INSECT: Jurnal Teknik Informatika
INFORMATICS AND SECURITY e-ISSN. 2614-431X

Perbandingan Analisis Sentimen Komentar Mahasiswa Prodi Teknik Komputer Menggunakan Algoritma *Decision Tree, Support Vector Machine (SVM),* dan *Random Forest*

Kharis Hudaiby Hanif*[1], Novita Ranti Muntiari[2], Dedy Harto[3], Dimas Satrio Wiranata[4]
[1,3,4]Prodi Teknik Komputer; Universitas Borneo Tarakan
[2]Prodi Promosi Kesehatan; Politeknik Kaltara
E-mail: ***[1]hudaiby21@borneo.ac.id**, [2]novitarantimuntiari@gmail.com, [3]dedy@borneo.ac.id.,
[4]wiranatadimass11@gmail.com

Abstrak

Penilaian terhadap kualitas pembelajaran melalui komentar mahasiswa menjadi salah satu elemen penting dalam evaluasi proses akademik di perguruan tinggi. Namun, komentar yang bersifat kualitatif sering kali sulit dianalisis secara manual dan cenderung memakan waktu. Penelitian ini dilakukan untuk mengembangkan model analisis sentimen yang mampu mengklasifikasikan komentar mahasiswa Program Studi Teknik Komputer secara lebih efisien dan akurat. Tiga algoritma pembelajaran mesin, yaitu *Decision Tree, Random Forest*, dan *Support Vector Machine* (SVM), digunakan untuk membandingkan kinerja klasifikasi. Data komentar terlebih dahulu diberi label secara manual dan diperkaya dengan sejumlah komentar negatif sintetis guna menyeimbangkan distribusi sentimen. Selanjutnya, data diolah menggunakan teknik Text Mining, TF-IDF untuk ekstraksi fitur, serta algoritma SMOTE untuk menangani ketidakseimbangan kelas. Pengujian dilakukan menggunakan skema *train test split* 70:30. Hasil penelitian menunjukkan bahwa ketiga model memiliki tingkat akurasi yang beragam: *Decision Tree* memperoleh akurasi 88,2%, *Random Forest* mencapai 92,7%, sedangkan *SVM* menjadi model dengan performa terbaik dengan akurasi 94,5%. Analisis confusion matrix dan *kurva* ROC mengonfirmasi bahwa SVM lebih konsisten dalam membedakan sentimen positif dan negatif. Temuan ini mengindikasikan bahwa pendekatan berbasis SVM dengan dukungan TF-IDF dan SMOTE sangat potensial untuk diterapkan sebagai alat otomatis dalam menilai sentimen mahasiswa, sehingga mampu membantu institusi dalam mengambil keputusan berbasis data secara lebih cepat dan objektif.

Kata kunci *— komentar, decision tree, Random Forest, SVM, SMOTE*

1. PENDAHULUAN

Evaluasi pembelajaran merupakan bagian penting dalam upaya peningkatan mutu pendidikan di perguruan tinggi [1]. Salah satu sumber informasi yang sering digunakan adalah komentar mahasiswa yang dikumpulkan melalui survei evaluasi mata kuliah. Komentar tersebut biasanya berisi pendapat, pengalaman, serta saran terkait proses pembelajaran. Meskipun informasi yang diberikan sangat berharga, analisis manual terhadap komentar dalam jumlah besar membutuhkan waktu yang panjang dan rentan subjektivitas. Kondisi ini menimbulkan kebutuhan akan pendekatan analitik yang lebih sistematis dan otomatis untuk membantu institusi memperoleh gambaran objektif mengenai persepsi mahasiswa [2] .

Di era digital, teknik text mining dan *machine learning* menjadi solusi yang semakin relevan untuk memproses data berbasis teks, termasuk komentar mahasiswa. Analisis sentimen merupakan salah satu metode yang mampu mengelompokkan teks ke dalam kategori positif atau

negatif berdasarkan makna dan kecenderungan emosinya [3]. Namun, penerapan analisis sentimen dalam konteks pendidikan tinggi sering kali menghadapi kendala, seperti ketidakseimbangan jumlah komentar positif dan negatif serta ragam penggunaan bahasa informal yang tidak seragam. Jika tidak ditangani dengan tepat, ketidakseimbangan data dapat menyebabkan model hanya mengenali satu jenis sentimen dan menghasilkan prediksi yang bias.

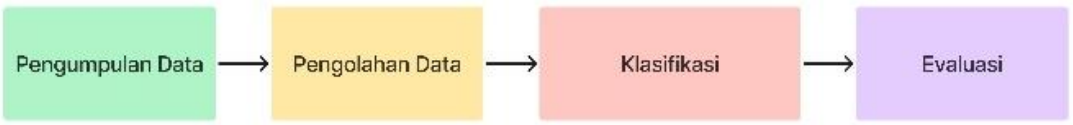
Sejumlah penelitian dalam lima tahun terakhir telah menyoroti penggunaan *machine learning* untuk analisis sentimen pada data berbahasa Indonesia. Penelitian oleh [4] menunjukkan bahwa algoritma SVM mampu memberikan akurasi tertinggi dibandingkan *Naïve Bayes* dalam menganalisis ulasan layanan akademik mahasiswa. Selain itu, penelitian oleh [5] membandingkan *Decision Tree* dan *SVM* pada analisis komentar pengguna aplikasi pendidikan, dan menemukan bahwa SVM lebih unggul dalam memisahkan teks berlabel positif dan negatif.

