JURNAL SESSION (*Software Development, Digital Business Intelligence, and Computer Engineering*)
Volume 04, Nomor 01, September 2025, Page 08-15
ISSN: 2963-9026 (Online). https://jurnal.poliwangi.ac.id/index.php/session
DOI: 10.57203/session.v4i01.2025.08-15

Komparasi Algoritma Klasifikasi untuk Analisis Sentimen Kinerja Dosen

Comparison of Classification Algorithms for Sentiment Analysis of Lecturer Performance

Hery Mustofa
Teknologi Informasi, Universitas Islam Negeri Walisongo Semarang, Semarang, Indonesia [1]
Email: herymustofa@walisongo.ac.id [1]

Abstract

In the digital era, text data originating from students' online reviews, comments and feedback can be a valuable source of information for understanding positive and negative perceptions of lecturer performance. This research aims to compare classification algorithms for sentiment analysis of student opinions regarding the performance of lecturers in higher education. In this research, a comparison of three classification algorithms was carried out, namely Naïve Bayes, Support Vector Machine, and Random Forest. The datasets used are student reviews of lecturer performance from various courses and lecturers. The dataset consists of 1254 students' critical comments and suggestions. The data consists of 839 positive comments and 415 negative comments. Next, classification is carried out using the Naïve Bayes algorithm, Support Vector Machine, and Random Forest. From the classification results, it is known that the Support Vector Machine algorithm gives the best results, followed by the Random Forest algorithm and finally the Naïve Bayes algorithm. It is known that the Support Vector Machine algorithm succeeded in getting an accuracy value of 82.24%, with a precision value of 84.66%, a recall value of 65.99% and an F1 score value of 79.81%.

Keywords: *clasification, sentiment analysis, lecturer performance, support vector machine, random forest, naive bayes;*

Abstrak

Dalam era digital, data teks yang berasal dari ulasan, komentar, dan *feedback online* mahasiswa dapat menjadi sumber informasi berharga untuk memahami persepsi positif dan negatif terhadap kinerja dosen. Penelitian ini bertujuan untuk melakukan komparasi algoritma klasifikasi untuk analisis sentimen opini atau persepsi mahasiswa terhadap kinerja dosen di pendidikan tinggi. Dalam penelitian ini, dilakukan perbandingan tiga algoritma klasifikasi yaitu *Naïve Bayes, Support Vector Machine*, dan *Random Forest.* Datasets yang digunakan berupa ulasan mahasiswa terhadap kinerja dosen dari berbagai mata kuliah dan dosen. *Datasets* terdiri dari 1254 data komentar kritik saran mahasiswa. Data tersebut terdiri dari 839 komentar positif dan 415 komentar negatif. Selanjutnya dilakukan klasifikasi menggunakan algoritma *Naïve Bayes, Support Vector Machine, serta Random Forest.* Dari hasil klasifikasi diketahui bahwa algoritma *Support Vector Machine* memberikan hasil yang paling baik kemudian disusul dengan algoritma *Random Forest* dan yang terakhir algoritma *Naïve Bayes.* Diketahui Algoritma *Support Vector Machine* berhasil mendapatkan nilai *accuracy* sebesar 82,24%, dengan nilai *precission* sebesar 84,66%, nilai *recall* sebesar 65,99% dan nilai *F1 score* sebesar 79,81%.

Kata Kunci: klasifikasi, analisis sentimen, kinerja dosen, support vector machine, random forest, naive bayes

I. PENDAHULUAN

Pendidikan tinggi memiliki peran penting dalam membentuk masa depan masyarakat dan negara. Dalam lingkungan pendidikan tinggi, peran dosen menjadi elemen kunci dalam memberikan pengetahuan, membimbing mahasiswa, dan memastikan kualitas pendidikan yang optimal [1]. Evaluasi kinerja dosen merupakan komponen integral dalam upaya untuk meningkatkan mutu pendidikan tinggi. Dalam era digital modern, mahasiswa semakin aktif dalam menyuarakan pandangan mereka terhadap pengalaman belajar mereka, termasuk kualitas pengajaran yang mereka terima [2].

Di dalam dunia pendidikan, proses evaluasi pengajaran ialah salah satu program atau kegiatannya adalah analisis sentimen [3]. Analisis sentimen adalah pendekatan yang efektif dalam memahami bagaimana mahasiswa menilai kinerja dosen mereka. Teknik analisis sentimen melibatkan penggalian informasi relevan dari kumpulan data Ulasan Pengguna Tidak Terstruktur (UUR) yang diambil dari *online* dan mengklasifikasikannya ke dalam komentar positif dan negatif yang sesuai untuk mengambil keputusan

Submitted: 05/06/2025; Accepted: 17/09/2025; Published: 30/09/2025

[4]. Ulasan, komentar, dan *feedback* yang dikumpulkan dari berbagai platform *online* mengandung beragam sentimen, sentimen tersebut bisa berupa sentimen positif maupun sentimen negatif, yang dapat memberikan wawasan berharga tentang pengalaman belajar mahasiswa. Penggunaan algoritma klasifikasi dalam analisis sentimen menjadi semakin penting dalam mengotomatisasi dan memudahkan proses ini.

