakan tahap penting dalam analisis sentimen. Algoritma pembelajaran mesin diimplementasikan pada data yang sudah dipersiapkan melalui *pre-processing* dan pelabelan untuk mengklasifikasikan sentimen [15]. Dalam penelitian ini, melibatkan penerapan tiga algoritma: *Support Vector Machine* (SVM), *K-Nearest Neighbor* (K-NN), serta *Naïve Bayes*. SVM bekerja dalam mencari hyperplane optimal, K-NN mengklasifikasikan berdasarkan kedekatan data, dan *Naïve Bayes* menggunakan pendekatan probabilistik. Ketiga metode tersebut dievaluasi untuk menentukan algoritma yang paling tepat dalam mengklasifikasikan sentimen ulasan pengguna.

1) Support Vector Machine Classifier

Salah satu algoritma yang diterapkan dalam proses pengelompokan adalah *Support Vector Machine* (SVM). Metode ini bertujuan untuk menentukan fungsi pemisah (hyperplane) yang paling maksimal dalam mengklasifikasikan dua kategori data. SVM memiliki ketahanan terhadap overfitting, terutama di ruang berdimensi tinggi, ketika jumlah dimensi melebihi jumlah sampel [16]. SVM menerapkan prinsip *Structural Risk Minimization* (SRM) untuk mengurangi kemungkinan terjadinya kesalahan dalam proses klasifikasi. Awalnya dirancang sebagai *linear classifier*, SVM kemudian dikembangkan untuk menangani kasus non-linear dengan memanfaatkan fungsi kernel. Beberapa jenis kernel yang sering diterapkan antara lain: *linear*, *polynomial*, *sigmoid*, dan *RBF* (*radial basis function*) [17]. Rumus dasar dari hyperplane dalam SVM:

$$f(x) = w^T x + b \quad (1)$$

Keterangan :

w = Vektor bobot

x = Vektor fitur input

b = Bias

2) K-Nearest Neighbors

K-Nearest Neighbor (K-NN) menjadi teknik klasifikasi dalam menentukan kelas suatu data berdasarkan k tetangga terdekatnya. Penentuan tetangga terdekat dilakukan dengan menghitung jarak antar data menggunakan persamaan 2. Kelas dengan frekuensi tertinggi di antara tetangga terdekat dipilih sebagai hasil prediksi data uji [17].

$$d(p, q) = \sqrt{\sum_{i=1}^n (pi - qi)^2} \quad (2)$$

Keterangan:

pi = Data untuk pelatihan

qi = Data untuk pengujian

i = Variabel data

n = Dimensi data

3) Naïve Bayes Classifier

Naïve Bayes merupakan salah satu pendekatan dalam algoritma klasifikasi probabilistik yang bersifat sederhana. Metode ini beroperasi melalui menghitung probabilitas dari sekumpulan data berdasarkan frekuensi kemunculan dan kombinasi nilai dalam dataset. *Naïve Bayes* mengasumsikan bahwa setiap atribut dalam kategori tersebut beroperasi secara terpisah, tanpa ada hubungan saling ketergantungan antar atribut [18]. Berikut merupakan penerapan rumus *Naïve Bayes*:

$$P(B) = \frac{P(B|A) \times P(A)}{P(B)} \quad (3)$$

Keterangan:

- $P(A|B)$ = Probabilitas A terjadi, mengingat B terjadi
 $P(B|A)$ = Probabilitas B terjadi, mengingat A terjadi
 $P(A)$ = Probabilitas A terjadi tanpa mempertimbangkan B
 $P(B)$ = Probabilitas B terjadi tanpa mempertimbangkan A

4) Pengaturan Hyperparameter Model

Konfigurasi hyperparameter untuk masing-masing algoritma ditetapkan berdasarkan pertimbangan karakteristik data dan tujuan analisis. Untuk Support Vector Machine, digunakan kernel linear dengan parameter regulasi C sebesar 1,0 dan solver Liblinear, yang dipilih karena kemampuannya dalam menangani data teks yang memiliki banyak fitur. Pada K-Nearest Neighbor, jumlah tetangga (k) ditetapkan sebesar 5 dengan metrik jarak Euclidean dan pembobotan uniform, yang umum digunakan dalam analisis sentimen teks untuk mencapai keseimbangan antara kompleksitas model dan akurasi prediksi. Sementara itu, pada Naïve Bayes menggunakan MultinomialNB dengan Laplace smoothing $\alpha = 1,0$ untuk menghindari probabilitas nol pada istilah yang jarang muncul.

