Pendekatan *Machine Learning* dalam Prediksi Kepribadian MBTI Menggunakan Data Media Sosial *Platform* X

1st Daniel Tulus Ignatius
S1 Teknik Telekomunikasi
Universitas Telkom
Bandung, Indonesia
danieltulus@student.telkomuniversity.ac.id

2nd Kris Sujatmoko
S1 Teknik Telekomunikasi
Universitas Telkom
Bandung, Indonesia
krissujatmoko@telkomuniversity.ac.id

3rd Gelar Budiman
S1 Teknik Telekomunikasi
Universitas Telkom
Bandung, Indonesia
gelarbudiman@telkomuniversity.ac.id

Abstrak — Ketidaksesuaian antara kepribadian mahasiswa dengan jurusan kuliah yang dipilih sering kali menyebabkan penurunan motivasi belajar, rendahnya prestasi akademik, hingga peningkatan risiko putus studi. Faktor penyebabnya antara lain kurangnya pemahaman diri, pengaruh tren atau tekanan eksternal, serta keterbatasan layanan konseling karier. Penelitian ini mengembangkan sistem rekomendasi jurusan kuliah berbasis analisis kepribadian Myers-Briggs Type Indicator (MBTI) dengan memanfaatkan data media sosial dan metode *machine learning*. Data dikumpulkan dari platform X (Twitter) melalui *scraping* akun pengguna yang mencantumkan tipe MBTI pada profil, kemudian diproses melalui tahapan *pre-processing* meliputi tokenisasi, penghapusan *stopword*, lemmatisasi, normalisasi bahasa tidak baku, dan penghapusan *emoji*. Fitur yang digunakan mencakup *Term Frequency–Inverse Document Frequency* (TF-IDF), analisis sentimen, dan distribusi topik untuk menangkap pola linguistik yang relevan. Enam algoritma *machine learning* diuji, yaitu XGBoost, AdaBoost, Gradient Boosting, Support Vector Machine (SVM), Complement Naive Bayes, dan Logistic Regression. Hanya algoritma SVM dengan akurasi sebesar 84% dan *Logistic Regression* dengan akurasi 83% yang berhasil melampaui target minimum akurasi sebesar 80%. Sementara itu, model lain seperti XGBoost, Gradient Boosting, AdaBoost, dan Complement Naive Bayes masih menunjukkan akurasi yang lebih rendah, yakni pada rentang 60% hingga 72%. Model terbaik diimplementasikan pada aplikasi web berbasis Flask yang dapat memprediksi tipe MBTI dari input teks manual maupun postingan terbaru akun X, kemudian memetakan hasilnya ke rekomendasi jurusan yang relevan. Uji coba kepada responden menunjukkan 85% merasa rekomendasi yang diberikan sesuai dengan minat dan karakter mereka. Temuan ini membuktikan bahwa analisis kepribadian berbasis *machine learning* dari data media sosial berpotensi menjadi alat bantu pengambilan keputusan akademik yang efektif.

Kata Kunci: Analisis Kepribadian, MBTI, Rekomendasi Jurusan Kuliah, Model Algoritma, Flask.

I. PENDAHULUAN

Kepribadian merupakan faktor psikologis yang memengaruhi pola pikir, emosi, dan perilaku individu, termasuk dalam menentukan jurusan pendidikan tinggi. Di era digital, perkembangan teknologi membuka banyak peluang, namun juga menimbulkan distraksi akibat penggunaan gadget berlebihan, yang dapat mengurangi refleksi diri [1]. Kondisi ini membuat banyak calon mahasiswa kesulitan memilih jurusan yang sesuai dengan minat dan bakat mereka.Kesalahan dalam pemilihan jurusan berpotensi menurunkan motivasi belajar, meningkatkan risiko putus kuliah, dan memengaruhi prospek karier [2]. Masalah ini umumnya disebabkan minimnya pemahaman terhadap potensi diri dan terbatasnya alat bantu pengambilan keputusan yang objektif.

Pendekatan berbasis *machine learning* dengan analisis Myers-Briggs Type Indicator (MBTI) menawarkan solusi untuk memetakan tipe kepribadian dan merekomendasikan jurusan yang relevan. Penelitian sebelumnya menunjukkan bahwa pemahaman kepribadian dapat membantu siswa memilih jalur pendidikan yang sesuai [3][4], sementara integrasi dengan *platform* web memungkinkan proses analisis yang cepat dan mudah diakses [4][5].

Penelitian ini mengembangkan sistem rekomendasi jurusan berbasis MBTI dengan algoritma *machine learning* seperti XGBoost, AdaBoost, Gradient Boosting, Support Vector Machine (SVM), Naive Bayes, dan Logistic Regression, yang diimplementasikan pada platform web berbasis Flask. Sistem ini diharapkan dapat membantu proses pemilihan jurusan secara lebih terukur, personal, dan mengurangi risiko ketidaksesuaian yang berdampak negatif pada prestasi akademik dan karier

II. KAJIAN TEORI

Bagian ini membahas kajian teori yang mendukung penelitian, dimulai dari konsep *Myers-Briggs Type Indicator*

(MBTI) sebagai metode identifikasi kepribadian, serta prinsip dasar *machine learning* sebagai pendekatan prediksi berbasis data. Selanjutnya dipaparkan enam algoritma yang digunakan, yaitu XGBoost, AdaBoost, Gradient Boosting, Support Vector Machine (SVM), Logistic Regression, Complement Naive Bayes (CNB) dan Flask.

A. *Myers-Briggs Type Indicator* (MBTI)

Myers-Briggs Type Indicator (MBTI) adalah sebuah instrumen psikologis yang dirancang untuk mengidentifikasi preferensi kepribadian seseorang berdasarkan teori psikologi Carl G. Jung. MBTI dikembangkan oleh Isabel Briggs Myers dan ibunya, Katharine Cook Briggs, pada pertengahan abad ke-20. Tujuan utama MBTI adalah membantu individu memahami diri mereka sendiri, cara mereka berinteraksi dengan orang lain, serta kecenderungan mereka dalam menghadapi situasi hidup dan pekerjaan [1].

