BAB II

LANDASAN TEORI

2.1 Penelitian Terdahulu

Tabel 2 1 Penelitian terdahulu

No	Jurnal	Judul	Penulis	Metode	Hasil
1	CESS (Journal of Computer Engineering System and Science) p-ISSN :2502-7131 Vol. 5 No. 2 Juli 2020	Analisis Sentimen Zoom Cloud Meetings di Play Store Menggunakan Naïve Bayes dan Support Vector Machine	Nuraeni Herlinawati, Yuri Yuliani, Siti Faizah, Windu Gata, Samudi Samudi	NB dan SVM	untuk NB nilai akurasi = 74,37% Sedangkan untuk algoritma SVM nilai akurasi = 81,22%. dapat diketahui bahwa tingkat akurasi yang didapatkan algoritma Support Vector Machine (SVM) lebih unggul 6,85% dibandingkan algoritma Naïve Bayes (NB).
2	Jurnal Teknik Informatika dan Sistem Informasi Vol.8,No.2,Juni 2021	Analisis Sentimen Pada Ulasan Pengguna Aplikasi Bibit Dan Bareksa Dengan Algoritma KNN	Alusius Dwiki Adhi Putra, Safitri Juanita	KNN	Akurasi untuk aplikasi Bibit 85,14%,91,91%,dan 76,44% Aplikasi Bareksa 81,70%,87,15%,75,73%

3	Unnes Journal of Mathematics Vol.10,No.2 Desember 2021	*Analisis Sentimen Aplikasi Gojek Menggunakan Support Vector Machine dan K Nearest Neighbour*	M.Nurul Muttaqin, Iqbal Kharisudin	SVM dan KNN	KNN memperoleh akurasi 82,14%,82,28%,dan 95,43% SVM memperoleh akurasi 87,98%,88,55% dan 95,43%
4	IJNMT (International Journal of New media Technology), Vol.8,No.1,2021	Sentiment Analysis about Indonesian LawyersClub Television Program Using K-Nearest Neighbor, Naïve Bayes Classifier, and Decision Tree	Nico Nathanael Wilim, Raymond Sunardi Oetama	*KNN, NB, dan DT*	Akurasi sebesar 76,94% diperoleh algoritma KNN. Namun,penggunaan dataset pada tahun yang berbeda menunjukkan bahwa akurasi tertinggi diperoleh algoritma NB
5	Sinkron:Jurnal dan Penelitian Teknik Informatika,Vol. 8, No.1, 2023	Sentiment Analysis on App Reviews Using Support Vector Machine and Naïve Bayes Classification	Marchenda Fayza Madjid,Dian Eka Ratnawati,Bayu Rahayudi Marchenda	*TF-IDF,NB dan SVM*	SVM memperoleh akurasi 94,29% dan NB memperoleh akurasi 93,97%

6	Jurnal Rekayasa Sistem dan Teknologi Informasi, Vol.7,No.1 ,2023	Naïve Bayes and TF-IDF for Sentiment Analysis of the Covid-19 Booster Vaccine	Imelda, Arief Ramdhan Kurnianto	*Naïve Bayes*	Naïve Bayes memperoleh akurasi 85,26%
7	International Research Journal of Engineering and Technology Vol.7,No.7,2020 20	Support Vector Machine versus Naïve Bayes Classifier : A Juxtaposition of Two Machine Learning Algorithms for Sentiment Analysis	Ananya Arora, Prayag Patel, Saud Shaikh , Prof. Amit Hatekar	*SVM, NB*	Naïve Bayes Sedikit lebih baik secara keseluruhan
8	Indonesian Journal of Computer Science Vol.12,No.1,2023	Analisa Sentimen pengguna social media Twitter terhadap perokok di indonesia	Dewi setiyawati , Nuri Cahyono	*NB,SVM dan Random Forest*	*Naïve Bayes* memiliki hasil akurasi terbesar dengan nilai akurasi tertinggi 62,1%