F. Evaluasi

Confusion matrix dimanfaatkan dalam proses evaluasi kinerja model klasifikasi, dengan metrik yang dihasilkan mencakup akurasi, presisi, recall, serta *F1-score*. Tabel ini memberikan representasi kuantitatif terhadap jumlah klasifikasi yang tepat berdasarkan data uji [19]. Berikut ini adalah formula yang digunakan untuk *accuracy*, *precision*, *recall*, dan *F1-score*:

$$\text{Accuracy} = \frac{TP+TN}{TP+TN+FP+FN} \times 100\% \quad (4)$$

$$\text{Precision} = \frac{TP}{TP+FP} \times 100\% \quad (5)$$

$$\text{Recall} = \frac{TP}{TP+FN} \times 100\% \quad (6)$$

$$\text{F1 Score} = \frac{2(Recall \times Precision)}{Recall+Precision} \times 100\% \quad (7)$$

Keterangan:

- TP = True Positive
 TN = True Negative
 FP = False Positive
 FN = False Negative

Accuracy digunakan dalam menghitung perbandingan prediksi secara tepat terhadap semua data. *Precision* merepresentasikan tingkatan ketepatan model ketika mengidentifikasi kelas positif, sedangkan *recall* menunjukkan kemampuan model dalam mendekripsi seluruh

data yang benar-benar termasuk dalam kelas positif. Adapun *F1-score* merupakan keseimbangan dari *precision* dan *recall*.

III. HASIL DAN PEMBAHASAN

A. Pengumpulan Data

Untuk tahapan pengumpulan data, didapatkan 1.491 data yang telah melalui proses *remove duplicate* dan penghapusan data kosong. Hasil scrapping data tersebut sebagaimana terlihat dalam gambar 2.

score	content	at
5	aplikasinya kerennnnnnn bgttt, lengkap juga mu...	2025-03-02 03:50:50
4	Saran: Perbanyak qari dan tafsir, serta tambah...	2025-03-16 09:14:58
5	Sekedar saran.. kalau bisa ditambahkan pilihan...	2025-03-17 12:49:10
5	aplikasi nya bagus cuma untuk audio pembacaan ...	2025-03-21 05:42:20
5	Di Versi 2.15.0 tulisan/ font di artikel berub...	2025-01-17 03:05:18

Gambar 2. Hasil Pengumpulan Data

B. Preprocessing Data

Preprocessing Data bertujuan untuk mengkonversi data mentah ke dalam format secara lebih tersusun juga terorganisir, sehingga data yang didapatkan dapat digunakan dalam proses analisis selanjutnya [20]. Adapun tahapan yang dilakukan dalam pra pemrosesan data meliputi data cleaning, case folding, normalisasi kata, tokenisasi, filtering, dan stemming. Seluruh proses tersebut ditampilkan dalam tabel berikut.

1) *Data cleaning* adalah tahap pengolahan data yang terdapat tujuan untuk menghapus komponen-komponen secara tidak relevan dalam teks, diantaranya angka, tanda baca, tag HTML, atau simbol lain, sehingga data menjadi lebih terstruktur dan konsisten.

TABEL I
DATA CLEANING

Sebelum Data Cleaning	Sesudah Data Cleaning
Aplikasi ini baik, bagus dan sangat membantu memudahkan saya sebagai warga Nahdliyyin yang belum paham tentang Islam ahlussunah wal jama'ah di Nusantara. Semua ada diaplikasi ini	Aplikasi ini baik bagus dan sangat membantu memudahkan saya sebagai warga Nahdliyyin yang belum paham tentang Islam ahlussunah wal jamaah di Nusantara. Semua ada diaplikasi ini

2) *Case folding* menjadi suatu tahapan dengan mengubah keseluruhan karakter ke dalam teks menjadi huruf kecil (*lowercase*) [21].