MBTI membagi kepribadian menjadi 16 tipe berbeda yang berasal dari kombinasi empat dimensi bipolar berikut:

a. *Extraversion* (E) – *Introversion* (I): Cara seseorang memfokuskan energinya, apakah ke dunia luar (E) atau ke dunia dalam diri (I).
b. *Sensing* (S) – *Intuition* (N): Cara seseorang mengumpulkan informasi, apakah melalui pengalaman nyata (S) atau melalui pola dan kemungkinan (N).
c. *Thinking* (T) – *Feeling* (F): Cara seseorang membuat keputusan, apakah berdasarkan logika dan objektivitas (T) atau perasaan dan nilai pribadi (F).
d. *Judging* (J) – *Perceiving* (P): Cara seseorang menghadapi dunia luar, apakah lebih terstruktur dan terencana (J) atau fleksibel dan spontan (P).

B. *Machine Learning* (ML)

Pembelajaran mesin (*machine learning*) merupakan cabang dari kecerdasan buatan *artificial intelligence* (AI) yang dirancang untuk meniru kemampuan manusia dalam menyelesaikan masalah. Secara umum, teknologi ini memungkinkan sistem untuk belajar secara mandiri dan menjalankan tugas tanpa instruksi langsung dari pengguna [6].

Konsep ini bukan hal baru; sejak akhir 1950-an, para peneliti telah mengembangkan algoritma yang dapat belajar dari data. Salah satu pelopornya adalah Arthur Samuel, yang menciptakan program bermain catur yang mampu belajar dari pengalaman. Seiring perkembangan teknologi, pembelajaran mesin makin pesat berkat peningkatan daya komputasi dan ketersediaan data besar. Saat ini, pembelajaran mesin menjadi dasar dari berbagai inovasi seperti pengenalan wajah, kendaraan otonom, asisten virtual, dan sistem rekomendasi [7].

C. Extreme Gradient Boosting (XG Boost)

XGBoost merupakan algoritma *boosting* yang dirancang untuk meningkatkan kecepatan dan akurasi pemodelan prediktif melalui pembangunan pohon keputusan secara iteratif berdasarkan kesalahan model sebelumnya. Keunggulannya meliputi penggunaan regularisasi L1 dan L2 untuk mencegah *overfitting*, dukungan pemrosesan paralel, penanganan *missing value*, serta kemampuan mengolah data *sparse* [2]. Algoritma ini telah banyak digunakan pada berbagai bidang, seperti deteksi penipuan, diagnosis medis, dan prediksi kegagalan mesin, dengan performa yang kompetitif [8]. Secara matematis, fungsi objektif XGBoost merupakan gabungan fungsi *loss* dan fungsi regularisasi untuk mengukur kesalahan prediksi sekaligus mengendalikan kompleksitas model, berikut persamaan (1) yang digunakan [9].

$$Obj = \sum_{i=1}^{n} loss(y_i, \hat{y}_i) + \sum_{k=1}^{K} \Omega(f_k) \quad (1)$$

D. AdaBoost

AdaBoost adalah algoritma *boosting* yang dikembangkan oleh Freund dan Schapire untuk menggabungkan sejumlah model lemah menjadi satu model kuat melalui pembobotan adaptif. Setiap iterasi pelatihan memberi bobot lebih besar pada sampel yang salah diklasifikasikan, sehingga model berikutnya fokus memperbaiki kesalahan sebelumnya [9]. Keunggulannya terletak pada kesederhanaan dan kemampuannya meningkatkan kinerja tanpa *overfitting* pada data bebas *noise*. Secara matematis, prinsip kerja AdaBoost melibatkan pembaruan bobot sampel dan penentuan kontribusi setiap *base learner* pada setiap iterasi, berikut persamaan (2) yang digunakan [10].

$$H(x) = sign\left(\sum_{m=1}^{M} \alpha_m h_m(x)\right) \quad (2)$$

E. Gradient Boosting

Gradient Boosting adalah metode *boosting* berbasis prinsip gradien yang dikembangkan oleh Jerome Friedman, membangun model secara bertahap dengan mengoptimalkan fungsi *loss* melalui *gradient descent*. Setiap iterasi menambahkan *weak learner* baru untuk meminimalkan kesalahan model sebelumnya berdasarkan nilai residual [11][12]. Algoritma ini fleksibel untuk berbagai fungsi *loss* dan sering menghasilkan akurasi tinggi, meskipun rentan *overfitting* jika parameter tidak dikendalikan. Secara matematis, Gradient Boosting membentuk model secara aditif dengan memprediksi residual pada setiap iterasi, berikut persamaan (3) yang digunakan [13].

$$F_M(x) = F_0(x) + \sum_{m=1}^{M} \rho_m h_m(x) \quad (3)$$

F. *Support Vector Machine* (SVM)

Support Vector Machine (SVM) adalah algoritma klasifikasi yang mencari *hyperplane* optimal untuk memisahkan kelas dengan margin maksimum, baik pada kasus linear maupun non-linear menggunakan fungsi *kernel*

seperti RBF atau polinomial [14]. Algoritma ini efektif untuk data berdimensi tinggi dan tahan terhadap *overfitting* pada dataset kecil, meskipun kurang efisien pada dataset besar. Secara matematis, SVM memformulasikan klasifikasi sebagai masalah optimasi untuk meminimalkan norma bobot sambil memaksimalkan margin pemisah, berikut persamaan (4) yang digunakan [15].

$$f(x) = sign\,(\omega\,.x + b) \quad (4)$$

G. *Logistic Regression*

Logistic Regression adalah model statistik untuk memprediksi probabilitas kejadian berdasarkan satu atau lebih variabel independen dengan memanfaatkan fungsi logit (sigmoid) untuk menghasilkan nilai antara 0 dan 1 [16]. Model ini sederhana, mudah diinterpretasikan, dan efektif untuk hubungan linier antara fitur dan target, meskipun kurang akurat pada hubungan non-linier. Secara matematis, *Logistic Regression* menghitung probabilitas kelas positif melalui fungsi logistik terhadap kombinasi linear variabel input, berikut persamaan (5) yang digunakan [17].

