BAB 2
TINJAUAN PUSTAKA

2.1 Penelitian Terdahulu

Beberapa penelitian yang terkait digunakan sebagai bahan pertimbangan untuk melakukan penelitian yang akan dilakukan sekarang mengenai implementasi pipeline big data menggunakan MongoDB dan *Sentiment Analysis* dengan menggunakan metode *Support Vector Machine* (SVM). Penelitian terdahulu yang akan digunakan sebagai perbandingan pada proyek akhir ini ada 5 penelitian yaitu:

Pada penelitian pertama yang dilakukan oleh (Rahat et al., 2020) dengan judul "*Comparison of Naive Bayes and SVM Algorithm based on Sentiment Analysis Using Review Dataset*" dimana penelitian ini bertujuan untuk membandingkan algoritma *machine learning* mana yang lebih cocok untuk membuat sentiment analysis dengan menggunakan data review pelayanan pesawat yang berasal dari Twitter .Hasil penelitian menunjukkan bahwa metode Naïve Bayes memiliki akurasi 76%, presisi 89%, recall 83%, dan F1-Score sebesar 86%. Sementara itu, metode SVM memiliki akurasi 82%, presisi 90%, recall 81%, dan F1-Score sebesar 85%. Penelitian ini menyimpulkan bahwa SVM lebih unggul untuk pengkategorian teks kompleks, meskipun Naïve Bayes juga memiliki kelebihan tersendiri.

(Seman & Razmi, 2020) juga melakukan penelitian terkait dengan sentiment analysis dengan membandingkan SVM, Decision Tree, Naive Bayes dengan menerapkan big data, penelitian yang berjudul "*Machine learning-based technique for big data sentiments extraction*", dimana di dalam penelitian ini peneliti mengumpulkan 3 buah dataset yaitu data Twitter terkait semEval(SE-T) , data Twitter terkait corpus (TS) dan dataset terkait malaysia yang berisikan 3 bahasa yaitu indonesia , singapore ,dan malaysia (OC) hasil dari penelitian ini menunjukan bahwa dari ketiga dataset dan ketiga algoritma ini , algoritma SVM memiliki akurasi yang paling tinggi dibandingkan dengan algoritma Naive Bayes (NB) dan Decision Trees (DT). Hal ini terlihat dari tingkat akurasi, presisi, recall, dan F-measure yang lebih tinggi pada tiga dataset yang digunakan, yaitu SE-T, TS, dan OC. Akurasi yang diperoleh menggunakan SVM mencapai 61,55%, 87,47%, dan 82,58% untuk masing-masing dataset, jauh lebih unggul dibandingkan algoritma

lainnya. Presisi keseluruhan SVM juga tercatat lebih tinggi, yaitu 64,96%, 71,26%, dan 91,25%.

Penelitian yang dilakukan oleh Oguine dkk (2022)dengan judul "*Big Data and Analytics Implementation in Tertiary Institutions to Predict Students Performance in Nigeria*" dimana pada penelitian ini bertujuan mengimplementasikan big data dan analitik di University of Abuja, Nigeria, untuk meningkatkan kualitas pendidikan, memprediksi kinerja mahasiswa, serta membangun *data-driven culture* di kampus. Pada penelitian ini mengunakan data dari mahasiswa kampus University of Abuja, Nigeria yang terdiri dari *column age*, *gender*, *department*, *ethnicity*, *academic records* and *lesson duration* dengan menerapkan analisis SWOT terhadap mahasiswa untuk mengidentifikasi kelebihan dan kekurangan mereka. Penelitian ini menggunakan lima algoritma machine learning, yaitu *Decision Trees, AdaBoost, Support Vector Regression, Naïve Bayes*, dan *Stochastic Gradient Descent*, yang dievaluasi menggunakan *cross-validation.* Hasil penelitian menunjukkan bahwa algoritma terbaik adalah Naïve Bayes dan AdaBoost dengan akurasi masing-masing sebesar 64% dan 65%.

Pada penelitian yang dilakukan oleh Purnomo dkk (2021) dengan judul "Implementasi Big Data Analytical Untuk Perguruan Tinggi Menggunakan Machine Learning", peneliti mengembangkan sebuah sistem berbasis big data analytical untuk meningkatkan pengambilan keputusan di perguruan tinggi, khususnya dalam menganalisis kinerja akademik mahasiswa dengan memanfaatkan data dari akademik publik yang terdiri dari 4 kolumn yaitu data absensi dan nilai 1, nilai 2 dan nilai 3. Penelitian ini menggunakan algoritma K-Means Clustering dengan dukungan Apache Spark dan Hadoop untuk mengelompokkan mahasiswa ke dalam lima cluster berdasarkan kinerja akademik mereka. Hasil penelitian menunjukkan bahwa sistem yang dikembangkan mampu memprediksi kinerja mahasiswa, di mana Cluster 1 merupakan mahasiswa dengan kinerja terbaik, sedangkan Cluster 5 adalah mahasiswa dengan kinerja terendah.

