Journal of Technology and Informatics (JoTI)
Vol. 7, No. 2, Bulan Oktober Tahun 2025
P-ISSN 2721-4842
E-ISSN 2686-6102

Analisis Sentimen Pengguna X terhadap Perempuan di Lingkungan Kerja Menggunakan Algoritma Machine Learning

Muhammad Davit Hilal Fahri[1*], Dedi Gunawan[2]
[1,2]Program Studi Teknik Informatika, Universitas Muhammadiyah Surakarta, Jawa Tengah, Indonesia
e-mail: I200210146@student.ums.ac.id[1*], dg163@ums.ac.id[2],

Informasi Artikel

***Article History*:**
Received : 12 Juni 2025
Revised : 4 Juli 2025
Accepted : 23 Juli 2025
Published : 30 Agustus 2025

***Korespondensi:**
I200210146@student.ums.ac.id

Keywords:
Flask, Labeling, Naïve Bayes, Random Forest, Support Vector Machine

10.37802/joti.v7i2.1087

Journal of Technology and Informatics (JoTI)
P-ISSN 2721-4842
E-ISSN 2686-6102
https://e-journals.dinamika.ac.id/index.php/joti

***Abstract*:**

Gender bias against women in the workplace persists, including within digital interactions on social media. This study analyzes user sentiment on Platform X regarding women in professional contexts using three machine learning algorithms: Naïve Bayes, Support Vector Machine (SVM), and Random Forest. A total of 2,336 tweets were collected using 14 gender-related keywords and labeled both automatically using the DistilBERT model and manually through contextual interpretation. The automatic dataset was imbalanced (1,823 negative, 479 positive), while the manual dataset was more balanced (1,196 negative, 1,106 positive). After preprocessing and TF-IDF feature extraction, the data were split using the train_test_split method. Evaluation metrics included accuracy, precision, recall, and F1-score. Random Forest achieved the highest accuracy (79%) on automatic labels but showed class imbalance (F1-score: 0.88 for negative, 0.08 for positive). Meanwhile, models trained on manual labels showed more balanced performance with accuracy between 57% and 59%. A web application prototype was developed using Flask to predict sentiment related to workplace gender issues. The findings highlight the importance of balanced labeling and appropriate algorithm selection to build fair and reliable sentiment analysis models, contributing to more inclusive digital discourse on gender equality.

PENDAHULUAN

Isu kesetaraan gender, terutama dalam lingkungan kerja, terus menjadi perbincangan kritis di era modern. Nilai-nilai sosial, agama, dan budaya yang seringkali dipahami secara dangkal menciptakan konstruksi patriarkal yang bersifat destruktif, bahkan memicu patologi sosial seperti kekerasan seksual [1]. Fenomena diskriminatif ini tidak hanya terjadi di dunia nyata, tetapi juga tercermin dalam interaksi digital, di mana pengguna aktif menyampaikan sentimen positif maupun negatif terkait peran perempuan di dunia profesional. Analisis terhadap sentimen ini menjadi penting, tidak hanya untuk memetakan persepsi publik, tetapi juga untuk mengidentifikasi bias gender yang tersirat dalam komunikasi sehari-hari di lingkungan kerja.

Volume data yang besar dan keragaman perspektif pengguna memungkinkan analisis sentimen yang komprehensif, baik yang bersifat eksplisit (pujian/keluhan) maupun implisit (sarkasme/ironi). Karakteristik ini menjadikan Platform X sebagai sumber data ideal untuk penelitian sentimen berbasis *machine learning*. Melalui analisis ini, dapat diidentifikasi pola sikap pengguna baik positif dan negatif terhadap perempuan dalam konteks professional [2].

Peran algoritma *machine learning* seperti *Naïve Bayes*, *Support Vector Machine* (SVM), dan *Random Forest* dalam analisis sentimen memiliki peran yang krusial. Ketiga algoritma ini mampu mengidentifikasi pola linguistik dan mengklasifikasikan sentimen (positif dan negatif) dari konten terkait perempuan di lingkungan kerja pada Platform X. Keunggulan masing-masing algoritma, seperti kemampuan *Naïve Bayes* menangani data *sparitas*, ketangguhan SVM pada data berdimensi tinggi, dan akurasi *Random Forest* pada *dataset* kompleks, memungkinkan ekstraksi wawasan mendalam dari ekspresi verbal baik yang eksplisit maupun implisit. Dengan dukungan *machine learning*, penelitian ini bertujuan memberikan gambaran lebih jelas terkait persepsi publik terhadap perempuan di lingkungan kerja [3].