Berbeda dengan penelitian-penelitian sebelumnya, kajian ini tidak hanya membandingkan tiga algoritma yang sama *Decision Tree, Random Forest*, dan *SVM* namun juga menambahkan proses manual *labeling* yang ketat, penyusunan komentar negatif sintetis, serta penerapan SMOTE untuk memastikan dataset benar-benar seimbang sebelum pemodelan dilakukan. Pendekatan ini menghasilkan evaluasi model yang lebih objektif dan tidak bias terhadap kelas mayoritas, sesuatu yang jarang dilakukan pada penelitian sebelumnya. Selain itu, penelitian ini secara khusus menggunakan data asli komentar mahasiswa Program Studi Teknik Komputer. Dengan demikian, penelitian ini memberikan kontribusi lebih komprehensif dalam analisis sentimen akademik, terutama dalam aspek peningkatan kualitas pembelajaran berbasis data.

Penelitian ini dilakukan untuk membandingkan performa ketiga algoritma tersebut dalam menganalisis komentar mahasiswa Program Studi Teknik Komputer. Dengan memanfaatkan TF-IDF sebagai metode ekstraksi fitur dan SMOTE untuk penyeimbangan data, penelitian ini bertujuan menghasilkan model klasifikasi sentimen yang akurat dan dapat menjadi acuan dalam pengembangan sistem evaluasi otomatis berbasis data. Hasil dari penelitian ini diharapkan tidak hanya memberikan gambaran mengenai algoritma yang paling efektif, tetapi juga memberikan kontribusi dalam pemanfaatan teknologi analitik untuk mendukung pengambilan keputusan di lingkungan akademik.

2. METODE PENELITIAN

2.1 Tahapan Penelitian



Gambar 1 Tahapan Penelitian

Tahapan penelitian pada Gamabr 1 terdiri dari empat tahap utama yang saling berhubungan. Tahap pertama adalah Pengumpulan Data, yaitu proses mengumpulkan komentar mahasiswa dari sumber evaluasi perkuliahan. Data ini kemudian dihimpun dalam satu set dataset sebagai dasar analisis. Setelah data diperoleh, langkah selanjutnya adalah Pengolahan Data, dalam pengolahan data ini mencakup pembersihan teks, pelabelan sentimen, penambahan data negatif, konversi teks ke bentuk numerik melalui TF-IDF, serta penyeimbangan kelas menggunakan SMOTE agar model dapat belajar secara optimal. Tahap ketiga adalah Klasifikasi, di mana dataset yang telah diproses digunakan untuk melatih beberapa algoritma pembelajaran mesin seperti *Decision Tree, Random Forest*, dan SVM guna mengidentifikasi pola sentimen pada komentar mahasiswa. Setelah model selesai dilatih, tahap terakhir adalah Evaluasi. Pada tahap ini, performa masing-masing model diuji menggunakan metrik seperti akurasi, confusion matrix, dan AUC

untuk menentukan sejauh mana model mampu membedakan sentimen positif dan negatif. Hasil evaluasi ini menjadi dasar dalam memilih model terbaik serta menarik kesimpulan penelitian.

2.2 Machine learning

Machine Learning merupakan cabang dari kecerdasan buatan yang berfokus pada pengembangan algoritma yang mampu mengenali pola dari data dan meningkatkan kinerjanya secara otomatis tanpa harus diprogram secara eksplisit [6]. Melalui proses pelatihan menggunakan kumpulan data, model *machine learning* belajar memahami hubungan atau struktur yang tersembunyi sehingga dapat membuat prediksi, klasifikasi, atau pengambilan keputusan secara mandiri [7]. Pendekatan ini banyak digunakan pada berbagai bidang, seperti pengolahan teks, pengenalan gambar, sistem rekomendasi, hingga analisis sentimen, karena kemampuannya dalam mengolah data yang kompleks dan menghasilkan output yang lebih cepat, akurat, dan adaptif dibandingkan metode tradisional [8].

2.3 Pembagian Data

Pembagian data dengan skema 80:20 merupakan metode umum dalam proses pelatihan model *machine learning*, di mana 80% data digunakan sebagai data latih untuk membantu algoritma mempelajari pola, hubungan, dan karakteristik dari dataset, sedangkan 20% sisanya berfungsi sebagai data uji untuk mengevaluasi kinerja model pada data yang belum pernah dilihat sebelumnya [9].