Pada penelitian yang dilakukan (Raviya & Mary Vennila, 2021) dengan judul "*An Implementation of Hybrid Enhanced Sentiment Analysis System using Spark ML Pipeline: A Big Data Analytics Framework*" penelitian ini dilakukan penelitian terhadap data "Amazon online product review". Dataset ini mencakup sekitar

100.000 ulasan produk dari berbagai kategori seperti elektronik, peralatan rumah tangga, dan buku yang dikumpulkan dari situs web Amazon.com dimana disimpan dalam big data dengan menggunakan apache Spark dimana pada penelitian ini dibuat sentiment analysis dengan memanfaatkan berbagai macam metode mulai dari naive bayes , logistic regression , random forest , svm , dan proposed SVM dimana hasil akhirnya diketahui bahwa rata rata akurasi tertinggi dipegang oleh metode proposed cnn-svm gabungan antara cnn dan svm dengan waktu sekitar 10-11 second dengan akurasi mencapai 0,95 ,dan diikuti dengan svm yang rata rata 15-16 second dan akurasi sebesar 0,89

Table 2. 1. Tabel Penelitian Terdahulu

Penelitian	Berjudul	tujuan	Model Dan teknologi	Hasil
Rahat dkk ,2020	*Comparison of Naive Bayes and SVM Algorithm based on Sentiment Analysis Using Review Dataset*	Membandingkan algrotima Naive Bayes dan SVM untuk review dataset	Naive Bayes, SVM	Naïve Bayes memiliki akurasi 76%, presisi 89%, recall 83%, dan F1-Score sebesar 86%. Sementara itu, metode SVM memiliki akurasi 82%, presisi 90%, recall 81%, dan F1-Score sebesar 85%. Penelitian ini menyimpulkan bahwa SVM lebih unggul untuk pengkategorian teks
Seman & Razmi, 2020	*Machine learning-based technique for big data sentiments extraction*	Membandingkan beberapa algrotima yang cocok untuk digunakna pada 3 dataset yaitu data twitter terkait semEval (SE-T), data twitter terkait corpus (TS) dan dataset terkait malaysia yang berisikan 3 bahasa	Naive Bayes (NB), Decision Trees (DT), SVM	Berdasarkan hasil analisis, algoritma SVM menunjukkan performa terbaik dibandingkan Naive Bayes (NB) dan Decision Trees (DT) dalam hal akurasi, presisi,

Penelitian	Berjudul	tujuan	Model Dan teknologi	Hasil
		yaitu indonesia, singapore ,dan malaysia (OC)		recall, dan F-measure pada tiga dataset (SE-T, TS, dan OC). SVM mencatatkan akurasi tertinggi, yaitu 61,55%, 87,47%, dan 82,58%, serta presisi keseluruhan yang unggul sebesar 64,96%, 71,26%, dan 91,25%. Hal ini mengindikasikan bahwa SVM lebih efektif dan andal dalam menangani data dibandingkan algoritma lainnya.
Oguine dkk, (2022)	*Big Data and Analytics Implementation in Tertiary Institutions to Predict Students Performance in Nigeria*	Mengimplementasi Big Data dan analitik dalam institusi pendidikan tinggi University of Abuja, Nigeria untuk meningkatkan kualitas pendidikan, memprediksi kinerja mahasiswa, serta mengimplementasikan *data-driven culture* di kampus tersebut.	Decision Trees, AdaBoost, Support Vector Regression, Naïve Bayes, dan Stochastic Gradient Descent. Dan dievaluasi dengan cross-validation	Dengan implementasi big data ini University of Abuja membantu menganalisis kelebihan dan kekurangan siswa di universitas dalam bentuk analisis SWOT dimana algrotima tertinggi dalam implementasi ini adalah Naive Bayess dan AdaBoost dengan akurasi 0,64% dan 0,65%
Purnomo	Implementasi	Mengimplementasikan	K-Means	Berdasarkan