Analisis sentimen merupakan bagian yang sangat penting dalam memaknai dan menganalisis opini publik, umpan balik, dan data yang berasal dari media sosial [5]. Terdapat berbagai macam algoritma *machine learning* yang dapat digunakan untuk analisis sentimen diantaranya yang paling sering banyak dipakai adalah *Bayesian, Support Vector Machine (SVM), Decission Tree*, dan *ensemble learning.* [6], [7]

Dalam penelitian Sanjay, melakukan perbandingan antar metode klasifikasi yaitu *algoritma Support Vector Machine (SVM)* dengan algoritma *Naïve Bayes*. Metode tersebut digunakan untuk melakukan klasifikasi analisis sentimen ulasan produk amazon. Diketahui bahwa metode SVM memberikan hasil F1-score lebih baik dari pada algoritma *Baïve Nayes.* SVM menghasilkan *F1-score* 0.83 sedangkan algoritma *Naïve Bayes* mempunya nilai sebesar 0.82 [8].

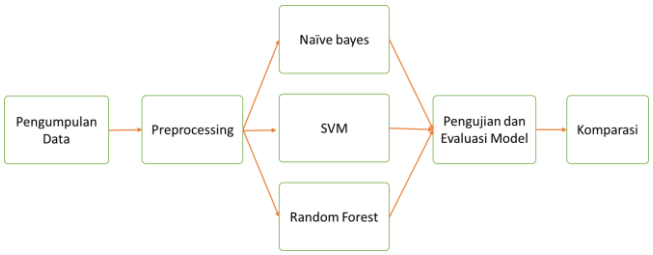
Analisis sentimen juga digunakan untuk melakukan analisis opini masyarakat Indonesia di *twitter* terhadap vaksin Covid-19. Data yang diambil dari kicauan twitter masyarakat Indonesia yang diambil dari *Drone Emprit Academic Streaming*. *Datasets* tersebut kemudian dilakukan klasifikasi dengan menggunakan algortima Naïve Bayes. Dari hasil analisis didapat *datasets* 6000 data tweet dengan 56% berlabel *tweet* negatif, dan 39% berlabel *tweet* positif, dan 1% berlabel *tweet* netral selama rentang waktu periode satu minggu [9].

Metode *Random Forest* bisa digunakan untuk mengklasifikasi sentimen analisis. Di dalam penelitian Fitri, *Random Forest* dipakai untuk melakukan klasifikasi analisis sentimen dengan *datasets* yang dipakai mengambil *datasets* media sosial. Penelitian tersebut berhasil melakukan klasifikasi sentimen media sosial kedalam kategori positif, negatif, dan netral. Dari hasil penelitian didapatkan hasil akurasi sebesar 82.91% [10].

Dari uraian di atas, dalam penelitian ini menganalisis opini kritik dan saran mahasiswa terhadap kinerja dosen, data tersebut kemudian diolah dan diklasifikasikan dengan pendekatan teks mining. Penelitian ini bertujuan untuk melakukan komparasi antara berbagai algoritma klasifikasi yang banyak dipakai untuk analisis sentimen terkait persepsi mahasiswa terhadap kinerja mengajar dosen. Algoritma klasifikasi yang digunakan untuk klasifikasi dalam penelitian adalah algoritma *Naïve Bayes,* algoritma *Support Vector Machine* (SVM), dan algoritma *Random Forest*. Penelitian ini mempunyai kontribusi menganalisis dan melakukan komparasi data kritik saran mahasiswa terhadap dosen dengan membandingkan tiga buah metode klasifikasi sehingga diketahui algoritma terbaik untuk melakukan klasifikasi sentimen opini mahasiswa terhadap kinerja mengajar dosen. Adapun hasil dari penelitian mempunyai manfaat bagi mahasiswa, dosen maupun pejabat kampus dalam melihat banyaknya sentimen positif dan negatif, hal tersebut berguna untuk pengambilan kebijakan ke depan yang lebih tepat dan akurat.

II. METODE PENELITIAN

Diagram alur dalam penelitian dapat di ilustrasikan pada gambar 1.1.



Gambar 1.1 Alur Penelitian

2.1. Data Collection

Dataset yang digunakan dalam penelitian ini bersifat privat dan dikumpulkan secara internal. Proses pengumpulan data dilakukan melalui penyebaran kuesioner daring kepada mahasiswa selama satu semester. Total data yang berhasil terkumpul adalah 1254 ulasan. Seluruh responden telah memberikan persetujuan untuk penggunaan data mereka secara anonim untuk tujuan penelitian ini. Untuk menjaga kerahasiaan dan etika, semua informasi pribadi yang dapat mengidentifikasi responden telah dihilangkan dari *dataset*. Data tersebut mencakup teks ulasan dan dapat sudah dilabeli dengan di anotasikan 2 sentimen, yaitu sentimen positif dan sentimen negatif. [11].

2.2 Preprocessing Data

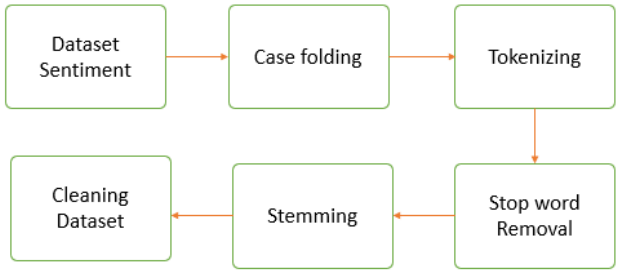
Preprocessing data merupakan proses yang sangat penting dalam melakukan analisis sentimen. Sebuah teks dapat berupa kalimat, kata, atau sebuah karakter, urutan karakter yang mempunyai makna. [12] *Datasets* yang dikumpulkan harus melalui tahap *preprocessing* untuk menghasilkan hasil yang optimal. Dalam penelitian ini, tahap *preprocessing* di lakukan seperti pada gambar 1.2. Pembersihan data yaitu menghapus karakter khusus, tanda baca, dan elemen yang tidak relevan.