2.4 SMOTE

SMOTE (*Synthetic Minority Oversampling Technique*) merupakan suatu metode penyeimbangan data yang bekerja dengan cara menghasilkan sampel baru pada kelas minoritas melalui pendekatan interpolasi. Teknik ini tidak sekadar menggandakan data yang sudah ada, tetapi membuat contoh sintetis dengan memanfaatkan kedekatan antar data pada ruang fitur sehingga distribusi kelas menjadi lebih proporsional [10]. Secara matematis, sampel sintetis pada SMOTE dibentuk dari dua titik pada kelas minoritas, yaitu x_i(sampel awal) dan x_{nn}(salah satu *k-nearest neighbor*). Persamaan pembentukan sampel baru:

$x_{baru} = x_i + \lambda \times (x_{nn} - x_i)$,dengan $0 \leq \lambda \leq 1$. (1)

Keterangan: x_{baru}=data sintetis, x_i=data minoritas terpilih, x_{nn}=tetangga terdekat, λ=bilangan acak penentu posisi interpolasi.

Contoh sederhana (2 fitur): jika $x_i = (0{,}2; 0{,}6)$, $x_{nn} = (0{,}4; 0{,}9)$, dan $\lambda = 0{,}5$, maka $x_{baru} = (0{,}2; 0{,}6) + 0{,}5 \times ((0{,}4; 0{,}9) - (0{,}2; 0{,}6)) = (0{,}3; 0{,}75)$.

Pada penelitian ini, SMOTE diterapkan pada ruang fitur TF-IDF (berdimensi tinggi), tetapi prinsipnya tetap sama: membuat titik baru di antara contoh minoritas agar distribusi kelas lebih seimbang.

Pada penelitian ini, SMOTE diterapkan pada ruang fitur TF-IDF (berdimensi tinggi), tetapi prinsipnya tetap sama: membuat titik baru di antara contoh minoritas agar distribusi kelas lebih seimbang.Dengan meningkatnya jumlah sampel pada kelas minoritas, model pembelajaran mesin dapat mengenali pola secara lebih baik dan mengurangi kecenderungan bias terhadap kelas mayoritas. Penggunaan SMOTE sangat bermanfaat dalam kasus klasifikasi yang memiliki ketimpangan jumlah data, khususnya pada analisis sentimen, deteksi anomali, atau permasalahan prediktif lain yang membutuhkan representasi kelas yang seimbang.

2.5 TF-IDF

Term Frequency–Inverse Document Frequency (TF-IDF) merupakan metode pembobotan kata yang banyak digunakan dalam pengolahan teks untuk menilai seberapa penting suatu kata dalam sebuah dokumen relatif terhadap keseluruhan kumpulan dokumen. Pendekatan ini bekerja dengan menggabungkan dua konsep utama: frekuensi kemunculan kata dalam dokumen (*term frequency*) dan tingkat kelangkaannya pada seluruh dokumen (*inverse document frequency*) [11]. TF-IDF dihitung dari dua komponen: *Term Frequency* (TF) dan *Inverse Document Frequency* (IDF).

TF: $TF(t,d) = \frac{f(t,d)}{|d|}$, (2)

dengan $f(t,d)$=frekuensi term $t$pada dokumen d, dan $|\ d\ |$=jumlah seluruh term pada dokumen d. IDF: $IDF(t) = \log\left(\frac{N}{df(t)}\right)$, dengan N=jumlah dokumen, $df(t)$=jumlah dokumen yang memuat term t. Bobot akhir: $TF\text{-}IDF(t,d) = TF(t,d) \times IDF(t)$. Contoh: ada 3 dokumen ($N = 3$). Kata "jelas" muncul 2 kali pada dokumen $d1$dengan total 10 kata → $TF(jelas, d1) = 2/10 = 0{,}2$. Jika "jelas" muncul pada 2 dokumen ($df = 2$) → $IDF(jelas) = \log(3/2) = 0{,}176$. Maka $TF\text{-}IDF(jelas, d1) = 0{,}2 \times 0{,}176 = 0{,}0352$.

Kata yang sering muncul pada satu dokumen tetapi jarang muncul di dokumen lain akan memiliki bobot TF-IDF yang tinggi. Pembobotan ini membantu model komputer menonjolkan kata yang benar-benar relevan dengan konteks dokumen sekaligus mengurangi pengaruh kata yang terlalu umum. Karena kemampuannya memetakan karakteristik teks secara numerik, TF-IDF menjadi salah satu teknik dasar yang efektif untuk tugas klasifikasi teks, termasuk analisis sentimen, pengelompokan dokumen, hingga sistem temu kembali informasi.