Penelitian	Berjudul	tujuan	Model Dan teknologi	Hasil
dkk (2021)	Big Data Analytical Untuk Perguruan Tinggi Menggunakan Machine Learning	big data analytical di perguruan tinggi untuk meningkatkan pengambilan keputusan, khususnya dalam menganalisis kinerja akademik mahasiswa dengan cara mengelompokan mahasiswa menjadi 5 cluster	Clustering, Apache Spark, Hadoop,	hasil peneltiian ini menghasil sebuah sistem yang dapat memprediksi kinerja mahasiswa ke dalam 5 cluster dimana cluster 1 adalah cluster terbaik dan cluster 5 adalah cluster terburuk
Raviya, & Mary, 2021	An Implementation of Hybrid Enhanced Sentiment Analysis System using Spark ML *Pipeline*: A Big Data Analytics Framework	Membandingkan algoritma pada *Machine learning* dan performanya terhadap data review dari amazon	naive bayes, logistic regression, random forest, svm, dan proposed SVM ,Apache Spark , keras	rata rata akurasi tertinggi dipegang oleh metode proposed cnn-svm gabungan antara cnn dan svm dengan waktu sekitar 10-11 second dengan akurasi mencapai 0.95, dan diikuti dengan svm yang rata rata 15-16 second dan akurasi sebesar 0.89
Deri (2025)	Implementasi *Pipeline big data* untuk Sentiment Analisis Survei Kepuasaan Mahasiswa Pada Politeknik Caltex Riau	Membuat suatu sistem untuk mengotomasikan proses visualisasi pada data umpan balik pembelajaran pada Politeknik Caltex Riau	SVM, *Python*, Big Data, *MongoDB*	-

Berdasarkan kajian terhadap beberapa penelitian terdahulu pada Tabel 2.1 (Politeknik Caltex Riau, 2026), diketahui bahwa penelitian-penelitian tersebut telah membahas topik yang relevan dengan penelitian ini, namun masih memiliki keterbatasan dari sisi implementasi otomatisasi analisis data umpan balik,

pengelolaan data berskala besar, maupun pengembangan visualisasi hasil analisis secara interaktif. Penelitian oleh Deri (2025) yang berjudul *Implementasi Pipeline Big Data untuk Sentiment Analisis Survei Kepuasaan Mahasiswa Pada Politeknik Caltex Riau* berfokus pada pembuatan sistem untuk mengotomatisasi proses visualisasi data umpan balik pembelajaran menggunakan metode SVM, Python, Big Data, dan MongoDB.

Penelitian ini memiliki perbedaan dengan penelitian sebelumnya karena mengkaji permasalahan analisis sentimen dengan pendekatan yang disesuaikan terhadap kebutuhan pengolahan data, visualisasi informasi, serta karakteristik objek penelitian yang digunakan. Dengan demikian, penelitian ini diharapkan dapat memberikan kontribusi dalam pengembangan sistem analisis sentimen berbasis Big Data yang diharapkan dapat mendukung pengambilan keputusan, serta melengkapi hasil-hasil penelitian yang telah ada sebelumnya.

2.2 Landasan Teori

2.2.1 Gambaran Umum Badan Perencanaan, Pengembangan dan Penjaminan Mutu (BP3M)

Badan Perencanaan, Pengembangan dan Penjaminan Mutu (BP3M) merupakan sebuah bidang di kampus Politeknik Caltex Riau yang fokus pada pengembangan dan penjaminan mutu di Politeknik Caltex Riau mulai dari mutu mahasiswa, staff pcr, prodi hingga visi misi kampus, dimana salah satu pendekatan yang dilakukan BP3M melalui kuesioner yang disebar melalui website https://survei.pcr.ac.id/ , yang berisikan kuesioner penjaminan mutu yang biasanya akan dilakukan pada setiap akhir semester pembelajaran.

2.2.2 Dashboard Business Intelligence

Dashboard Business Intelligence (BI) merupakan sistem visualisasi data yang dirancang untuk mengumpulkan, menganalisis, dan memvisualisasikan data dalam bentuk yang mudah dipahami oleh pemangku kepentingan(Chaudhuri et al., 2011). Dalam lingkungan bisnis yang kompetitif, organisasi memerlukan alat yang dapat mengubah data mentah menjadi wawasan yang bermakna. Dashboard BI memungkinkan integrasi berbagai sumber data, analisis prediktif, serta penyajian metrik utama dalam bentuk grafik atau indikator performa(Few, 2006). Implementasi dashboard BI tidak hanya meningkatkan efisiensi operasional tetapi

juga memberikan visibilitas yang lebih baik terhadap tren bisnis, memungkinkan para pemimpin organisasi untuk mengambil keputusan berbasis (Busulwa, 2022). Dengan demikian, pemanfaatan dashboard BI menjadi elemen krusial dalam transformasi digital perusahaan, di mana aksesibilitas data yang cepat dan akurat dapat menjadi keunggulan kompetitif dalam industri yang dinamis.