TABEL II
CASE FOLDING

Sebelum Case Folding	Sesudah Case Folding
Aplikasi ini baik, bagus dan sangat membantu memudahkan saya sebagai warga Nahdliyyin yang belum paham tentang Islam ahlussunah wal jama'ah di Nusantara Semua ada diaplikasi ini	aplikasi ini baik bagus dan sangat membantu memudahkan saya sebagai warga nahdliyyin yang belum paham tentang islam ahlussunah wal jamaah di nusantara semua ada diaplikasi ini

3) *Normalisasi* kata bertujuan untuk mengganti kata-kata tidak baku atau slang dengan bentuk standarnya untuk menyelaraskan variasi penulisan kata yang sama [22].

TABEL III
NORMALISASI KATA

Sebelum Normalisasi	Sesudah Normalisasi
Aplikasi ini baik, bagus dan sangat membantu memudahkan saya sebagai warga Nahdliyyin yang belum paham tentang Islam ahlussunah wal jama'ah di Nusantara Semua ada diaplikasi ini	aplikasi ini baik bagus dan sangat membantu memudahkan saya sebagai warga nahdliyyin yang belum paham tentang islam ahlussunah wal jamaah di nusantara semua ada diaplikasi ini

4) *Tokenization* merupakan proses memisahkan teks menjadi bagian-bagian kecil, seperti token atau kata [23].

TABEL IV
TOKENIZATION

Sebelum Tokenization	Sesudah Tokenization
Aplikasi ini baik, bagus dan sangat membantu memudahkan saya sebagai warga Nahdliyyin yang belum paham tentang Islam ahlussunah wal jama'ah di Nusantara Semua ada diaplikasi ini	['aplikasi', 'ini', 'baik', 'bagus', 'dan', 'sangat', 'membantu', 'memudahkan', 'saya', 'sebagai', 'warga', 'nahdliyyin', 'yang', 'belum', 'paham', 'tentang', 'islam', 'ahlussunah', 'wal', 'jamaah', 'di', 'nusantara', 'semua', 'ada', 'diaplikasi', 'ini']

5) *Filtering* menjadi tahapan penyaringan istilah yang tidak memiliki peran penting untuk analisis, diantaranya “dan”, “atau”, atau “yang”, untuk memastikan fokus analisis lebih terarah dalam kata-kata dengan lebih relevan atau signifikan.

TABEL V
FILTERING

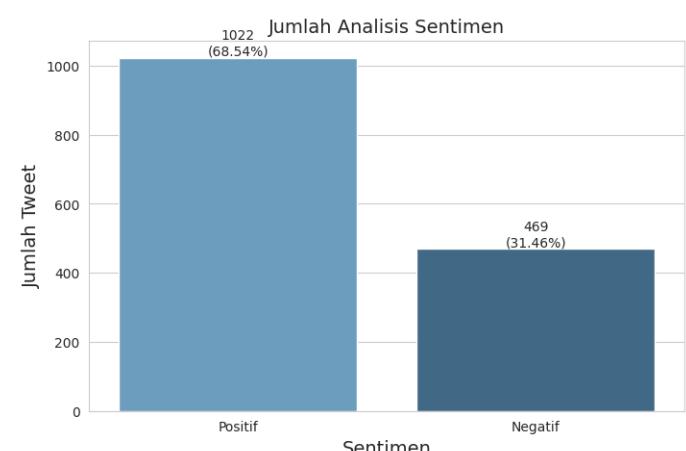
Sebelum Filtering	Sesudah Filtering
Aplikasi ini baik, bagus dan sangat membantu memudahkan saya sebagai warga Nahdliyyin yang belum paham tentang Islam ahlussunah wal jamaah di Nusantara Semua ada diaplikasi ini	['aplikasi', 'bagus', 'membantu', 'memudahkan', 'warga', 'nahdliyyin', 'paham', 'islam', 'ahlussunah', 'wal', 'jamaah', 'nusantara', 'diaplikasi']

6) *Stemming* adalah proses untuk menemukan bentuk dasar dari setiap kata, dengan cara menghapus imbuhan atau mengembalikannya ke akar katanya menggunakan aturan tertentu [20].