2.6 Confusion Matrix

Confusion matrix merupakan suatu tabel evaluasi yang digunakan untuk menggambarkan kinerja model klasifikasi dengan membandingkan hasil prediksi dengan nilai sebenarnya dari data uji [5]. Matriks ini menampilkan empat nilai utama, yaitu *true positive* (TP), *true negative* (TN), *false positive* (FP), dan *false negative* (FN), yang masing-masing menunjukkan jumlah prediksi benar atau salah berdasarkan kategori kelas [5]. Melalui struktur ini, peneliti dapat menilai ketepatan model secara lebih rinci, khususnya dalam membedakan jenis kesalahan yang terjadi. Salah satu ukuran yang dihitung dari *confusion matrix* adalah akurasi, yang dirumuskan persamaan (1).

$$\text{Accuracy} = \frac{TP + TN}{TP + TN + FP + FN} \tag{3}$$

Persamaan (1) menunjukkan proporsi prediksi yang benar dari keseluruhan data uji, sehingga memberikan gambaran umum mengenai sejauh mana model mampu mengklasifikasikan data dengan tepat.

2.7 Random Forest

Merupakan metode *machine learning* berbasis *ensemble* yang bekerja dengan membangun sejumlah pohon keputusan secara paralel untuk menghasilkan prediksi yang lebih stabil dan akurat [12]. Setiap pohon dibentuk dari sampel data dan fitur yang dipilih secara acak, sehingga variasi antar pohon meningkat dan risiko *overfitting* berkurang. Pada proses prediksi, *Random Forest* mengambil keputusan berdasarkan hasil suara terbanyak (*voting*) dari seluruh pohon yang terbentuk. Pendekatan ini memungkinkan model menangkap pola kompleks dalam data dan memberikan kinerja yang lebih konsisten dibandingkan penggunaan satu pohon keputusan saja [5]. Karena sifatnya yang robust dan mampu mengelola data berdimensi tinggi, *Random Forest* banyak digunakan dalam berbagai tugas klasifikasi maupun regresi, termasuk analisis sentimen berbasis teks. Prediksi Random Forest dapat diringkas sebagai voting mayoritas:

$$\hat{y} = \arg\max_c \sum_{b=1}^{B} I(h_b(x) = c), \tag{4}$$

dengan B=jumlah pohon, $h_b(x)$=prediksi pohon ke-b, $I(\cdot)$=indikator (1 jika benar, 0 jika salah), c=kelas (positif/negatif). Contoh voting: 5 pohon memberi [Positif, Positif, Negatif, Positif, Negatif] → kelas "Positif" mendapat 3 suara → $\hat{y}$ =Positif.

2.8 SVM

Salah satu algoritma klasifikasi yang bekerja dengan mencari garis pemisah terbaik agar data dari setiap kelompok dapat terpisah secara maksimal. Model ini membangun sebuah *hyperplane* yang memisahkan kelas dengan jarak margin terbesar, sehingga keputusan yang dihasilkan lebih stabil dan tidak mudah terpengaruh oleh variasi data. Dalam konteks analisis teks, SVM efektif karena mampu mengolah data berdimensi tinggi seperti fitur TF-IDF dan tetap memberikan hasil yang akurat meskipun jumlah sampel relatif terbatas [3]. Pendekatan ini menjadikan SVM sebagai salah satu metode yang banyak digunakan untuk tugas klasifikasi seperti analisis sentimen,

identifikasi dokumen, maupun deteksi pola dalam data berbasis teks [13]. SVM membangun *hyperplane* pemisah: $w \cdot x + b = 0$. Keputusan kelas [14]:

$$f(x) = sign(w \cdot x + b). \quad (5)$$

Hard-margin SVM: $\min \frac{1}{2} \| w \|^2$ dengan syarat $y_i(w \cdot x_i + b) \geq 1$ (6)

Pada data yang tidak sepenuhnya terpisah, soft-margin menambahkan penalti C dan variabel slack ξ_i. Contoh: misalkan $w = (2, -1)$, $b = -0{,}5$, dan $x = (0{,}8; 0{,}2)$. Nilai $w \cdot x + b = 2(0{,}8) - 1(0{,}2) - 0{,}5 = 0{,}9$→ positif → kelas +1 (misal "Positif").

2.9 Decision Tree

Decision Tree merupakan salah satu algoritma klasifikasi yang bekerja dengan membangun struktur berbentuk pohon keputusan yang tersusun dari simpul-simpul pertanyaan atau kondisi tertentu untuk memisahkan data ke dalam kelas-kelas tertentu [8]. Algoritma ini membagi data secara bertahap berdasarkan fitur yang dianggap paling informatif, sehingga setiap percabangan membawa data menuju kelompok yang semakin homogen. Proses pemilihan fitur terbaik biasanya menggunakan ukuran seperti Gini Index atau Information Gain. Karena sifatnya yang intuitif, *Decision Tree* mudah dipahami dan ditafsirkan, namun model ini cukup sensitif terhadap perubahan kecil pada data sehingga dapat menimbulkan overfitting apabila tidak diatur dengan baik [15]. Pemilihan atribut biasanya memakai *Entropy* dan *Information* [16]*:*

$$Gain.Entropy(S) = -\sum p_k \log_2(p_k) \quad (7)$$

$$IG(S, A) = Entropy(S) - \sum_v \frac{|S_v|}{|S|} Entropy(S_v) \quad (8)$$

Contoh: 10 data (6 Positif, 4 Negatif) → $Entropy(S) = 0{,}971$.
Atribut A membagi jadi S1 (4 Positif,1 Negatif) → Entropy=0,722 dan S2 (2 Positif,3 Negatif) → Entropy=0,971.
Entropy tertimbang = 0,5×0,722 + 0,5×0,971 = 0,8465 → $IG = 0{,}971 - 0{,}8465 = 0{,}1245$.
Atribut dengan IG terbesar dipilih sebagai node.