9	Jurnal Nasional Teknik Elektro dan Teknologi Informasi Vol.10,No.2,2021	Analisis Sentimen Masyarakat terhadap Tindakan Vaksinisasi dalam Upaya Mengatasi Pandemi Covid-19	Brian Laurensz, Eko Sediyono	*NB,SVM*	Hasil akurasi tertinggi diraih oleh *Naïve Bayes* dengan angka rata-rata 85,59% berbanding sedikit dengan SVM dengan angka rata-rata 84,41%
10	Jurnal Sistem Komputer dan Informatika Vol.4,No.2,2022	Analisis Sentimen Terhadap Program Kampus Merdeka Menggunakan Naïve Bayes dan Support Vector Machine	Irma Putri Rahayu, Ahmad Fauzi, Jamaludin Indra	*NB* dan *SVM*	hasil menunjukan bahwa metode SVM memperoleh tingkat akurasi terbesar dengan angka sebesar 93%

Berdasarkan tinjauan dari sepuluh artikel jurnal terkait analisis sentimen, terlihat bahwa algoritma yang paling sering digunakan adalah *Naïve Bayes* (NB) dan *Support Vector Machine* (SVM), serta beberapa algoritma lain seperti K-*Nearest Neighbor* (KNN) dan *Decision Tree* (DT). Dalam penelitian pertama yang menganalisis ulasan Zoom di Google Play Store, SVM mencapai akurasi 81,22%, lebih tinggi dibandingkan NB yang memperoleh 74,37%. Penelitian kedua tentang review marketplace menunjukkan SVM sebagai yang paling unggul dengan akurasi 93,65%, mengalahkan NB dan KNN. Penelitian ketiga yang memprediksi arah

pergerakan pasar saham menggunakan SVM menghasilkan akurasi sebesar 89,93%. Dalam penelitian keempat mengenai program TV *Indonesian Lawyers Club*, KNN mencetak akurasi 76,94%, tetapi pada dataset yang berbeda, NB menunjukkan akurasi tertinggi. Penelitian kelima terhadap aplikasi Allo Bank menunjukkan akurasi NB sebesar 93,97%, sedikit lebih rendah dari SVM dengan 94,29%. Penelitian keenam menggunakan NB pada data Twitter tentang vaksin booster dan mendapatkan akurasi sebesar 85,26%. Penelitian ketujuh menunjukkan bahwa NB sedikit lebih unggul dibanding SVM secara keseluruhan. Pada penelitian kedelapan, analisis terhadap isu perokok di Twitter menunjukkan bahwa NB memperoleh akurasi tertinggi sebesar 62,1%, mengalahkan SVM dan DT. Penelitian kesembilan mengenai sentimen vaksinasi menunjukkan NB unggul tipis dengan rata-rata akurasi 85,59%, dibandingkan SVM yang mencapai 84,41%. Terakhir, penelitian kesepuluh tentang program Kampus Merdeka menunjukkan bahwa SVM memberikan akurasi tertinggi sebesar 93%. Secara keseluruhan, meskipun kedua algoritma sering digunakan, SVM cenderung memberikan hasil akurasi yang lebih tinggi dalam sebagian besar penelitian, meski ada beberapa kasus di mana NB lebih unggul.