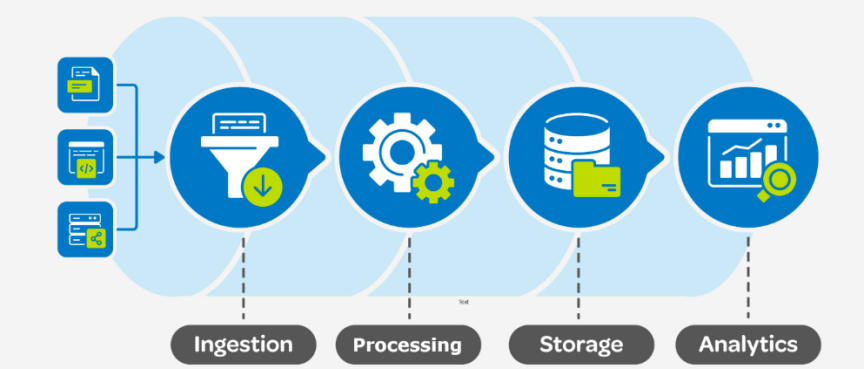
2.2.3 Python

Python merupakan bahasa pemrograman tingkat tinggi yang dikenal karena sintaksisnya yang sederhana dan fleksibilitasnya dalam berbagai domain aplikasi, termasuk pengelolaan dan analisis data (Rossum & Drake, 2006). Bahasa pemerograman ini memiliki dukungan pustaka yang luas, seperti Pandas, NumPy, dan Matplotlib, Python pengguna untuk memungkinkan mengolah, menganalisis, serta memvisualisasikan data dengan lebih efisien(Mckinney, 2017). Dalam pengelolaan data, Python dapat digunakan untuk membaca dan menulis berbagai format file (CSV, Excel, JSON), membersihkan data dari nilai yang hilang atau tidak valid, serta melakukan agregasi data untuk mendapatkan wawasan yang lebih mendalam (Lutz, 2015). Selain itu, Python juga mendukung analisis data skala besar dengan pustaka seperti Dask dan PySpark, memungkinkan pemrosesan data dalam jumlah besar secara lebih optimal. Dengan kemampuannya dalam otomatisasi, integrasi dengan database, dan dukungan komunitas yang luas, Python menjadi alat yang sangat berharga dalam proses pengolahan data.

2.2.4 Sentiment Analysis

Sentiment analysis adalah teknik dalam pemrosesan bahasa alami (*Natural Language Processing*/NLP) yang digunakan untuk mengidentifikasi, mengklasifikasikan, dan menganalisis opini atau kecenderungan sentimen masyarakat terhadap suatu topik tertentu, baik positif, negatif, maupun netral (Pranata, B., & Susanti. (2023). Metode ini sering diterapkan dalam berbagai bidang, seperti analisis media sosial, ulasan produk, dan survei kepuasan pelanggan, dengan memanfaatkan algoritma pembelajaran mesin dan model NLP untuk mengekstrak makna dari teks. Dengan sentiment analysis, perusahaan dan peneliti dapat memahami opini publik secara lebih mendalam dan mengambil keputusan yang lebih berbasis data.

2.2.5 Pipeline Big Data



Gambar 2. 1. Pipeline Big Data
sumber, (Datamation, 2021)

Pipeline big data merupakan rangkaian proses pengolahan data dalam skala besar yang terdiri dari berbagai komponen pemrosesan yang saling terhubung secara berurutan. Arsitektur pipeline ini dirancang untuk memastikan aliran data berjalan secara efisien melalui beberapa tahapan seperti yang kita lihat pada Gambar 2.1. Proses diawali dengan pengambilan data dari berbagai sumber (*ingestion*), kemudian dilanjutkan dengan pemrosesan dan pembersihan data menggunakan metode serta algoritma tertentu untuk meningkatkan akurasi dan kualitasnya (*processing*). Setelah data diproses dan dibersihkan, hasilnya disimpan dalam database online (*storage*) agar dapat diakses untuk analisis lebih lanjut. Data yang telah tersimpan kemudian divisualisasikan dalam bentuk informasi yang mudah dipahami, sehingga dapat membantu pengguna dalam mengambil keputusan yang lebih tepat (Israel Mnsen et al., 2024).

2.2.6 Sastrawi

Sastrawi adalah pustaka (library) Python yang dirancang khusus untuk bahasa Indonesia, berdasarkan algoritma yang dikembangkan oleh Naziem dan Adrian. Fungsi utamanya adalah melakukan Stemming, yaitu mengubah kata berimbuhan menjadi bentuk dasarnya. Selain itu, Sastrawi juga dapat digunakan untuk menghitung polaritas sentimen dalam teks, dengan memanfaatkan sumber seperti kateglo.com untuk menentukan nilai polaritas kata-kata tertentu (Dharmendra et al., 2022). Dalam analisis sentimen, proses *Stemming* membantu mengurangi kompleksitas data dengan menyamakan berbagai bentuk kata menjadi satu representasi standar, sehingga meningkatkan akurasi klasifikasi sentimen. Sastrawi

berperan penting dalam meningkatkan efisiensi dan akurasi meningkatkan efisiensi dan akurasi analisis sentimen terhadap teks berbahasa Indonesia.