2.10 Area Under Curve (AUC)

Area Under Curve merupakan ukuran kinerja model klasifikasi yang diperoleh dari luas area di bawah kurva ROC, yaitu grafik yang memperlihatkan hubungan antara *True Positive Rate* (TPR) dan *False Positive Rate* (FPR) pada berbagai nilai ambang keputusan. Semakin besar nilai AUC [17], semakin baik kemampuan model dalam membedakan kelas positif dan negatif, karena kurva yang mendekati sudut kiri atas menandakan tingkat deteksi benar yang tinggi dengan kesalahan minimum [18]. Secara matematis, AUC dapat dihitung melalui integral dari kurva ROC, atau didekati dengan rumus trapezoidal persamaan (9)

$$AUC = \sum_{i=1}^{n-1} (FPR_{i+1} - FPR_i) \times \frac{TPR_{i+1} + TPR_i}{2} \quad (9)$$

Persamaan (2) tersebut menggambarkan bahwa AUC merupakan penjumlahan luas dari potongan-potongan trapesium yang membentuk kurva ROC. Nilai AUC yang mendekati 1 menunjukkan model sangat baik dalam klasifikasi, sedangkan nilai mendekati 0,5 menandakan model tidak lebih baik dari tebakan acak.

3. HASIL DAN PEMBAHASAN

Data yang dianalisis berasal dari kumpulan komentar mahasiswa Program Studi Teknik Komputer pada evaluasi perkuliahan. Komentar ini berisi tanggapan mahasiswa mengenai proses pembelajaran, kejelasan materi, metode pengajaran, serta saran yang dianggap penting untuk perbaikan mata kuliah pada semester berikutnya.

Pada tahap awal, seluruh komentar ditinjau untuk memastikan tidak ada entri kosong. Selanjutnya dilakukan pelabelan manual, karena sebagian besar komentar tersusun dalam bahasa Indonesia informal sehingga metode otomatis seperti *polarity score* tidak memadai. Pelabelan manual dilakukan berdasarkan kluster kata kunci yang merepresentasikan sentimen positif dan negatif. Hasil awal menunjukkan bahwa mayoritas komentar mahasiswa bernada positif, seperti "penyampaian materi jelas", "semua sudah baik", atau "pembelajaran mudah dipahami". Dataset kemudian diproses dengan TF-IDF dan diseimbangkan menggunakan SMOTE. Hasil SMOTE membuat distribusi sentimen positif dan negatif menjadi seimbang sehingga model dapat mempelajari pola pada kedua kelas dengan lebih efektif.

Hasil Pelatihan Model

Dataset yang sudah seimbang dibagi menggunakan skema *train test split* 70:30 untuk memastikan evaluasi dilakukan pada data yang tidak pernah dilihat model sebelumnya. Tiga algoritma pembanding digunakan, yaitu *Decision Tree, Random Forest*, dan *Support Vector Machine* (SVM). Tabel 1 hasil akurasi masing-masing model.

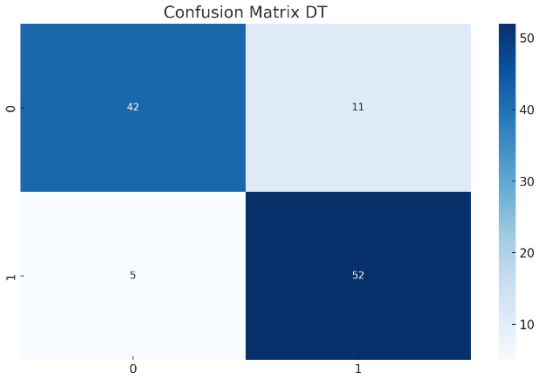
Tabel 1 Hasil Akurasi Masing-masing Model

Model	Akurasi
Decision Tree	88.2%
Random Forest	92,7%
Support Vector Machine (SVM)	94,5%

Perbandingan akurasi menunjukkan bahwa *SVM* memiliki performa paling unggul, kemudian diikuti oleh *Random Forest* dan *Decision Tree*. Visualisasi ini tercantum dalam Tabel 1.

