

BAB II

LANDASAN TEORI

2.1 Merdeka Belajar Kampus Merdeka (MBKM)

Merdeka Belajar Kampus Merdeka (MBKM), merupakan kebijakan dari Kementerian Pendidikan, Kebudayaan, Riset, dan Teknologi Republik Indonesia yang bertujuan memberikan kebebasan kepada mahasiswa untuk belajar di luar program studi guna mengembangkan potensi diri secara maksimal (Bhakti et al., 2022). Program ini mencakup berbagai bentuk kegiatan seperti pertukaran pelajar, magang, studi independen, penelitian, kampus mengajar, kewirausahaan, dan proyek kemanusiaan. Implementasi program ini menuntut adanya sistem administrasi yang responsif, terstruktur, dan efisien untuk mendukung kebutuhan mahasiswa dan institusi.

Tujuan utama dari MBKM adalah memberikan pengalaman belajar yang lebih kontekstual, adaptif terhadap dunia kerja, dan memperluas wawasan mahasiswa lintas disiplin ilmu. Hal ini sesuai dengan Permendikbud No. 3 Tahun 2020 tentang Standar Nasional Pendidikan Tinggi, yang menyatakan bahwa mahasiswa memiliki hak untuk belajar di luar program studi selama tiga semester, termasuk satu semester di luar perguruan tinggi asal (Kuncoro et al., 2022).

Seiring dengan penerapan program MBKM, dibutuhkan sistem administrasi yang responsif dan efisien dalam pengelolaan informasi. Administrasi MBKM meliputi seluruh proses layanan seperti pendaftaran program, pemilihan jenis kegiatan (misalnya magang atau pertukaran pelajar), pemrosesan dokumen, pemantauan pelaksanaan program, hingga konversi nilai. Seluruh tahapan ini menuntut adanya layanan yang terstruktur, informatif, dan mudah diakses oleh mahasiswa (Bhakti et al., 2022).

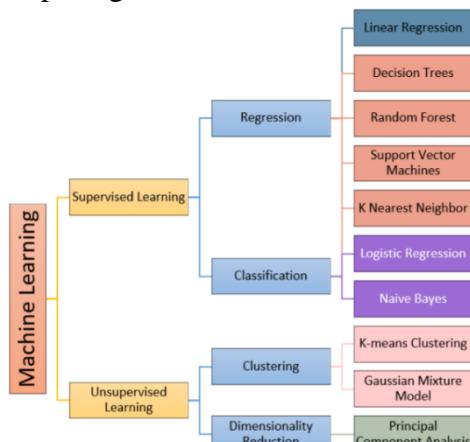
Namun, dalam praktiknya, proses administrasi MBKM di banyak institusi pendidikan tinggi, termasuk di Institut Informatika dan Bisnis Darmajaya, masih menghadapi tantangan seperti keterbatasan tenaga administratif, waktu respon yang lambat, dan kurangnya sistem otomatisasi pelayanan informasi. Oleh karena itu,

diperlukan solusi berbasis teknologi untuk mendukung transformasi digital dalam pelayanan administrasi MBKM agar informasi dapat tersampaikan dengan lebih cepat, akurat, dan efisien (Kuncoro et al., 2022). Saat ini program MBKM telah berubah menjadi Kampus berdampak yang memiliki program Magang Berdampak

2.2 Machine Learning

Machine Learning (ML) merupakan cabang dari *Artificial Intelligence* yang memanfaatkan teknik statistik dan algoritma untuk memungkinkan mesin mengambil keputusan layaknya manusia berdasarkan pengalaman dari analisis data dalam jumlah besar (Faisal et al., 2025). ML telah menjadi teknologi penting dalam berbagai bidang seperti *Speech Recognition*, *Natural Language Processing* (NLP), dan *Computer Vision*.

Pada *Speech Recognition*, ML memungkinkan komputer untuk mentranskripsi ucapan dan mengenali instruksi suara, yang kini digunakan pada asisten virtual, perangkat kendali suara, layanan transkripsi, hingga komunikasi hands-free. Pada *Natural Language Processing* NLP, ML mendukung berbagai aplikasi seperti analisis sentimen, *topic modeling*, *chatbot*, penerjemahan bahasa, deteksi iklan palsu, dan pembuatan teks otomatis. Sementara itu, dalam *Computer Vision*, ML dapat digunakan untuk deteksi objek, klasifikasi gambar, pengenalan gerakan tubuh, dan manajemen lalu lintas (Faisal et al., 2025). (Faisal et al., 2025) menjelaskan bahwa *Machine Learning* terdiri atas dua kategori utama, yaitu *Supervised Learning* dan *Unsupervised Learning*. Ilustrasi skema Machine Learning dapat dilihat pada gambar 2.1 berikut:



Gambar 2. 1 Taksonomi Model Machine Learning

Sumber: (Faisal et al., 2025)

Supervised Learning adalah metode pembelajaran mesin yang menggunakan data berlabel, artinya setiap data latih (*training data*) memiliki pasangan antara *input* dan *output* atau nilai target yang ingin diprediksi. Dalam proses ini, model dilatih untuk mempelajari hubungan atau fungsi yang memetakan input ke output berdasarkan contoh-contoh yang telah disediakan (Faisal et al., 2025). Tujuan utama dari pendekatan ini adalah menghasilkan model prediksi yang akurat untuk data baru yang belum pernah dilihat sebelumnya.