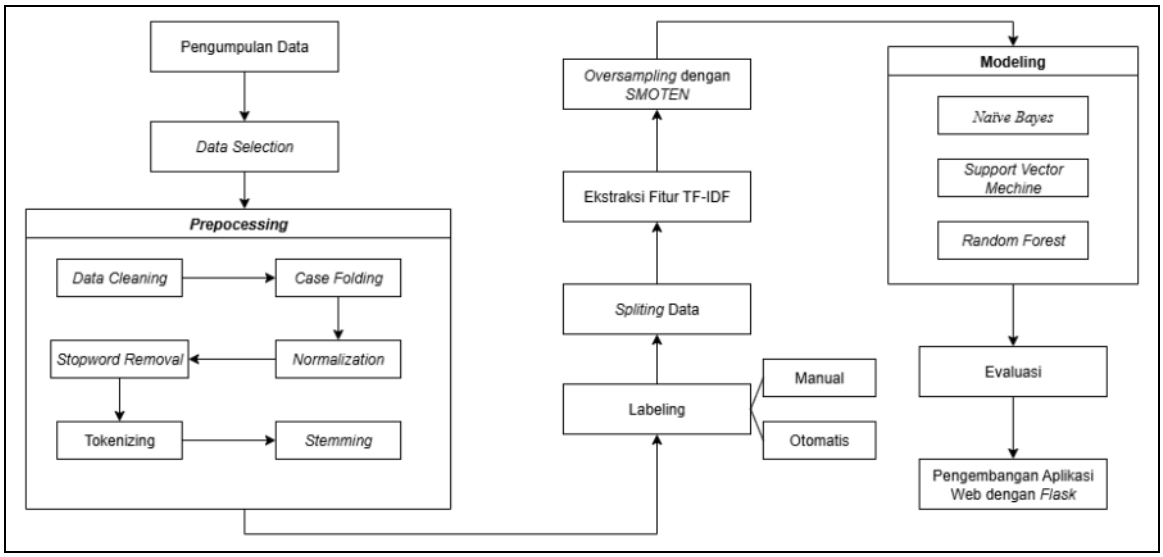
Penelitian oleh [4], diimplementasikan algoritma *Random Forest* untuk mendeteksi *hate speech* dan *abusive language* pada platform X berbahasa Indonesia. Penelitian ini mengumpulkan data dari 13.167 *tweet* yang berisi ujaran kebencian dan bahasa kasar. Metode *Random Forest* dipilih sebagai pengklasifikasi karena kemampuannya dalam menangani data yang kompleks dan memberikan akurasi yang baik. Hasil pengujian menunjukkan bahwa algoritma ini mampu mencapai akurasi sebesar 75,96% dengan menggunakan fitur dan parameter terbaik, sehingga diharapkan dapat membantu pengguna media sosial untuk lebih bijak dalam berinteraksi di dunia maya.

Penelitian oleh [5], membahas deteksi seksisme online menggunakan algoritma SVM dan *Naïve Bayes* pada 16.000 kalimat berbahasa Inggris dari media sosial. Penelitian ini menekankan pentingnya klasifikasi kalimat '*sexist*' dan '*not sexist*' serta membandingkan dua pendekatan: tanpa penanganan data tidak seimbang dan dengan teknik *SMOTE*. Hasilnya, SVM dengan *SMOTE* menghasilkan *F1-score* tertinggi sebesar 0,90, menunjukkan efektivitasnya dalam menangani ketidakseimbangan data.

Berdasarkan kajian dari penelitian sebelumnya, penelitian ini bertujuan untuk menganalisis sentimen positif dan negatif pengguna Platform X terkait isu perempuan di lingkungan kerja dengan mengimplementasikan tiga algoritma *machine learning*, yaitu *Naïve Bayes*, *Support Vector Machine* (SVM), dan *Random Forest*. Penelitian ini mengembangkan model klasifikasi sentimen (positif/negatif) yang diintegrasikan dalam sebuah aplikasi web berbasis *Flask* yang memungkinkan pengguna untuk memasukkan sebuah kalimat dan mendapatkan hasil analisis berupa prediksi sentimen.

METODE

Metode penelitian ini menggambarkan tahapan sistematis dalam menganalisis sentimen terhadap perempuan di lingkungan kerja. Proses dimulai dari pengumpulan *tweet* relevan, dilanjutkan *preprocessing*, ekstraksi fitur menggunakan TF-IDF, dan penyeimbangan data dengan *SMOTEN*. Data kemudian dibagi menjadi data latih dan uji, lalu dimodelkan menggunakan tiga algoritma *machine learning: Naïve Bayes*, SVM, dan *Random Forest*. Evaluasi dilakukan dengan metrik akurasi. Penelitian ini juga mencakup pelabelan data secara manual dan otomatis, serta penerapan model ke dalam aplikasi web menggunakan *Flask*. Alur lengkap metode ditampilkan pada Gambar 1.



Gambar 1. Tahapan metodologi penelitian

Pengumpulan Data

Data opini pengguna di Platform X dikumpulkan secara acak melalui metode *crawling* menggunakan *API*. Proses ini bertujuan mengunduh *tweet* yang relevan dengan pengalaman perempuan di lingkungan kerja secara efisien dan sistematis, sehingga lebih efektif dibandingkan pengumpulan manual [6].

Data Selection

Tahap ini merupakan tahap persiapan dalam pemilihan data. Data yang diperoleh melalui proses *crawling* dari platform X akan melalui seleksi atribut sebelum digunakan dalam tahap *preprocessing* [7].

Praprocessing

Proses praproses merupakan tahap penting untuk mengatasi hambatan dalam pengolahan data, dengan cara membersihkan dan mentransformasi data mentah agar siap dianalisis [8]. Tahapan ini mencakup beberapa langkah sebagai berikut.

1. *Data Cleaning*

   Data *Cleaning* merupakan salah satu tahap dalam *preprocessing* data yang bertujuan untuk membersihkan data. Proses ini mencakup penghapusan data duplikat, data kosong, karakter atau tanda baca yang tidak diperlukan, *Uniform Resource Locator* (URL), kode *HyperText Markup Language* (HTML), serta emotikon dan mention yang tidak relevan [9].
2. *Case Folding*

   Case folding merupakan proses transformasi seluruh huruf dalam dokumen menjadi bentuk huruf kecil (*lowercase*) guna memastikan konsistensi penulisan dan menghindari perbedaan makna akibat penggunaan huruf kapital [10].
3. *Normalization*

   Normalisasi teks bertujuan menyederhanakan bentuk penulisan agar seragam. Proses ini mencakup penghapusan huruf berulang seperti "seeeenaaang" menjadi "senang", serta konversi kata tidak baku atau singkatan seperti "gk" menjadi "tidak" dan "sy" menjadi "saya" [11].
4. *Filtering (Stopword Removal)*

   Stopword removal adalah proses penghapusan kata-kata dengan makna rendah dalam

analisis, seperti konjungsi atau penghubung, yang sering muncul berulang dalam teks. *Stopword* diperkirakan mencakup 20–30% dari total dokumen. Penghapusan ini bertujuan meningkatkan efisiensi pemodelan dengan memfokuskan analisis pada kata-kata yang lebih bermakna [12].