Tokenization yaitu memisahkan teks menjadi token atau kata-kata individu [13], [14], [15]. Penghapusan kata-kata berhenti (*stop words*) yaitu menghapus berbagai kata yang umum dan tidak memberikan kontribusi signifikan pada analisis sentimen. *Stop word* umum digunakan dalam bahasa Inggris dan variasi huruf besar/kecil juga telah dihapus dari teks sebelum proses tokenisasi untuk mengurangi redundansi kata. [16]. Dalam penelitian ini akan menggunakan stopword khusus untuk Bahasa Indonesia yang diambil dari *library* NLTK.

Stemming yaitu mengubah kata-kata menjadi bentuk kata

dasar (*root word*) untuk mengurangi variasi kata yang mirip sehingga merurangi redudansi kata. *Stemming* adalah proses mengolah perubahan kata yang mempunyai imbuhan diubah menjadi kata dasar (*root word*). [17] Dalam bahasa Indonesia, *stemming* menjadi sangat penting. Beberapa kata dalam Bahasa Indonesia biasanya mempunyai prefiks, sufiks, infiks, dan konfiks. [18] Hal tersebut membuat pencarian kata dasar menjadi lebih rumit. *Stemming* Bahasa Indonesia, mempunyai perbedaan dengan *stemming* Bahasa Inggris atau Bahasa asing lainya. Dalam bahasa Indonesia, *stemming* dapat menggunakan metode *Stemming Berbasis Confix-stripping (CS)*. [19] Setelah pra-pemrosesan data, dilakukan extraksi fitur-fitur dari teks ulasan yang digunakan dalam proses klasifikasi sentimen. Fitur-fitur ini dapat mencakup representasi vektor kata (*word embeddings*).

Setelah dilakukan *preprocessing* maka akan dilakukan visualisasi data dengan *wordcloud. Wordcloud* adalah gambaran visual frekuensi kata yang tersusun dalam visualisasi teks. [20], [21] *Wordcloud* juga disebut sebagai metode visualisasi yang menggabungkan *trend chart*. *Wordcloud* dapat digunakan untuk melakukan ilustrasi evolusi konten temporal dalam sekumpulan dokumen. [22] Dengan *wordcloud* kata yang paling sering muncul akan memiliki bobot lebih, kemudian akan digambarkan dengan *font size* yang besar, sehingga lebih terlihat. Terdapat tiga algoritma yang dijadikan visualisasi *wordcloud* adalah *frequency, category*, dan *mixed*. [23]



Gambar 1.2 Proses *Preprocessing*

2.3 Algoritma *Naive Bayes*

Naïve Bayes merupakan model algoritma klasifikasi menggunakan penerapan teorema *Bayes. Naive bayes* merupakan algoritma yang banyak digunakan untuk analisis sentimen teks. [10] *Naïve Bayes* menggunakan asumsi atau anggapan yang disederhanak terhadap suatu komponen nilai atribut yang mempunyai kondisional tidak terikat atau bebas, jika diberikan suatu nilai *output.* Dalam naive bayes harus memperhatikan nilai *priori.* Nilai *priori* adalah sebuah nilai keluaran yang mempunyai probabilitas atau kemungkinan secara berbarengan pada suatu produk dari hipotesis sebuah sampel. Sedangkan *eveidence* atau bukti dari kemungkinan data *training* dinotasikan dengan notasi *P(X).* Kemudian nilai *P(H|X)* merupakan nilai kemungkinan atau probabilitas *H* yang mempengaruhi *posterior density* yang dapat dinotasikan dengan notasi X. Sehingga nilai *P(X|H)* merupakan nilai probabilitas nilai X terhadap H, nilai tersebut disebut sebagai *likelihood.* Teorema bayes dapat disebut sebagai proses klasifikasi yang terdiri dari sejumlah petunjuk untuk menetapkan *class* yang paling sesuai dengan *sample* yang di analisis. Sehingga teorema Baye dapat dinotasikan seperti persamaan 1.

$$P(C|f_1 \ldots f_n) = \frac{P(C)P(f_1 \ldots f_n|C)}{P(f_1 \ldots fn)} \qquad (1)$$

Variabel *C* merupakan representasi sebuah kelas, sedangkan variabel *F1 ..... Fn* menunjukkan representasi dari sebuah klasifikasi yang mempunyai karakteristik. Persamaan tersebut di atas merepresentasikan probabilitas *input sample* pada karakteristik dari kelas C atau yang disebut sebagai *posterior.* Probabilitas kelas C disebut sebagai *posterior*, di mana *posterio*r adalah *prior* dikalikan dengan likehood kemudian di bagi dengan *evident.* Hal tersebut dapat dinotasikan dalam persamaan 2 [24]

$$Posterior = \frac{Prior\ x\ likehold}{evident} \qquad (2)$$

Dari persamaan tersebut di atas dapat disimpulkan bahwa asumsi *naive independence* tersebut memunculkan terjadinya prasyarat peluang menjadi lebih sederhana, sehingga perhitungan dapat dilakukan menjadi sederhan. Kemudian, langkah selanjutnya adalah penjabaran $P(C|F1 \ldots Fn)$ yang dapat dinotasikan pada persamaan 4 dan persamaan 5.

$$P(C|f_1 \ldots f_n) = P(C)P(F1|C)P(F2|C) \qquad (4)$$
$$P(C|f_1 \ldots f_n) = P(C)\prod_{i=1}^{n} P(Fi|C) \qquad (5)$$

Persamaan 4 dan 5 tersebut merupakan sebuah model dari teorema *Naive Bayes* yang digunakan sebagai proses klasifikasi. Sedangkan, untuk klasifikasi dengan data kontinu digunakan persamaan rumus *Densitas Gauss* yang dapat dinotasikan sesuai persamaan 6.

