BAB II

LANDASAN TEORI

2.1 Penelitian Terdahulu

Dalam melakukan penelitian ini tidak dapat berjalan dengan sendirinya. Terdapat beberapa penelitian terdahulu yang dapat dijadikan sebagai acuan dan perbandingan hasil yang didapat. Berikut penelitian terdahulu yang digunakan dalam penelitian:

Tabel 2. 1 Penelitian Terdahulu

Penelitian 1 [13]	
Judul	Perbandingan Metode Klasifikasi Naïve Bayes, Decision Tree, Random Forest Terhadap Analisis Sentimen Kenaikan Biaya Haji 2023 Pada Media Sosial Youtube
Peneliti dan Tahun	Muhammad Yasir, Robertus Suraji, 2023
Sumber	Jurnal Cahaya Mandalika
Tujuan	Melakukan analisis untuk mengetahui opini masyarakat terkait usulan kenaikan biaya haji lebih cenderung sentimen positif atau negatif. serta tujuan lain adalah untuk melakukan pengujian tingkat akurasi Metode *Naïve bayes*, *Decision Tree*, *Random Forest*
Metode	*Naïve bayes*, *Desicion Tree*, *Random Forest*
Hasil	Pada penelitian yang dilakukan pada opini masyarakat terhadap usulan kenaikan biaya haji mendapatkan hasil 346 komentar negatif dan 320 komentar positif. nilai akurasi yang didapatkan menggunakan metode *naïve bayes* sebesar 90% sedangkan metode *decision tree* mendapatkan nilai akurasi sebesar 83%, serta akurasi *random forest* sebesar 87%
Penelitian 2 [14]	
Judul	Analisis Sentimen Terhadap Implementasi Program Merdeka Belajar Kampus Merdeka Menggunakan *Naïve Bayes*, *K-Nearest Neighbors* Dan *Decision Tree*
Peneliti dan Tahun	Abdul Rozaq, Yessi Yunitasari, Kelik Sussolaikah, Eka Resty Novieta Sari, Restyono Ilham Syahputra, 2022
Sumber	Jurnal Media Informatika Budidarma Volume 6, Nomor 2
Tujuan	Mengetahui opini publik mengenai implementasi program merdeka belajar kampus merdeka mendapatkan nilai positif atau negatif. juga melakukan

	perbandingan beberapa metode untuk mengetahui metode mana yang memiliki nilai akurasi tertinggi
Metode	*Naïve Bayes*, *K-Nearest Neighbor* Dan *Decision Tree*
Hasil	Metode *Naïve bayes* mendapatkan nilai akurasi yang lebih baik dengan nilai akurasi 99.22% sedangkan metode KNN nilai akurasi 96.90%, metode D*ecision tree* mendapatkan nilai akurasi paling kecil dengan nilai 37,21%.
Penelitian 3 [15]	
Judul	Penerapan *Na ïve Bayes Classifier*, *K-Nearest Neighbor* (KNN) dan *Decision Tree* untuk Menganalisis Sentimen pada Interaksi Netizen dan Pemeritah
Peneliti dan Tahun	M. Khairul Anam, Bunga Nanti Pikir, Muhammad Bambang Firdaus, Susi Erlinda, Agustin
Sumber	Jurnal Manajemen, Teknik Informatika, dan Rekayasa Komputer Vol. 21, No. 1
Tujuan	Melakukan pengelompokan opini publik yang dibagi menjadi tiga kelas yaitu positif, negatif, dan netral terhadap interaksi pengguna media sosial dengan pemerintahan, serta melakukan perbandingan kinerja algoritma Naïve bayes dan SVM.
Metode	*Na ïve Bayes Classifier*, *K-Nearest Neighbor* (KNN)
Hasil	Hasil yang didapatkan pada penelitian ini setelah melakukan Terkait penerapan teknologi yang dilakukan Pemerintah Kota Metropolitan Pekanbaru mendapatkan hasil bahwa algoritma *naïve bayes* memiliki nilai akurasi sebesar 100%, algoritma KNN mendapatkan nilai akurasi sebesar 98,25%, dan algoritma yang memiliki akurasi terkecil adalah *decision tree* dengan nilai 62,28%.
Penelitian 4 [16]	
Judul	Analisis Sentimen Terhadap Kontroversi Fatwa MUI Nomor 83 Tahun 2023 Tentang Pemboikotan Produk yang Terafiliasi Israel
Peneliti dan Tahun	Muhammad Yasir, Marissa Grace Haque, Robertus Suraji, Istianingsih, 2024
Sumber	Jurnal Ekonomi Manajemen Sistem Informasi Vol. 5, No. 4
Tujuan	Penelitian ini memiliki tujuan untuk mengembakan model klasifikasi menggunakan lima metode berbeda, untuk memahami respon opini masyarakat terhadap fatwa tersebut
Metode	*Naïve Bayes*, *Decision Tree*, *Random Forest*, *Support Vector Machine* (SVM), dan *K-Nearest Neighbor*(KNN)