5. *Tokenizing*

Tokenisasi merupakan proses memisahkan teks menjadi unit-unit terkecil yang bermakna, seperti kata, frasa, atau simbol tertentu. Tahap ini bertujuan untuk mengubah teks mentah menjadi daftar token sehingga dapat diproses oleh algoritma *machine learning* yang dapat dianalisis lebih lanjut [13].

6. *Stemming*

Stemming adalah proses praproses dalam pemrosesan bahasa alami *Natural Language Processing* (NLP) yang bertujuan mengembalikan kata-kata ke bentuk dasarnya atau akar katanya dengan menghilangkan imbuhan seperti awalan dan akhiran [10].

Labeling

Tahap labeling merupakan proses pemberian kelas pada data, di mana setiap data dikategorikan ke dalam label positif atau negatif sesuai dengan karakteristik isi yang dianalisis [14]. Penandaan sentimen dilakukan melalui dua pendekatan, yakni manual dan otomatis.

Splitting Data

Proses *splitting* data dalam klasifikasi bertujuan untuk memisahkan *dataset* menjadi dua *subset*, yaitu data latih dan data uji, yang masing-masing digunakan untuk membangun serta mengevaluasi performa model klasifikasi [15].

Ekstraksi Fitur dengan TF-IDF

Term Frequency-Inverse Document Frequency (TF-IDF) adalah metode untuk menghitung bobot kata berdasarkan frekuensinya dalam satu dokumen dibandingkan dengan seluruh korpus. Bobot ini diperoleh dari hasil perkalian antara TF dan IDF, di mana TF mengukur frekuensi kemunculan suatu kata dalam dokumen, dan IDF menilai kelangkaan kata tersebut di seluruh dokumen. TF-IDF bertujuan mengubah kata menjadi representasi numerik agar dapat diproses oleh algoritma machine learning [16]. Rumus TF-IDF dapat ditulis dalam persamaan 1,2 dan 3 [17].

- TF mengukur seberapa sering *term t* muncul dalam dokumen *d*, dibandingkan dengan total *term*.

$$TF(t,d) = \frac{\text{Jumlah } term\ t \text{ dalam } d}{\text{Total } term \text{ dalam } d} \quad (1)$$

- IDF menilai kelangkaan term *t* dalam seluruh dokumen. Nilai logaritmik digunakan untuk memberi bobot lebih tinggi pada *term* yang jarang muncul.

$$IDF(t,D) = \log\left(\frac{\text{Jumlah dokumen}}{\text{Dokumen yang memuat } t} + 1\right) \quad (2)$$

- TF-IDF merupakan hasil kali TF dan IDF yang menghasilkan bobot representatif bagi setiap *term* dalam dokumen.

$$TF - IDF(t,d,D) = TF(t,d) \times IDF(t,D) \quad (3)$$

Oversampling dengan *SMOTEN*

SMOTEN merupakan varian hibrida dari metode *SMOTE* yang tidak hanya melakukan *oversampling*, tetapi juga menghapus data dari kelas mayoritas yang berada dekat dengan kelas minoritas. Proses ini dimulai dengan mengidentifikasi dan menghilangkan *nearest*

neighbors dari kelas mayoritas, guna mengurangi potensi *noise*, sebelum menambahkan data sintetis pada kelas minoritas [18].

Modeling

1. *Naïve Bayes*

Algoritma *Naïve Bayes Classification* merupakan metode klasifikasi yang menggunakan data statistik untuk memprediksi probabilitas setiap anggota dalam suatu kelas. Algoritma *Naïve Bayes* terbukti memiliki kecepatan dan akurasi tinggi ketika diterapkan pada kumpulan data berukuran besar [19]. Rumus perhitungan model *Naïve Bayes* ditunjukkan dalam persamaan 4 [20].

$$P(H|X) = \frac{P(X|H)\ .\ P(H)}{P(X)} \quad (4)$$

Keterangan:
X : data sampel dengan kelas (label) yang tidak diketahui
H : hipotesis bahwa data X merupakan suatu kelas spesifik
P(H|X): peluang hipotesis H berdasarkan kondisi X (*posteriori probability*)
P(H) : peluang hipotesis H (*prior probability*)
P(X) : peluang data sampel yang diamati
P(X|H): peluang X berdasarkan kondisi hipotesis

2. *Support Vector Machine (SVM)*

Support Vector Machine (SVM) adalah algoritma pembelajaran mesin yang umum digunakan untuk klasifikasi dan regresi, dengan prinsip utama mencari *hyperplane* optimal yang memaksimalkan margin antar kelas [21]. Algoritma ini dipilih karena kemampuannya mengklasifikasikan data teks berdasarkan pola distribusi fitur numerik, seperti bobot kata dari TF-IDF, sehingga relevan untuk analisis sentimen. Dalam hal ini, rumus dasar untuk SVM linear dapat dinyatakan pada Persamaan 5 [22].

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

Dimana *F(x)* merupakan hasil prediksi, w menyatakan vektor normal terhadap *hyperplane*, *x* adalah vektor fitur input, dan b merupakan nilai bias (*intercept*) yang memengaruhi posisi *hyperplane* terhadap asal koordinat.