$$P(X_i = x_i|\, Y = \, y_i) = \frac{1}{\sqrt{2\pi\vartheta ij}}\, e^{\frac{(x_i-\mu_i)^2}{2\sigma^2 ij}} \qquad (6)$$

2.4 Support Vector Machine (SVM)

SVM adalah suatu metode dalam *machine learning* yang dapat digunakan untuk melakukan klasifikasi dan regresi. SVM bekerja dengan cara mencari *hyperplane* terbaik yang dapat memisahkan dua kelas dalam ruang fitur data dengan margin maksimal, di mana margin adalah jarak antara *hyperplane* dan titik terdekat dari masing-masing kelas. [25] Tujuan utama SVM adalah menemukan *hyperplan*e yang memaksimalkan *margin* dengan cara memastikan bahwa semua titik data dari kelas yang berbeda berada di sisi yang benar dari *hyperplane*. SVM juga dapat mengatasi masalah ketidaklinieran dengan menggunakan fungsi kernel, yang memetakan data ke dalam dimensi yang lebih tinggi untuk

memungkinkan pemisahan yang lebih baik. SVM dikenal karena keandalan dan kinerjanya yang baik dalam mengatasi masalah klasifikasi pada data yang kompleks dan nonlinier. [26] SVM dikenal dengan memiliki konsep sentral dalam melakukan klasifikasi dengan cara mencari *hyperlane* terbaik dalam memilah dua kelas. [24]

2.5 Random Forest

Algoritma *Random Forest* diusulkan secara resmi pada tahun 2001 oleh Leo Breiman dan Adèle Cutler, merupakan bagian dari teknik pembelajaran otomatis. Algoritma ini menggabungkan konsep subruang acak dan *bagging.* Algoritma *Random Forest* melatih beberapa pohon keputusan yang digerakkan pada subkumpulan data yang sedikit berbeda. [27]. Algoritma Random Forest merupakan pengembangan dari algortima *Classification and Regression Trees* (CART). [28]

Random Forest merupakan algoritma pembelajaran *ensemble* dalam bidang *machine learning* yang memanfaatkan konsep keputusan berbasis pohon. Algoritma ini bekerja dengan cara membangun sejumlah besar pohon keputusan yang acak selama pelatihan, dan kemudian menggabungkan hasil prediksi dari setiap pohon untuk menghasilkan *output akhir.* Setiap pohon dihasilkan dengan menggunakan subset acak dari data pelatihan dan subset acak dari fitur, sehingga mencegah *overfitting* dan meningkatkan generalisasi model. *Random Forest* juga dapat memberikan estimasi kepentingan fitur, memungkinkan analisis yang lebih baik terhadap kontribusi masing-masing fitur terhadap prediksi keseluruhan. Kombinasi keputusan dari pohon-pohon tersebut membuat Random Forest tangguh, toleran terhadap noise, dan efektif dalam menangani data berdimensi tinggi. [24]

2.6 Pelatihan dan Evaluasi Model

Setelah dilakukan *preprocessing data,* kemudian dibagi menjadi dua *sub set,* yaitu data pelatihan atau disebut sebagai *training data* dan data pengujian atau yang disebut sebagai *testing data*. Kemudian klasifikasi model dilatih menggunakan *data training*, dan kinerja model tersebut kemudian dilakukan evaluasi menggunakan *data testing*. Dalam penelitian ini hasil klasifikasi menghasilkan label *multiclass*. Oleh karena ini pengujian atau evaluasi model menggunakan *Confusion Matrix*, dengan parameter evaluasi mencakup *accuracy, precision, recall* dan dan *F1-score*. [29]

Persamaan perhitungan *accuracy, precision, recall* dan dan *F1-score* dapat dirumuskan seperti pada persamaan 7, 8, 9 dan 10. Sedangkan tabel *Confusion Matrix* dapat terlihat pada tabel 2.1. [30], [31], [32]

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} x\ 100\% \quad (7)$$

$$Precision = \frac{TP}{TP+FP}\ x\ 100\% \quad (8)$$

$$Recall = \frac{TP}{TP+FN}\ x\ 100\% \quad (9)$$

$$F1Score = 2\ x\ \frac{Recall+Precission}{Recall+Precission}\ x\ 100\% \quad (10)$$

Tabel 2.1 Conffusion Matrix

	TERKLASIFIKASI POSITIF	**TERKLASIFIKASI NEGATIF**
POSITIF	TP (True Positive)	FN (False Negative)
NEGATIF	FP (False Positive)	TN (True Negative)

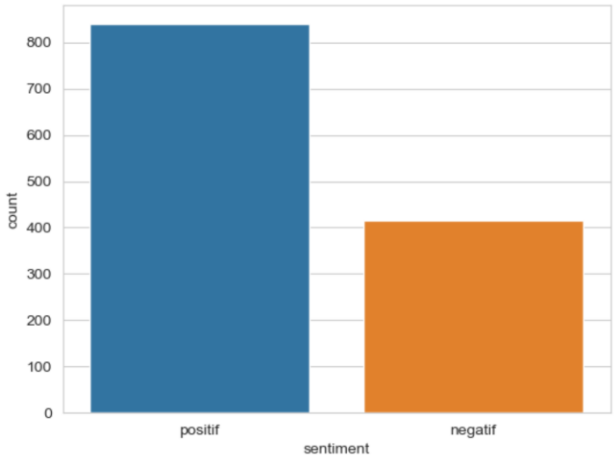
2.7 Analisis Hasil

Hasil dari komparasi algoritma klasifikasi dianalisis secara mendalam, termasuk identifikasi algoritma yang paling efektif dalam mengklasifikasikan sentimen terkait kinerja dosen. Selain itu, juga dilakukan analisis faktor-faktor yang memengaruhi kinerja algoritma.