Untuk menghitung *polarity score*, langkah pertama adalah melakukan *stemming* terhadap teks menggunakan Sastrawi agar setiap kata dikonversi ke bentuk dasarnya. Setelah itu, setiap kata yang telah *distemming* dicocokkan dengan kamus polaritas yang berisi daftar kata positif dan negatif beserta nilai polaritasnya. Nilai polaritas dari setiap kata dalam teks kemudian dijumlahkan untuk mendapatkan skor akhir. Jika hasilnya positif, teks cenderung memiliki sentimen positif, sedangkan jika hasilnya negatif, teks lebih condong ke arah sentimen negatif. Dengan pendekatan ini, Sastrawi tidak hanya berperan dalam normalisasi teks tetapi juga dapat meningkatkan akurasi analisis sentimen terhadap bahasa Indonesia.

2.2.7 Pembobotan Kata TF-IDF

TF-IDF (*Term Frequency-Inverse Document Frequency*) adalah metode pembobotan kata yang digunakan dalam pemrosesan teks untuk menilai seberapa penting suatu kata dalam suatu dokumen dibandingkan dengan kumpulan dokumen lainnya (Hanani, 2023). Metode ini sering diterapkan dalam analisis teks, pencarian informasi, dan pemodelan dokumen.

Pembobotan kata dengan TF-IDF terdiri dari dua komponen utama:

1) Term Frequency (TF)

   Term Frequency (TF) mengukur seberapa sering suatu kata muncul dalam sebuah dokumen dibandingkan dengan jumlah total kata dalam dokumen tersebut. Rumusnya adalah:

$$TF(t,d) = \frac{F_{(t,d)}}{N_d}$$

Rumus 2. 1. Rumus Term Frequency

   Di mana:

   - $F_{(t,d)}$ Merupakan jumlah kata t pada dokument / kalimat d
   - N_d merupakan jumlah seluruh kata pada dokument /kalimat d

2) Inverse Document Frequency (IDF)

   Inverse Document Frequency (IDF) mengukur seberapa jarang suatu kata muncul di seluruh dokumen. Kata yang sering muncul di banyak dokumen akan memiliki IDF yang lebih rendah, sedangkan kata yang

jarang muncul akan memiliki IDF yang lebih tinggi. Rumusnya adalah:

$$IDF(t) = \log(\frac{N}{DF(T)})$$

Rumus 2. 2. Rumus Inverse Document Frequency

Di mana:

- N merupakan jumlah total dokument/kalimat yang digunakan
- DF(T) merupakan jumlah dokument/kalimat yang mengandung kata t

3) Pembobotan akhir

Bobot akhir dari suatu kata dihitung dengan mengalikan TF dan IDF:

$$W = TF(t,d)x\ IDF(t)$$

Rumus 2. 3. Rumus Pembobotan Akhir

Dimana:

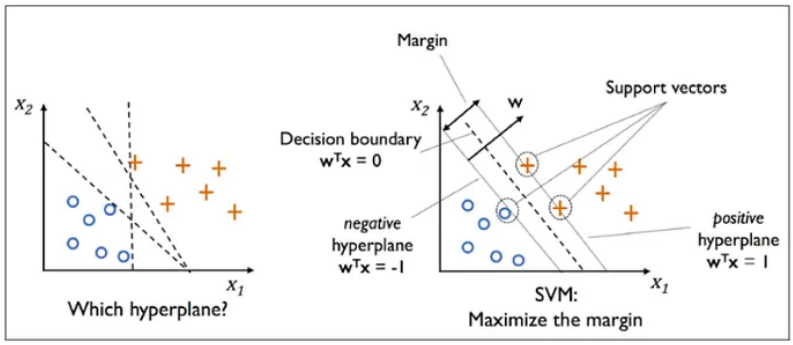
- TF (t, d) merupakan term frequency dari kata t terhadap dokument d
- IDF(t) merupakan *inverse document frequency* (IDF) dari kata t terhadap seluruh dokument

Semakin tinggi nilai pembobotan TF-IDF, semakin penting kata tersebut dalam dokumen tertentu dibandingkan dengan kumpulan dokumen lainnya. Metode ini efektif dalam menghilangkan kata-kata umum yang kurang bermakna dan lebih menonjolkan kata-kata yang signifikan dalam suatu konteks tertentu (Hanani, 2023).

2.2.8 Ngram

N-gram adalah model probabilistik dalam pemrosesan *Natural Language Processing* yang menganalisis urutan kata berdasarkan sekumpulan kata berurutan dengan ukuran n tertentu (Martin., 2001). N-gram membantu menangkap pola kata yang mencerminkan polaritas sentimen serta meningkatkan representasi fitur dengan mempertimbangkan kombinasi kata yang sering muncul, sehingga meningkatkan akurasi model klasifikasi teks (Medhat et al., 2014) sehingga sering dikombinasikan dengan TF-IDF untuk meningkatkan efektivitas pemrosesan teks.