2.2 Teori Penelitian

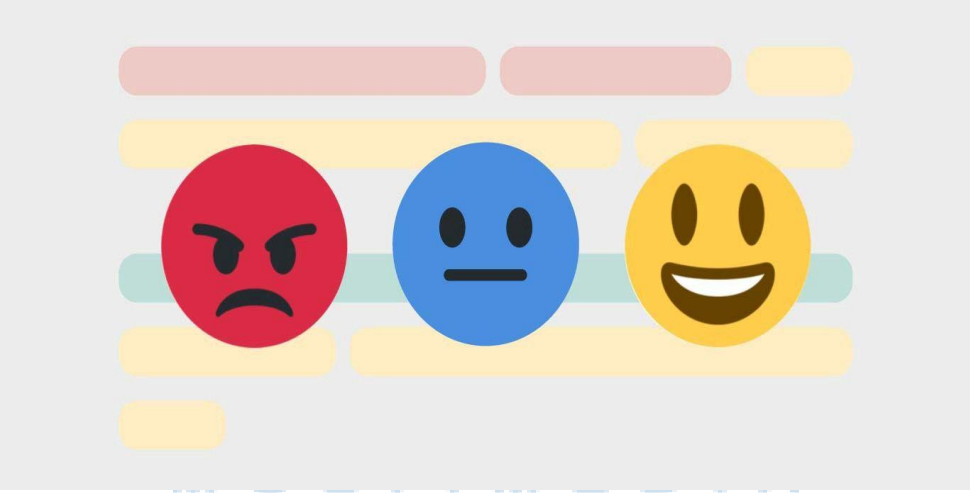
2.2.1 Analisis Sentimen

Analisis sentimen adalah cabang dari pengolahan bahasa alami (*Natural Language Processing*/NLP) yang berfokus pada identifikasi sikap, opini, atau emosi yang terkandung dalam teks. Tujuan utamanya adalah untuk menentukan apakah pendapat atau perasaan yang terkandung dalam suatu teks adalah positif, negatif, atau netral. Analisis ini banyak digunakan untuk memahami bagaimana pengguna merespons produk, layanan, atau aplikasi, serta untuk menggali informasi terkait persepsi publik terhadap suatu objek atau entitas[21].

Proses analisis sentimen dimulai dengan pengumpulan data teks yang dapat diperoleh dari berbagai sumber, seperti ulasan pengguna,

komentar di media sosial, atau forum *online*. Setelah data terkumpul, tahap berikutnya adalah prapemrosesan data. Prapemprosesan meliputi beberapa langkah, antara lain tokenisasi, yang membagi teks menjadi kata atau kalimat; penghapusan *stopwords*, yang menghilangkan kata-kata tidak bermakna seperti "dan" atau "di"; dan *stemming*, yang mengubah kata menjadi bentuk dasarnya. Proses ini bertujuan untuk mempersiapkan data agar lebih mudah dianalisis.

Setelah data diproses, langkah selanjutnya adalah penerapan algoritma klasifikasi untuk mengidentifikasi sentimen dalam teks. Beberapa algoritma yang umum digunakan dalam analisis sentimen termasuk *Naïve Bayes*, *K-Nearest Neighbors* (KNN), *Support Vector Machine* (SVM), *Random Forest*, dan *Decision Tree*. Algoritma-algoritma ini bekerja dengan menganalisis pola dalam teks dan memprediksi apakah sentimen yang terkandung di dalamnya bersifat positif, negatif, atau netral.



Gambar 2 1 Analisis Sentimen[22]

2.2.2 Teori Bayes

Teorema Bayes adalah sebuah prinsip dalam teori probabilitas yang digunakan untuk memperbarui probabilitas suatu peristiwa berdasarkan informasi baru yang diperoleh. Teorema Bayes menghubungkan dua

probabilitas—probabilitas awal (sebelum ada bukti baru) dan probabilitas yang diperbarui (setelah memperhitungkan bukti baru)[23].

Teorema Bayes dinyatakan dengan rumus berikut:

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

Penjelasan setiap komponen dalam rumus tersebut adalah sebagai berikut:

- P(A|B)P(A|B): Probabilitas kejadian AA terjadi setelah diketahui bahwa BB telah terjadi. Ini disebut **probabilitas bersyarat** dari AA mengingat BB.
- P(B|A)P(B|A): Probabilitas kejadian BB terjadi jika diketahui bahwa AA terjadi. Ini juga disebut sebagai **likelihood**, yaitu seberapa besar kemungkinan bahwa bukti BB diberikan peristiwa AA.
- P(A)P(A): Probabilitas awal dari kejadian AA, atau **prior probability**, yang menunjukkan seberapa besar kemungkinan AA terjadi sebelum bukti BB diperhitungkan.
- P(B)P(B): Probabilitas dari kejadian BB, atau **marginal likelihood**, yang berfungsi sebagai faktor normalisasi agar hasilnya merupakan probabilitas yang valid.