3. *Random Forest*

Algoritma *Random Forest Classifier* merupakan metode *homogeneous ensemble learning* yang membangun banyak *decision tree* melalui teknik *bootstrap aggregating (bagging)*, yaitu pengambilan sampel acak dengan penggantian [23]. Setiap pohon dibentuk dengan menghitung nilai *entropy* untuk mengukur ketidakmurnian atribut, lalu memilih atribut terbaik berdasarkan nilai *information gain*. Perhitungan *entropy* dan *information gain* ditunjukkan pada Persamaan 6 dan 7 [24].

$$Entropy\ (Y) = -\sum ip(c|Y)log^2p(c|Y), \quad (6)$$

Keterangan:
Y adalah himpunan data atau kasus,
p(c|Y) menyatakan proporsi data *Y* yang termasuk dalam kelas c.

Information Gain dari atribut *a* terhadap *Y* dapat dihitung dengan:

$$Information\ Gain\ (Y, a) = Entropy\ (Y) - \sum v \in Values(a) \frac{|Yv|}{|Ya|} Entropy\ (Yv) \quad (7)$$

Keterangan:
Values(a) adalah sekumpulan nilai yang mungkin dimiliki atribut a,
Y_v adalah *subset* dari *Y* yang memiliki nilai atribut a sebesar v,
Y_a merupakan seluruh data yang memiliki nilai pada atribut a.

Evaluasi

Tahapan akhir dalam penelitian ini adalah evaluasi. Evaluasi dilakukan dengan menghitung performa model menggunakan metrik seperti *accuracy, precision, recall, dan F1-score*. Metrik ini bertujuan untuk mengukur seberapa baik model dalam mengklasifikasikan data dengan benar [25]. Rumus yang digunakan untuk menghitung metrik tersebut adalah sebagai berikut [25].

$$Accuracy = \frac{(TP+TN)}{(TP+TN+FN)} \quad (8)$$

$$Precision = \frac{TP}{(TP+FP)} \quad (9)$$

$$Recall = \frac{TP}{(TP+FN)} \quad (10)$$

$$F1\text{-}score = \frac{(recallxpresisi)}{(presisi+recall)} \quad (11)$$

Di mana:

TP (*True Positive*) : Jumlah data positif yang terklasifikasi dengan benar.

TN (*True Negative*) : Jumlah data negatif yang terklasifikasi dengan benar.

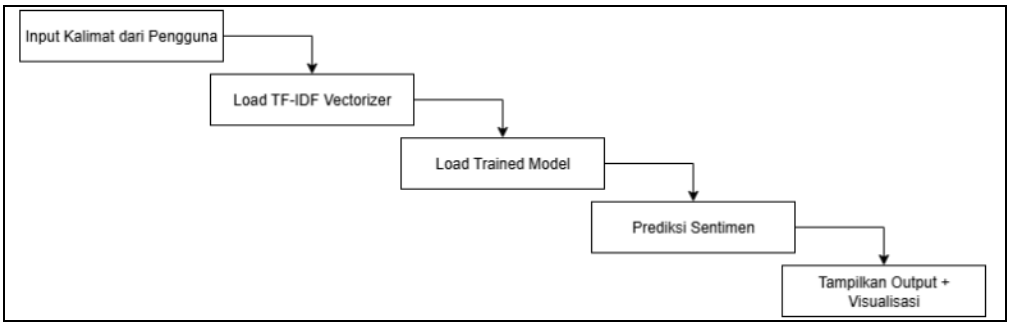
FP (*False Positive*) : Jumlah data negatif yang salah diklasifikasikan sebagai positif.

FN (*False Negative*) : Jumlah data positif yang salah diklasifikasikan sebagai negatif.

Pengembangan Aplikasi Web dengan *Flask*

Pengembangan aplikasi web dilakukan sebagai implementasi dari model klasifikasi sentimen yang sebelumnya telah dibangun dan dievaluasi menggunakan *Google Colab*. Seluruh proses awal mulai dari pengumpulan data hingga evaluasi dilakukan secara terpisah. Model *machine learning* dan file TF-IDF *vectorizer* yang telah dilatih disimpan dalam format *.pkl*, lalu dimuat ulang dalam aplikasi berbasis *Flask* untuk melakukan prediksi secara instan terhadap input baru dari pengguna. Dengan demikian, aplikasi ini hanya berfungsi sebagai alat inferensi, bukan pelatihan.

Flask sendiri merupakan *framework Python* yang ringan dan minimalis, dibangun dengan *Werkzeug* untuk manajemen permintaan dan *Jinja2* untuk antarmuka dinamis [26]. Melalui aplikasi ini, pengguna dapat memasukkan teks dan memperoleh hasil klasifikasi sentimen berdasarkan model yang telah dilatih. Alur proses web ditunjukkan pada Gambar 2.



Gambar 2. Alur proses dalam aplikasi web *flask*

HASIL DAN PEMBAHASAN

Pengumpulan Data

Pengumpulan data dilakukan menggunakan *Python* dengan *search query* berisi 14 kata kunci yang mencerminkan *stereotip* terhadap perempuan di dunia kerja. Proses *crawling* menghasilkan 2.336 tweet berbahasa Indonesia yang relevan, dikumpulkan dalam rentang 25

September 2010 hingga 13 Maret 2025. Data yang diambil terbatas pada isi teks cuitan tanpa menyertakan metadata lain, dan selanjutnya digunakan untuk proses klasifikasi sentimen.

Data Selection

Dari keseluruhan data yang awalnya terdiri dari 15 kolom, akan dilakukan seleksi variabel sesuai dengan kebutuhan penelitian. Kolom yang digunakan dalam penelitian ini adalah *full_text* dan *username*, sementara kolom lainnya dihapus karena tidak relevan dengan tujuan analisis.