2.2.9 Support Vector Machine (SVM)



Gambar 2. 2. Ilustrasi Support Vector Machine

sumber, (Samsudiney, 2020)

Support Vector Machine (SVM) adalah salah satu metode machine learning yang digunakan dalam analisis data, termasuk analisis sentimen, dengan mencari hyperplane terbaik yang memisahkan data dari berbagai kelas (Vapnik, 1995). Seperti yang ditunjukkan pada Gambar 2.2, SVM bekerja dengan menentukan decision boundary ($w^Tx = 0$) yang memisahkan dua kelas data serta membangun dua support hyperplane ($w^Tx = \pm 1$) yang berada pada jarak maksimal dari titik data terdekat (support vectors). Hyperplane ini berfungsi sebagai batas keputusan yang memaksimalkan margin antar kelas, sehingga meningkatkan kemampuan model dalam melakukan klasifikasi (Vapnik & N., 1995). Dalam kasus multi-class classification (memiliki label lebih dari 2) memiliki dua metode yaitu

1) One-Versus-One (OVO)

   One-Versus-One (OVO) adalah metode klasifikasi multi-kelas yang membagi setiap pasangan kelas menjadi submasalah biner. Dengan N kelas, OVO membangun ((N(N-1))/2) classifier, masing-masing membedakan dua kelas. Selama prediksi, setiap model memberikan suara, dan kelas dengan suara terbanyak dipilih sebagai hasil akhir. OVO efektif menangani distribusi kelas yang berbeda, tetapi memerlukan lebih banyak model, sehingga lebih lambat dan membutuhkan sumber daya komputasi lebih besar.

2) One-Versus-All (OVA)

   One-Versus-All (OVA) menangani klasifikasi multi-kelas dengan membangun satu classifier per kelas. Setiap model mengenali satu kelas sebagai positif dan lainnya sebagai negatif, menghasilkan N classifier

untuk N kelas. Pada prediksi, kelas dengan skor tertinggi dipilih. OVA lebih efisien dibanding OVO, tetapi bisa kesulitan jika data tidak terdistribusi baik atau terdapat ketidakseimbangan antar kelas.

Dalam penelitian ini, SVM akan diterapkan untuk mengklasifikasikan umpan balik mahasiswa di Politeknik Caltex menjadi sentimen positif, negatif, dan netral. Dengan memanfaatkan fitur-fitur yang diekstraksi dari teks umpan balik, SVM dapat secara efektif memodelkan hubungan antara fitur tersebut dan kategori sentimen, sebagaimana prinsip yang terlihat dalam Gambar.2.2.

2.2.10 MongoDB

MongoDB adalah salah satu database *NoSQL* yang dirancang untuk menangani data dalam format dokumen (*document-oriented database*). MongoDB menggunakan struktur data BSON (Binary JSON), yang memungkinkan penyimpanan data semi-terstruktur dan fleksibel. Database ini sangat populer untuk aplikasi modern, terutama yang membutuhkan skalabilitas tinggi dan fleksibilitas dalam pengelolaan data (Chodorow, 2013).

2.2.11 PyMongo

PyMongo merupakan pustaka Python yang digunakan untuk berinteraksi dengan basis data NoSQL MongoDB. Pustaka ini menyediakan antarmuka pemrograman yang memungkinkan pengguna untuk melakukan operasi basis data, seperti menyimpan, memperbarui, membaca, dan menghapus dokumen di dalam koleksi MongoDB (MongoDB, 2023). Dengan menggunakan PyMongo, pengembang dapat memanfaatkan model data berbasis dokumen MongoDB yang fleksibel, mendukung berbagai tipe data kompleks, serta memungkinkan pengelolaan data dalam skala besar.

2.2.12 Application Programming Interface (API)

Application Programming Interface (API) adalah mekanisme yang memungkinkan dua aplikasi atau lebih untuk saling berkomunikasi dan bertukar data. Dalam pengembangan sistem berbasis *machine learning*, API sering digunakan untuk menghubungkan model dengan aplikasi frontend atau layanan lain. API bertindak sebagai perantara yang menerima permintaan data dari klien, memproses data tersebut melalui model, dan mengembalikan hasil prediksi atau klasifikasi kepada klien.

2.2.13 Flask

Flask adalah salah satu *framework* Python yang ringan dan fleksibel, sering digunakan untuk membangun aplikasi web, termasuk API yang mendukung pengembangan sistem berbasis machine learning. Flask memungkinkan pengembang untuk mengintegrasikan model *machine learning*, seperti SVM, ke dalam layanan berbasis web yang dapat menerima input data, memprosesnya menggunakan model, dan mengembalikan hasil klasifikasinya secara *real-time*. Dalam konteks ini, Flask digunakan untuk membuat API yang memungkinkan data baru dikirimkan ke model SVM untuk diklasifikasikan. Hasil klasifikasi tersebut kemudian dapat diteruskan ke database dan sistem visualisasi untuk memperbarui informasi yang ditampilkan kepada pengguna. Keunggulan Flask terletak pada desainnya yang sederhana, modular, dan mudah diimplementasikan, menjadikannya pilihan yang populer untuk aplikasi berbasis machine learning (Grinberg, 2014).