III. HASIL DAN PEMBAHASAN

3.1 Data Collection

Pada penelitian menggunakan *datasets* sebanyak 1254 data komentar kritik saran dari mahasiswa. Data komentar tersebut sudah mempunyai label di mana berisi 839 komentar positif dan 415 komentar negatif. Sebaran data komentar dapat dilihat pada Grafik 3.1.



Grafik 3.1 Sebaran Data

Datasets dibagi menjadi dua bagian, yaitu *data training* dan *data testing*. Proporsional yang digunakan dalam penelitian ini yaitu 80% *datasets* digunakan sebagai *data training*, dan 20% *datasets* digunakan sebagai *data testing*.

Pada tabel 3.1 merupakan contoh data sentimen positif dan negatif yang berasal dari data kritik dan saran perkuliahan dari mahasiswa.

Tabel 3.1 Contoh Sentimen Positif dan Negatif

ID	**Komentar**	**Sentimen**

102	Untuk alat praktikum ada beberapa yang kekurangan mungkin bisa lebih di tambah alatnya	negatif
104	Lab praktikum ruangan agak engap, kurang udara, kamar mandi harus kelantai 3	negatif
204	menurut saya sangat baik, dosen maupun staf membimbing dengan baik ketika kita ada kesulitan	positif
501	Metode pengajaran yang digunakan jelas dan efektif dalam penyampaian materi kepada mahasiswa	positif

3.2 Preprocessing

Preprocessing merupakan tahapan proses pembersihan data atau *cleaning data*. Tahapan yaitu dimulai *case folding, tokenize, stopword removal, dan stemming*. Selanjutnya didalam proses *stopword removal* digunakan kamus *stopwords* yang di ambil dari *library NLTK* Indonesia. Kemudian pada tahapan *stemming* dengan menggunakan *library* sastrawi. Berikut ini merupakan hasil dari tahapan *preprocessing.*

Tabel 3.2 Tahapan *Preprocessing*

Preprocessing	**Sebelum**	**Sesudah**
Case Folding	Untuk alat praktikum ada beberapa yang kekurangan mungkin bisa lebih di tambah alatnya	untuk alat praktikum ada beberapa yang kekurangan mungkin bisa lebih di tambah alatnya
Tokenize	untuk alat praktikum ada beberapa yang kekurangan mungkin bisa lebih di tambah alatnya	[‘untuk’, ’alat’, ‘praktikum’, ‘ada’, ’beberapa’, ‘yang’, ‘kekurangan’, ‘mungkin’, ‘bisa’, ‘lebih’, ‘ditambah’, ‘alatnya’]
Stopword Removal	untuk alat praktikum ada beberapa yang kekurangan mungkin bisa lebih di tambah alatnya	[’alat’, ‘praktikum’, ’beberapa’, ‘kekurangan’, ‘mungkin’, ‘bisa’, ‘lebih’, ‘ditambah’, ‘alatnya’]
Stemming	[’alat’, ‘praktikum’, ’beberapa’, ‘kekurangan’, ‘mungkin’, ‘bisa’, ‘lebih’, ‘ditambah’, ‘alatnya’]	[’alat’, ‘praktik’, ’beberapa’, ‘kurang’, ‘mungkin’, ‘bisa’, ‘lebih’, ‘tambah’, ‘alat’]

3.3 Word Cloud

Hasil analisis visualisasi *Word Cloud* pada sentimen positif dan sentimen negatif dapat terlihat pada grafik 3.2 Hasil *wordcloud.* Dari *Word Cloud* dapat dilihat bahwa hasil *Word Cloud* komentar positif kata mahasiswa, kuliah, ajar materi terlihat jelas itu menandakan kata tersebut sering muncul dalam komentar positif mahasiswa. Sedangkan pada *Word Cloud* komentar negatif, kata praktikum, kuliah, kurang kelihatan jelas, itu menandakan kata-kata tersebut sering dituliskan dalam komentar negatif mahasiswa dalam kinerja dosen.

Grafik 3.2 Hasil Wordcloud

3.4 Klasifikasi dan Evaluasi

Dalam penelitian ini, peneliti melakukan komparasi antara tiga algoritma klasifikasi yang berbeda, yaitu algoritma *Naïve Bayes*, algortima *Support Vector Machine (SVM)*, dan algoritma *Random Forest*, untuk melakukan klasifikasi sentimen terhadap kinerja dosen. *Datasets* yang digunakan terdiri dari ulasan dan umpan balik mahasiswa yang telah dikategorikan menjadi dua kategori sentimen, yaitu sentimen positif dan sentimen negatif.

Dataset yang digunakan yaitu 80% *dataset training*, atau sebanyak 1003 opini komentar mahasiswa dan 20% *datasets testing* atau sebanyak 241 komentar opini

mahasiswa. Kemudian dilakukan pengujian dengan *Confusion Matrix*. Dari hasil pengujian didapatkan hasil *Confusion Matrix* seperti pada tabel 3.3, tabel 3.4, dan tabel 3.5.