Prepocessing

1. *Data Cleaning*

Dataset awal yang terdiri dari 2.336 data dibersihkan dengan menghapus duplikat dan data kosong, menyisakan 2.302 data. Proses *cleaning* dilanjutkan dengan menghilangkan karakter unik, tautan, tanda baca, spasi berlebih, *retweet, hashtag, mention*, dan karakter khusus dalam teks. Contoh hasil pembersihan ditampilkan pada Tabel 1.

Tabel 1. Contoh Hasil Data *Cleaning*

Sebelum	Sesudah
@Tita83079013 Klo menurut hukum syariah perempuan dilarang bekerja	Klo menurut hukum syariah perempuan dilarang bekerja

2. *Case Folding*

Case folding merupakan proses konversi seluruh karakter dalam teks menjadi huruf kecil guna menyeragamkan format penulisan. Proses ini bertujuan untuk meningkatkan konsistensi data sehingga lebih siap untuk dianalisis. Contoh hasil dari proses *case folding* dapat dilihat pada Tabel 2.

Tabel 2. Contoh Hasil *Case Folding*

Sebelum	Sesudah
Klo menurut hukum syariah perempuan dilarang bekerja	klo menurut hukum syariah perempuan dilarang bekerja

3. *Normalization*

Penelitian ini diawali dengan pembuatan kamus normalisasi untuk memastikan bahwa kata-kata yang diubah lebih sesuai dengan kebutuhan penelitian serta lebih akurat dalam konteks kalimat. Contoh hasil dari proses normalisasi disajikan pada Tabel 3.

Tabel 3. Contoh Hasil *Normalization*

Sebelum	Sesudah
klo menurut hukum syariah perempuan dilarang bekerja	kalo menurut hukum syariah perempuan dilarang bekerja

4. *Filtering (Stopword Removal)*

Stopword removal merupakan proses menghapus kata yang tidak memiliki makna signifikan guna mempertahankan kata relevan dalam dokumen. Seleksi dilakukan dengan menggabungkan daftar *stopword* dari NLTK dan Sastrawi sebagai acuan untuk menghilangkan kata umum yang tidak memuat informasi sentimen, sehingga meningkatkan akurasi

representasi data. Contoh hasilnya ditampilkan pada Tabel 4.

Tabel 4. Contoh Hasil *Filtering*

Sebelum	Sesudah
kalo menurut hukum syariah perempuan dilarang bekerja	kalo hukum syariah perempuan dilarang

5. *Tokenizing*

Tokenisasi merupakan proses pemisahan kalimat menjadi unit kata yang disebut *token*. Setiap *token* merepresentasikan kata atau elemen penting dalam teks yang dapat digunakan untuk analisis lebih lanjut. Proses ini memungkinkan identifikasi setiap kata secara terpisah, sehingga model dapat memahami pola serta hubungan antar kata dalam data. Hasil dari proses tokenisasi dapat dilihat pada Tabel 5.

Tabel 5. Contoh Hasil *Tokenizing*

Sebelum	Sesudah
kalo hukum syariah perempuan dilarang	['kalo', 'hukum', 'syariah', 'perempuan', 'dilarang']

6. *Stemming*

Stemming dilakukan untuk mengubah kata berimbuhan menjadi bentuk dasar menggunakan pustaka Sastrawi, guna menyederhanakan variasi kata dan mempermudah analisis sentimen. Setiap kata diproses secara individual lalu digabungkan kembali, dan hasilnya disimpan dalam format CSV, dikombinasikan dengan data lain, serta diperiksa agar bebas dari data kosong. Contoh hasil *stemming* ditampilkan pada Tabel 6.

Tabel 6. Contoh Hasil *Stemming*

Sebelum	Sesudah
['kalo', 'hukum', 'syariah', 'perempuan', 'dilarang']	kalo hukum syariah perempuan larang

Labeling

Tahap pelabelan sentimen dilakukan dengan dua pendekatan: otomatis menggunakan model Transformer dan manual melalui interpretasi kontekstual. Hasil pelabelan manual menunjukkan distribusi yang seimbang, dengan 1.106 tweet positif dan 1.196 negatif. Sebaliknya, pelabelan otomatis menghasilkan 479 tweet positif dan 1.823 negatif, menunjukkan kecenderungan bias terhadap sentimen negatif.

Spitting Data

Pembagian data dilakukan dengan fungsi *train_test_split*, menggunakan 75% data untuk pelatihan dan 25% untuk pengujian. Kolom "*tweet*" berfungsi sebagai fitur, sementara label berasal dari "*sentiments"* sesuai jenis pelabelan. Parameter *random_state* digunakan untuk menjaga konsistensi hasil.

Ekstraksi Fitur TF-IDF

Ekstraksi fitur dilakukan dengan metode TF-IDF (*Term Frequency–Inverse Document Frequency*) untuk mengubah data teks menjadi representasi numerik. Proses ini diterapkan

pada data berlabel manual dan otomatis, dengan TF-IDF di-*fit* pada data latih dan ditransformasikan ke data uji. Hasilnya, matriks manual berukuran 1.726 × 5.817 dengan 28.982 elemen non-nol, sementara matriks otomatis berukuran 1.726 × 5.816 dengan 28.196 elemen non-nol. Setiap baris merepresentasikan *tweet*, dan setiap kolom menunjukkan kata unik dari hasil praproses, di mana elemen non-nol menunjukkan term yang muncul dan memiliki bobot.

Oversampling dengan *SMOTEN*

Distribusi label pada data pelabelan otomatis tidak seimbang, dengan dominasi kelas negatif (1.823) dibanding positif (479), yang berpotensi menimbulkan bias model. Oleh karena itu, dilakukan *oversampling* data latih menggunakan metode *SMOTEN* untuk menyeimbangkan distribusi kelas melalui sintesis data minoritas berbasis kemiripan fitur. Sementara itu, data pelabelan manual tidak memerlukan *oversampling* karena distribusinya sudah seimbang.