2.2.3 Text Mining

Text Mining adalah proses untuk mengekstraksi informasi yang bermanfaat dan pola dari teks yang tidak terstruktur dengan menggunakan berbagai teknik dari pengolahan bahasa alami (NLP) dan pembelajaran mesin. Tujuan utama dari *text mining* adalah untuk menggali wawasan dari data teks yang luas dan tidak terstruktur, yang sulit dianalisis secara langsung oleh manusia[24].

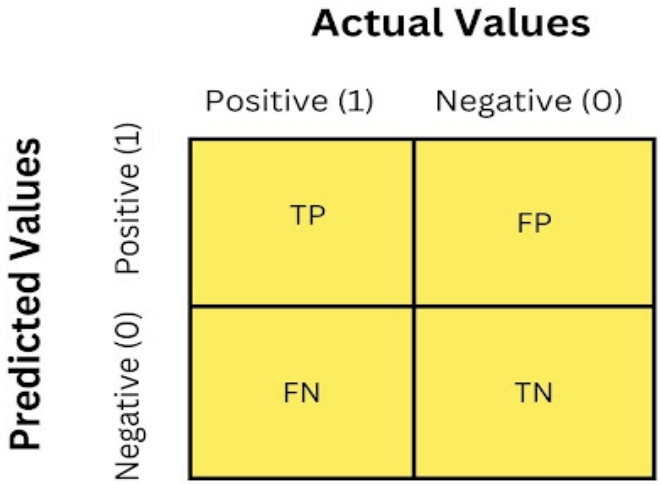
Proses *text mining* dimulai dengan prapemprosesan teks, yang mencakup langkah-langkah seperti tokenisasi, di mana teks dibagi menjadi kata atau kalimat untuk mempermudah analisis. Selanjutnya,

stopwords yang tidak bermakna dihapus, seperti kata "dan", "atau", atau "di". Setelah itu, *stemming* dilakukan untuk mengubah kata menjadi bentuk dasar atau akarnya, misalnya "memesan" menjadi "pesan". Selain itu, *lemmatization* juga digunakan untuk mengubah kata-kata menjadi bentuk dasar berdasarkan konteks dalam kalimat.

Text mining banyak digunakan dalam analisis sentimen, pengelompokan dokumen, dan pemahaman topik dalam teks, serta berperan penting dalam membantu mengekstraksi informasi berharga dari data teks yang besar.

2.2.4 Confusion Matrix

Confusion Matrix adalah sebuah alat yang dapat digunakan untuk mengukur performa dari hasil klasifikasi sebuah *machine learning*, Bentuk dari *confusion matrix* adalah tabel ringkasan yang dinamakan matriks 2 dimensi dan berisikan jumlah prediksi yang benar dan salah dari sebuah model klasifikasi[25].



Gambar 2 2 Confusion Matrix

Gambar 2.2[26] adalah bentuk tabel *confusion matrix* paling sederhana yang terdiri dari 4 kolom. 4 kolom tersebut akan terbentuk apabila *output* dari prediksi model hanya 2 buah sedangkan, apabila hasil keluaran dari prediksi model lebih dari 2 buah maka kolom dan

baris pada tabel juga akan bertambah banyak. Pada dasarnya, tabel *confusion matrix* memiliki makna sebagai berikut:

a. *True Positive* (TP): jumlah prediksi positif dan tepat

b. *False Positive* (FP): jumlah prediksi yang positif dan salah

c. *False Negative* (FN): jumlah prediksi yang negatif dan salah

d. *True Negative* (TN): jumlah prediksi negatif yang tepat

Confusion matrix memiliki 4 rumus antara lain:

1. *Precision*

   Precision menyatakan rasio prediksi positif yang tepat dari seluruh prediksi positif yang ada. Rumus 2.1 merupakan rumus *precision.*

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

Rumus 2 1Precision

2. *Recall*

   Recall menyatakan rasio jumlah prediksi positif dari kelas positif yang dapat diprediksi dengan tepat oleh model. Rumus 2.2 merupakan rumus *Recall.*