Supervised Learning umumnya digunakan dalam dua jenis masalah utama, yaitu klasifikasi dan regresi. Pada masalah klasifikasi, variabel keluaran bersifat diskrit, seperti menentukan apakah email termasuk kategori *spam* atau *non-spam*. Sedangkan pada masalah regresi, variabel keluaran bersifat kontinu, seperti memprediksi harga rumah berdasarkan ukuran dan lokasi (Rahaman, 2024).

Kelebihan utama dari Supervised Learning adalah kemampuannya menghasilkan prediksi yang akurat apabila dilatih dengan data yang memadai dan representatif. Namun, metode ini memerlukan dataset berlabel yang besar, yang seringkali sulit dan mahal untuk dikumpulkan.

Unsupervised Learning adalah metode pembelajaran mesin yang digunakan untuk menganalisis data yang tidak memiliki label atau kategori yang telah ditentukan sebelumnya. Tidak seperti supervised learning yang membutuhkan output sebagai panduan, unsupervised learning bertujuan menemukan pola, struktur, atau hubungan tersembunyi di dalam data tanpa bantuan label (Rahaman, 2024).

Metode ini umum digunakan untuk tugas seperti *clustering*, *dimensionality reduction*, *association rule mining*, dan *anomaly detection*. Tantangan utama dalam unsupervised learning adalah sulitnya mengevaluasi hasil, karena tidak ada ground truth atau jawaban benar yang dapat dijadikan pembanding (Nurhalizah et al., 2024).

Kelebihan unsupervised learning adalah tidak memerlukan data berlabel, sehingga dapat digunakan pada data mentah yang jumlahnya besar. Namun, kelemahannya adalah interpretasi hasil yang lebih sulit dan kebutuhan untuk memilih algoritma serta parameter yang tepat agar pola yang ditemukan benar-benar relevan.

Secara keseluruhan, baik supervised maupun unsupervised learning memiliki peran penting dalam pengembangan sistem berbasis kecerdasan buatan, termasuk pada penelitian ini yang memanfaatkan algoritma supervised learning, yaitu Multinomial Naïve Bayes, untuk mengklasifikasikan pertanyaan mahasiswa terkait layanan administrasi MBKM.

2.3 Chatbot

Chatbot merupakan perangkat lunak yang dirancang untuk meniru interaksi dengan pengguna melalui media teks atau suara. Teknologi ini memungkinkan terjadinya interaksi dua arah antara manusia dan sistem komputer, di mana *chatbot* dapat memberikan informasi, menjawab pertanyaan, dan melakukan tindakan tertentu sesuai dengan pengetahuan atau data yang telah diprogram sebelumnya (Nurmahya & Diana, 2023). Dalam beberapa tahun terakhir, penggunaan *chatbot* semakin meluas dalam berbagai sektor seperti pelayanan publik, pendidikan, bisnis, kesehatan, hingga perbankan, karena kemampuannya dalam memberikan layanan yang cepat, efisien, dan dapat diakses 24 jam (Heryati et al., 2023).

Secara umum, *chatbot* dapat dikelompokkan menjadi dua kategori utama berdasarkan pendekatan pengembangannya. Pertama, *rule-based chatbot*, yaitu *chatbot* yang beroperasi mengikuti alur atau aturan tertentu yang telah ditentukan oleh pengembang. Jenis ini relatif lebih mudah dikembangkan, stabil, dan cocok digunakan untuk melayani pertanyaan yang bersifat berulang serta terstruktur, seperti *Frequently Asked Questions* (FAQ). Kedua, *chatbot* berbasis *machine learning* atau *AI-based chatbot*, yang memanfaatkan teknologi kecerdasan buatan, khususnya *Natural Language Processing* (NLP), untuk memahami konteks, maksud, dan struktur bahasa alami pengguna. Jenis ini memiliki fleksibilitas lebih tinggi dan mampu belajar dari data untuk menangani percakapan yang lebih kompleks, namun pengembangannya memerlukan dataset berukuran besar, sumber daya komputasi, dan waktu pelatihan yang lebih panjang sehingga tidak selalu cocok untuk kebutuhan institusi dengan pola pertanyaan yang relatif konsisten (Deden Moh Alfiansyah et al., 2025).

2.4 Natural Language Processing (NLP)

Natural Language Processing (NLP) atau Pemrosesan Bahasa Alami merupakan salah satu cabang dari *Artificial Intelligence* (AI) yang berfokus pada interaksi antara komputer dan bahasa manusia. NLP memberikan kemampuan kepada komputer untuk memproses bahasa yang digunakan oleh manusia dalam percakapan sehari-hari. Kemampuan ini menjadi dasar bagi *chatbot* untuk memahami pesan dari pengguna, bahkan ketika terdapat kesalahan ejaan atau tata bahasa, dan kemudian memberikan respons yang tepat. NLP memegang peranan krusial karena menjadi fondasi bagi sistem dalam memproses pertanyaan pengguna yang tidak terstruktur dan menghasilkan jawaban yang relevan (Hikmah et al., 2023).