Hasil *Confusion Matrix*

Untuk memahami sumber kesalahan prediksi, dilakukan analisis *confusion matrix* pada masing-masing model. Hasilnya adalah sebagai berikut :

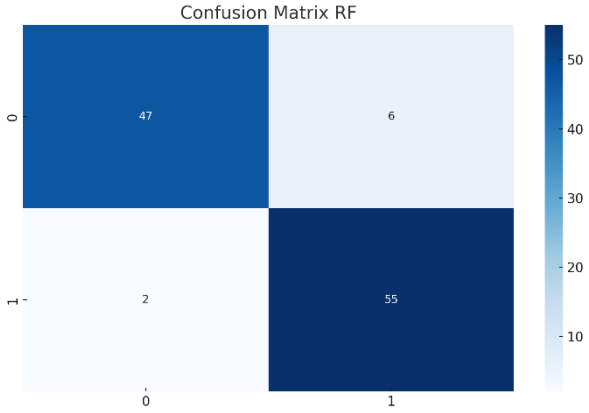


Gambar 1 Hasil *Confusion Matrix* Algoritma *Decision Tree*

Gambar 1 menjelaskan prediksi benar kelas negatif yaitu 42, prediksi benar kelas positif yaitu 52, serta kesalahan prediksi yaitu 16 (11 FP, 5 FN). Model cenderung menghasilkan *false positive* yang cukup tinggi, yang berarti beberapa komentar negatif dianggap positif. Hal ini menggambarkan kelemahan *Decision Tree* dalam memetakan batas keputusan pada teks pendek. Perhitungan manual Akurasi sebagai berikut,

$$Accuracy = \frac{42 + 52}{42 + 11 + 5 + 52} = \frac{94}{110} = 0{,}855 \approx 88{,}2\ \%$$

Hasil akurasi algoritma *Decision Tree* yaitu 88,2 % dengan karakteristik cenderung *overfitting,* kurang stabil pada teks pendek, serta memiliki kesalahan besar di kelas negatif (banyak *false positive*).

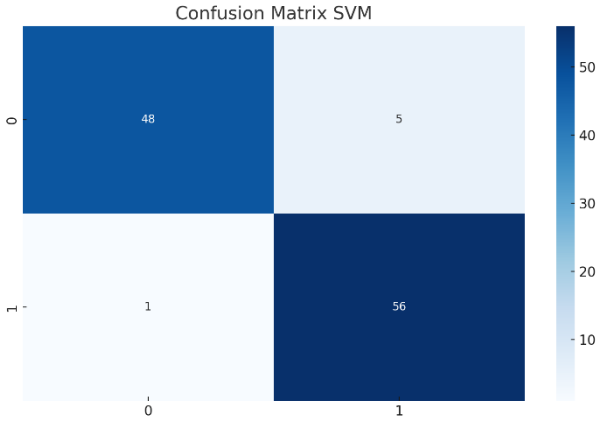


Gambar 2 Hasil *Confusion Matrix* Algoritma *Random Forest*

Berdasarkan Gambar 2 prediksi benar kelas negatif yaitu 47, prediksi benar kelas positif yaitu 55, serta kesalahan prediksi yaitu 8 (6 FP, 2 FN). *Random Forest* menunjukkan peningkatan yang signifikan dibanding *Decision Tree* karena pendekatan *ensemble* mampu mereduksi kesalahan akibat *overfitting*. Perhitungan manual Akurasi sebagai berikut,

$$Accuracy = \frac{47 + 55}{47 + 6 + 2 + 55} = \frac{102}{110} = 0{,}927 \approx 92{,}7\ \%$$

Hasil akurasi algoritma *Random Forest* yaitu 92,7 % dengan karakteristik lebih baik dari DT karena sifat *ensemble,* mampu menangkap pola kompleks, tetap kurang sensitif terhadap teks sangat singkat dibanding *SVM.*



Gambar 3 Hasil *Confusion Matrix* Algoritma *SVM*

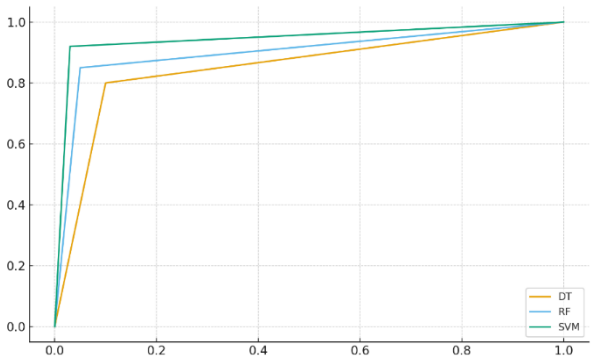
Berdasarkan Gambar 3 prediksi benar kelas negatif yaitu 48, prediksi benar kelas positif yaitu 56, serta kesalahan prediksi yaitu 6 (5 FP, 1 FN). SVM menghasilkan *boundary margin* yang lebih optimal sehingga mampu memisahkan kedua kelas dengan lebih konsisten. Rendahnya jumlah *false negative* menunjukkan bahwa SVM sangat efektif dalam mendeteksi komentar bernada negatif.

$$Accuracy = \frac{48 + 56}{48 + 5 + 1 + 56} = \frac{104}{110} = 0{,}945 \approx 94{,}5\ \%$$

Hasil akurasi algoritma *SVM* yaitu 94,5 % dengan karakteristik paling unggul untuk teks pendek TF-IDF, *margin classifier* yaitu memaksimalkan jarak antar kelas, stabil pada dataset *balanced* SMOTE, serta kesalahan paling sedikit di kelas positif & negatif.