2.2.14 Confusion Matrix

Confusion Matrix merupakan alat yang efektif dalam mengidentifikasi performa model klasifikasi, terutama dalam kasus ketidakseimbangan kelas yang signifikan dengan cara memetakan hasil klasifikasi yang dihasilkan oleh algoritma SVM dengan hasil aktualnya (Kotsiantis et al., 2006) dengan klasifikasi seperti Tabel dibawah ini

Table 2. 2. Tabel Confusion Matrix

	Prediksi			
Aktual	Kelas	Positif	Negatif	Netral
	Positif	True Positif (TP)	False Negative (FN)	Netral Positif (NP)
	Negatif	False Positif (FP)	True Negatif (TNG)	Negatif Netral (NGN)
	Netral	Positif Netral (PN)	Netral Negatif (NNG)	True Netral (TN)

Dari Tabel 3.3.1 ini dapat dijelaskan bahwa

1) *True Negatif* (TP) ini menunjukan dalam model memprediksi data dengan label positif dengan benar
2) *False Negatif* (FN) ini menunjukan dalam model memprediksi data dengan label Positif, padahal label aktual adalah Negatif
3) *Netral Positif* (NP) Ini menunjukkan bahwa model memprediksi data

dengan label Netral, padahal label aktualnya adalah Positif.

4) *False Positif* (FP) Ini menunjukkan bahwa model memprediksi data dengan label Positif, padahal label aktualnya adalah Negatif.
5) *True Negatif* (TNG) Ini menunjukkan bahwa model memprediksi data dengan label Negatif dengan benar.
6) *Negatif Netral* (NGN)Ini menunjukkan bahwa model memprediksi data dengan label Netral, padahal label aktualnya adalah Negatif.
7) *Positif Netral* (PN) Ini menunjukkan bahwa model memprediksi data dengan label Positif, padahal label aktualnya adalah Netral.
8) *Netral Negatif* (NNG) Ini menunjukkan bahwa model memprediksi data dengan label Negatif, padahal label aktualnya adalah Netral.
9) *True Netral* (TN) Ini menunjukkan bahwa model memprediksi data dengan label Netral dengan benar.

Dari pengujian ini Suatu model dapat dikatakan memiliki performa yang baik jika :

1) Pada Perhitungan Confusion Matrix ini nilai *True Positif* (TP), *True Negative* (TNG), dan *True Neutral* (TN) memiliki jumlah yang tinggi
2) Distribusi kesalahan yang rendah di semua kelas
3) Metrik evaluasi Accuracy dan F1-Score bernilai Tinggi
4) Precision, dan Recall bernilai seimbang

Dengan memahami distribusi prediksi model terhadap label aktual, kita dapat mengevaluasi tingkat akurasi serta keseimbangan antara Precision, Recall, dan F1-Score. Model yang baik tidak hanya memiliki tingkat akurasi yang tinggi tetapi juga distribusi kesalahan yang rendah di semua kelas. Oleh karena itu, analisis *Confusion Matrix* menjadi langkah krusial dalam memastikan model yang dihasilkan memiliki performa yang optimal dan dapat digunakan secara efektif dalam berbagai aplikasi.

2.2.15 End-to-End Testing (E2E)

End-to-End Testing (E2E) merupakan bagian penting dari testing pipeline dalam siklus pengembangan perangkat lunak, terutama dalam *Continuous Integration/Continuous Deployment* (CI/CD). *Testing pipeline* adalah serangkaian tahapan otomatis yang memastikan kualitas perangkat lunak sebelum dirilis ke

lingkungan produksi. E2E testing biasanya ditempatkan pada tahap akhir pipeline setelah pengujian unit (*unit testing*) dan pengujian integrasi (*integration testing*), bertujuan untuk mensimulasikan skenario dunia nyata guna memastikan bahwa seluruh sistem bekerja dengan baik secara keseluruhan. Dengan mengintegrasikan E2E testing ke dalam pipeline, tim pengembang dapat mendeteksi bug lebih awal, memastikan kompatibilitas antar komponen, dan meningkatkan stabilitas aplikasi sebelum sampai ke pengguna akhir. Menurut (Abdelfattah et al., 2023) dalam jurnal "*End-to-End Test Coverage Metrics in Microservice Systems: An Automated Approach*", pendekatan otomatis dalam pengujian E2E sangat penting dalam arsitektur mikroservis untuk meningkatkan efektivitas pipeline pengujian dan memastikan cakupan pengujian yang optimal.