Agar komputer dapat mengolah data teks secara efektif, terutama sebelum digunakan oleh algoritma *machine learning*, diperlukan serangkaian tahapan yang dikenal sebagai *text preprocessing*. Proses ini merupakan bagian dari teknik NLP yang bertujuan menerjemahkan bahasa manusia menjadi format yang dapat dimengerti oleh komputer. Tujuan utamanya adalah membersihkan dan mengubah data teks yang tidak terstruktur menjadi format terstruktur yang siap dianalisis lebih lanjut, sehingga secara signifikan mempengaruhi kualitas hasil pada tahap klasifikasi teks (Muhidin et al., 2023).

2.3.1 Tahapan Preprocessing Teks

Preprocessing merupakan tahapan penting dalam NLP untuk mempersiapkan data teks sebelum diproses oleh algoritma machine learnin. Tahapan *preprocessing* yang umum digunakan meliputi:

1. *Tokenizing* (Tokenisasi)

Tokenisasi merupakan proses memecah teks panjang menjadi unit-unit yang lebih kecil seperti kata atau frasa. Proses ini biasanya menggunakan spasi sebagai pemisah untuk menghasilkan kata-kata yang berdiri sendiri, sehingga dapat dianalisis secara individual.

2. *Case Folding*

Tahap ini bertujuan untuk menyeragamkan teks dengan mengubah seluruh karakter dalam dokumen menjadi huruf kecil (lowercase). Proses ini diperlukan untuk menghindari ambiguitas sistem terhadap kata yang identik. (Muhidin et al., 2023).

3. *Stopword*

Tahap ini bertujuan untuk menghilangkan kata-kata umum yang memiliki frekuensi tinggi namun tidak memberikan kontribusi signifikan terhadap makna semantik kalimat. *Stopwords* dalam Bahasa Indonesia meliputi kata-kata seperti "yang", "di", "dan", "adalah", "untuk" (Muhidin et al., 2023).

4. *Stemming*

merupakan tahapan untuk menemukan bentuk dasar dari suatu kata. Proses ini bertujuan menghapus variasi morfologis yang menempel pada kata tersebut, seperti awalan atau akhiran, sehingga menghasilkan kata dasar yang sesuai dengan kaidah morfologi dalam bahasa Indonesia (Muhidin et al., 2023).

2.3.2 Ekstraksi Fitur Teks

Komputer tidak dapat memahami teks dalam bentuk mentah secara langsung. Oleh karena itu, data teks harus dikonversi terlebih dahulu ke dalam bentuk representasi numerik agar dapat diproses oleh algoritma pembelajaran mesin. Proses konversi ini dikenal sebagai ekstraksi fitur teks (*text feature extraction*), dan salah satu metode yang paling umum digunakan adalah *Term Frequency-Inverse Document Frequency* (TF-IDF) (Muhidin et al., 2023).

TF-IDF adalah metode yang digunakan untuk menilai tingkat kepentingan suatu kata dalam sebuah dokumen relatif terhadap seluruh kumpulan dokumen. Prinsip dasarnya adalah: semakin sering sebuah kata muncul dalam suatu dokumen tetapi jarang muncul dalam dokumen lainnya, maka kata tersebut dianggap lebih penting dan akan diberikan bobot yang lebih tinggi (Muhidin et al., 2023).

Term Frequency (TF) mengukur seberapa sering suatu kata (*term*) muncul dalam sebuah dokumen. Nilai TF dihitung menggunakan rumus

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

TF-IDF digunakan untuk mengukur pentingnya suatu kata dalam sebuah dokumen relatif terhadap seluruh kumpulan dokumen. Semakin sering suatu kata muncul dalam sebuah dokumen, namun jarang muncul di dokumen lain, maka kata tersebut dianggap lebih penting dan memiliki bobot tinggi.

2.5 Multinomial Naive Bayes

Multinomial Naive Bayes merupakan salah satu algoritma dalam pembelajaran mesin (machine learning) yang secara luas digunakan untuk tugas klasifikasi teks. Algoritma ini merupakan varian dari Naive Bayes yang menggunakan model probabilistik multinomial, yaitu dengan menghitung probabilitas kemunculan kata dalam dokumen untuk menentukan kemungkinan dokumen tersebut termasuk ke dalam suatu kelas tertentu. Multinomial Naive Bayes banyak diterapkan dalam berbagai bidang, seperti klasifikasi dokumen, analisis sentimen, deteksi spam, sistem rekomendasi berbasis konten, serta sistem *chatbot* berbasis teks. Algoritma ini dinilai efektif karena mampu menangani data dengan jumlah fitur yang besar, seperti dalam pemrosesan teks alami (Toyibah et al., 2024).

Algoritma ini bekerja dengan asumsi bahwa fitur-fitur (dalam hal ini, kata-kata dalam teks) bersifat independen secara kondisional terhadap kelas. Asumsi ini dikenal dengan istilah *naive* (naif), karena dalam praktiknya, kata-kata dalam bahasa alami sering kali saling bergantung. Namun demikian, asumsi ini justru menyederhanakan perhitungan probabilitas dan memungkinkan algoritma ini bekerja dengan cepat dan cukup akurat, terutama dalam konteks klasifikasi teks yang melibatkan data dalam jumlah besar (Toyibah et al., 2024).