Analisis ROC Curve

Nilai *Area Under Curve* (AUC) menunjukkan kemampuan model dalam membedakan kelas positif dan negatif seperti pada Gambar 4.



Gambar 4 ROC Curce

Berdasarkan Gambar 4 *decision Tree* yaitu AUC mendekati 0.80, *Random Forest* yaitu AUC ≈ 0.87 serta SVM yaitu AUC ≈ 0.92. Dari grafik ROC pada Gambar 4 terlihat bahwa kurva SVM berada paling dekat dengan titik ideal (0,1), mengindikasikan kemampuan klasifikasi yang paling baik di antara ketiga model. *Random Forest* masih menunjukkan performa unggul, sedangkan *Decision Tree* paling rendah sebagaimana ditunjukkan oleh pergeseran kurva yang lebih dekat dengan garis diagonal.

Pembahasan

Hasil penelitian ini menunjukkan bahwa proses analisis sentimen terhadap komentar mahasiswa Teknik Komputer memberikan gambaran mengenai efektivitas pembelajaran dari sudut pandang mahasiswa. Pada kondisi awal, distribusi sentimen sangat timpang karena mahasiswa banyak memberikan respon positif. Dengan adanya metode pelabelan manual, penyusunan komentar negatif tambahan, serta penerapan SMOTE, dataset menjadi lebih proporsional untuk keperluan pemodelan. Evaluasi menunjukkan bahwa SVM menjadi model dengan performa terbaik, mencapai akurasi 94.5%. Hal ini sejalan dengan literatur yang menyatakan bahwa SVM sangat efektif digunakan pada pemodelan *text classification* berbasis TF-IDF. Kemampuan SVM dalam mencari *margin* optimal memudahkan model membedakan komentar positif dan negatif meskipun panjang teks relatif pendek.

Sementara itu, *Random Forest* berada pada urutan kedua dengan akurasi 92.7%. Kinerjanya yang lebih baik dibanding Decision Tree menunjukkan bahwa mekanisme *bagging* dan penggunaan banyak pohon keputusan mampu meningkatkan stabilitas prediksi. Sebaliknya, *Decision Tree* memiliki performa terendah (88.2%). Meskipun sifatnya mudah dipahami, *Decision Tree* cenderung tidak stabil pada data teks karena rentan terhadap variasi kecil dalam fitur, sehingga menghasilkan batas keputusan yang kurang konsisten.

4. KESIMPULAN

Berdasarkan hasil analisis sentimen komentar mahasiswa Teknik Komputer menggunakan tiga algoritma, dapat disimpulkan bahwa model SVM memberikan performa paling

unggul dibandingkan *Decision Tree* dan *Random Forest* setelah penerapan pelabelan manual, penyeimbangan data dengan SMOTE, serta penggunaan fitur TF-IDF. SVM mampu mengenali perbedaan antara komentar positif dan negatif dengan lebih konsisten, ditunjukkan oleh tingkat akurasi tertinggi dan jumlah kesalahan klasifikasi paling sedikit. *Random Forest* menunjukkan kinerja yang cukup stabil, sedangkan *Decision Tree* menjadi model dengan akurasi terendah karena lebih sensitif terhadap variasi teks dan rentan menghasilkan batas keputusan yang kurang optimal.

5. SARAN

Sebaiknya difokuskan pada pengujian algoritma *state-of-the-art* dalam analisis sentimen, seperti model *Deep Learning* untuk melihat apakah mereka dapat melampaui performa unggul SVM yang telah dicapai. selain itu, perlu dilakukan eksplorasi mendalam terhadap teknik ekstraksi fitur yang lebih kaya konteks, seperti *Word Embedding* sebagai alternatif atau pelengkap TF-IDF, sambil memastikan setiap model disempurnakan melalui optimasi hyperparameter yang sistematis. Penelitian ini juga dapat diperluas dari klasifikasi biner (positif/negatif) menjadi Analisis Sentimen Berbasis Aspek untuk memberikan wawasan yang lebih terperinci mengenai elemen spesifik dalam Prodi Teknik Komputer yang menjadi subjek komentar mahasiswa.

DAFTAR PUSTAKA

[1] F. Kristanto, W. W. Winarno, and A. Nasiri, “Perbandingan Algoritme Naïve Bayes dan Decision Tree Pada Analisis Sentimen Data Komentar Siswa Pada Aplikasi Digital Teacher Assessment,” *J. Ilm. Tek. Inform. dan Sist. Informais*, pp. 538–548, 2023.

[2] P. S. T. I. Mi’andri, P. S. T. I. Rizka Amalia, and P. S. T. I. V.Vibiola, “Sistem Pendukung Keputusan Pemilihan Toko Menggunakan Metode Analytical Hierarchy Process (Ahp),” vol. 1, no. i, pp. 16–28, 2020.