Modeling

Penelitian ini menggunakan tiga algoritma klasifikasi, yaitu *Multinomial Naïve Bayes* (MNB), *Support Vector Machine* (SVM), dan *Random Forest*. Ketiga model dilatih pada data hasil pelabelan manual dan otomatis yang telah diekstraksi menggunakan TF-IDF. Untuk data otomatis, diterapkan *oversampling* dengan *SMOTEN* guna menyeimbangkan distribusi kelas, sementara data manual tidak memerlukan langkah ini karena sudah seimbang.

MNB dipilih karena efektif untuk data teks dan sederhana dalam asumsi independensi antar fitur. SVM digunakan karena kemampuannya memisahkan data berdimensi tinggi secara optimal, sedangkan *Random Forest* dipilih untuk keunggulannya dalam menangani data kompleks dan mengurangi overfitting melalui kombinasi pohon keputusan. Ketiga model kemudian dievaluasi menggunakan metrik *Accuracy, Precision, Recall, dan F1-score* untuk mengukur performa klasifikasi sentimen terhadap perempuan di lingkungan kerja.

Evaluasi

Bagian evaluasi ini menyajikan perbandingan performa tiga algoritma *Naïve Bayes*, *Support Vector Machine* (SVM), dan *Random Forest* berdasarkan metrik *accuracy, precision, recall, dan F1-score* pada dua jenis pelabelan, yakni otomatis dan manual. Tujuan dari perbandingan ini adalah untuk mengidentifikasi model yang tidak hanya akurat, tetapi juga seimbang dan andal dalam mengklasifikasikan kedua kelas sentimen, sebagaimana ditampilkan pada Tabel 7.

Tabel 7. Hasil Evaluasi 3 Algoritma Klasifikasi Sentimen pada l*abeling* Otomatis dan Manual

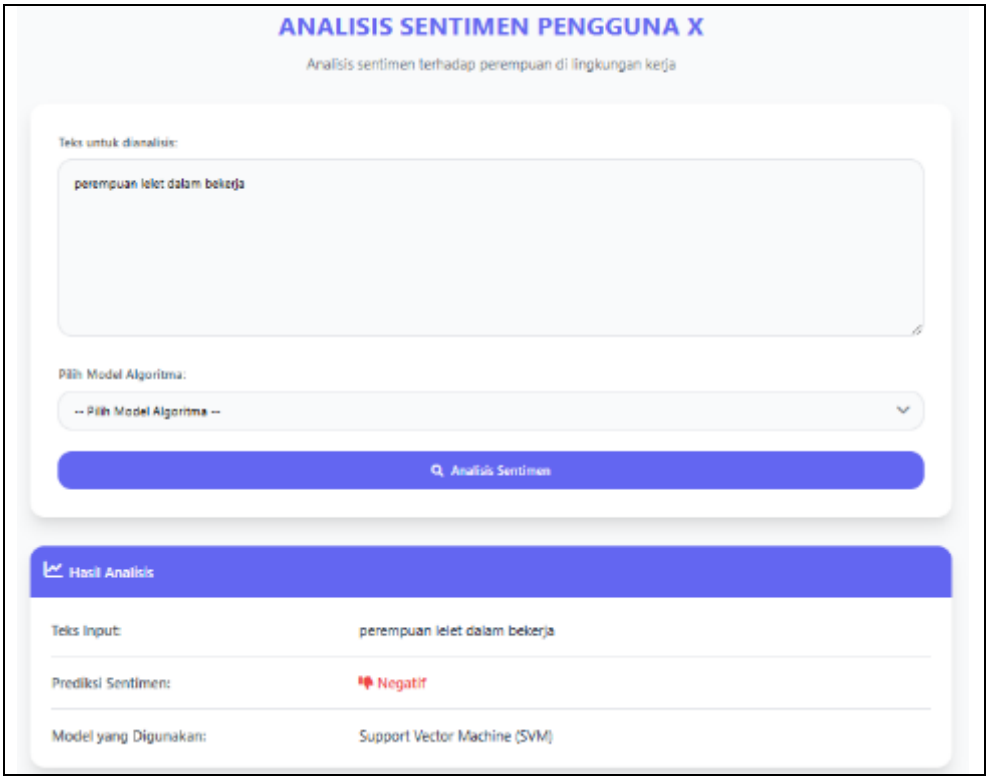
Algoritma	Jenis Labeling	*Accuracy*	*Precision* (Pos/Neg)	*Recall* (Pos/Neg)	*F1-Score* (Pos/ Neg)
Naïve Bayes	Otomatis	0.78	0.26/0.81	0.07/0.95	0.11/0.87
Naïve Bayes	Manual	0.59	0.63/0.56	0.41/0.76	0.50/0.65
SVM	Otomatis	0.74	0.30/0.82	0.24/0.86	0.26/0.84
SVM	Manual	0.59	0.60/0.58	0.55/0.63	0.57/0.60
Random Forest	Otomatis	0.79	0.28/0.81	0.04/0.97	0.08/0.88
Random Forest	Manual	0.57	0.61/0.55	0.40/0.74	0.49/0.63

Tabel 7 menyajikan hasil evaluasi performa tiga algoritma klasifikasi *Naïve Bayes, SVM,* dan *Random Forest* pada data yang dilabeli secara otomatis dan manual. Berdasarkan tabel, pelabelan otomatis cenderung menghasilkan akurasi yang lebih tinggi dibanding pelabelan

manual, dengan *Random Forest* mencapai akurasi tertinggi sebesar 0,79. Namun, performa tersebut tidak merata pada kedua kelas sentiment, terlihat dari nilai *F1-score* yang timpang antara kelas negatif dan positif, misalnya pada *Random Forest* otomatis, *F1-score* kelas negatif mencapai 0,88 sedangkan positif hanya 0,08. Sementara itu, algoritma pada pelabelan manual seperti *Naïve Bayes* dan SVM menunjukkan distribusi metrik yang lebih seimbang, walaupun dengan akurasi yang lebih rendah, masing-masing 0,59. Hal ini mengindikasikan bahwa meskipun pelabelan otomatis unggul dari segi akurasi, pelabelan manual menghasilkan distribusi metrik evaluasi yang lebih proporsional antara kelas, sehingga memberikan gambaran performa model yang lebih representatif.