Multinomial Naive Bayes didasarkan pada Teorema Bayes, yang secara matematis dinyatakan sebagai berikut:

$$P(A|B) = \frac{P(A|B) \cdot P(A)}{P(B)}$$

Keterangan:

- $P(A|B)$: Probabilitas posterior, yaitu probabilitas sebuah dokumen masuk ke dalam suatu kelas.
- $P(B|A)$: Likelihood, yaitu probabilitas kemunculan dokumen jika diketahui kelasnya.
- $P(A)$: Probabilitas prior, yaitu probabilitas kemunculan kelas tersebut secara umum.
- $P(B)$: Probabilitas marginal, yaitu probabilitas kemunculan dokumen.

Rumus Multinomial Naive Bayes untuk Klasifikasi Teks

$$P(c|d) \propto P(c) \pi_{k=1}^n P(t_k|c)$$

Keterangan:

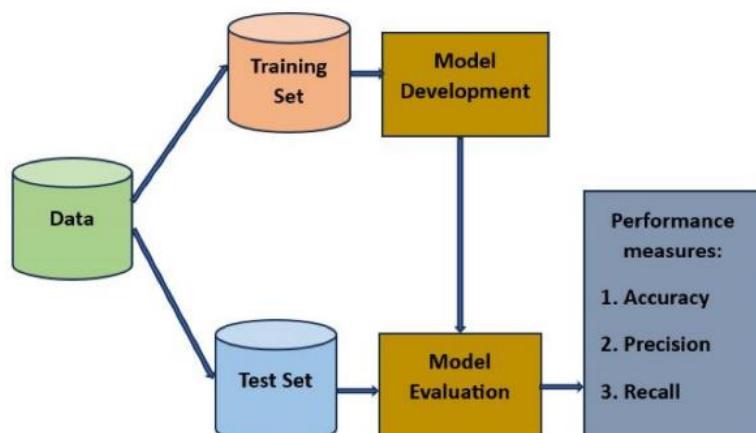
$P(c|d)$: Probabilitas dokumen d berada di kelas c

$P(c)$: Prior probability suatu dokumen berada di kelas c

$\{t_1, t_2, t_3, \dots, t_n\}$: Token dalam dokumen d yang merupakan bagian dari vocabulary dengan jumlah n

$P(t_k|c)$: Probabilitas bersyarat term t_k berada di dokumen pada kelas c

penerapan algoritma Multinomial Naïve Bayes untuk klasifikasi dilakukan melalui beberapa tahapan mulai dari pengumpulan data hingga evaluasi model. Proses ini digambarkan pada Gambar 2.2 berikut.



Gambar 2.2 Langkah-langkah Algoritma Multinomial Naïve Bayes

Sumber: (Keerthana et al., 2024)

Langkah-langkah Algoritma Multinomial Naïve Bayes (Keerthana et al., 2024):

1. Pengumpulan Data

Dataset berasal dari sumber publik atau disusun secara mandiri, lalu dibagi menjadi training set dan testing set.

2. Pembersihan Data (*Data Cleaning*)

Menghapus nilai kosong dan data yang tidak relevan.

3. Normalisasi Data

Menyesuaikan skala fitur agar kontribusi setiap variabel seimbang.

4. Pengkodean Fitur (*Feature Encoding*)

Mengubah variabel kategorikal menjadi representasi numerik dengan one-hot encoding.

5. Pembangunan Model (*Model Building*)

Melatih model menggunakan algoritma Multinomial Naïve Bayes dan algoritma pembanding lainnya.

6. Perhitungan Probabilitas

Menghitung probabilitas tiap kelas dan memilih kelas dengan probabilitas tertinggi sebagai hasil prediksi.

7. Evaluasi Model

Menggunakan confusion matrix, akurasi, presisi, recall, dan F1-score untuk menilai kinerja model.

2.6 Confusion Matrix

Confusion Matrix adalah alat evaluasi yang digunakan untuk mengukur kinerja model klasifikasi dalam *machine learning*. Matriks ini menyajikan informasi tentang jumlah prediksi yang benar dan salah yang dilakukan oleh model dalam empat kategori: *True Positive* (TP), *False Positive* (FP), *True Negative* (TN), dan *False Negative* (FN). Evaluasi kinerja model klasifikasi sangat penting untuk mengetahui seberapa efektif model dalam menghasilkan prediksi yang akurat (Toyibah et al., 2024). Selain itu, matriks ini membantu mengidentifikasi distribusi kesalahan (*misclassifications*) dan prediksi yang benar (*correct predictions*),

sehingga dapat menjadi acuan dalam perbaikan kinerja model (Muhidin et al., 2023). Dapat dilihat pada gambar 2.3 berikut:

		Positive (1)	Negative (0)
Predicted Values	Positive (1)	TP	FP
	Negative (0)	FN	TN

Gambar 2. 3 Confussion matrix
Sumber: (Khatib Sulaiman et al., n.d.)