Tabel 3.3 Hasil *Confusion Matrix* dengan *Naive Bayes*

	POSITIF	NEGATIF	TOTAL
POSITIF	31	64	95
NEGATIF	1	155	156
TOTAL	32	219	251

Berdasarkan Tabel 3.3, hasil *confusion matrix* dari algoritma *Naive Bayes* menunjukkan bahwa dari total 251 data uji, model berhasil mengklasifikasikan 155 komentar negatif dengan benar *(True Negative)* dan 31 komentar opini positif dengan benar *(True Positive).* Namun, terdapat kesalahan klasifikasi yang signifikan, di mana model gagal mengenali 64 komentar yang sebenarnya positif dan salah mengklasifikasikannya sebagai negatif *(False Negative).* Di sisi lain, kesalahan prediksi untuk kelas negatif sangat rendah, dengan hanya 1 data negatif yang salah diklasifikasikan sebagai positif *(False Positive).* Hasil ini mengindikasikan bahwa model *Naive Bayes* memiliki kecenderungan yang kuat untuk memprediksi kelas mayoritas (negatif) dan kurang efektif dalam mengidentifikasi kelas minoritas (positif).

Tabel 3.4 Hasil *Confusion Matrix* dengan *Support Vector Machine (SVM)*

	POSITIF	NEGATIF	TOTAL
POSITIF	58	37	95
NEGATIF	7	149	156
TOTAL	65	186	251

Berdasarkan Tabel 3.4, hasil klasifikasi opini mahasiswa menggunakan algoritma *Support Vector Machine (SVM)* menunjukkan kinerja yang jauh lebih seimbang. Model ini berhasil mengidentifikasi 58 opini positif dan 149 opini negatif dengan benar. Meskipun masih terdapat 37 opini positif yang keliru diklasifikasikan sebagai negatif *(False Negative),* angka ini merupakan peningkatan signifikan dalam kemampuan mengenali sentimen positif dibandingkan model sebelumnya. Di sisi lain, kesalahan pada kelas negatif tetap terkendali dengan hanya 7 opini yang salah diberi label positif *(False Positive).* Secara keseluruhan, *confusion matrix* ini menggambarkan bahwa SVM mampu mengurangi bias terhadap kelas mayoritas (negatif) dan memberikan keseimbangan yang lebih baik dalam mengklasifikasikan kedua jenis sentimen opini mahasiswa.

Tabel 3.5 Hasil Confusion Matrix dengan Random Forest

	POSITIF	NEGATIF	TOTAL
POSITIF	58	37	95
NEGATIF	7	149	156
TOTAL	65	186	251

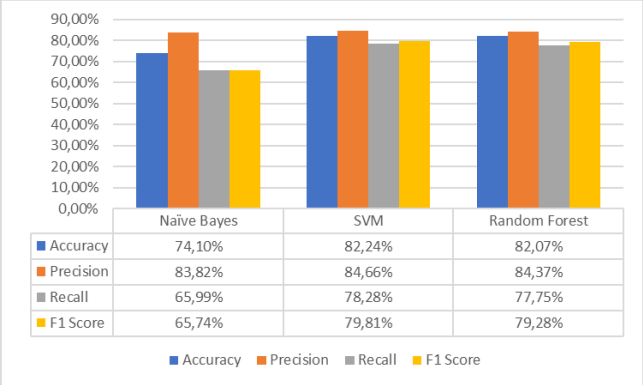
Tabel 3.5 menyajikan hasil klasifikasi opini mahasiswa menggunakan algoritma *Random Forest,* yang menunjukkan kinerja sangat mirip dengan SVM. Model ini berhasil mengidentifikasi 58 opini positif *(True Positive)* dan 149 opini negatif *(True Negative)* dengan tepat. Kesalahan klasifikasi terjadi pada 37 opini positif yang dianggap negatif *(False Negative)* dan 7 opini negatif yang dianggap positif *(False Positive).* Sebuah temuan yang sangat menarik adalah bahwa hasil confusion matrix ini identik dengan yang dihasilkan oleh *Support Vector Machine (SVM).* Hal ini mengindikasikan bahwa pada *dataset* ini, kedua model tersebut mencapai batas kinerja yang serupa dan memiliki kemampuan yang setara dalam menyeimbangkan prediksi antara sentimen positif dan negatif dari opini mahasiswa.

Langkah selanjutnya yaitu dilakukan komparasi algoritma klasifikasi tersebut, dengan melakukan evaluasi pada *Accuracy, Precision, Recall dan F1 Score*, dari hasil pengukuran didapatkan hasil seperti pada grafik 3.3.

Dari hasil klasifikasi Algoritma *Naïve Bayes* memperoleh nilai *accuracy* 74,10 %, nilai *precision* 83,82%, nilai *recall* 65.99% dan nilai *F1 Score* 65,74%. Sedangkan klasifikasi algorima *Support Vector machine (SVM)* mendapatkan hasil *accuracy* 82.24%, *precision* 84.66%, *recall* 78,28%, dan *F1 Scoce* 79,81%. Yang terakhir yaitu algoritma *random forest* menghasilkan nilai *accuracy* 82,07%, *precision* 84.37%, *recall* 77, 75%, dan nilai *F1 Score* 79.28%.

Dapat diketahui bahwa algoritma klasifikasi terbaik digunakan untuk klasifikasi komentar mahasiswa terhadap penilaian kinerja dosen, dengan *datasets* 1254 komentar, kemudian dilakukan *spliting* data 80% *data training* dan 20% *data testing* diketahui bahwa algoritma *Support Vector Machice (SVM)* memberikan nilai *accuracy, preccion, recall, dan f1 score* paling baik. Kemudian yang kedua yaitu algoritma *Random Forest*, dan yang ketiga yaitu *algoritma naive bayes.*

Fenomena menarik yang teramati adalah kinerja *SVM* dan *Random Forest* yang sangat berdekatan, bahkan menghasilkan *confusion matrix* yang identik. Kemiripan ini dapat dijelaskan secara teoritis dari karakteristik kedua algoritma. SVM bekerja dengan mencari hyperplane optimal yang memaksimalkan margin antar kelas, membuatnya sangat efektif dalam ruang fitur berdimensi tinggi seperti data teks hasil vektorisasi. Di sisi lain, *Random Forest* adalah model ensemble yang membangun banyak *decision tree* pada *subset* data yang berbeda dan menggabungkan hasilnya, yang memberikannya robustisitas tinggi terhadap *overfitting* dan kemampuan menangani interaksi fitur yang kompleks.