Hasil	Hasil dari penelitian menghasilkan 293 sentimen positif yang setuju atau mendukung terkait dengan fatwa, sedangkan terdapat 251 sentimen negatif yang tidak mendukung dengan fatwa. Algoritma *naïve bayes* mendapatkan nilai akurasi sebesar 74.80% mengungguli algoritma lain seperti *decision tree* yang mendapatkan nilai akurasi 64%, *random forest* memiliki akurasi 66%, SVM dengan akurasi 62%, dan algoritma yang mendapatkan nilai akurasi terkecil adalah KNN dengan nilai sebesar 52%
Penelitian 5 [9]	
Judul	Analisis Sentimen Terhadap Kepuasan Pelanggan Perbankan Digital di Indonesia
Peneliti dan Tahun	Bramanthyo Andrian, Tiarma Simanungkalit, Indra Budi, Alfan Farizki Wicaksono, 2022
Sumber	International Journal of Advanced Computer Science and Applications, Vol. 13, No. 3
Tujuan	Melakukan analisis terkait dengan kepuasan pelanggan perbankan digital indonesia, untuk mengetahui apakah kepuasan apakah cenderung positif atau negatif. data yang dikumpulkan berasal dari sosial media *twitterr* dan tingkat kepuasan dari tiga digital bank berbeda yaitu Jenius, Jago, dan Blu. Serta membandingkan prediksi dengan lima algoritma berbeda, untuk menentukan algoritma yang terbaik berdasarkan perhitungan nilai akurasinya.
Metode	*Naïve bayes*, *Logistic Regression*, SVM, *Decision Tree*, KNN, *Random Forest*
Hasil	Berdasarkan penelitian yang telah dilakukan terkait dengan kepuasan pelanggan bank digital di indonesia. dari 22.572 tweet yang terkumpul mendapatkan hasil 12.504 tweet merupakan sentimen positif dimana pengguna merasa puas dengan bank digital di indonesia, kemudian 5.603 tweet merupakan sentimen netral, dan 4.465 tweet merupakan sentimen negatif yang mana pengguna merasa tidak puas dengan bank digital di indonesia. Algoritma yang memiliki performa terbaik dalam memprediksi adalah *Logistic Regression* dengan nilai akurasi 74.48%, disusul oleh *SVM* dengan nilai akurasi 74.29%, *Random Forest* dengan nilai akurasi 74.19%. lalu *naïve bayes* dengan nilai 73.81%, decision tree dengan nilai akurasi 66.33%, dan algoritma yang memiliki nilai akurasi terkecil adalah *KKN* dengan nilai 40.52%
Penelitian 6 [17]	

Judul	Analisis Sentimen Pengguna Platform Belajar Online *Coursera* menggunakan *Random Forest* dengan Metode Ekstraksi Fitur *Word2vec*
Peneliti dan Tahun	Muhammad Jazaal Aufa, Anita Qoiriah, 2022
Sumber	Journal of Informatics and Computer ScienceVolume 04 Nomor 02
Tujuan	Mengetahui performa algoritma *Random Forest* dengan menggunakan fitur ekstraksi fitur *word2vec* pada analisis sentimen Pengguna Platform Belajar Online *Coursera.* Mengetahui sentimen pengguna cenderung positif, negatif, atau netral
Metode	*Random Forest*
Hasil	Hasil akurasi terbaik pada skenario data *train* dan *test* 80 : 20 mendapatkan hasil akuarasi sebesar 91%, *precision* 81%, *reacall* 89%, *f1_score* 84,6%. Sentimen positif sebesar 114043, sentimen netral 20400, negatif sebesar 6923. Berdasarkan jumlah hasil sentimen penggua aplikasi *cousera* cenderung memiliki sentimen positif.
Penelitian 7 [18]	
Judul	Perbandingan Metode *Support Vector Machine* dan *Decision Tree* Untuk Analisis Sentimen *Review* Komentar Pada Aplikasi Transportasi *Online*
Peneliti dan Tahun	Khoirul Abbi Rokhman, Berlilana, Primandani Arsi, 2021
Sumber	Jurnal Of Information System Management
Tujuan	Tujuan dari penelitian adalah untuk mengetahui akurasi klasifikasi sentiment pegguna gojek menggunakan metode SVM dan Decision Tree
Metode	*Support Vector Machine* dan *Decision Tree*
Hasil	Nilai akurasi yang di dapat algoritma Support Vectore Machine(SVM) sebesar 90.20%, sedangkan metode Decision Tree mendapatkan nilai akurasi sebesar 89.80%
Penelitian 8 [19]	
Judul	Implementasi Algoritma *Naive Bayes*, *Support Vector Machine*, dan *K-Nearest Neighbors* Untuk Analisa Sentimen Aplikasi *Halodoc*
Peneliti dan Tahun	Elly Indrayuni , Acmad Nurhadi, Dinar Ajeng Kristiyanti. 2021
Sumber	Faktor Exacta
Tujuan	Mengetahui sentimen masyarakat terhadap aplikasi halodoc cenderung ke arah sentimen positif atau negatif. melakukan komparasi algoritma untuk mengetahui kinerja algoritma yang paling baik dalam sentimen analisis