Komponen-komponen dalam Confusion Matrix terdiri dari empat kategori utama sebagai berikut:

- True Positive (TP): Kasus di mana model memprediksi positif dengan benar (misalnya, memprediksi kelas yang benar sebagai positif).
- True Negative (TN): Kasus di mana model memprediksi negatif dengan benar (misalnya, memprediksi kelas yang benar sebagai negatif).
- False Positive (FP): Kasus di mana model memprediksi positif tetapi salah (misalnya, memprediksi kelas negatif sebagai positif).
- False Negative (FN): Kasus di mana model memprediksi negatif tetapi salah (misalnya, memprediksi kelas positif sebagai negatif).

Berdasarkan nilai-nilai pada Confusion Matrix, beberapa metrik evaluasi yang penting dapat dihitung, antara lain:

1. Akurasi

Akurasi mengukur seberapa sering model memprediksi dengan benar.

$$Akurasi = \frac{TP + TN}{TP + TN + FP + FN}$$

2. *Precision*

Presisi mengukur ketepatan prediksi positif yang dilakukan oleh model.

$$Presisi = \frac{TP}{TP + FP}$$

3. *Recall*

Mengukur sejauh mana model mampu menangkap seluruh data aktual positif.

$$Recall = \frac{TP}{TP + FN}$$

4. *F1-Score*

Merupakan rata-rata harmonis dari presisi dan recall. Cocok digunakan jika diperlukan keseimbangan antara keduanya.

$$F1 - Score = 2 \times \frac{Presisi \times Recall}{Presisi + Recall}$$

2.7 Software dan Tools untuk Pembuatan Model Klasifikasi

2.7.1 Framework

1. Scikit-learn

Scikit-learn merupakan pustaka *machine learning* berbasis Python yang menyediakan modul untuk *preprocessing*, *model selection*, dan *performance metrics*. Dalam penelitian ini, *Scikit-learn* digunakan untuk mengimplementasikan algoritma Multinomial Naïve Bayes, melakukan *training*, *testing*, serta mengevaluasi kinerja model (Fahmi, 2023).

2. NLTK (Natural Language Toolkit)

NLTK digunakan untuk text preprocessing seperti tokenization, stopword removal, dan stemming. Untuk bahasa Indonesia, NLTK dapat diintegrasikan dengan pustaka Sastrawi untuk melakukan stemming dan penghapusan stopword (Heryati et al., 2023).

2.7.2 Bahasa Pemrograman

1. Python

Python adalah bahasa pemrograman *open source* yang populer di bidang data science dan NLP karena sintaksnya yang sederhana, ekosistem pustaka

yang kaya, serta dukungan komunitas yang luas (Heryati et al., 2023). Bahasa ini mampu mengintegrasikan proses text preprocessing serta mengelola seluruh alur kerja pengembangan *chatbot*, mulai dari pengolahan data, implementasi algoritma, hingga evaluasi model, sehingga memudahkan peneliti maupun pengembang dalam membangun sistem yang efektif dan efisien.

2.7.3 Pustaka Python yang Digunakan

1. Pandas

Digunakan untuk mengelola dan membaca data dalam format .csv, .txt, .json (Retnoningsih & Pramudita, 2020).

2. NumPy

Menyediakan fungsi komputasi numerik dan manipulasi array yang efisien untuk pemrosesan data berukuran besar dan kecil (Retnoningsih & Pramudita, 2020). NumPy digunakan dalam proses representasi numerik teks dan perhitungan vektor kata.

3. Matplotlib

Digunakan untuk membuat visualisasi data, termasuk confusion matrix dan grafik evaluasi model. Visualisasi ini mempermudah analisis performa algoritma yang digunakan (Retnoningsih & Pramudita, 2020).

4. NLTK

Digunakan untuk pemrosesan teks berbahasa Indonesia, termasuk stemming dan penghapusan stopword. Stemming adalah proses mengubah kata berimbuhan menjadi bentuk dasarnya, sedangkan stopword removal merupakan penghapusan kata-kata umum yang tidak memiliki kontribusi signifikan terhadap pemaknaan teks (Heryati et al., n.d.).

2.7.4 Tools

1. Google Colab

Google Colab adalah platform berbasis cloud yang memungkinkan eksekusi kode Python tanpa memerlukan instalasi lokal, serta dilengkapi dukungan GPU untuk mempercepat proses pelatihan model (Gelar Guntara, 2023).

2. Visual Studio Code (VS Code)

Visual Studio Code adalah editor kode sumber yang mendukung berbagai bahasa pemrograman dan terintegrasi dengan sistem version control, sehingga memudahkan pengembangan dan pengelolaan proyek secara terstruktur.

2.8 Metode Penelitian Campuran (*Mixed Methods*)

Metode penelitian mixed methods merupakan pendekatan penelitian yang menggabungkan metode kualitatif dan kuantitatif dalam satu rangkaian penelitian secara sistematis untuk memperoleh pemahaman yang lebih komprehensif terhadap suatu permasalahan. Pendekatan ini memanfaatkan kelebihan masing-masing metode, di mana metode kuantitatif memberikan data numerik yang dapat dianalisis secara statistik, sedangkan metode kualitatif memberikan kedalaman pemahaman terhadap fenomena melalui data naratif (Vebrianto et al., 2020).