TABEL VI
STEMMING

Sebelum Stemming	Sesudah Stemming
Aplikasi ini baik, bagus dan sangat membantu memudahkan saya sebagai warga Nahdliyyin yang belum paham tentang Islam ahlussunah wal jamaah di Nusantara Semua ada diaplikasi ini	aplikasi bagus bantu mudah warga nahdliyyin paham islam ahlussunah wal jamaah nusantara aplikasi

7) *Labelling* merupakan proses klasifikasi untuk menganalisis sentimen pada setiap ulasan, yang dikategorikan ke dalam sentimen positif maupun negatif. Proses pelabelan ini menggunakan pendekatan *lexicon-based*, yaitu metode yang memanfaatkan daftar kosakata yang telah diklasifikasikan berdasarkan sentimen tertentu sebagai acuan untuk mengidentifikasi polaritas masing-masing ulasan.



Gambar 3. Hasil Labelling Lexicon Based

C. Pembagian Dataset

Dataset dibagi dalam dua bagian, 80% pada data *training* dengan 20% pada data *testing*. Hasil rasio pembagian ini ditampilkan pada tabel berikut.

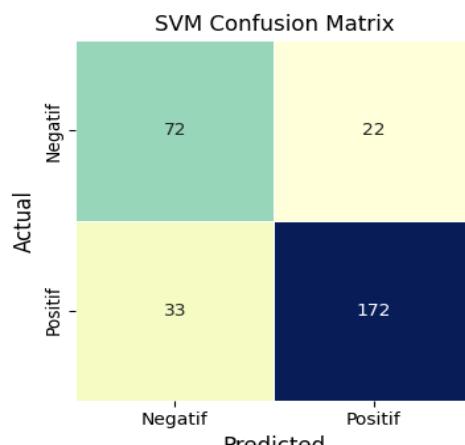
TABEL VII
PEMBAGIAN DATASET

Jenis Data	Presentase	Jumlah
Data Training	80%	1192
Data Testing	20%	299
Total	100%	1491

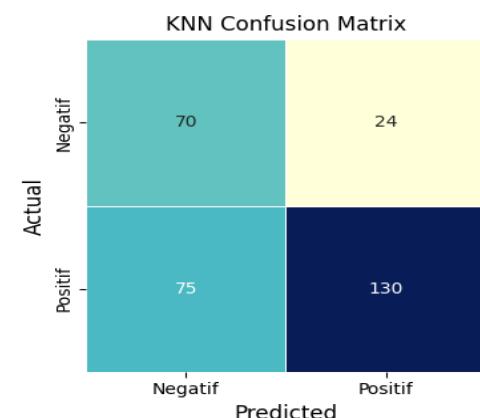
D. Evaluasi Model

Untuk mengevaluasi kinerja model klasifikasi diantaranya algoritma *Naïve Bayes*, *K-Nearest Neighbor* (K-NN), serta *Support Vector Machine* (SVM), digunakan *confusion matrix* yang menghasilkan metrik evaluasi berupa *accuracy*, *precision*, *recall*, serta *F1-score*. Evaluasi ini dilakukan guna menilai performa masing-masing model dan mengidentifikasi algoritma dengan hasil klasifikasi terbaik. Proses evaluasi dilakukan dengan pendekatan hold-out validation, di mana data dibagi secara acak menjadi dua subset, yaitu 80% untuk data pelatihan dan 20% untuk data pengujian.

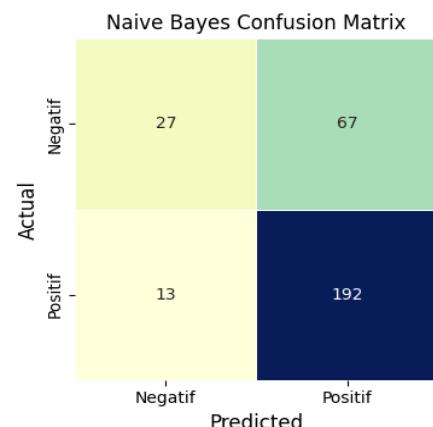
Visualisasi hasil klasifikasi ditampilkan pada Gambar 4, 5, dan 6 secara berurutan menampilkan *confusion matrix* dari model SVM, *K-Nearest Neighbor* (K-NN), serta *Naïve Bayes*. Pada masing-masing gambar, nilai *actual* menggambarkan label asli dari ulasan, sementara nilai *predicted* menunjukkan hasil prediksi yang diberikan oleh model. Melalui pengujian terhadap data dengan sentimen positif juga negatif, diperoleh empat kombinasi hasil klasifikasi seperti berikut.