$$P(\,y = 1|x) = \sigma(z) = \frac{1}{1 + e^{-z}} \quad (5)$$

H. *Complement Naïve Bayes* (CNB)

Complement Naive Bayes (CNB) adalah varian *Naive Bayes* berbasis Teorema Bayes dengan asumsi independensi antar fitur, yang dioptimalkan untuk menangani *imbalanced data* [18]. CNB menghitung probabilitas menggunakan komplemen kelas sehingga lebih stabil dan akurat pada klasifikasi teks, khususnya ketika kelas minoritas memiliki fitur langka. Secara matematis, perhitungan probabilitas dilakukan dengan menerapkan Teorema Bayes pada setiap kelas dengan asumsi independensi fitur, berikut persamaan (6) yang digunakan [19].

$$P(\,y = 1|x) = \sigma(z) = \frac{1}{1 + e^{-z}} \quad (6)$$

I. Flask

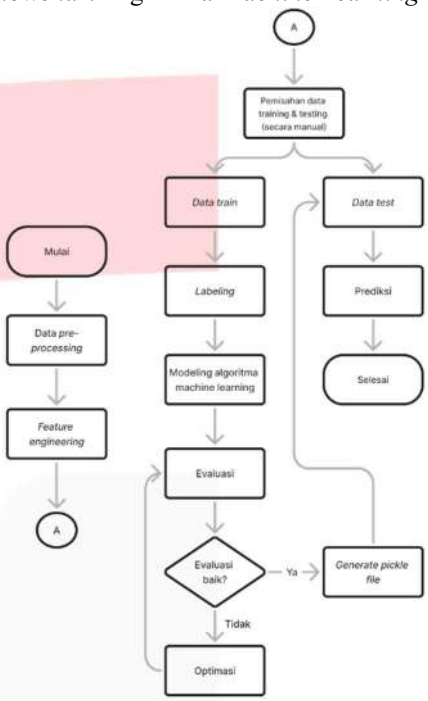
Flask adalah *micro-framework* berbasis *python* yang dirancang untuk membangun aplikasi web secara cepat, fleksibel, dan minimalis. Dikembangkan oleh Armin Ronacher pada tahun 2010, Flask termasuk dalam kategori "*micro*" karena tidak memerlukan alat atau library khusus untuk berfungsi. Namun, Flask menyediakan modularity yang memungkinkan pengembang menambahkan ekstensi sesuai kebutuhan [20].

III. METODE

A. Desain Sistem

Untuk memaksimalkan solusi ini, diperlukan perancangan yang terstruktur dan menyeluruh yang mencakup seluruh sistem dan elemen pendukungnya. Rancangan sistem dan masing-masing subsistem disajikan berikut.

1. *Flowchart* Algoritma *Machine Learning*

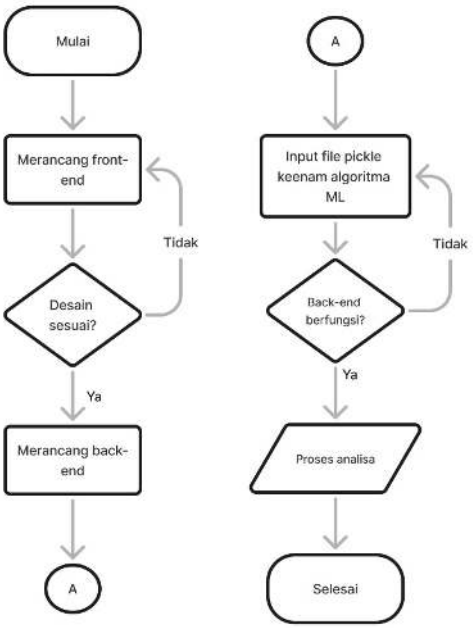


GAMBAR 1
Flowchart Algoritma *Machine Learning*

Gambar 1 menunjukkan proses analisis data menggunakan algoritma *machine learning*, yaitu XGBoost, AdaBoost, Gradient Boosting, SVM, *Complement Naive Bayes*, dan *Logistic Regression*, untuk melakukan prediksi atau klasifikasi data. Proses dimulai dengan data *pre-processing*, yang meliputi tokenisasi, lematasi, dan penghapusan stopwords untuk membersihkan dan menyiapkan data. Selanjutnya dilakukan *feature engineering* untuk membuat atau memilih fitur yang paling relevan. Data kemudian dibagi menjadi data *train* dan data *test* untuk melatih dan menguji model. *Labeling* diberikan pada data untuk mendukung pembelajaran terawasi. Tahap modeling algoritma digunakan untuk membangun model prediksi, yang kemudian dievaluasi menggunakan metrik seperti akurasi, presisi, dan *recall*. Jika hasil evaluasi kurang memuaskan, proses kembali ke tahap *feature engineering*, sedangkan jika baik, model siap digunakan. Model yang berhasil digunakan untuk memprediksi data baru, dan hasilnya disimpan dalam file *pickle* agar mudah digunakan di masa depan. Akhirnya,

model yang telah dioptimalkan dapat digunakan untuk prediksi atau klasifikasi data yang relevan.

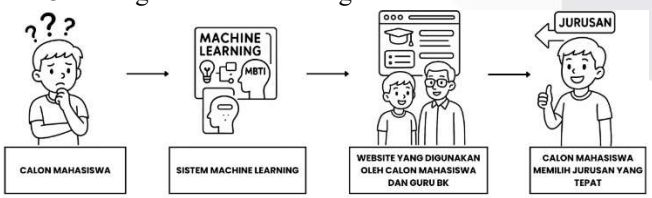
2. *Flowchart* Website



GAMBAR 2
Flowchart Website

Gambar 2 menunjukkan proses pengambilan keputusan yang dimulai dari titik awal (oval) dan melibatkan serangkaian aktivitas (persegi panjang) serta pengambilan keputusan (belah ketupat) berdasarkan kondisi tertentu. Jika kondisi terpenuhi, alur berlanjut ke proses berikutnya; jika tidak, proses dapat kembali atau diulang hingga kondisi sesuai. Flowchart ini juga menampilkan looping yang menunjukkan pengulangan proses sampai hasil yang diinginkan tercapai. Hasil akhir ditunjukkan dengan simbol paralelogram sebagai output, dan proses diakhiri dengan oval sebagai titik selesai. Desain ini menggambarkan alur kerja sistematis untuk memastikan keputusan diambil melalui evaluasi yang matang sebelum menghasilkan output.