Munurut (Hakim Nasution et al., 2024), penelitian kombinasi (mixed methods) adalah metode penelitian campuran yang memadukan teknik, konsep, dan perspektif dari pendekatan kualitatif dan kuantitatif secara terintegrasi sehingga menghasilkan temuan baru. Pendekatan ini dirancang untuk memberikan pemahaman yang lebih menyeluruh terhadap fenomena atau masalah penelitian dan dapat diterapkan dalam berbagai disiplin ilmu seperti pendidikan, kesehatan, teknologi informasi, dan ilmu sosial.

2.8.1 Tujuan Penggunaan Mixed Methods

Penggunaan metode mixed methods memiliki beberapa tujuan utama, yaitu:

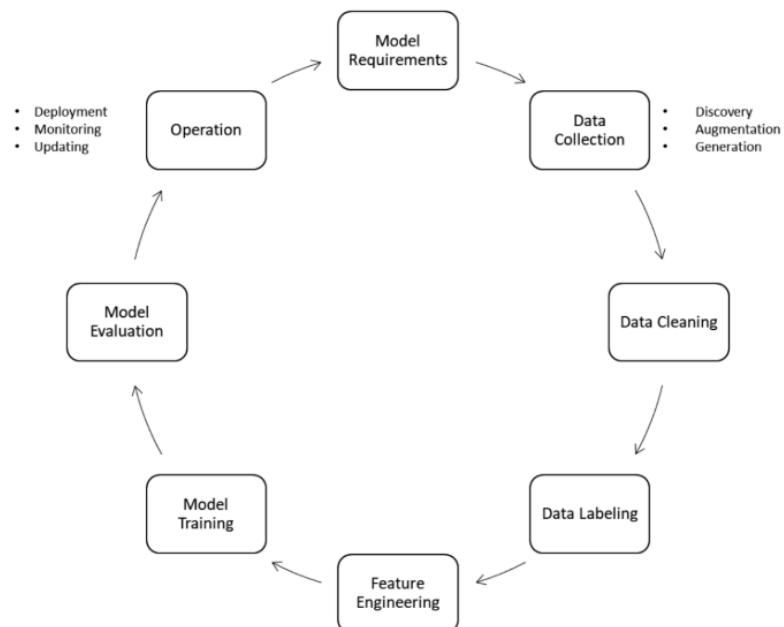
1. Komplementaritas - Hasil dari satu metode digunakan untuk melengkapi atau memperjelas hasil dari metode lainnya (contohnya, data wawancara mendukung temuan dari survei kuantitatif).
2. Pengembangan - Hasil dari satu metode digunakan untuk mengembangkan metode lainnya, termasuk dalam pemilihan sampel dan penyusunan instrumen penelitian.
3. Inisiasi - Menemukan ketidaksesuaian hasil dari kedua metode untuk menghasilkan wawasan atau hipotesis baru.

4. Ekspansi - Memperluas cakupan penelitian dengan menjawab berbagai aspek dari pertanyaan penelitian yang memerlukan pendekatan berbeda.
5. Triangulasi - Menggabungkan hasil dari kedua metode untuk meningkatkan validitas temuan penelitian.

Melalui tujuan-tujuan ini, pendekatan mixed methods membantu peneliti memperoleh hasil yang lebih akurat, menyeluruh, dan relevan dengan konteks permasalahan yang dikaji (Diantoni et al., 2024; Vebrianto et al., 2020).

2.9 Machine Learning Life Cycle

Berikut adalah metode pembuatan model klasifikasi *Machine Learning* dengan menggunakan *Machine Learning life cycle*:



Gambar 2. 4 Machine Learning Life Cycle

Sumber: (Gärtler et al., 2021)

Proses pembuatan model klasifikasi Machine Learning mengikuti tahapan yang terdapat pada Machine Learning Life Cycle. Menurut (Gärtler et al., 2021), tahapan tersebut meliputi beberapa langkah utama sebagai berikut:

1. Model Requirements

Pada fase model requirements dilakukan pemilihan jenis data, model algoritma, pengukuran kinerja dan teknologi atau platform yang akan digunakan kemudian akan diimplementasikan ke dalam *Machine Learning*.

2. Data Collection

Pada fase data collection ini yang dilakukan adalah mengumpulkan dataset yang akan digunakan untuk melakukan pengklasifikasian, dalam fase data collection ini terdapat tiga teknik dalam pengumpulan data collection yaitu, *data discovery* (data yang sudah ada di repository platform online), *data augmentation* (pengumpulan data tambahan yang dilakukan oleh peneliti) dan *data generation* (pengolahan data dari *data discovery* dan *data augmentation* agar sampel data yang telah dikumpulkan lebih berkualitas dan menghindari terjadinya kehilangan data secara acak).

3. Data Cleaning

Data Cleaning digunakan untuk memastikan kualitas dari sampel seperti akurasi kelengkapan, konsistensi, keunikan dan integritas. Pada *text processing*, *data cleaning* sangat diperlukan untuk standarisasi ukuran dari suatu text yang ada dalam dokumen dataset.

4. Data Labeling

Data labeling adalah proses memberikan tanda, kategori, atau anotasi pada data mentah agar dapat digunakan dalam proses pembelajaran mesin. Label ini berfungsi sebagai informasi tambahan yang membantu algoritma mengenali pola dan melakukan klasifikasi dengan lebih akurat.