Gambar 4. SVM Confusion Matrix



Gambar 5. K-NN Confusion Matrix



Gambar 6. Naïve Bayes Confusion Matrix

- 1) *True Positive* menunjukkan prediksi benar untuk kelas positif. Model SVM berhasil mengklasifikasikan 172 ulasan positif secara akurat, diikuti oleh K-NN sebanyak 130 ulasan, dan *Naïve Bayes* sebanyak 192 ulasan.
- 2) *False Positive* menunjukkan prediksi salah untuk kelas negatif yang diklasifikasikan sebagai positif. Model SVM menghasilkan 22 ulasan prediksi salah, K-NN sebanyak 24 ulasan, dan *Naïve Bayes* sebanyak 67 ulasan.
- 3) *False Negative* menunjukkan prediksi salah untuk kelas positif yang diklasifikasikan sebagai negatif. Model SVM mengklasifikasikan 33 ulasan positif sebagai negatif, K-NN sebanyak 75 ulasan, dan *Naïve Bayes* sebanyak 13 ulasan.
- 4) *True Negative* menunjukkan prediksi benar untuk kelas negatif, sebanyak 72 data ulasan berhasil dikenali secara tepat sebagai ulasan negatif oleh model SVM, diikuti oleh K-NN sebanyak 70 ulasan, dan *Naïve Bayes* sebanyak 27 ulasan.

Berdasarkan evaluasi menggunakan *confusion matrix*, kesalahan klasifikasi lebih sering terjadi pada model *Naïve Bayes* dan *K-Nearest Neighbor* dibandingkan dengan SVM. Model *Naïve Bayes* menghasilkan 67 kesalahan *false positive*, lebih banyak dibandingkan K-NN yang mencatatkan 24 kesalahan, dan SVM yang hanya menghasilkan 22 kesalahan. Hal ini mengindikasikan bahwa *Naïve Bayes* cenderung salah mengklasifikasikan ulasan negatif sebagai

positif, terutama ketika ulasan tersebut mengandung kata-kata dengan konotasi positif.

Di sisi lain, K-Nearest Neighbor menunjukkan tingkat kesalahan false negative yang lebih tinggi, dengan 75 ulasan positif yang diklasifikasikan sebagai negatif. Nilai ini melebihi jumlah kesalahan yang dihasilkan oleh SVM, yang mencatatkan 33 kesalahan, serta Naïve Bayes yang mencatatkan 13 kesalahan. Hal ini mengindikasikan bahwa K-NN kurang optimal dalam mengidentifikasi sentimen positif yang tidak disampaikan secara eksplisit atau tersirat dalam teks.

Dari evaluasi performa model dengan memanfaatkan *confusion matrix* untuk membandingkan dan menganalisis tiga algoritma klasifikasi, yaitu *Support Vector Machine* (SVM), *Naïve Bayes*, serta *K-Nearest Neighbor* (K-NN). Perbandingan kinerja masing-masing model dianalisis berdasarkan metrik evaluasi yang meliputi *accuracy*, *precision*, *recall*, dan *F1-score*.

TABEL VIII

PERBANDINGAN HASIL EVALUASI KINERJA MODEL KLASIFIKASI

Model Klasifikasi	SVM	KNN	Naïve Bayes
Accuracy	81,6 %	66,9 %	73,2 %
Precision	79,0 %	66,0 %	71,0 %
Recall	80,0 %	69,0 %	61,0 %
F1-Score	79,0 %	66,0 %	62,0 %

Evaluasi performa model klasifikasi dilakukan dengan mengacu pada metrik *accuracy*, *precision*, *recall*, dan *F1-score*. Hasil evaluasi mengindikasikan bahwa algoritma *Support Vector Machine* (SVM) terdapat kinerja paling baik dalam tingkat akurasi sejumlah 82,0%, *precision* 79,0%, *recall* 80,0%, dan *F1-score* 79,0%. Hasil tersebut mengindikasikan bahwa SVM mampu melakukan klasifikasi secara konsisten dan akurat.

Model *Naïve Bayes* menempati posisi kedua dengan *accuracy* 73,24%, *precision* 71,0%, *recall* 61,0%, dan *F1-score* 62,0%, yang menunjukkan bahwa meskipun prediksinya cukup tepat, model ini belum optimal dalam mendeteksi seluruh data yang relevan. Model *K-Nearest Neighbor* (K-NN) menunjukkan performa terendah dengan *accuracy* 66,89%, *precision* 66,0%, *recall* 69,0%, dan *F1-score* 66,0%.