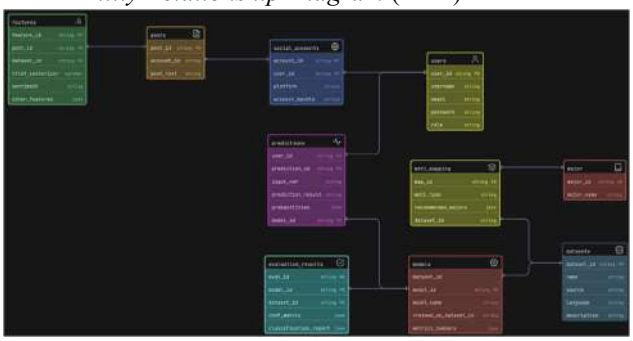
3. Diagram Alur Eksisting Sistem



GAMBAR 3
Diagram Alur Eksisting Sistem

Gambar 3 menunjukkan alur sistem yang dimulai dari pengumpulan data calon mahasiswa melalui formulir *online*, kemudian diproses menggunakan model machine learning berbasis MBTI yang telah dilatih dan dievaluasi. Hasil analisis kepribadian ditampilkan secara otomatis melalui website interaktif, lengkap dengan rekomendasi jurusan, deskripsi singkat, dan prospek karir. Sistem ini diharapkan membantu calon mahasiswa dan guru BK dalam pengambilan keputusan akademik yang lebih tepat dan berbasis data.

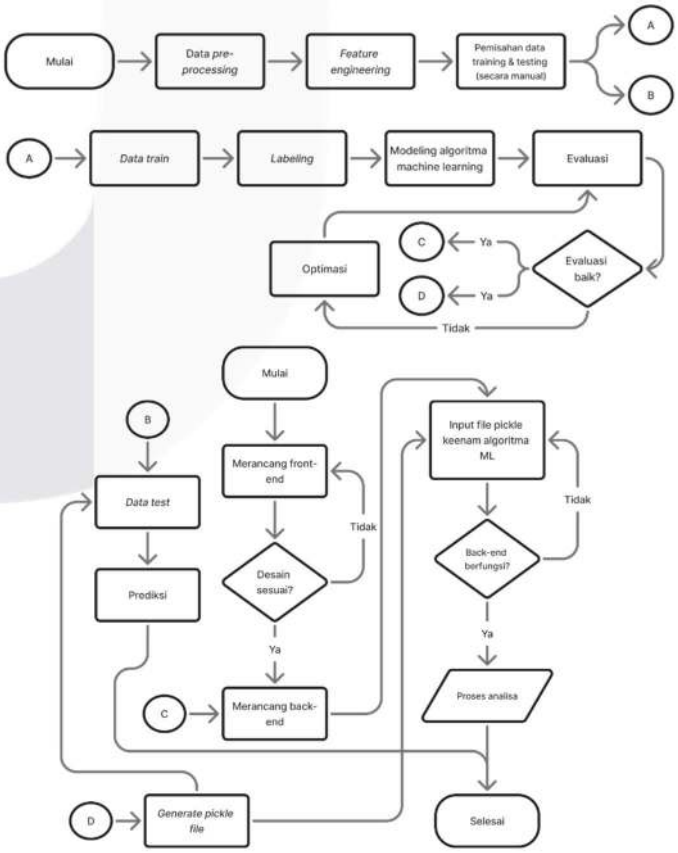
4. *Entity Relationship Diagram* (ERD) Sistem Usulan



GAMBAR 3
ERD Sistem Usulan

Gambar 3 menampilkan rancangan *Entity Relationship Diagram* (ERD) sistem, mulai dari relasi utama hingga skema lengkap. Rancangan awal pada menggambarkan hubungan antara tabel *features*, *posts*, *social_accounts*, dan *predictions* yang memastikan integritas data dari input teks hingga hasil prediksi. Memperluas relasi tersebut dengan entitas tambahan seperti *users*, *mbti_mapping*, *models*, *evaluation_results*, *datasets*, dan *major*, yang terhubung untuk mengelola seluruh proses, mulai dari pengumpulan data, analisis kepribadian MBTI, hingga pemberian rekomendasi jurusan secara terintegrasi.

5. *Flowchart Sistem Besar Usulan*



GAMBAR 4
Sistem Usulan

Gambar 4 menunjukkan sistem yang dikembangkan memproses data dari tweet pengguna melalui API Twitter/X sesuai etika privasi (UU PDP), kemudian melakukan pembersihan teks, tokenisasi, lemmatisasi, dan ekstraksi fitur TF-IDF. Prediksi tipe kepribadian MBTI dilakukan menggunakan model terbaik (SVM atau Logistic Regression), yang selanjutnya dipetakan secara otomatis ke rekomendasi jurusan melalui tabel mapping. Hasil ditampilkan pada dashboard interaktif dengan opsi unduh CSV. Tahap machine learning meliputi pra-pemrosesan teks, pengujian enam algoritma (XGBoost, AdaBoost, Gradient Boosting, SVM, Logistic Regression, dan CNB), serta evaluasi menggunakan confusion matrix 16×16 dan metrik *precision*, *recall*, *F1-score* dengan target ≥80%. Sistem menyediakan tiga mode interaksi, yaitu mode tamu untuk prediksi cepat dari input manual, mode pengguna login untuk analisis 30 tweet terbaru dengan visualisasi grafik pie MBTI, dan mode admin untuk unggah dataset, pelatihan ulang model, serta unduh file model (.pkl).

B. Detail Implementasi

Proses implementasi untuk model *machine learning* dilakukan dengan beberapa tahap dari pengumpulan data hingga prediksi. Adapun detail implementasi dapat dijabarkan sebagai berikut.