5. Feature Engineering

Feature engineering adalah proses pemeriksaan dataset untuk meningkatkan fitur dari sistem yang digunakan yang nantinya hasil dari proses feature engineering ini akan direpresentasikan dalam bentuk *numeric data* atau *data matrix*.

6. Model Training

Model training mencakup aktivitas untuk melatih dan memilih model *Machine Learning* kemudian menyesuaikan hyperparameter pada data yang sudah dikumpulkan, dibersihkan dan dilabelkan.

7. Model Evaluation

Model evaluation Pada tahap ini akan dilakukan analisa untuk mendapatkan dan mengkonfirmasi apakah model *Machine Learning* yang telah dibuat sudah sesuai dengan tujuan ataukah belum berdasarkan hasil analisa dari pendapat ahli bahasa Indonesia.

8. Operation/Deployment

Operation merupakan tahap di mana model *Machine Learning* yang telah dikembangkan diimplementasikan ke dalam produksi dan mulai digunakan untuk memproses data secara *real-time*.

2.10 Penelitian Terkait

Penelitian terdahulu dijadikan sebagai pertimbangan oleh penulis untuk membantu penulisan dan sebagai referensi, sehingga memudahkan dalam pelaksanaan penelitian. Penelitian-penelitian sebelumnya dapat dilihat pada table berikut.

Tabel 2. 1 Penelitian Terkait

No	Peneliti, Tahun	Objek Penelitian	Metode Preprocessing	Modeling	Evaluasi	Akurasi
1	Muliyono(Muliyono, 2021)	Chatbot pelayanan akademik di Fakultas Teknik	Case Folding, Tokenizing, Filtering	NLP berbasis Dialogflow	Uji beta (skor 3,55/4), uji akurasi 40 pertanyaan	92,5%
2	Muhammad Izzuddin Mubarok, Mahardika Abdi(Mubarok & Abdi, 2024)	Chatbot akademik berbasis NLP untuk FIKTI UMSU	Lowercase, punctuation removal, lemmatization, tokenization, padding, label encoding	Deep Learning (LSTM)	Blackbox testing, MAP (Mean Average Precision)	MAP 97%, 8/9 jawaban tepat
3	Alvin Christoper Sebayang(Christoper Sebayang et al., 2024)	Chatbot layanan pelanggan web Mangcoding (PT Anugrah Kreasi Digital)	Case Folding, Tokenizing, Filtering, Lemmatizing (menggunakan NLTK)	NLP + Naive Bayes (Scikit-learn)	Confusion Matrix, Blackbox, Usability Test	Akurasi 100%, Precision, Recall, F1 = 1
4	Kesuma Dwi Ningtyas,(Dwi Ningtyas &	Chatbot info layanan kantor lurah	Split sentence, Tokenizing, Stopword	Naive Bayes + TF-IDF	Probabilitas	Probabilitas

No	Peneliti, Tahun	Objek Penelitian	Metode Preprocessing	Modeling	Evaluasi	Akurasi
	Kurniawan, 2023)				manual, matching	bilitas tertinggi: 0.00016 6891
5	Nimatul Mamuriyah, Haeruddin2, Hero(Mamuriyah et al., 2024)	Chatbot interaktif untuk informasi stok barang (GUI Python)	Tokenisasi, matching pertanyaan dengan difflib, JSON-based KB	Naive Bayes + get_close_matches (difflib)	Uji coba data latih dan uji, analisis akurasi dan respons	Akurasi data latih: 90% Data uji: 85% Respons < 1 detik
6	Gugi Gustaman, Khoirida Aelani(Gustaman & Aelani, n.d.)	Chatbot untuk layanan informasi penerimaan mahasiswa baru	Tokenisasi, TF-IDF (pembobotan kata)	Multi-nomial Naïve Bayes	Cross-validation, uji langsung ke pengguna, confusion matrix	Akurasi: 99,3% Akutasi uji pengguna: 65% Kesalahan: 35%
7	Asep Muhidin, Muhtajuddin Danny, Elkin Rilvani (Muhidin et al., 2023)	Chatbot layanan informasi berbasis teks untuk membantu Customer Service	NLP (tokenisasi, pengolahan teks)	Multi-nomial Naïve Bayes	Pengujian 60 percakapan berbeda; perbandingan dengan metode lain	96% (train), 84% (test)
8	Cahya Diantoni, Ratna Mufidah, Heru Triana(Diantoni et al., 2024)	Chatbot untuk informasi Magang dan Studi Independen (MSIB) di Fakultas Ilmu	Preprocessing teks	Naïve Bayes	Evaluasi kinerja model dengan CRISP-DM; implementasi di Telegram	88,9%