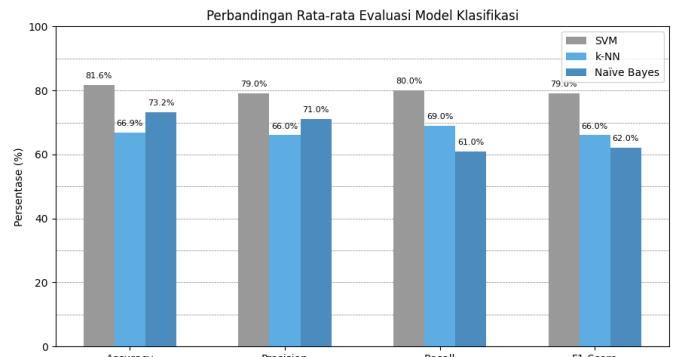
Berdasarkan hasil perbandingan tersebut, mampu diperoleh kesimpulan mengenai SVM merupakan algoritma klasifikasi yang paling unggul untuk analisis sentimen terhadap data yang digunakan sebagai objek analisis dalam penelitian ini.

Perbedaan performa antar algoritma dapat dijelaskan berdasarkan karakteristik masing-masing model dalam mengelola data teks. Algoritma *Support Vector Machine* (SVM) menunjukkan hasil terbaik karena kemampuannya dalam menangani data berdimensi tinggi melalui pembentukan hyperplane optimal yang memisahkan kelas dengan margin maksimum. Hal ini membuat SVM unggul dalam klasifikasi data teks yang secara alami memiliki banyak fitur dan sparsitas tinggi.

Sebaliknya, algoritma *K-Nearest Neighbor* (K-NN) menunjukkan performa yang paling rendah. Kinerja tersebut kemungkinan besar dipengaruhi oleh sensitivitas K-NN terhadap data high-dimensional dan sparse seperti representasi bag-of-words yang digunakan dalam penelitian ini. Pada ruang vektor berdimensi tinggi, metrik jarak seperti Euclidean menjadi kurang efektif karena distribusi data cenderung menyebar merata sehingga kedekatan antar titik data menjadi kurang bermakna. Akibatnya, proses penentuan tetangga terdekat menjadi tidak akurat.

Sementara itu, algoritma *Naïve Bayes* menghasilkan performa sedang. Pendekatan probabilistik yang digunakan oleh *MultinomialNB* relatif efisien dalam memproses data teks, terutama karena mempertimbangkan frekuensi kata. Namun, asumsi independensi antar fitur yang menjadi dasar metode ini sering kali tidak sepenuhnya terpenuhi dalam konteks nyata, sehingga berdampak pada penurunan akurasi dibandingkan SVM.

Visualisasi hasil evaluasi kinerja ketiga model secara keseluruhan terlihat pada Gambar 7.



Gambar 7. Evaluasi Nilai Rata-rata Kinerja Model Klasifikasi

IV. KESIMPULAN

Penelitian ini membandingkan performa algoritma SVM, *Naïve Bayes*, dan K-NN dalam klasifikasi sentimen ulasan pengguna terhadap aplikasi NU Online. Dari 1.491 ulasan yang dianalisis, SVM dengan kernel linear menunjukkan performa terbaik memperoleh akurasi 81,6%, Diikuti oleh *Naïve Bayes* (*MultinomialNB*) dengan akurasi 73,2% dan K-NN ($k=5$) yang mencatatkan akurasi terendah sebesar 66,9%. SVM terbukti lebih unggul dalam mengklasifikasikan sentimen dengan akurat karena kemampuannya membentuk batas keputusan yang optimal di ruang fitur berdimensi tinggi, yang umum ditemukan pada data teks. Sebaliknya, *Naïve Bayes* dan K-NN menunjukkan keterbatasan dalam mengenali konteks semantik dan ekspresi sentimen yang tersirat, sehingga lebih rentan terhadap kesalahan klasifikasi, khususnya pada ulasan yang ambigu atau tidak secara eksplisit menyatakan sentimen.