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

Rumus 2 2Recall

3. FI-*score*

   FI-*score* memberikan rasio mengenai gabungan *precision dan recall* yang biasanya memiliki hubungan terbaik. Rumus 2.3 merupakan rumus FI-*score.*

$$F1score = 2 \frac{Precision \times Recall}{Precision + Recall}$$

Rumus 2 3F1-Score

4. *Accuracy*

   Accuracy atau akurasi akan memberikan informasi terkait jumlah prediksi yang tepat dari total seluruh prediksi yang dilakukan. Rumus 2.4 merupakan rumus *Accuracy.*

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

Rumus 2 4Accuracy

2.2.5 Web Scraping

Web Scraping adalah teknik untuk mengumpulkan data dari situs web secara otomatis dengan menggunakan program atau skrip. Teknik ini memungkinkan pengguna untuk mengekstrak informasi yang ada di halaman web seperti teks, gambar, tabel, dan elemen lainnya. Dengan menggunakan alat seperti Python dan pustaka seperti *BeautifulSoup* atau *Scrapy*, *web scraping* dapat mengambil data dalam jumlah besar dan mengubahnya menjadi format yang lebih terstruktur, seperti CSV atau database, untuk analisis lebih lanjut atau penggunaan lainnya[27].

2.2.6 TF-IDF

TF-IDF adalah metode yang digunakan untuk mengukur signifikansi relatif suatu kata dalam suatu dokumen dalam kumpulan dokumen[28]. *Term Frequency* (TF) mengukur seberapa sering suatu kata muncul dalam dokumen, sementara *Inverse Document Frequency* (IDF) menilai seberapa unik kata tersebut dalam seluruh kumpulan dokumen. Dengan mengalikan TF dan IDF, kita mendapatkan nilai TF-IDF, yang mengidentifikasi kata-kata kunci dalam dokumen.

Penggunaan TF-IDF umumnya untuk ekstraksi fitur dalam analisis teks, pengelompokan dokumen, dan pencarian informasi relevan. Metode ini membantu memahami pentingnya suatu kata dalam konteks dokumen tertentu. TF-IDF dapat menemukan kata-kata yang paling sering dicantumkan dalam sebuah ulasan seperti “baik” , “bagus”, “buruk” , “jelek” , dan lainnya. Pada umumnya penggunaan *feature extraction* seperti TF-IDF ini dapat meningkatkan performa akurasi algoritma yang digunakan dalam melakukan analisis sentiment[29].

2.3 Framework dan Algoritma Penelitian

2.3.1 Knowledge Discovery in Databases (KDD)

Framework KDD (Knowledge Discovery in Databases) adalah proses yang digunakan untuk menemukan pengetahuan atau pola yang berguna dalam data yang besar dan kompleks. KDD melibatkan serangkaian langkah untuk mengekstraksi informasi yang dapat digunakan dari basis data atau kumpulan data yang belum diproses. Proses ini mencakup berbagai teknik dari data mining, statistik, dan pembelajaran mesin untuk menemukan hubungan, tren, atau pola tersembunyi dalam data[30].

Secara umum, framework KDD terdiri dari beberapa tahap, yaitu:

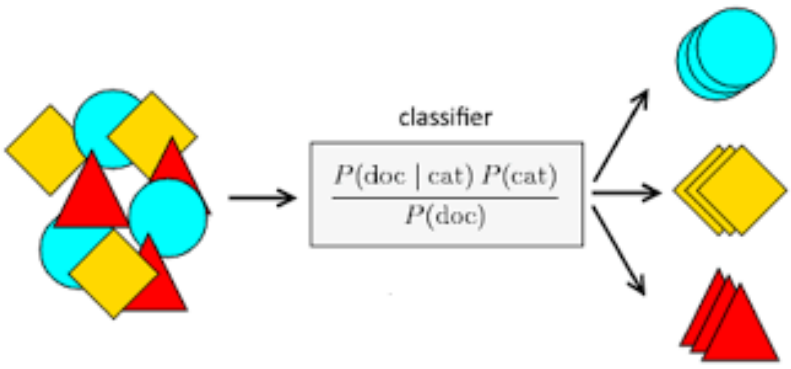
- *Selection*: Memilih data yang relevan dari berbagai sumber untuk analisis.
- *Preprocessing:* Membersihkan dan mempersiapkan data agar dapat dianalisis dengan lebih baik, seperti mengatasi data yang hilang, duplikasi, atau kesalahan.
- *Transformation*: Mengubah data ke dalam format yang sesuai atau membentuk fitur baru untuk analisis lebih lanjut.
- *Data Mining*: Menerapkan teknik seperti klasifikasi, *clustering*, atau regresi untuk menemukan pola dalam data.