Metode	*Naive Bayes*, *Support Vector Machine*, dan KNN
Hasil	Berdasarkan hasil penelitian di dapatkan bahwa naïve bayes mendapatkan hasil akurasi sebesar 92.50% pada nilai n-gram = 4. Pada algoritma SVM mendapatkan hasil akurasi yang lebih baik dengan 93%. Sedangkan algoritma KNN mendapatkan hasil akurasi terbesar denagn nilai 95%
Penelitian 9 [20]	
Judul	Perbandingan Algoritma *Random Forest*, *Naïve Bayes*, dan *Support Vector Machine* PadaAnalisis Sentimen *Twitter* Mengenai Opini Masyarakat Terhadap Penghapusan Tenaga Honorer
Peneliti dan Tahun	Akhmad Miftahusalam, Adinda Febby Nuraini, Awalia Agustina Khoirunisa, Hasih Pratiwi. 2022
Sumber	Seminar Nasional Official Statistics
Tujuan	mengklasifikasikan opini yang beredar di masyarakat berupa opini positif, negatif, ataupun netral melalui media sosial Twitter
Metode	*Random Forest*, *Naïve Bayes*, dan *Support Vector Machine*
Hasil	Berdasarkan hasil analisis sentimen Twitter mengenai opini masyarakat terhadap penghapusan tenaga honorer didapatkan bahwa klasifikasi menggunakan metode *Random Forest* dengan penanganan data imbalanced menggunakan random overs ampling menghasilkan tingkat akurasi lebih tinggi yaitu sebesar 66,67% daripada menggunakan metode SVM dan *Naïve Bayes*. Pada penelitian ini diperoleh bahwa sentimen masyarakat mengenai kebijakan pemerintah dalam menghapus tenaga honorer pada tahun 2023 mendapat keseimbangan opini baik negatif, netral, ataupun positif.
Penelitian 10 [21]	
Judul	*Support Vector Machine VS Information Gain: Analisis* Sentimen *Cyberbullying* di *Twitter* Indonesia
Peneliti dan Tahun	Christevan Destitus, Wella, Suryasari. 2020
Sumber	Ultima InfoSys
Tujuan	Mengetahui akurasi metode *Super Vector Machine* dan *Information Gain* pada sentimen analisis tweet yang mengandung cyber bully
Metode	*Super Vector Machine* dan *Information Gain*
Hasil	Berdasarkan penelitian yang telah dilakukan algoritma SVM mendapatkan hasil matrix yang cukup tinggi dengan hasil akurasi 80%, *precision* 81%, *recall* 95%, *f-measure* 87%.

Pada penelitian terdahulu [13][14][15] belum terdapat penelitian yang mebandingkan tiga algoritma Random Forest, Decision Tree, KNN. Pada peneletian sebelumnya juga belum ada penelitian yang melakukan sentimen analisis terhadap aplikasi IDN. Berdasarkan peneltian terdahulu yang telah di temukan pada penelitian ini menggunakan algoritma Random Forest, Decision Tree, KNN. Pada penelitian terdahulu. Pada penelitian terdahulu [9] dengan skenario pembagian data penelitian 80% *data train* dan 20% *data test*. memiliki hasil akurasi menggunakan algoritma *Random Forest* sebesar 74.19%, algoritma *Decision Tree* sebesar 63%, KNN sebesar 40.52%. hasil akurasi yang di dapatkan terbilang cukup kecil. Pada penelitian ini ingin menguji ketiga kinerja algoritma agar mendapatkan hasil akurasi yang lebih tinggi. Berdasarkan saran yang di berikan penelitian terdahulu [15] kinerja algoritma dapat di tingkatkan menggunakan *feature selection* menggunakan *Chi-Square*.

2.2 Analisis Sentimen

Analisis sentimen merupakan proses memahami, mengekstrak dan mengolah data tekstual secara otomatis untuk mendapatkan informasi sentimen yang terkandung dalam kalimat opini [22].Tugas dasar yang dilakukan oleh analisis sentimen adalah mengelompokkan polaritas yang terdapat pada suatu dokumen teks, apakah pendapat yang dikemukakan bersifat negatif, positif, atau netral. Analisis sentimen juga dapat digunakan sebagai cara untuk mengumpulkan opini dari pengguna aplikasi. Terdapat 3 *level* pada analsisi sentimen yaitu[23]:

1. *Level* Dokumen

Melakukan analisis dan klasifikasi pada suatu dokumen untuk mengetahui memiliki sentimen positif atau negatif. *level* ini memiliki asumsi bahwa dokumen hanya terdapat opini tentang satu entitas saja. *Level* ini baik digunakan untuk perbandingan yang memiliki lebih dari satu entitas.

2. *Level* Kalimat

Pada *level* ini terdapat analisis terhadap kalimat untuk menentukan kalimat bermuat nilai positif, negatif atau netral. Kalimat yang bernilai positif atau negatif

bisa disebut sebagai opini sedangkan kalimat yang memiliki nilai netral tidak dapat dikatakan sebagai sebuah opini.