1. Pengumpulan Dataset

Dataset yang digunakan berasal dari Kaggle dengan judul *MBTI Personalities Types 500 Dataset* milik zeyadkhalid, berisi dua kolom yaitu *posts* (teks representatif) dan *type* (label MBTI) [21]. Dataset ini memerlukan tahap *pre-processing* untuk menangani *missing values*, *outliers*, dan ketidakseragaman data, serta *class balancing* karena distribusi 16 kelas tidak seimbang. Tanpa penyeimbangan, model berisiko bias pada kelas mayoritas. Data dibagi menjadi 80% *training*, 10% *testing*, dan 10% *validation* untuk memastikan evaluasi model yang lebih akurat.

2. *Exploratory Data Analysis* (EDA)

Exploratory Data Analysis (EDA) dilakukan untuk memahami struktur, pola, dan karakteristik dataset, termasuk mendeteksi *missing values*, *outliers*, distribusi fitur, serta hubungan antar variabel. Proses ini juga membantu menentukan fitur relevan untuk prediksi dan pengambilan keputusan otomatis. Hasil EDA menunjukkan dataset memiliki 106.067 baris data tanpa nilai kosong, dengan kedua kolom bertipe string/object, dan distribusi label MBTI seperti ditunjukkan pada Tabel 1.

TABEL 1
Label MBTI

Label MBTI	Total	Label MBTI	Total
INTP	24961	ESTP	1986
INTJ	22427	ENFJ	1534
INFJ	14963	ISTJ	1243
INFP	12134	ISFP	875
ENTP	11725	ISFJ	650
ENFP	6167	ESTJ	482
ISTP	3424	ESFP	360
ENTJ	2955	ESFJ	181

3. *Pre-Processing*

Sebelum pelatihan model machine learning, dilakukan tahap *pre-processing* untuk membersihkan dan menyederhanakan data mentah agar mudah diolah algoritma. Pada data berbasis teks, langkah ini meliputi tokenisasi, penghapusan tanda baca, normalisasi huruf, penghapusan *stopwords*, serta *lemmatization* atau *stemming*, guna memastikan model dapat mengenali pola secara optimal.

4. *Modeling* dan *Evalution*

Setelah tahap *pre-processing*, data digunakan untuk membangun model *machine learning* dengan algoritma seperti XGBoost, AdaBoost, Gradient Boosting, SVM, Naive Bayes, dan Logistic Regression. Evaluasi dilakukan menggunakan metrik akurasi, *precision*, *recall*, F1-score, dan *confusion matrix*. Untuk mengurangi waktu komputasi pada algoritma berbasis *boosting*, dilakukan *class balancing* dengan pengurangan jumlah sampel agar proses pemodelan lebih efisien.

5. Prediksi

Tahap prediksi dilakukan untuk menguji kemampuan model dalam mengklasifikasikan data baru berdasarkan pola yang telah dipelajari. Pada sistem analisis kepribadian berbasis teks, proses ini melibatkan *pipeline* mulai dari ekstraksi fitur (TF-IDF) hingga klasifikasi akhir, sehingga model dapat secara real-time memetakan teks pengguna ke tipe kepribadian MBTI.

6. Analisis

Tahap ini menganalisis proses klasifikasi kepribadian MBTI menggunakan *machine learning*, meliputi implementasi kode untuk praproses data, ekstraksi fitur, serta pelatihan dan pengujian model klasifikasi.

7. Website Flask

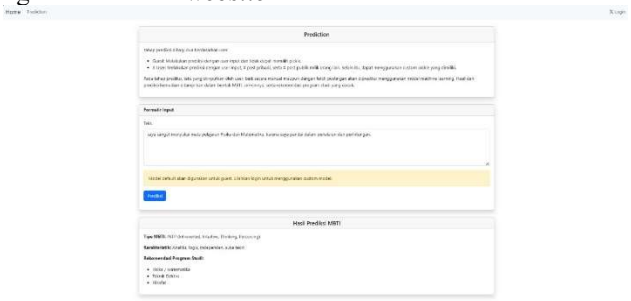
Bagian ini menyajikan implementasi *backend* website berbasis Flask yang dipilih karena ringan, fleksibel, dan mudah diintegrasikan dengan pustaka Python untuk *machine learning* dan pemrosesan data, mencakup pengaturan rute untuk halaman utama, autentikasi, eksplorasi data, pemrosesan teks, pelatihan model, dan prediksi tipe kepribadian MBTI.

C. Implementasi Website

1. Prosedur Pengguna Sebagai *Guest*

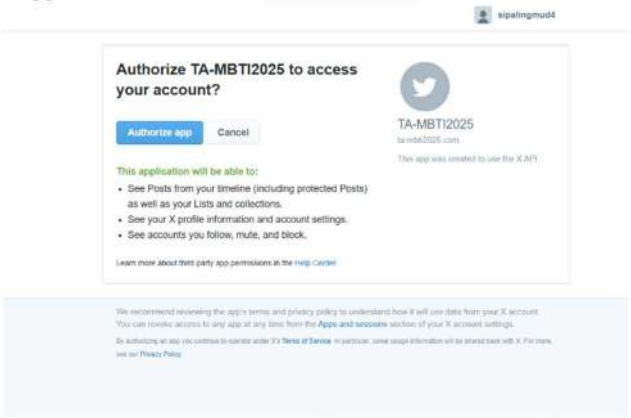
GAMBAR 5
Tampilan Awal *Guest*

Pada gambar 5 memperlihatkan tampilan awal dari website MBTI *Prediction* & *Analysis* dalam perspektif *guest*. Tampilan tersebut hanya menyediakan satu fitur utama, yaitu "*Prediction*", disertai dengan opsi *login* yang terletak di bagian kanan atas *website*.