	Naïve Bayes	SVM	Random Forest
Accuracy	74,10%	82,24%	82,07%
Precision	83,82%	84,66%	84,37%
Recall	65,99%	78,28%	77,75%
F1 Score	65,74%	79,81%	79,28%

Grafik 3.3 Hasil Komparasi Pengukuran

IV. KESIMPULAN

Dalam penelitian ini, peneliti telah berhasil melakukan komparasi terhadap tiga algoritma klasifikasi yang berbeda, yaitu *Naïve Bayes, Support Vector Machine (SVM),* dan *Random Forest* untuk melakukan klasifikasi sentimen terhadap kinerja dosen dalam pendidikan tinggi. Ketiga algoritma tersebut dapat melakukan klasifikasi sentimen mahasiswa ke dalam dua kategori yaitu sentimen positif dan sentimen negatif. Dari hasil penelitian diketahui bahwa algoritma *Support Vector Mechine (SVM)* memberikan hasil yang paling bagus dibandingkan dengan 2 algoritma lainya. Algoritma *Support Vector Mechine (SVM)* berhasil menghasilkan nilai *accuracy* 82,24%, dengan nilai *precision* 84,665, nilai *recal*l 65,99% dan nilai *F1 score* 79,81%.

REFERENSI

[1] C. R. Glass, K. A. Godwin, and R. M. Helms, "Toward Greater Inclusion and Success," 2021.

[2] J. E. Chukwuere, "Student Voice in an Extended Curriculum Programme in the Era of Social Media: A systematic Review of Academic Literature," *IJHE*, vol. 10, no. 1, p. 147, Oct. 2020, doi: 10.5430/ijhe.v10n1p147.

[3] C. A. Pacol and T. D. Palaoag, "Enhancing Sentiment Analysis of Textual Feedback in the Student-Faculty Evaluation using Machine Learning Techniques," *Eur. j., eng. sci., tech.*, vol. 4, no. 1, pp. 27–34, Dec. 2021, doi: 10.33422/ejest.v4i1.604.

[4] N. Saraswathi, T. Sasi Rooba, and S. Chakaravarthi, "Improving the accuracy of sentiment analysis using a linguistic rule-based feature selection method in tourism reviews," *Measurement: Sensors*, vol. 29, p. 100888, Oct. 2023, doi: 10.1016/j.measen.2023.100888.

[5] S. Nayak, Savita, and Y. K. Sharma, "A modified Bayesian boosting algorithm with weight-guided optimal feature selection for sentiment analysis," *Decision Analytics Journal*, vol. 8, p. 100289, Sept. 2023, doi: 10.1016/j.dajour.2023.100289.

[6] Han, Jiawei, Kamber, Michelin, and Pei, Jian, *Data Mining: Concepts and Techniques*. Elsevier, 2012. doi: 10.1016/C2009-0-61819-5.

[7] W. F. Satrya, R. Aprilliyani, and E. H. Yossy, "Sentiment analysis of Indonesian police chief using multi-level ensemble model," *Procedia Computer Science*, vol. 216, pp. 620–629, 2023, doi: 10.1016/j.procs.2022.12.177.

[8] S. Dey, S. Wasif, D. S. Tonmoy, S. Sultana, J. Sarkar, and M. Dey, "A Comparative Study of Support Vector Machine and Naive Bayes Classifier for Sentiment Analysis on Amazon Product Reviews," in *2020 International Conference on Contemporary Computing and Applications (IC3A)*, Lucknow, India: IEEE, Feb. 2020, pp. 217–220. doi: 10.1109/IC3A48958.2020.233300.

[9] Pristiyono, M. Ritonga, M. A. A. Ihsan, A. Anjar, and F. H. Rambe, "Sentiment analysis of COVID-19 vaccine in Indonesia using Naïve Bayes Algorithm," *IOP Conf. Ser.: Mater. Sci. Eng.*, vol. 1088, no. 1, p. 012045, Feb. 2021, doi: 10.1088/1757-899X/1088/1/012045.

[10] V. A. Fitri, R. Andreswari, and M. A. Hasibuan, "Sentiment Analysis of Social Media Twitter with Case of Anti-LGBT Campaign in Indonesia using Naïve Bayes, Decision Tree, and Random Forest Algorithm," *Procedia Computer Science*, vol. 161, pp. 765–772, 2019, doi: 10.1016/j.procs.2019.11.181.

[11] Riccosan and K. E. Saputra, "Multilabel multiclass sentiment and emotion dataset from indonesian mobile application review," *Data in Brief*, vol. 50, p. 109576, Oct. 2023, doi: 10.1016/j.dib.2023.109576.

[12] D. Ramachandran and R. Parvathi, "Analysis of Twitter Specific Preprocessing Technique for Tweets," *Procedia Computer Science*, vol. 165, pp. 245–251, 2019, doi: 10.1016/j.procs.2020.01.083.

[13] D. Rifaldi, Abdul Fadlil, and Herman, "Teknik Preprocessing Pada Text Mining Menggunakan Data Tweet 'Mental Health,'" *Decode*, vol. 3, no. 2, pp. 161–171, Apr. 2023, doi: 10.51454/decode.v3i2.131.