Pengembangan Aplikasi Web dengan Flask

Aplikasi web berbasis *Flask* ini dikembangkan untuk menganalisis sentimen teks terkait perempuan di lingkungan kerja, sesuai fokus penelitian. Antarmuka aplikasi menyediakan kolom input teks dan opsi algoritma klasifikasi, yaitu *Naïve Bayes*, SVM, dan *Random Forest*. Setelah pengguna memilih algoritma dan menekan tombol "Analisis", aplikasi memproses teks menggunakan model dan *vectorizer* yang telah dilatih. Hasil ditampilkan dalam bentuk tabel berisi teks masukan dan prediksi sentimennya positif /negatif, seperti terlihat pada Gambar 3.



Gambar 3. Tampilan antarmuka aplikasi web Flask saat menampilkan hasil prediksi teks

KESIMPULAN DAN SARAN

Penelitian ini menganalisis sentimen pengguna Platform X terhadap perempuan di lingkungan kerja menggunakan pelabelan manual dan otomatis, serta tiga algoritma *machine learning: Naïve Bayes,* SVM*, dan Random Forest*. Hasil evaluasi menunjukkan bahwa pelabelan manual menghasilkan klasifikasi yang lebih seimbang meskipun akurasinya lebih rendah, sedangkan pelabelan otomatis, terutama dengan *DistilBERT*, menunjukkan bias terhadap sentimen negatif akibat sensitivitas model terhadap ekspresi negatif yang eksplisit. *Random Forest* mencatat akurasi tertinggi pada data otomatis (79%), namun disertai ketimpangan antar

kelas dengan *F1-score* 0,88 (negatif) dan 0,08 (positif). Sebaliknya, SVM dan *Naïve Bayes* pada data manual memberikan akurasi lebih rendah (59%) namun distribusi metriknya lebih stabil. Komparasi ini menegaskan bahwa akurasi tinggi tidak selalu menunjukkan performa yang adil, terutama pada data tidak seimbang. Ketimpangan antara *precision* dan *recall* menyebabkan turunnya *F1-score* pada kelas minoritas, menjadikan *F1-score* metrik yang lebih representatif dalam mengevaluasi keandalan model. Oleh karena itu, keseimbangan distribusi data, pemilihan algoritma, dan kepekaan terhadap konteks sosial perlu diperhatikan dalam membangun sistem analisis sentimen berbasis isu gender.

Penelitian ini memiliki potensi untuk dikembangkan lebih lanjut melalui perluasan *dataset* dari segi jumlah, variasi topik, serta keberagaman gaya bahasa dapat dipertimbangkan untuk meningkatkan representasi konteks sosial yang lebih luas. Penggunaan algoritma lain yang sesuai dengan karakteristik data juga dapat dieksplorasi lebih lanjut guna memperoleh hasil klasifikasi yang lebih akurat dan adaptif terhadap dinamika bahasa di media sosial. Aplikasi web yang telah dikembangkan juga dapat ditingkatkan dengan penambahan fitur visualisasi hasil yang lebih interaktif serta integrasi input secara langsung dari media sosial. Penelitian lanjutan diharapkan tidak hanya memetakan kecenderungan sentimen, tetapi juga menghasilkan rekomendasi berbasis data untuk mendorong kesetaraan gender dalam komunikasi digital dan ruang kerja profesional.

DAFTAR PUSTAKA

[1] Y. Afrillia, L. Rosnita, and D. Siska, "Analisis Sentimen Pengguna Twitter terhadap Isu Kesetaraan Gender dalam Penerapan Permendikbudristek Nomor 30 Tahun 2021 Menggunakan Textblob Analysis of Twitter User Sentiment Towards to Issue of Gender Equality in the Implementation of Permendikbudristek Number 30 of 2021 using Textblob," *Journal of Informatics and Computer Science*, vol. 8, no. 2, 2022.

[2] B. A. Yuniarossy, K. Maulida Hindrayani, and A. T. Damaliana, "Analisis Sentimen Terhadap Isu Feminisme di Twitter Menggunakan Model Convolutional Neural Network (CNN)," vol. 5, no. 1, 2024, doi: 10.46306/lb.v5i1.

[3] A. Muzakir, K. Adi, and R. Kusumaningrum, *Penerapan Konsep Machine Learning & Deep Learning*. 2024.

[4] A. Amri, "Implementasi Algoritma Random Forest untuk Mendeteksi Hate Speech dan Abusive Language pada Twitter Bahasa Indonesia," 2020.

[5] D. Shabira *et al.*, "MIND (Multimedia Artificial Intelligent Networking Database Deteksi Seksisme Online menggunakan Support Vector Machine dan Naïve Bayes," *Journal MIND Journal | ISSN*, vol. 8, no. 2, pp. 254–266, 2023, doi: 10.26760/mindjournal.v8i2.254-266.

[6] M. Ilmar Rifaldi, Y. Raymond Ramadhan, and I. Jaelani, "Analisis Sentimen terhadap Aplikasi Chatgpt pada Twitter Menggunakan Algoritma Naïve Bayes," 2023.