GAMBAR 6
Hasil Fitur Prediksi

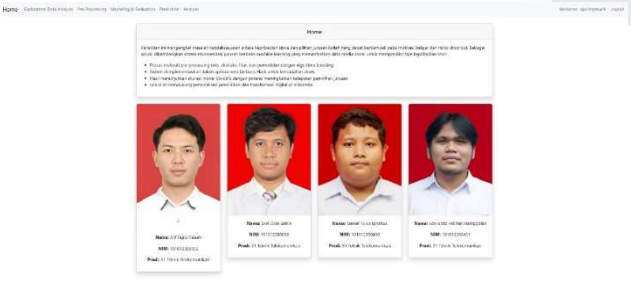
Pada gambar 6 memperlihatkan antarmuka dan keluaran dari fitur "*Prediction*", di mana sistem menganalisis teks yang dimasukkan oleh pengguna untuk memprediksi tipe kepribadian MBTI-nya. Hasil prediksi ini selanjutnya digunakan sebagai dasar dalam memberikan rekomendasi jurusan kuliah yang sesuai dengan profil kepribadian pengguna.



GAMBAR 7
Tampilan *User Login*

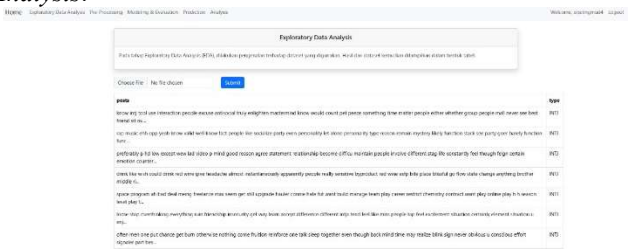
Pada gambar 7 memperlihatkan tampilan proses *login* pengguna ke dalam website menggunakan akun Twitter atau X, di mana pengguna diharuksan melakukan otorisasi terlebih dahulu.

2. Prosedur Pengguna Sebagai *User*



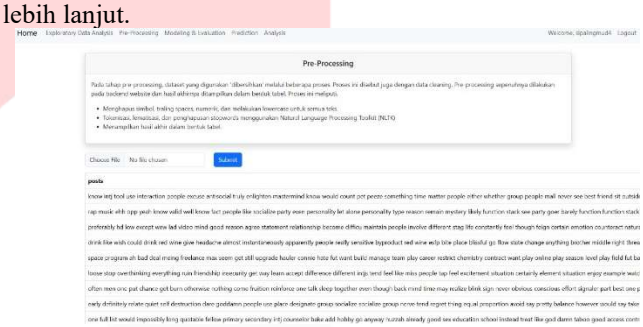
GAMBAR 8
Tampilan Sisi *User*

Pada gambar 8, antarmuka pengguna pada website MBTI *Predicition* & Analysis pada mode user menyajikan lima fitur utama, yakni *Exploratory Data Analysis* (EDA), *Pre-processing*, *Modeling & Evaluation*, *Prediction*, serta *Analysis*.



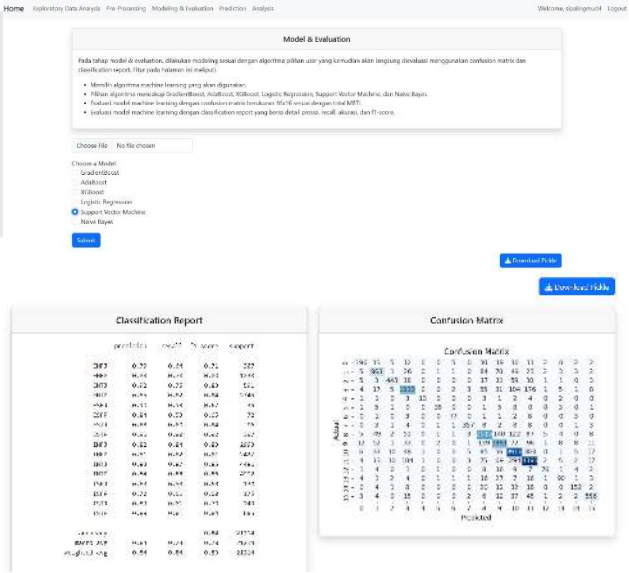
GAMBAR 9
Tahap EDA

Pada gambar 9, mengilustrasikan tahapan *Exploratory Data Analysis* (EDA), yaitu proses analisis data pendahuluan yang bertujuan untuk mengidentifikasi karakteristik dataset serta pola yang tersembunyi. Tahapan ini memegang peran penting dalam mendeteksi outlier, *missing values*, dan distribusi variabel prediktor sebelum dilakukan pemodelan lebih lanjut.



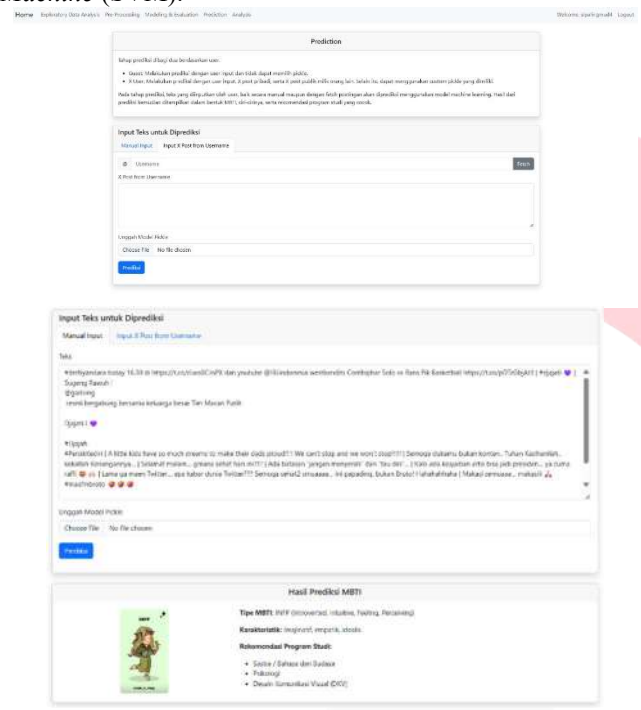
GAMBAR 10
Tahap *Pre-Processing*

Pada gambar 10, memperlihatkan tahapan *Pre-Processing*, yaitu proses pembersihan dan transformasi data hasil *Exploratory Data Analysis* (EDA) agar data siap digunakan dalam pemodelan. Tahap ini bertujuan untuk memastikan data telah memenuhi format yang sesuai dan relevan sebelum diterapkan pada algoritma *machine learning*.