[14] E. Rosenberg *et al.*, "Sentiment analysis on Twitter data towards climate action," *Results in Engineering*, vol. 19, p. 101287, Sept. 2023, doi: 10.1016/j.rineng.2023.101287.

[15] N. Tri Romadloni and N. Dwi Septiyanti, "Optimasi Feature Selection Pada Komentar Media Sosial Terhadap Peralihan Tv Digital Menggunakan Naïve Bayes, Support Vector Machine dan K-Nearest Neighbor," *Decode*, vol. 3, no. 2, pp. 151–160, Apr. 2023, doi: 10.51454/decode.v3i2.121.

[16] D. Sudigyo, A. A. Hidayat, R. Nirwantono, R. Rahutomo, J. P. Trinugroho, and B. Pardamean, "Literature study of stunting supplementation in Indonesian utilizing text mining approach," *Procedia Computer Science*, vol. 216, pp. 722–729, 2023, doi: 10.1016/j.procs.2022.12.189.

[17] N. H. Jeremy and D. Suhartono, "Automatic personality prediction from Indonesian user on twitter using word embedding and neural networks," *Procedia Computer Science*, vol. 179, pp. 416–422, 2021, doi: 10.1016/j.procs.2021.01.024.

[18] M. Adriani, J. Asian, B. Nazief, S. M. M. Tahaghoghi, and H. E. Williams, "Stemming Indonesian: A confix-stripping approach," *ACM Transactions on Asian Language Information Processing*, vol. 6, no. 4, pp. 1–33, Dec. 2007, doi: 10.1145/1316457.1316459.

[19] H. Dwiharyono and S. Suyanto, "Stemming for Better Indonesian Text-to-Phoneme," *Ampersand*, vol. 9, p. 100083, 2022, doi: 10.1016/j.amper.2022.100083.

[20] R. L. Atenstaedt, "Word cloud analysis of historical changes in the subject matter of public health practice in the United Kingdom," *Public Health*, vol. 197, pp. 39–41, Aug. 2021, doi: 10.1016/j.puhe.2021.06.010.

[21] M. A. Hearst, E. Pedersen, L. Patil, E. Lee, P. Laskowski, and S. Franconeri, "An Evaluation of Semantically Grouped Word Cloud Designs," *IEEE TRANSACTIONS ON VISUALIZATION AND COMPUTER GRAPHICS*.

[22] Y. W. W. Cui, "Context Preserving Dynamic Word Cloud Visualization".

[23] Y. Jin, "Development of Word Cloud Generator Software Based on Python," *Procedia Engineering*, vol. 174, pp. 788–792, 2017, doi: 10.1016/j.proeng.2017.01.223.

[24] N. B. Putri and A. W. Wijayanto, "Analisis Komparasi Algoritma Klasifikasi Data Mining Dalam Klasifikasi Website Phishing," *Komputika*, vol. 11, no. 1, pp. 59–66, Jan. 2022, doi: 10.34010/komputika.v11i1.4350.

[25] A. P. Nardilasari, A. L. Hananto, S. S. Hilabi, T. Tukino, and B. Priyatna, "Analisis Sentimen Calon Presiden 2024 Menggunakan Algoritma SVM Pada Media Sosial Twitter," *JOINTECS*, vol. 8, no. 1, p. 11, Mar. 2023, doi: 10.31328/jointecs.v8i1.4265.

[26] D. Wang and Y. Zhao, "Using News to Predict Investor Sentiment: Based on SVM Model," *Procedia Computer Science*, vol. 174, pp. 191–199, 2020, doi: 10.1016/j.procs.2020.06.074.

[27] Y. Al Amrani, M. Lazaar, and K. E. El Kadiri, "Random Forest and Support Vector Machine based Hybrid Approach to Sentiment

Analysis," *Procedia Computer Science*, vol. 127, pp. 511–520, 2018, doi: 10.1016/j.procs.2018.01.150.

[28] M. Ozcan and S. Peker, "A classification and regression tree algorithm for heart disease modeling and prediction," *Healthcare Analytics*, vol. 3, p. 100130, Nov. 2023, doi: 10.1016/j.health.2022.100130.

[29] M. M. Ghiasi and S. Zendehboudi, "Decision tree-based methodology to select a proper approach for wart treatment," *Computers in Biology and Medicine*, vol. 108, pp. 400–409, May 2019, doi: 10.1016/j.compbiomed.2019.04.001.

[30] Q. Gu, L. Zhu, and Z. Cai, "Evaluation Measures of the Classification Performance of Imbalanced Data Sets," in *Computational Intelligence and Intelligent Systems*, vol. 51, Z. Cai, Z. Li, Z. Kang, and Y. Liu, Eds., in Communications in Computer and Information Science, vol. 51. , Berlin, Heidelberg: Springer Berlin Heidelberg, 2009, pp. 461–471. doi: 10.1007/978-3-642-04962-0_53.

[31] M. Roshani *et al.*, "Evaluation of flow pattern recognition and void fraction measurement in two phase flow independent of oil pipeline's scale layer thickness," *Alexandria Engineering Journal*, vol. 60, no. 1, pp. 1955–1966, Feb. 2021, doi: 10.1016/j.aej.2020.11.043.

[32] D. Valero-Carreras, J. Alcaraz, and M. Landete, "Comparing two SVM models through different metrics based on the confusion matrix," *Computers & Operations Research*, vol. 152, p. 106131, Apr. 2023, doi: 10.1016/j.cor.2022.106131.