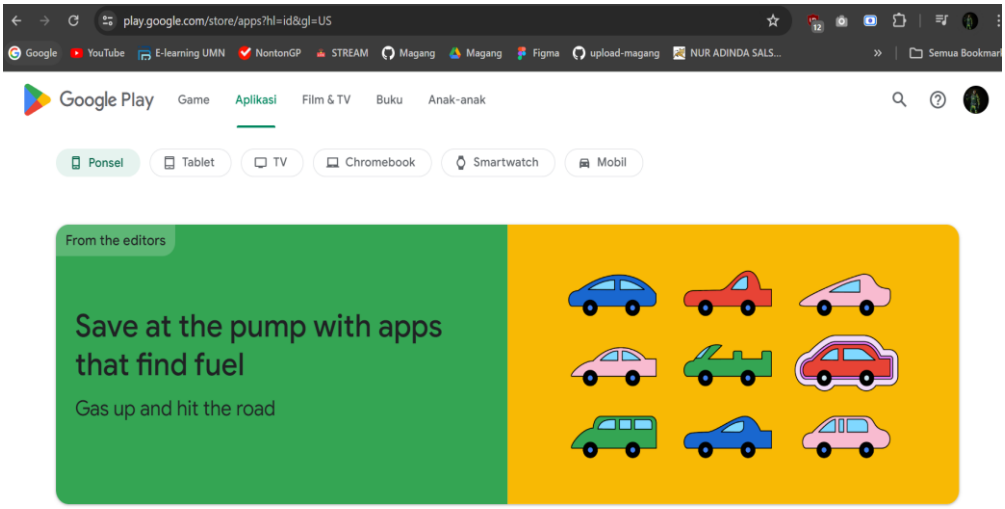
3. *Level* Entitas dan Aspek.

Pada Level ini tidak melakukan sebuah analisa kepada konstruksi bahasa (paragraf, kalimat, atau klausa) akan tetapi pada sebuah opini.

2.3 *Google Play store*

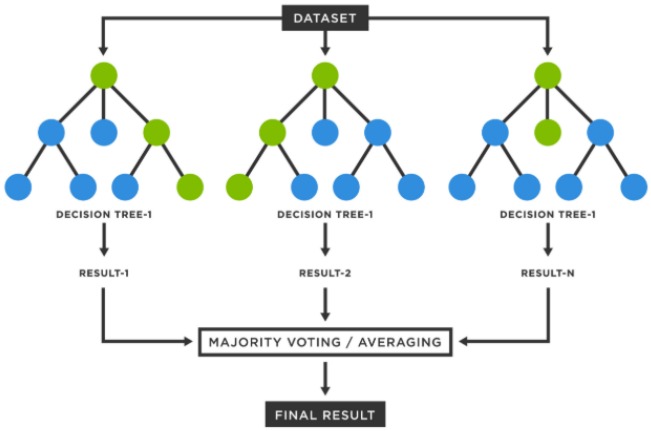
Google Play Store merupakan platform yang digunakan untuk mengundung berbagai macam aplikasi. *Google Play Store* terdapat pada *handphone* yang memiliki basis *operating system (OS)* Android. Terdapat *Google Play Store* dengan *website*. *Google Play Store* adalah sebuah *platform* dimiliki dan dioperasikan oleh *Google*. Platform ini dirancang khusus untuk perangkat berbasis *Android*, seperti *smartphone* dan *tablet*[24]. *Google Play Store* menyediakan berbagai jenis konten digital, termasuk aplikasi (aplikasi *mobile*, game, dan aplikasi lainnya), film, acara TV, buku, musik, dan majalah. Pengguna Android dapat mengakses *Google Play Store* langsung dari perangkat mereka atau melalui *web browser*. Fitur yang ada pada *Google Play Store*:

1. Mengunduh aplikasi dan game. *Google Play Store* merupakan tempat utama untuk menemukan, mengunduh, dan menginstal aplikasi dan game untuk perangkat Android. Pengguna dapat mencari aplikasi berdasarkan kategori, popularitas, ulasan pengguna, dan banyak lagi.
2. Memberikan ulasan aplikasi. Pengguna dapat memberikan penilaian dan ulasan untuk aplikasi dan game yang mereka unduh. Ini membantu pengguna lain dalam memutuskan apakah suatu aplikasi atau game layak untuk diunduh.



Gambar 2. 1 Tampilan Googleplaystore Pada Website

2.4 *Random Forest*



Gambar 2. 2 Cara kerja algoritma *Random Forest*

Sumber:[25]

Random Forest merupakan sebuah metode yang dapat digunakan untuk melakukan klasifikasi pada data dalam jumlah besar. Teknik *Random Forest* menggabungkan beberapa model *Decision Tree* keputusan yang bagus menjadi satu model. Semakin banyak pohon yang digunakan, semakin baik pula akurasinya[26]. Setiap pohon dalam model ini terdiri dari kumpulan variabel acak yang tersusun secara terstruktur. random forest menghubungkan setiap pohon dari *Decision tree* menjadi sebuah model. Jumlah *tree* yang digunakan memiliki dampak langsung terhadap nilai akurasi,presisi, *recall*, dan *f1_score* yang akan

didapatkan dari model ini. Proses pemilihan pohon dari model *decision tree* dimulai dengan mengevaluasi nilai entropy, dan *tree* yang paling baik yang ditemukan akan digunakan dalam model *Random Forest*. Cara kerja algoritma *Random Forest* yaitu [26]:

1. Langkah pertama algoritma adalah mengambil sampel acak dari dataset yang telah tersedia.
2. Kemudian, untuk setiap sampel yang terpilih, sebuah *Decision Tree* akan dibangun. Hasil rediksi berasal dari setiap *Decision Tree* yang telah dibuat.
3. Setelah itu, dilakukan voting untuk memilih hasil prediksi dari setiap pohon. Untuk klasifikasi, nilai *modus* akan dijadikan prediksi akhir, sementara untuk masalah regresi, nilai rata-rata (*mean*) akan dijadikan prediksi akhir.
4. algoritma akan memutuskan untuk memilih hasil prediksi yang mendapatkan *vote*. Prediksi yang memiliki vote terbanyak merupakan hasil prediksi akhir.