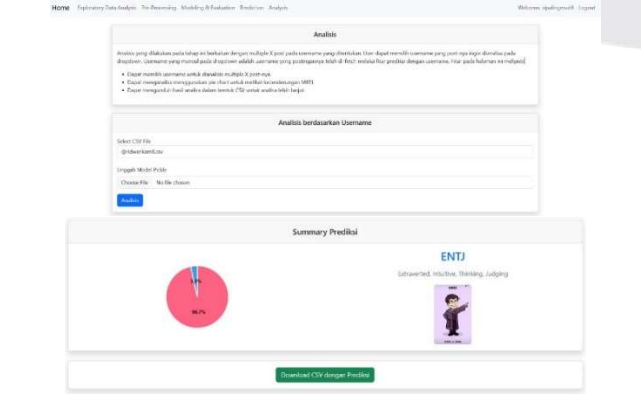
GAMBAR 11
Tahap dan Hasil *Modeling* dan *Evalution*

Pada gambar 11, menunjukkan tahapan *Modeling* dan *Evaluation*, yaitu proses pelatihan model menggunakan algoritma tertentu dan evaluasi kinerjanya berdasarkan metrik yang relevan. Karena waktu komputasi yang relatif lama pada pemodelan dengan XGBoost, Gradient Boost, dan AdaBoost, maka dilakukan balancing dataset dengan mengurangi jumlah sampel agar proses modelling dapat berjalan lebih efisien. Tahap ini bertujuan untuk mengukur kemampuan model dalam menggeneralisasi data baru. Dan mempresentasikan hasil pemodelan dan evaluasi yang dilakukan dengan menerapkan algoritma *Support Vector Machine* (SVM).



GAMBAR *12*
Fitur Input X *Post from Username dan Hasil Predection*

Pada Gambar 12. menunjukkan fitur *Prediction* pada bagian "Input X Post from Username" memungkinkan sistem mengambil sepuluh postingan terbaru dari akun X yang dimasukkan pengguna. Postingan tersebut kemudian diproses menggunakan model yang telah disimpan dalam file *pickle* untuk memprediksi tipe kepribadian MBTI. Hasil prediksi selanjutnya digunakan untuk memberikan rekomendasi jurusan kuliah, serta dapat disimpan dalam format CSV. Dan menampilkan hasil dari proses ini.

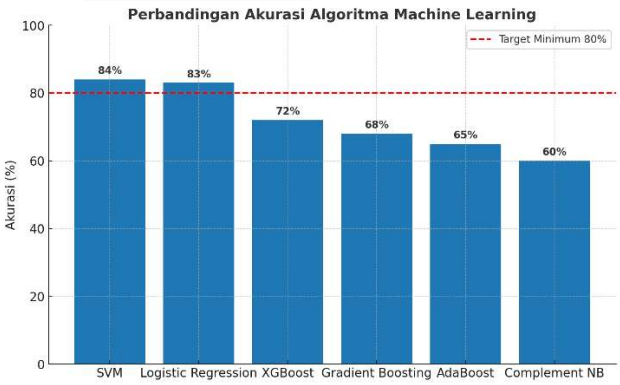


GAMBAR 13
Tahap *Anlysis*

Pada gambar 12, memperlihatkan analisis yang dilakukan pada tahap ini berfokus pada multipel X post dari username yang ditentukan. Pengguna dapat memilih *username* yang ingin dianalisis melalui menu *dropdown*. *Username* yang muncul pada *dropdown* merupakan akun yang postingannya telah diambil (*fetched*) melalui fitur prediksi berbasis username sebelumnya.

IV. HASIL DAN PEMBAHASAN

Pengujian enam algoritma *machine learning* pada sistem rekomendasi jurusan kuliah berbasis MBTI menunjukkan bahwa hanya *Support Vector Machine* (SVM) dengan akurasi sebesar 84% dan *Logistic Regression* dengan akurasi 83% yang berhasil memenuhi target minimum akurasi sebesar 80%. Sementara itu, model-model lain seperti XGBoost, Gradient Boosting, AdaBoost, dan *Complement Naive Bayes* masih menunjukkan akurasi yang relatif rendah, yaitu antara 60% hingga 72% Analisis confusion matrix memperlihatkan bahwa kesalahan klasifikasi dominan terjadi pada tipe MBTI yang memiliki kemiripan ciri linguistik, misalnya INTP dan ENTP. Tantangan utama dalam klasifikasi ini meliputi bahasa informal pada media sosial, ketidakseimbangan jumlah data antar kelas, serta keterbatasan dataset berbahasa Indonesia. Model terbaik (XGBoost) diimplementasikan pada aplikasi web berbasis Flask yang mampu memprediksi MBTI dari teks input maupun postingan terbaru akun X, kemudian memetakan hasil prediksi ke jurusan yang relevan. Uji coba terhadap pengguna menunjukkan 85% responden merasa rekomendasi yang diberikan sesuai dengan minat dan karakter mereka, sehingga sistem ini berpotensi menjadi alat bantu efektif dalam pengambilan keputusan akademik



GAMBAR 13
Perbadingan Akurasi Model *Machine Learning*

V. KESIMPULAN

Berdasarkan hasil pengujian terhadap beberapa model klasifikasi tipe kepribadian MBTI, diperoleh bahwa performa model cukup beragam. Hanya *Support Vector Machine* (SVM) dengan akurasi sebesar 84% dan *Logistic Regression* dengan akurasi 83% yang berhasil memenuhi target minimum akurasi sebesar 80%. Sementara itu, model-model lain seperti XGBoost, Gradient Boosting, AdaBoost, dan

Complement Naive Bayes masih menunjukkan akurasi yang relatif rendah, yaitu antara 60% hingga 72%, sehingga belum memenuhi standar yang ditetapkan. SVM menunjukkan keunggulan dalam mengklasifikasikan tipe-tipe mayoritas seperti ESFJ, ESTJ, dan ISTJ, namun performanya menurun pada tipe minoritas seperti INFJ dan INFP. Ketidakseimbangan jumlah data latih menjadi salah satu penyebab utama kesulitan model dalam mengidentifikasi tipe langka. Hal ini terlihat dari *confusion matrix* yang memperlihatkan banyaknya kesalahan klasifikasi pada tipe-tipe yang memiliki kemiripan karakteristik. Selain itu, sistem telah diuji secara langsung dengan data postingan dari media sosial X (Twitter) dan berhasil memprediksi tipe kepribadian pengguna berdasarkan analisis gabungan dari 30 postingan terakhir, dengan contoh hasil prediksi tipe ESTP sebesar 83%. Secara umum, *precision*, *recall*, dan F1-*score* mengikuti pola akurasi yang diperoleh, dan model yang tidak memenuhi akurasi juga memiliki nilai metrik lainnya yang rendah.