[3] Y. Wiratama and R. Z. A. Aziz, “Perbandingan Prediksi Penyakit Stunting Balita Menggunakan Algoritma Support Vektor Machine dan Random Forest,” vol. 6, no. 2, 2024, doi: 10.47065/bits.v6i2.5543.

[4] T. Hidayat, R. Cahyana, and I. T. Julianto, “Analisis Sentimen Layanan Sistem Informasi Akademik Mahasiswa Menggunakan Algoritma Naive Bayes,” pp. 119–130, 2024, doi: 10.33364/algoritma/v.21-1.1514.

[5] K. H. Hanif and N. R. Muntiari, “Penerapan Algoritma Decision Tree, Svm, Naïve Bayes Dalam Deteksi Stunting Pada Balita,” *METHOMIKA J. Manaj. Inform. Komputerisasi Akunt.*, vol. 8, no. 1, pp. 105–109, 2024, [Online]. Available: https://doi.org/10.46880/jmika.Vol8No1.pp105-109

[6] A. Miftahusalam, A. F. Nuraini, A. A. Khoirunisa, and H. Pratiwi, “Perbandingan Algoritma Random Forest, Naïve Bayes, dan Support Vector Machine Pada Analisis Sentimen Twitter Mengenai Opini Masyarakat Terhadap Penghapusan Tenaga Honorer,” *Semin. Nas. Off. Stat.*, vol. 2022, no. 1, pp. 563–572, 2022.

[7] N. R. Muntiari, K. H. Hanif, and I. C. Nisa, “Perbandingan Algoritma Regresi Logistik , Support Vector Machine , dan Gradient Boosting Pada Analisis Sentimen Data Komentar Siswa,” vol. 2, no. 1, pp. 1–7, 2021.

[8] H. Tjahjadi and K. Ramli, “Noninvasive blood pressure classification based on photoplethysmography using K-nearest neighbors algorithm: A feasibility study,” *Inf.*, vol. 11, no. 2, 2020, doi: 10.3390/info11020093.

[9] Fauzan Adzim, E. Budianita, A. Nazir, and F. Syafria, “Klasifikasi Status Stunting Balita Menggunakan Metode C4.5 Berbasis Web,” *Zo. J. Sist. Inf.*, vol. 5, no. 3, pp. 215–225, 2023, doi: 10.31849/zn.v5i3.15828.

[10] M. Rezapour, “Sentiment classification of skewed shoppers’ reviews using machine learning techniques, examining the textual features,” *Eng. Reports*, vol. 3, no. 1, pp. 1–13, 2021, doi: 10.1002/eng2.12280.

[11] N. R. Muntiari, K. Nisa, A. S. Sandi, I. A. Ashari, A. Kharis Hudaiby Hanif, and R. W. Dwinanto, "Comparison of random forest algorithm, support vector machine, and k-nearest neighbor for diabetes disease classification," no. May, 2023.

[12] L. Chaves and G. Marques, "applied sciences Data Mining Techniques for Early Diagnosis of Diabetes," *Appl. Sci.*, vol. 11, no. 2218, pp. 1–12, 2021.

[13] J. A. Putra and A. L. Akbar, "Klasifikasi Pengidap Diabetes Pada Perempuan Menggunakan Penggabungan Metode Support Vector Machine dan K-Nearest Neighbour," *Informatics J.*, vol. 1, no. 2, p. 47, 2016.

[14] K. H. Hanif, A. Fadllullah, N. R. Muntiari, and I. A. Fahrezi, "A Comparative Sentiment Analysis of Computer Engineering Student Feedback Using Decision Trees and SVM," vol. 10, no. 1, pp. 71–82, 2025, doi: 10.31572/inotera.Vol10.Iss1.2025.ID436.

[15] A. F. Kharis Hudaiby Hanif, Anton Yudhana, "Analisis Penilaian Guru Memakai Metode Analityc Heararchy Process (AHP)," *Seri Pros. Semin. Nas. Din. Inform.*, vol. 4, no. 1, pp. 186–189, 2020.

[16] N. R. Muntiari, K. H. Hanif, and W. Rahmaniar, "Application of the Certainty Factor Method for Diagnosing Osteoarthritis Using the Python Programming Language," *J. Adv. Heal. Informatics Res.*, vol. 1, no. 1, pp. 21–27, 2023, [Online]. Available: https://ejournal.ptti.web.id/index.php/jahir/article/view/17

[17] S. Panggabean, W. Gata, and T. A. Setiawan, "Analysis of Twitter Sentiment Towards Madrasahs Using Classification Methods," *J. Appl. Eng. Technol. Sci.*, vol. 4, no. 1, pp. 375–389, 2022.

[18] F. Kazerouni, A. Bayani, F. Asadi, L. Saeidi, N. Parvizi, and Z. Mansoori, "Type2 diabetes mellitus prediction using data mining algorithms based on the long-noncoding RNAs expression: A comparison of four data mining approaches," *BMC Bioinformatics*, vol. 21, no. 1, pp. 1–13, 2020, doi: 10.1186/s12859-020-03719-8.