[7] F. Alghifari and D. Juardi, "Fauzan Alghifari Penerapan Data Mining pada Penerapan Data Mining pada Penjualan Makanan dan Minuman Menggunakan Metode Algoritma Naïve Bayes," 2021.

[8] H. Al, F.; Endang, and W. Pamungkas, "Pendeteksian Ujaran Seksisme pada Platform X dengan Algoritma Machine Learning Tradisional," 2024.

[9] B. Ramadhani and R. R. Suryono, "Komparasi Algoritma Naïve Bayes dan Logistic Regression untuk Analisis Sentimen Metaverse," *JURNAL MEDIA INFORMATIKA BUDIDARMA*, vol. 8, no. 2, p. 714, Apr. 2024, doi: 10.30865/mib.v8i2.7458.

[10] M. P. Firdaus, "Perbandingan Algoritma K-Nearest Neighbor (KNN) dan Naïve Bayes Classifier (NBC) dengan pelabelan Transformers serta Ektraksi Fitur TF-IDF dan N-Gram," 2023.

[11] S. S. Aljameel *et al.*, "A sentiment analysis approach to predict an individual's awareness of the precautionary procedures to prevent covid-19 outbreaks in Saudi Arabia," *Int J Environ Res Public Health*, vol. 18, no. 1, pp. 1–12, Jan. 2021, doi: 10.3390/ijerph18010218.

[12] T. H. J. Hidayat, Y. Ruldeviyani, A. R. Aditama, G. R. Madya, A. W. Nugraha, and M. W. Adisaputra, "Sentiment analysis of twitter data related to Rinca Island development using Doc2Vec and SVM and logistic regression as classifier," in *Procedia Computer Science*, Elsevier B.V., 2021, pp. 660–667. doi: 10.1016/j.procs.2021.12.187.

[13] S. Sohail and F. Amjad, "Investigation of Feminism Trends Through Sentiment Analysis Using Machine Learning and Natural Language Processing," 2024.

[14] I. Afdhal *et al.*, "Penerapan Algoritma Random Forest untuk Analisis Sentimen Komentar di YouTube Tentang Islamofobia," *Jurnal Nasional Komputasi dan Teknologi Informasi*, vol. 5, no. 1, 2022.

[15] F. Andy Kusuma and E. Wahyu Pamungkas, "Pendeteksian Hate Speech pada Sosial Media Indonesia dengan Algoritma Support Vector Machine (Svm) dan Decision Tree," 2022.

[16] F. Novianti and K. R. N. Wardani, "Analisis Sentimen Masyarakat terhadap Data Tweet Traveloka Selama Rapid Test Antigen Menggunakan Algoritma Naïve Bayes," *JIPI (Jurnal Ilmiah Penelitian dan Pembelajaran Informatika)*, vol. 8, no. 3, pp. 922–933, Aug. 2023, doi: 10.29100/jipi.v8i3.3973.

[17] P. Kumala Sari and R. Randy Suryono, "Komparasi Algoritma Support Vector Machine dan Random Forest untuk Analisis Sentimen Metaverse," 2024.

[18] A. Alabrah, "An Improved CCF Detector to Handle the Problem of Class Imbalance with Outlier Normalization Using IQR Method," *Sensors*, vol. 23, no. 9, May 2023, doi: 10.3390/s23094406.

[19] A. Ridwan, "Penerapan Algoritma Naïve Bayes untuk Klasifikasi Penyakit Diabetes Mellitus," 2020.

[20] J. Lemantara, "Penerapan Algoritma Naïve Bayes dan ID3 untuk Memprediksi Segmentasi Pelanggan pada Penjualan Mobil," *Journal of Technology and Informatics (JoTI)*, vol. 4, no. 1, pp. 31–40, Oct. 2022, doi: 10.37802/joti.v4i1.265.

[21] H. S. W. Wafa, A. I. Hadiana, and F. R. Umbara, "Prediksi Penyakit Diabetes Menggunakan Algoritma Support Vector Machine (SVM) INFORMASI ARTIKEL ABSTRAK," 2022. [Online]. Available: https://e-journal.unper.ac.id/index.php/informatics

[22] J. Ipmawati, S. Saifulloh, and K. Kusnawi, "Analisis Sentimen Tempat Wisata Berdasarkan Ulasan pada Google Maps Menggunakan Algoritma Support Vector Machine," *MALCOM: Indonesian Journal of Machine Learning and Computer Science*, vol. 4, no. 1, pp. 247–256, Jan. 2024, doi: 10.57152/malcom.v4i1.1066.

[23] H. Santoso, R. A. Putri, and S. Sahbandi, "Deteksi Komentar Cyberbullying pada Media Sosial Instagram Menggunakan Algoritma Random Forest," *Jurnal Manajemen Informatika (JAMIKA)*, vol. 13, no. 1, pp. 62–72, Mar. 2023, doi: 10.34010/jamika.v13i1.9303.

[24] G. Arther Sandag, "Prediksi Rating Aplikasi App Store Menggunakan Algoritma Random Forest Application Rating Prediction on App Store Using Random Forest Algorithm," *Cogito Smart Journal* |, vol. 6, no. 2, 2020, [Online]. Available: https://www.kaggle.com/

[25] R. Wirawan, "Analisa terhadap Tingkat Keaktifan Siswa pada Kegiatan Pembelajaran Sekolah dengan Metode Naïve Bayes," 2022.
[26] F. Dias Puspitasari and S. Anardani, "Aplikasi Klasifikasi Huruf Hijaiyah Menggunakan Algoritma Convolutional Neural Network dan Random Forest," *Seminar Nasional Teknologi Informasi dan Komunikasi*, 2023.