Kelebihan algoritma *Random Forest* yaitu[27]:

1. efektif dalam menangani dataset yang besar dengan banyak fitur dan observasi.
2. Dengan menggunakan proses *voting* atau *averaging* dari berbagai pohon keputusan, *Random Forest* cenderung memberikan prediksi yang lebih stabil dan anda.
3. Memliki fleksibilitas karena dapat digunakan untuk tugas klasifikasi dan regresi. Ini membuatnya berguna dalam berbagai aplikasi, baik itu dalam memprediksi kategori maupun nilai numerik.

Berikut merupakan rumus *Random Forest*:

$$\hat{c}_{rf}^{B}(x) = majority\ vote(C_b(x))_1^B$$

Rumus 2. 1 *Random Forest*

Keterangan variable:

$\hat{c}_{rf}^{B}(x) =$ Kelas prediksi dari pohon $Random\ Forest$ ke – b.

2.5 *Decision Tree*

Decision Tree adalah salah satu teknik klasifikasi yang banyak dikenal karena kemudahan interpretasinya bagi manusia. Algoritma ini mengadopsi model prediksi dengan struktur pohon atau hierarki[28]. Konsep dasar algoritma *Decision Tree* adalah membuat pohon keputusan yang didalamnya terdapat sebuah data. Keunggulan *Decision Tree* terletak pada kemampuannya untuk mengurai proses keputusan yang rumit menjadi langkah yang sederhana.

Algoritma *Decision Tree* disebut sebagai pohon keputusan karena strukturnya menyerupai bentuk pohon. Data yang terdapat dalam *Decision Tree* ditampilkan dengan bentuk tabel yang memiliki atribut dan catatan. Dalam *Decion Tree* terdapat node yang mewakili atribut, cabang dari pohon keputusan merupakan hasil dari pengujian, dan node daun mewakilkan kelompok dari sebuah kelas. Terdapat tiga jenis node dalam *Decision Tree*[20], yaitu:

1. *Root*, *Node* yang paling atas dan tidak terdapat *input* dan outputnya bisa lebih dari Satu
2. *Internal Node*, merupakan cabang node dan hanya memiliki satu *input* dan outputnya minimal dua
3. *Leaf Node*, *node* yang paling akhir hanya memiliki satu *input* dan tidak terdapat output

Berikut merupakan rumus *Decision Tree*[29]:

$$Entropi(s) = \sum_{j=1}^{k} - P_j \log_2 P_j$$

Rumus 2. 2 Decision Tree

Keterangan variable:

S = Himpunan dataset

K = Jumlah partisi S

P_j = Hasil probabilitas keputusan “ya”

2.6 *K Nearest Neighbors*

*K-Nearest Neighbors (*KNN) adalah algoritma yang digunakan untuk klasifikasi dalam *supervised learning*. Algoritma ini mengklasifikasikan data berdasarkan pembelajaran (*train datasets*) dan memilih nilai K tetangga terdekatnya (*nearest neighbors*). Dalam KNN, nilai K menunjukkan jumlah tetangga terdekat yang digunakan untuk memprediksi kelas suatu data [30]. Algoritma ini menggunakan pendekatan *Memory-based Classification*, dimana *training example* digunakan langsung saat waktu eksekusi. Berikut merupakan rumus KNN:

$$Distance = \sqrt{\sum_{i=1}^{n} (x_{training}^{i} - x_{testing})^2}$$

Rumus 2. 3 KNN

Keterangan variable:

$x_{training}^{i}$ = Data *train* ke- i

$x_{testing}$ = Data *testing*

i=1 = Baris ke- dari table

n = Jumlah data *train*

2.7 *Jupyter NoteBook*

Jupyter Notebook merupakan sebuah aplikasi yang bisa digunakan setiap orang yang dapat digunakan untuk membuat dokumen phyton[31]. Aplikasi Jupyter notebook di ciptakan oleh Perez dan Granger. Nama awal aplikasi ini adalah *Ipython Notebook*. Aplikasi ini dapat dijalankan menggunakan *web browser*. Tedapat beberapa produk dari *Jupyter* [32]:

1. *Jupyter Notebook*

Merupakan web base yang dapat digunakan untuk membuat dokumen phyton.

2. *Jupyter Hub*

Merupakan sebuah server yang menyimpan dan menjalankan *jupyter notebook*.

2.8 *Anaconda Navigator*

Anaconda Navigator adalah sebuah aplikasi desktop yang menawarkan antarmuka grafis (GUI) untuk mengelola lingkungan pengembangan dan proyek data ilmiah menggunakan distribusi *Python* yang dikenal sebagai *Anaconda* [33]. *Anaconda Navigator* mempermudah proses instalasi, manajemen, dan penggunaan berbagai paket dan alat yang umum digunakan dalam pengembangan dan analisis data, seperti *Jupyter Notebook*, *JupyterLab*, *Spyder*, dan lainnya. *Anaconda Navigator* dapat membuat pengguna dapat dengan mudah membuat dan mengatur lingkungan virtual *Python*, memungkinkan mereka untuk memisahkan dependensi dan versi paket yang digunakan dalam setiap proyek. Ini membantu menghindari konflik paket dan memastikan konsistensi dalam pengembangan. Anaconda Navigator menyediakan akses mudah ke berbagai paket dan perangkat lunak yang sering digunakan dalam analisis data dan ilmu data, seperti *numpy*, *pandas*, *matplotlib*, *scikit-learn*, dan banyak lagi. Anaconda Navigator juga terintegrasi dengan lingkungan pengembangan yang populer seperti *Jupyter Notebook* dan *Spyder*.