- *Interpretation/Evaluation:* Menginterpretasikan hasil yang ditemukan dan mengevaluasi kualitasnya untuk memastikan pola yang ditemukan memiliki kegunaan praktis dan dapat diterapkan.

KDD banyak digunakan dalam berbagai bidang seperti analisis pasar, kesehatan, keuangan, dan lainnya, di mana data yang kompleks perlu dianalisis untuk mendapatkan wawasan yang berguna.

2.3.2 Naïve Bayes

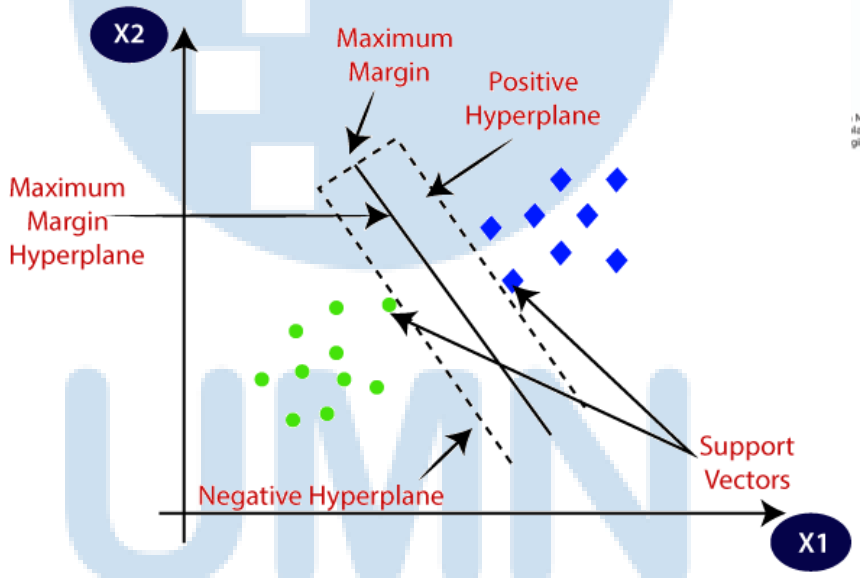
Naïve Bayes adalah algoritma klasifikasi probabilistik yang berdasarkan teorema Bayes. Dalam *text mining*, *Naïve Bayes* sering digunakan untuk analisis sentimen dan kategorisasi dokumen. Meskipun sederhana, algoritma ini efektif dan cepat. Menggunakan asumsi kemandirian naif, *Naïve Bayes* menganggap fitur-fitur dalam teks tidak memiliki ketergantungan satu sama lain. Dengan melibatkan perhitungan probabilitas, *Naïve Bayes* dapat mengklasifikasikan teks ke dalam kategori atau kelas yang sesuai, menjadikannya pilihan populer dalam pemrosesan teks untuk tugas-tugas seperti klasifikasi dokumen atau analisis sentimen.