Dengan demikian, SVM dapat dipertimbangkan sebagai algoritma yang efektif dalam menganalisis sentimen untuk konteks aplikasi NU Online. Penelitian ini memberikan kontribusi praktis bagi pengembangan aplikasi NU Online

dalam mengevaluasi persepsi pengguna melalui pendekatan analisis sentimen berbasis machine learning, serta kontribusi ilmiah terhadap kajian pemrosesan bahasa alami (NLP) dalam bahasa Indonesia. Untuk memperoleh generalisasi hasil yang lebih kuat, penelitian selanjutnya direkomendasikan untuk memperluas cakupan data dan menerapkan teknik validasi yang lebih komprehensif, seperti k-fold cross-validation.

DAFTAR PUSTAKA

- [1] H. Fajrussalam, I. Dwiyanti, N. F. Salsabila, R. Aprillionita, and S. Auliakhasanah, "Pemanfaatan Media Sosial sebagai Media Dakwah Islam dalam Kemajuan Perkembangan Teknologi," *ASSABIQUN*, vol. 4, no. 1, pp. 102–114, Mar. 2022, doi: 10.36088/assabiqun.v4i1.1686.
- [2] B. Liu, "Sentiment Analysis and Opinion Mining," *Synth. Lect. Hum. Lang. Technol.*, vol. 5, no. 1, pp. 1–167, May 2012, doi: 10.2200/S00416ED1V01Y201204HTL016.
- [3] C. Cortes and V. Vapnik, "Support-vector networks," *Mach. Learn.*, vol. 20, no. 3, pp. 273–297, Sep. 1995, doi: 10.1007/BF00994018.
- [4] N. S. Altman, "An Introduction to Kernel and Nearest-Neighbor Nonparametric Regression," *Am. Stat.*, vol. 46, no. 3, pp. 175–185, Aug. 1992, doi: 10.1080/00031305.1992.10475879.
- [5] N. Azizah, R. Goejantoro, and Sifriyani, "Metode Naive Bayes Dengan Pendekatan Distribusi Gauss Untuk Klasifikasi Peminatan Peserta Didik," *Pros. Semin. Nas. Mat. dan Stat.*, vol. 1, pp. 1–7, 2019, [Online]. Available: <https://jurnal.fmipa.unmul.ac.id/index.php/SNMSA/article/view/520/217>
- [6] J. W. Iskandar and Y. Nataliani, "Perbandingan Naïve Bayes, SVM, dan k-NN untuk Analisis Sentimen Gadget Berbasis Aspek," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 5, no. 6, pp. 1120–1126, Dec. 2021, doi: 10.29207/resti.v5i6.3588.
- [7] M. R. Elfansyah, Rudiman, and F. Yulianto, "Perbandingan Metode K-Nearest Neighbor (Knn) Dan Naïve Bayes Terhadap Analisis Sentimen Pada Pengguna E-Wallet Aplikasi Dana Menggunakan Fitur Eksstraksi Tf-Idf," *J. Teknol. Inf. J. Keilmuan dan Apl. Bid. Tek. Inform.*, vol. 8, no. 2, pp. 139–159, 2024.
- [8] H. S. Utama, D. Rosiyadi, B. S. Prakoso, and D. Ariadarma, "Analisis Sentimen Sistem Ganjil Genap di Tol Bekasi Menggunakan Algoritma Support Vector Machine," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 3, no. 2, pp. 243–250, Aug. 2019, doi: 10.29207/resti.v3i2.1050.
- [9] 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, Aug. 2020, doi: 10.33096/ilkom.v12i2.597.154-161.
- [10] Y. Zhao, M. Mamat, A. Aysa, and K. Ubul, "Multimodal sentiment system and method based on CRNN-SVM," *Neural Comput. Appl.*, vol. 35, no. 35, pp. 24713–24725, Dec. 2023, doi: 10.1007/s00521-023-08366-7.
- [11] M. Rahardi, A. Aminuddin, F. F. Abdulloh, and R. A. Nugroho,
- [12] "Sentiment Analysis of Covid-19 Vaccination using Support Vector Machine in Indonesia," *Int. J. Adv. Comput. Sci. Appl.*, vol. 13, no. 6, 2022, doi: 10.14569/IJACSA.2022.0130665.