2.9 *Visual Studio*

Visual Studio adalah sebuah lingkungan pengembangan terintegrasi (IDE) yang dikembangkan oleh Microsoft. IDE ini digunakan oleh para pengembang perangkat lunak untuk membuat aplikasi berbasis Windows, aplikasi *web*, aplikasi mobile, dan banyak lagi menggunakan berbagai bahasa pemrograman seperti C#, Visual Basic, C++, Python, dan lainnya. Salah satu keunggulan Visual Studio adalah kaya fitur [34]. IDE ini menyediakan berbagai alat bantu yang membantu pengembang dalam setiap tahap pengembangan, mulai dari menulis kode, debugging, hingga merilis aplikasi. Fitur-fitur yang dimaksud yaitu penyorotan *sintaks*, pemeriksa kode (*code inspection*), refaktorisasi, pengelolaan versi, integrasi dengan sistem kontrol versi seperti Git, dan banyak lagi.

2.10 *Scraping*

Scraping adalah sebuah metode yang memiliki fungsi untuk mendapatkan data atau informasi yang berasal dari sebuah *website* serta dilakukan secara proses dilakukan secara otomatis. Scraping merupakan teknik yang dapat digunakan untuk

menggali sebuah informasi dari sebuah *website* [35]. cara kerja *web* scraping adalah dengan cara menelusuri dokumen HTML dari sebuah *web*. Tujuan dari penggunaan scraping adalah untuk mendapatkan sebuah data secara otomatis serta dalam waktu yang cepat, data yang terkumpul dapat disimpan dengan format *CSV*.

2.11 *Data Preprocessing*

Data preprocessing merupakan sebuah teknik yang dapat digunakan pada data mining yang memiliki fungsi mengolah dan mempersiapkan data yang awalnya masih belum terstruktur menjadi data yang terstruktur dan siap untuk digunakan dalam penelitian [36]. Teknik data preprocesing membersihkan seluruh data dengan cara mengeluarkan data yang tidak diperlukan dalam penelitian. Dalam teknik data preprocessing terdapat tahapan yang dapat dilakukan yaitu [36]:

1.Case Folding

Pada tahap ini proses merubah seluruh karakter pada data yang awalnya huruf kapital menjadi huruf kecil.

2.Cleaning

Pada tahap ini merupakan proses untuk menghilangkan atribut/elemen yang tidak diperlukan untuk penelitian. Contoh atribut yang akan dihilangkan seperti emoji, tanda baca, dan karakter kosong.

3.Normalization

Pada tahap ini merupakan proses untuk memperbaiki sebuah kata. Kata yang diperbaiki adalah kesalahan pada sebuah ejaan kata atau *typo* agar kata tersebut memiliki makna yang dimaskud.

4.*Stopword Removal*

Pada tahapan ini merupakan proses untuk menghilangkan/menghapus kata-kata yang tidak memiliki pengaruh terhadap penelitian.

5.Stemming

Pada tahapan ini merupakan proses untuk menghilangkan imbuhan pada sebuah kata sehingga kata tersebut menjadi kata dasar.

2.12 *Term Frequency - Inverse Document Frequency (TF-IDF)*

TF-IDF adalah sebuah teknik yang dapat berfungsi memberi bobot/nilai kepada kata yang terdapat di dokumen. Tujuan untuk menggunakan teknik TF-IDF adalah melakukan evaluasi terhadap suatu kata untuk mengukur seberapa penting kata tersebut dalam sebuah dokumen. Metode *TF-IDF* sangat umum untuk dijumpai pada penelitian sentimen analisis karena mudah untuk digunakan dan hasil yang akurat. Tedapat dua faktor penting dalam *TF-IDF* yaitu [37]:

1. *Term Frequency* (TF) memberikan fokus untuk mengukur seberapa sering kata muncul pada sebuah data/dokumen. Cara untuk mendapatkan TF adalah dengan melakukan perhitungan dengan rumus total kata yang muncul dibagi dengan total keseluruhan kata pada dokumen.
2. *Inverse Document Frequency* (IDF) memberikan fokus untuk mengukur seberapa penting kata dalam dokumen. Kata-kata dalam dokumen yang lebih sedikit memiliki IDF yang tinggi. Cara untuk mendapatkan IDF adalah dengan perhitungan membagi jumlah dokumen dalam koleksi dengan jumlah dokumen yang mengandung kata tersebut.

2.13 *Python*

Python adalah adalah sebuah bahasa pemrograman yang sudah banyak dikenal yang dapat digunakan untuk mendukung pemrograman yang berorientasi pada objek. Phyton dapat digunakan pada berbagai macam operating system seperti windows, Unix, MacOs[27]. Bahasa pemrograman *pyhton* dapat digunakan oleh seorang developer untuk membuat sebuah aplikasi, membuat perintah komputer, serta melakukan analisa data.