Gambar 2 3 Naive Bayes[31]

2.3.3 Support Vector Machine (SVM)

Support Vector Machine (SVM) adalah algoritma machine learning yang digunakan untuk klasifikasi dan regresi. Dalam *text mining*, SVM digunakan untuk klasifikasi dokumen dan analisis sentimen. SVM menciptakan *hyperplane* yang memaksimalkan margin antara kelas-kelas, memisahkan data teks ke dalam kategori yang berbeda[32]. SVM efektif dalam mengatasi dimensi tinggi dan mampu menangani data teks kompleks. Kelebihan utamanya adalah kemampuan generalisasi yang baik dan ketangguhan terhadap *overfitting*. Dengan menerapkan konsep ini, SVM menjadi alat yang kuat dalam memahami pola dan mengklasifikasikan teks dalam berbagai konteks.

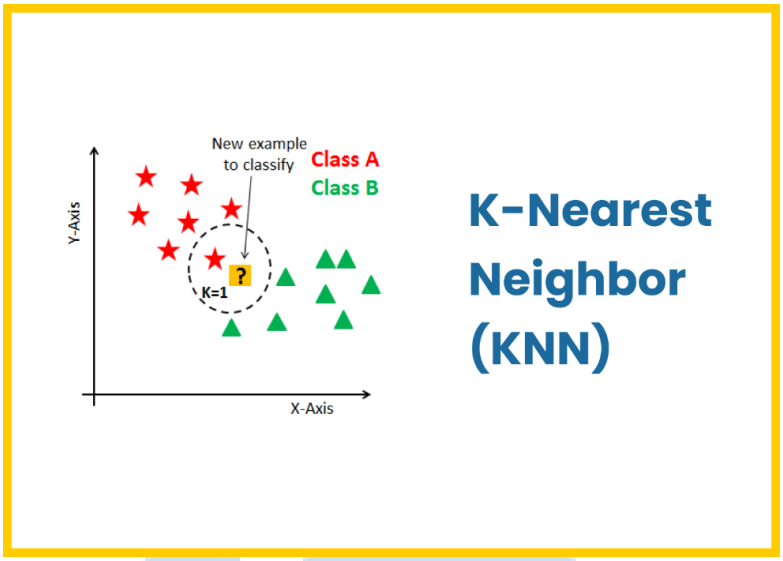


Gambar 2 4 Support Vector Machine[33]

2.3.4 K-Nearest Neighbors (KNN)

K-Nearest Neighbors (KNN) adalah algoritma pembelajaran mesin yang digunakan untuk tugas klasifikasi dan regresi. KNN bekerja dengan cara menghitung jarak antara data yang akan diprediksi dengan titik data lainnya dalam ruang fitur menggunakan metrik seperti *Euclidean Distance*. Setelah itu, algoritma ini mencari 'k' tetangga terdekat dan memprediksi kelas atau nilai berdasarkan mayoritas kelas dari tetangga-terdekat tersebut (untuk klasifikasi) atau rata-rata nilai dari

tetangga-terdekat (untuk regresi)[34]. KNN tidak membutuhkan fase pelatihan khusus, namun memerlukan banyak daya komputasi saat membuat prediksi untuk dataset besar, karena harus menghitung jarak untuk setiap titik data dalam dataset.



Gambar 2 5 K-Nearest Neighbour[35]

2.3.5 Decision Tree

Decision Tree adalah algoritma pembelajaran mesin yang digunakan untuk klasifikasi dan regresi. Algoritma ini membangun model dalam bentuk pohon keputusan, di mana setiap node internal mewakili kondisi atau pertanyaan yang dibagi berdasarkan atribut data, dan setiap cabang mewakili hasil atau keputusan yang diambil berdasarkan kondisi tersebut[15]. Daun pohon menunjukkan kelas atau nilai prediksi.

Proses pembentukan decision tree dilakukan dengan membagi dataset ke dalam subset yang lebih kecil menggunakan kriteria tertentu, seperti *Gini impurity*, *Entropy*, atau *Information Gain*, untuk menemukan pemisah terbaik. *Decision Tree* sangat mudah dipahami dan diinterpretasikan, tetapi cenderung *overfit* jika pohon terlalu dalam atau kompleks.

$$\text{Entropy (S)} = \sum^{n} - pi * \log_2 pi$$

Rumus 2 5 Rumus Decision Tree

2.3.6

Random Forest

Random Forest adalah algoritma pembelajaran mesin berbasis *ensemble* yang digunakan untuk klasifikasi dan regresi. Algoritma ini membangun beberapa pohon keputusan (*decision trees*) selama proses pelatihan dan menghasilkan prediksi berdasarkan mayoritas suara (untuk klasifikasi) atau rata-rata (untuk regresi) dari hasil pohon-pohon keputusan tersebut. Setiap pohon dalam *random forest* dibangun dengan memilih subset acak dari data pelatihan dan subset acak dari fitur untuk menentukan pembagian node[36].