- [13] A. E. Karyawati, P. A. Utomo, and I. G. A. Wibawa, "Comparison of SVM and LIWC for Sentiment Analysis of SARA," *IJCCS (Indonesian J. Comput. Cybern. Syst.)*, vol. 16, no. 1, p. 45, Jan. 2022, doi: 10.22146/ijccs.69617.
- [14] C. P. D. Cyril, J. R. Beulah, N. Subramani, P. Mohan, A. Harshavardhan, and D. Sivabalaselvamani, "An automated learning model for sentiment analysis and data classification of Twitter data using balanced CA-SVM," *Concurr. Eng.*, vol. 29, no. 4, pp. 386–395, Dec. 2021, doi: 10.1177/1063293X211031485.
- [15] R. A. A. Renal, Syariful Alam, and Moch Hafid T, "Komparasi Payment Digital Untuk Analisis Sentimen Berdasarkan Ulasan Di Google Playstore Menggunakan Metode Support Vector Machine," *STORAGE J. Ilm. Tek. dan Ilmu Komput.*, vol. 2, no. 3, pp. 118–128, Aug. 2023, doi: 10.55123/storage.v2i3.2337.
- [16] M. Apriliyani, M. I. Musyaffaq, S. Nur'Aini, M. R. Handayani, and K. Umam, "Implementasi analisis sentimen pada ulasan aplikasi Duolingo di Google Playstore menggunakan algoritma Naïve Bayes," *AITI*, vol. 21, no. 2, pp. 298–311, Sep. 2024, doi: 10.24246/aiti.v21i2.298-311.
- [17] S. S.K.Rastogi, R. Singhal, and A. Kumar, "An Improved Sentiment Classification using Lexicon into SVM," *Int. J. Comput. Appl.*, vol. 95, no. 1, pp. 37–42, Jun. 2014, doi: 10.5120/16562-6226.
- [18] P. D. Silitonga and R. Damanik, "Perbandingan Algoritma k-Nearest Neighbors (k-NN) dan Support Vector Machines (SVM) untuk Klasifikasi Pengenalan Citra Wajah," *J. ICT Inf. Commun. Technol.*, vol. 20, no. 1, pp. 186–191, Jul. 2021, doi: 10.36054/jictikmi.v20i1.354.
- [19] R. N. Devita, H. W. Herwanto, and A. P. Wibawa, "Perbandingan Kinerja Metode Naïve Bayes dan K-Nearest Neighbor untuk Klasifikasi Artikel Berbahasa indonesia," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 5, no. 4, pp. 427–434, Oct. 2018, doi: 10.25126/jtiik.201854773.
- [20] A. Desiani *et al.*, "Penerapan Metode Support Vector Machine Dalam Klasifikasi Bunga Iris," *Indones. J. Appl. Informatics*, vol. 7, no. 1, p. 12, Apr. 2023, doi: 10.20961/ijai.v7i1.61486.
- [21] L. A. Susanto, "Komparasi Model Support Vector Machine Dan K-Nearest Neighbor Pada Analisis Sentimen Aplikasi Polri Super App," *J. Inform. dan Tek. Elektro Terap.*, vol. 12, no. 2, Apr. 2024, doi: 10.23960/jitet.v12i2.4152.
- [22] E. Fitri, "Analisis Sentimen Terhadap Aplikasi Ruangguru Menggunakan Algoritma Naïve Bayes, Random Forest Dan Support Vector Machine," *J. Transform.*, vol. 18, no. 1, pp. 71–80, Jul. 2020, doi: 10.26623/transformatica.v18i1.2317.
- [23] A. M. Ndapamuri, D. Manongga, and A. Iriani, "Analisis Sentimen Ulasan Aplikasi Tripadvisor Dengan Metode Support Vector Machine, K-Nearest Neighbor, Dan Naïve Bayes," *INOVTEK Polbeng - Seri Inform.*, vol. 8, no. 1, p. 127, Jun. 2023, doi: 10.35314/isi.v8i1.3260.
- [24] M. I. Fikri, T. S. Sabrina, and Y. Azhar, "Perbandingan Metode Naïve Bayes dan Support Vector Machine pada Analisis Sentimen Twitter," *SMATIKA J.*, vol. 10, no. 02, pp. 71–76, Dec. 2020, doi: 10.32664/smatika.v10i02.455.