Terdapat *natural language toolkit* pada *python* yang memiliki fungsi untuk proses klasifikasi.Bahasa pemrograman ini juga memiliki library untuk membantu untuk data klasifikasi, *data mining.* Kerangka kerja *Python* dalam analisis data yaitu [27]:

1. *Numpy* adalah kerangka kerja untuk perhitungan numerik.
2. *SciPy* adalah sebuah modul yang digunakan untuk matematika.

3. *Scikit-Learn* adalah machine learning yang digunakan untuk mengambil data, sehingga bisa melakukan *preprocessing*, dan klasifikasi.

Kelebihan yang dimiliki *python* yaitu:

1. Penembangan program yang lebih efisien karena lebih cepat dan coding sedikit.
2. *Python* dapat digunakan pada berbagai macam platform.
3. Pengelolaan memori yang dilakukan secara otomatis.
4. Memiliki sifat *Object Oriented Programing (OOP)*.

2.14 *Confusion Matrix*

Confusion Matrix adalah sebuah teknik yang dapat digunakan dalam penelitian. *Confusion matrix* digunakan untuk mengetahui hasil akurasi pada konsep *data mining* [38]. Untuk menguji keakuratan hasil maka akan dilakukan evaluasi yang bertujuan mendapatkan hasil *recall, precision, f1_score*. Metode *Confusion Matrix* akan diketahui model yang digunakan bekerja dengan baik atau tidak. *Confusion matrix* dapat digunakan sebagai alat untuk menentukan seberapa baik *clasifier* mengenali tuple yang berasal dari kelas yang berbeda. TP dan TN menunjukan bahwa *classifier* benar, kemudian FP dan FN menunjunkan bahwa *classifier* salah. Tabel confusion matrix yaitu [38]:

Tabel 2. 2 TP, FP, FN, TN

	Actual Positive	***Actual Negative***
Predicted Positive	*True Positive* (TP)	*False Positive (FP)*
Predicted Negative	*False Negative* (FN)	*True Negative (TN)*

Keterangan pada istilah tabel 2.11 sebagai berikut:

a) *True Positive* (TP): banyaknya data dari kelas sesungguhnya yang memiliki nilai positif dan hasil prediksi memiliki nilai positif.
b) *True Negative* (TN): banyaknya data dari kelas sesungguhnya yang memiliki nilai positif dan hasil prediksi memiliki nilai negatif
c) *False Negative* (FN): banyaknya data dari kelas sesungguhnya yang memiliki nilai negatif dan hasil prediksi memiliki nilai positif
d) *False Positive* (FP): banyaknya data dari kelas sesungguhnya yang memiliki nilai negatif dan hasil prediksi memiliki nilai negatif

Dalam melakukan perhitungan akurasi terdapat rumus untuk melakukan perhitungan. Berikut rumus dalam confusion matrix yaitu:

Tabel 2. 3 Rumus Confusion Matrix

Accuracy	$\frac{(\mathbf{TP} + \mathbf{TN})}{(\mathbf{TP} + \mathbf{FP} + \mathbf{TN} + \mathbf{FN})}$
Recall	$\frac{\mathbf{TP}}{\mathbf{TP} + \mathbf{FN}}$
Precision	$\frac{\mathbf{TP}}{\mathbf{TP} + \mathbf{FP}}$
F1_Score	$\frac{(\mathbf{TP})}{\mathbf{TP} + \mathbf{1/2}(\mathbf{FP} + \mathbf{FN})}$

Keterangan pada tabel 2.12 yaitu:

1. Akurasi: $\frac{(\mathbf{TP + TN})}{(\mathbf{TP + FP + TN + FN})}$

Rumus 2. 4 Akurasi *Confusion Matrix*

Hasil dari perhitungan akurasi menggunakan rumus diatas akan mendapatkan rasio data yang benar terdeteksi dalam pengujian. Nilai akurasi dapat menunjukan seberapa dekat nilai prediksi dengan nilai aktual.

2. Recall: $\frac{\mathbf{TP}}{\mathbf{TP + FN}}$

Rumus 2. 5 Recall *Confusion Matrix*

Rumus recall bertujuan untuk mendapatkan nilai positif yang berhasil di prediksi dari keseluruhan nilai positif yang sebenarnya. Nilai recall dapat menunjukan seberapa besar kelas positif.

3. Precision: $\frac{\mathbf{TP}}{\mathbf{TP + FP}}$

Rumus 2. 6 *Precision Confusion Matrix*

Rumus precision diatas bertujuan untuk mendapatkan nilai positif dari prediksi data yang mengembalikan nilai positif. Recall dapat menunjukan tingkat keberhasilan dan sensitivitas model dalam menemukan informasi.

4. F1_Score: $\frac{(\mathbf{TP})}{\mathbf{TP} + \mathbf{1/2}(\mathbf{FP} + \mathbf{FN})}$

Rumus 2. 7 *F1_Score Confusion Matrix*

Rumus F1_Score diatas bertujuan untuk mendapatkan hasil nilai rata-rata dari hasil precision dan recall. Hasil F1 score dapat dijadikan sebagai perbandingan rata-rata nilai keduanya.