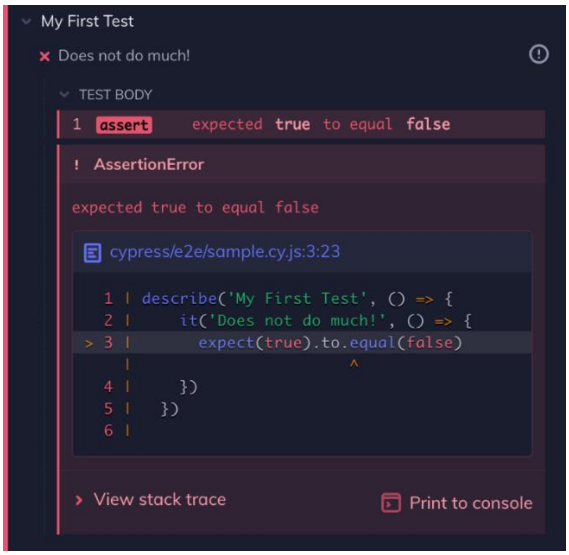
Dalam penelitian ini, Cypress akan digunakan untuk menguji tampilan dan fungsionalitas website, sedangkan Selenium dan *WebDriver* akan digunakan dalam pipeline pengujian berbasis Python untuk mengotomatiskan proses pengujian dan memastikan stabilitas aplikasi. Adapun tahapan yang akan dilakukan dalam E2E testing ini yaitu

1) Membuat rancangan skenario pengujian yang mencakup berbagai alur kerja pengguna, seperti login, navigasi, pengisian formulir, dan interaksi lainnya.
2) Menyiapkan lingkungan pengujian dengan menginstal dan mengonfirgurasi alat pengujian yang diperlukan serta data uji dan lingkungan pengujian yang sesuai
3) Membuat skrip pengujian mengunakan Cypress, Celenium dan webDriver untuk pengujian sistem sesuai dengan skenario yang telah dibuat
4) Melakukan uji coba skrip pengujian kita secara lokola untuk memastikan semua fungsi bekerja

Gambar 2. 3 Cypress testing berhasil

sumber , (Samsudiney, 2020)

Jika hasil testing menampilkann seperti Gambar 2.3 maka dapat lanjut ke tahapan skenario selanjutnya sementara jika



Gambar 2. 4. Cypress testing error
sumber, (Samsudiney, 2020)

Hasil pengujian seperti Gambar 2.4 dimana ditemukan bug atau kegagalan dalam pengujian maka pengembang akan memperbaiki kode dan mengulang proses pengujian untuk memastikan semua fungsi berjalan dengan baik

Dengan menerapkan tahapan ini, pengujian E2E dapat memastikan bahwa seluruh sistem bekerja dengan optimal, sehingga aplikasi lebih stabil dan siap digunakan oleh pengguna.

2.2.16 User Acceptance Testing (UAT)

User Acceptance Testing(UAT) merupakan proses verifikasi yang menegaskan bahwa solusi yang telah dikembangkan dalam sistem telah memenuhi persyaratan yang diinginkan oleh pengguna. Proses ini berbeda dengan pengujian sistem (yang bertujuan untuk memastikan kestabilan perangkat lunak dan kesesuaian dengan dokumen permintaan pengguna), karena UAT berfokus pada kepastian bahwa solusi dalam sistem akan berfungsi sesuai harapan pengguna (yakni, uji coba bagaimana pengguna merespons solusi yang ada di dalam sistem). UAT biasanya dilaksanakan oleh klien atau pengguna akhir, dan umumnya tidak menitikberatkan pada masalah-masalah kecil seperti kesalahan pengejaan, ataupun masalah serius yang dapat menghentikan proses, seperti *crash* perangkat lunak. Pada tahap awal pengujian fungsionalitas, pengujian integrasi, dan tahap pengujian

sistem, penguji dan pengembang bekerja sama untuk mengidentifikasi dan memperbaiki masalah-masalah semacam itu

Pelaksanaan UAT meliputi beberapa langkah yaitu:

1) Perencanaan UAT dilakukan dengan menyusun skenario pengujian dan kriteria keberhasilan
2) Lingkungan pengujian disiapkan agar menyerupai kondisi operasional.
3) klien atau penguna akhir menguji fitur sistem sesuai skenario uji dan mencatat kendala yang ditemukan.
4) Hasil pengujian didokumentasikan dan dihitung tingkat keberhasilanya melalui rumus

$$Persentase = \frac{Jumlah\ Skor}{Jumlah\ Maksimal}\ x\ 100\%$$

Rumus 2. 4. Perhitungan Persentase UAT

5) Perbaikan dilakukan berdasarkan temuan UAT. Terakhir, penguji memutuskan apakah sistem dapat diterapkan atau