No	Peneliti, Tahun	Objek Penelitian	Metode Preprocessing	Modeling	Evaluasi	Akurasi
		Komputer Unsika				
9	Deden Moh Alfiansyah, Willys, Lila Setiyani, Devi Fajar Wati, Dedi (Deden Moh Alfiansyah et al., 2025)	Web untuk layanan informasi akademik di Horizon University	Rule-based (pencocokan kata kunci)	Rule-based Chatbot	Black-box testing, survei kepuasan pengguna	85% responden puas; respon < 2 detik
10	Rena Cahya Hutama, Fauziah, Ratih Titi Komalasari (Cahya Hutama & Titi Komalasari, n.d.)	Aplikasi Chatbot Berbasis Teks Menggunakan Algoritma Naive Bayes Classifier Faq Grabads	Lower-casing, Tokenization, Stopword Removal, Lemmatization, Stemming, Bag of Words	Naive Bayes Classifier	Evaluasi melalui split data training dan testing dengan 5 rasio berbeda (0.5–0.9);	93.33% (split 0.8)
11	Dedi Kundana, Chairani (Kundana & Chairani, 2023)	Analisis sentimen wacana kenaikan harga tiket Candi Borobudur	Normalisasi, Case Folding, Cleansing, Tokenisasi, Stopword Removal, Transform Cases, Labeling	Naïve Bayes, Decision Tree	Confusion Matrix	Naïve Bayes: 100%, Decision Tree: 35,97%
12	Fitria, Rifad Sobah, Chairani, Septilia, Suci, & Siti (Fitria et al., 2021)	Prediksi ketidakpartisan sipasian pemilih pada Pemilu di Provinsi Lampung	Pemilihan variabel (Tempat Lahir, Status, Gender, Alamat, Informasi)	Naïve Bayes Classifier	Probabilitas (perhitungan manual & visualisasi dengan Tableau)	Tidak disebutkan spesifik (%)
13	Syahputri Syahputri, Winarto, Sherli Trisnawati, & Taufik (2025)(Rahmalia Syahputri et al., 2025)	Deteksi penyakit daun melon berbasis Android	Data cleaning, labeling (one-hot encoding), train-test split (80:20)	Convolutional Neural Network (CNN) dengan TensorFlow	Accuracy, Precision, Recall, F1-score, ROC-AUC	Training : 91,5%; Validasi: 71,9%; Implementasi: 86,3%

No	Peneliti, Tahun	Objek Penelitian	Metode Preprocessing	Modeling	Evaluasi	Akurasi
14	Syahputri, Halawa, Trisnawati, Nurfiana, & Taufik (2024) (Rahmalia Syahputri et al., 2024)	Sistem e-voting berbasis pengenalan wajah (FaceVoting)	Ekstraksi fitur Haar, training gambar positif & negatif	Haar Cascade + LBPH (Local Binary Pattern Histogram)	Validasi jarak, sudut, objek, ekspresi; UAT (110 responden)	Tingkat kepuasan pengguna 96,5%
15	Ari Rohmawati, Sri Lestari (Rohmawati A & Lestari S, 2021)	Pemetaan kinerja UMKM di Kabupaten Pesawaran	Pemilihan atribut (jenis usaha, nama usaha, izin, aset, omzet)	Naïve Bayes Classifier	Split Validation (80:20)	98,68%

Penelitian yang penulis laksanakan memiliki perbedaan dengan penelitian-penelitian terdahulu pada Tabel 2.1. Sebagian besar penelitian sebelumnya berfokus pada pengembangan *chatbot* untuk berbagai kebutuhan seperti layanan akademik, layanan informasi umum, maupun layanan pelanggan. Model yang digunakan juga bervariasi, mulai dari *rule-based*, *Naïve Bayes*, hingga *deep learning* dengan LSTM. Dari sisi preprocessing, penelitian terdahulu umumnya menggunakan tahapan seperti *case folding*, tokenisasi, *stopword removal*, stemming, maupun lemmatization untuk meningkatkan kualitas data teks. Selain itu, metode evaluasi yang diterapkan juga beragam, seperti *confusion matrix*, *cross-validation*, *black-box testing*, hingga survei kepuasan pengguna, dengan hasil akurasi yang cukup bervariasi mulai dari 85% hingga mencapai 100%.

Perbedaan utama penelitian ini dengan penelitian terdahulu terletak pada objek dan fokus penelitian. Jika penelitian sebelumnya banyak mengembangkan *chatbot* untuk layanan akademik secara umum, customer service, maupun layanan desa, maka penelitian ini secara khusus diarahkan untuk membangun *chatbot* berbasis NLP yang membantu layanan administrasi MBKM di Institut Informatika dan Bisnis Darmajaya. Selain itu, penelitian ini menggunakan tahapan preprocessing yang disesuaikan dengan karakteristik bahasa Indonesia, yaitu *case folding*, tokenisasi, *stopword removal*, dan stemming menggunakan library NLTK serta Sastrawi. Dari sisi pemodelan, penelitian ini menerapkan algoritma *Multinomial Naïve Bayes* dengan pendekatan TF-IDF sebagai metode klasifikasi intent, berbeda

dengan penelitian terdahulu yang cenderung mengandalkan LSTM maupun pendekatan berbasis aturan.

Pada aspek evaluasi, penelitian ini memanfaatkan kombinasi melalui *confusion matrix*, penghitungan metrik akurasi, precision, recall, F1-score, serta *user acceptance testing (UAT)* untuk menilai tingkat penerimaan pengguna. Pendekatan ini memberikan gambaran yang lebih komprehensif dibandingkan penelitian sebelumnya yang hanya menekankan pada uji teknis atau survei sederhana. Dengan demikian, penelitian ini tidak hanya melanjutkan tren penelitian *chatbot* berbasis NLP, tetapi juga menghadirkan fokus baru pada konteks administrasi MBKM di perguruan tinggi, yang hingga saat ini masih jarang diteliti.