Keunggulan utama *Random Forest* adalah kemampuannya untuk mengurangi risiko *overfitting* yang sering terjadi pada pohon keputusan tunggal. Selain itu, algoritma ini sangat baik dalam menangani data yang besar, memiliki banyak fitur, dan sangat kuat dalam menghasilkan prediksi yang akurat.

2.4 Tools dan Software Penelitian

2.4.1 Microsoft Excel

Gambar 2 6 Microsoft Excel

Meskipun Microsoft Excel bukanlah alat khusus untuk analisis sentimen, namun dapat digunakan sebagai pendekatan awal untuk pemrosesan dan visualisasi data teks[37]. Dalam konteks analisis sentimen, Excel dapat digunakan untuk mengelola dataset, menghitung statistik deskriptif, dan membuat grafik untuk mengidentifikasi tren sentimen.

Dengan menggunakan rumus dan fungsi Excel, pengguna dapat menghitung nilai rata-rata, deviasi standar, dan persentase sentimen positif, negatif, dan netral. Fitur filter dan pengelompokan juga dapat membantu dalam mengkategorikan dan menganalisis data sentimen. Meskipun Excel memiliki keterbatasan dalam pemahaman konteks teks dan analisis yang mendalam, kelebihannya terletak pada kemudahan penggunaan dan ketersediaan umum. Sementara untuk analisis sentimen yang lebih canggih, terutama untuk pemrosesan bahasa alami, pustaka Python seperti TextBlob atau NLTK mungkin lebih efektif.

2.4.2Python

Gambar 2 7 Python

Python adalah bahasa pemrograman yang populer dalam pengembangan alat analisis sentimen. Untuk analisis sentimen, pustaka seperti TextBlob, NLTK, dan VADER sering digunakan[38]. TextBlob menyediakan antarmuka sederhana dengan kemampuan analisis sentimen menggunakan model yang telah dilatih. NLTK, sebagai toolkit pemrosesan bahasa alami, menawarkan berbagai algoritma klasifikasi untuk analisis sentimen. VADER, fokus pada intensitas dan konteks sentimen, cocok untuk teks informal. SpaCy, pustaka pemrosesan bahasa alami, dan scikit-learn, dengan algoritma klasifikasi, juga digunakan. Pilihan alat tergantung pada kebutuhan proyek, sumber data, dan jenis teks. Integrasi dengan metode pemrosesan teks dan pemilihan model yang tepat adalah kunci keberhasilan analisis sentimen menggunakan Python.

2.4.3 Google Colabpratory

Gambar 2 8 Google Colaboratory

Google Colaboratory (Colab) adalah platform penelitian dan pengembangan berbasis web yang membawa kemudahan dan kekuatan Google ke dunia pemrograman Python[39]. Colab menyediakan lingkungan penulisan dan eksekusi kode secara daring, memungkinkan pengguna untuk membuat notebook interaktif yang dapat dijalankan di cloud. Keuntungan utama Colab adalah akses gratis ke unit pemrosesan grafis (GPU) yang mempercepat pelatihan model machine learning. Dengan integrasi yang mulus ke *Google Drive*, pengguna dapat menyimpan, berbagi, dan mengelola proyek mereka dengan mudah. Kemudahan penggunaan, GPU gratis, dan ketersediaan di berbagai perangkat menjadikan Colab alat yang sangat berguna untuk penelitian, pembelajaran, dan pengembangan proyek Python yang melibatkan analisis data dan *machine learning*.