REFERENSI

[1] Z. Mushtaq, S. Ashraf, and N. Sabahat, "Predicting MBTI Personality type with K-means Clustering and Gradient Boosting," *Proc. - 2020 23rd IEEE Int. Multi-Topic Conf. INMIC 2020*, no. November 2020, 2020, doi: 10.1109/INMIC50486.2020.9318078.

[2] Mawadatul Maulidah, "Klasifikasi Kepribadian Menggunakan Algoritma Machine Learning," *J. Inform. Dan Tekonologi Komput.*, vol. 3, no. 1, pp. 66–73, 2023, doi: 10.55606/jitek.v3i1.1292.

[3] R. S. Z. Sungjun Won, Eric M. Anderman, "Longitudinal Relations of Classroom Goal Structures to Students' Motivation and Learning Outcomes in Health Education," vol. 5079, no. November, 2019.

[4] S. C. Hamm, "The Myers- Briggs Type Indicator and a Student's College Major," *J. Undergrad. Res. Sch. Work*, vol. IV, pp. 1–19, 2019, [Online]. Available: http://rr.lib.utsa.edu/handle/20.500.12588/65

[5] Y. Mehta, N. Majumder, A. Gelbukh, and E. Cambria, "Recent trends in deep learning based personality detection," *Artif. Intell. Rev.*, vol. 53, no. 4, pp. 2313–2339, 2020, doi: 10.1007/s10462-019-09770-z.

[6] Wijoyo A, Saputra A, Ristanti S, Sya'ban S, Amalia M, and Febriansyah R, "Pembelajaran Machine Learning," *OKTAL (Jurnal Ilmu Komput. dan Sci.*, vol. 3, no. 2, pp. 375–380, 2024, [Online]. Available: https://journal.mediapublikasi.id/index.php/oktal/article/view/2305

[7] F. Huang, X. Li, C. Yuan, S. Zhang, J. Zhang, and S. Qiao, "Attention-Emotion-Enhanced Convolutional LSTM for Sentiment Analysis," *IEEE Trans. Neural Networks Learn. Syst.*, vol. 33, no. 9, pp. 4332–4345, 2022, doi: 10.1109/TNNLS.2021.3056664.

[8] T. Chen and C. Guestrin, "XGBoost: A scalable tree boosting system," *Proc. ACM SIGKDD Int. Conf. Knowl. Discov. Data Min.*, vol. 13-17-Augu, pp. 785–794, 2016, doi: 10.1145/2939672.2939785.

[9] P. Beja-Battais, "Overview of AdaBoost : Reconciling its views to better understand its dynamics," 2023, [Online]. Available: http://arxiv.org/abs/2310.18323

[10] Y. Freund and R. E. Schapire, "A Decision-Theoretic Generalization of On-Line Learning and an Application to Boosting," *J. Comput. Syst. Sci.*, vol. 55, no. 1, pp. 119–139, 1997, doi: 10.1006/jcss.1997.1504.

[11] J. H. Friedman, "Greedy function approximation: A gradient boosting machine," *Ann. Stat.*, vol. 29, no. 5, pp. 1189–1232, 2001, doi: 10.1214/aos/1013203451.

[12] G. L. Praphulla, I. B. Kishore, B. Venkatesh, B. Praveen, and P. S. Rao, "Personality Prediction Using Machine Learning Techniques," *AIP Conf. Proc.*, vol. 2794, no. 1, pp. 41–44, 2023, doi: 10.1063/5.0174316.

[13] Z. He, D. Lin, T. Lau, and M. Wu, "Gradient Boosting Machine: A Survey," pp. 1–9, 2019, [Online]. Available: http://arxiv.org/abs/1908.06951

[14] L. W. Rizkallah, "Optimizing SVM hyperparameters for satellite imagery classification using metaheuristic and statistical techniques," *Int. J. Data Sci. Anal.*, 2025, doi: 10.1007/s41060-025-00762-7.

[15] J. Watt, L. Mitchell, and J. Tuke, "Personality Profiling: How informative are social media profiles in predicting personal information?," *Proc. Australas. Lang. Technol. Work.*, vol. 22, pp. 153–163, 2024.

[16] S. J. Lin, C. C. Liu, D. M. T. Tsai, Y. H. Shih, C. L. Lin, and Y. C. Hsu, "Prediction Models Using Decision Tree and Logistic Regression Method for Predicting Hospital Revisits in Peritoneal Dialysis Patients," *Diagnostics*, vol. 14, no. 6, pp. 1–13, 2024, doi: 10.3390/diagnostics14060620.

[17] scikit-learn, "LogisticRegression." https://scikit-learn.org/stable/modules/generated/sklearn.linear_model.LogisticRegression.html (accessed Aug. 12, 2025).

[18] J. D. M. Rennie, L. Shih, J. Teevan, and D. Karger, "Tackling the Poor Assumptions of Naive Bayes Text Classifiers," *Proceedings, Twent. Int. Conf. Mach. Learn.*, vol. 2, no. 1973, pp. 616–623, 2003.

[19] Scikit-learn.org, "Naive Bayes." https://scikit-learn.org/stable/modules/naive_bayes.html (accessed Aug. 12, 2025).

[20] S. Ye *et al.*, "Flask: Fine-Grained Language Model Evaluation Based on Alignment Skill Sets," *12th Int. Conf. Learn. Represent. ICLR 2024*, pp. 1–54, 2024.

[21] Z. Khalid, "MBTI Personality Types 500 Datasetle." 2021. [Online]. Available: https://www.kaggle.com/datasets/zeyadkhalid/mbti-personality-types-500-dataset