2.15 *Chi-Square*

Chi-square adalah sebuah teknik statistik yang digunakan untuk menentukan apakah terdapat hubungan antara dua variabel kategori atau lebih dalam suatu populasi [39]. Chi-Square adalah prosedur statistik untuk menentukan perbedaan antara data yang diamati dan yang diharapkan. Tes ini juga dapat digunakan untuk menentukan apakah berkorelasi dengan variabel kategori dalam data kita. Hal ini membantu untuk mengetahui apakah perbedaan antara dua variabel kategori disebabkan oleh kebetulan atau adanya hubungan di antara keduanya.

2.16 *Select K best*

Metode *SelectKBest* adalah salah satu metode dalam *feature selection* yang digunakan dalam analisis data dan *machine learning* untuk memilih fitur-fitur yang paling penting dari *dataset* [40]. Tujuannya adalah untuk mengurangi dimensi *dataset* dengan memilih subset fitur-fitur yang paling relevan atau informatif untuk memprediksi variabel target. Cara kerja selectkbest adalah dengan menetukan nilai k. nilai “k” merupakan jumlah fitur yang ingin diseleksi.

2.17 *N-Gram*

N-Gram adalah salah satu teknik yang dapat digunakan untuk menyimpan kata-kata yang telah dopotong dari sebuah kalimat berdasarkan jumlah karakternya [41]. Penerapan metode terdapat tiga gram yaitu: *unigram* (n=1), *bigram* (n=2), dan *trigram* (n=3). Dengan menggunakan pendekatan ini, teks dibagi menjadi bagian-bagian kata atau karakter berdasarkan jumlah yang ditentukan, memungkinkan algoritma untuk memperoleh pemahaman yang lebih baik tentang struktur dan konteks dari data teks. Hal ini dapat membantu meningkatkan akurasi dan efisiensi dalam proses klasifikasi, memungkinkan sistem untuk mengenali pola yang lebih

kompleks dalam data. Contoh dari penggunaan *Unigram*, *Bigramm* , dan *Trigram* Misalnya, kalimat "Saya suka makan nasi" yaitu:

1. *Unigram* (N=1):[Saya, suka, makan, nasi]
2. *Bigram* (N=2): [Saya suka, suka makan, makan nasi]
3. *Trigram* (N=3): [Saya suka makan, suka makan nasi]

2.18 *Streamlit*

Streamlit adalah sebuah *framework open-source* yang digunakan untuk membuat aplikasi *web* interaktif dengan menggunakan *Python* secara cepat dan mudah. Pengguna dapat membuat antarmuka pengguna (UI) untuk aplikasi *web* tanpa perlu memiliki pengetahuan mendalam tentang pemrograman web atau *HTML/CSS* [42]. *Streamlit* memliki keunggulan yang terletak pada kesederhanaannya. Dengan menggunakan sintaks yang mirip dengan menulis skrip Python biasa, pengguna dapat membuat aplikasi web interaktif dengan cepat. Streamlit menyediakan berbagai komponen dan *widget* yang dapat digunakan untuk membuat berbagai jenis elemen *UI* seperti tombol, *input* teks, *plot*, dan tabel data.

Salah satu fitur utama Streamlit adalah kemampuannya untuk secara otomatis melakukan *refresh* ketika ada perubahan dalam kode *Python*, sehingga pengguna dapat melihat perubahan langsung pada *web* tanpa perlu melakukan *reload* halaman secara manual. Streamlit sangat cocok digunakan untuk berbagai keperluan, mulai dari visualisasi data, *prototyping* aplikasi, hingga pembuatan dashboard interaktif.

2.19 *CRISP-DM*

CRISP-DM, singkatan dari *Cross-Industry Standard Process for Data Mining*, adalah sebuah kerangka kerja yang populer digunakan untuk proyek-proyek data mining dan analitik [43]. Ia memberikan pendekatan terstruktur dalam mengarahkan tahapan-tahapan proyek, dimulai dari pemahaman masalah bisnis hingga penerapan model akhir. Terdapat enam tahap dalam *CRISP-DM* [43]:

1. Pemahaman Bisnis: pada tahap ini adalah memahami tujuan dan kebutuhan proyek dari sudut pandang bisnis.

2. Pemahaman Data: pada tahap ini melibatkan pengumpulan dan eksplorasi data yang akan digunakan untuk analisis. Data dikaji dari berbagai sumber, dan evaluasi kualitasnya dilakukan. Pemahaman awal tentang struktur, konten, dan hubungan data diperoleh.
3. Persiapan Data: Setelah pemahaman, langkah selanjutnya adalah mempersiapkan data untuk pemodelan. Pada tahap ini akan dilakukan pembersihan data, transformasi ke format yang sesuai, dan seleksi variabel yang relevan.
4. Pemodelan: Pada tahap ini, berbagai teknik pemodelan diterapkan pada data yang telah dipersiapkan untuk membangun dan mengevaluasi model-model prediktif atau deskriptif.
5. Evaluasi: Model yang dipilih dievaluasi untuk memastikan kesesuaian dengan tujuan bisnis dan kinerja yang baik pada data yang tidak terlihat sebelumnya. Penilaian model menggunakan teknik evaluasi yang sesuai.
6. Implementasi: Setelah evaluasi model, model yang terpilih diimplementasikan dalam lingkungan operasional. Ini melibatkan integrasi model dalam proses bisnis, pemantauan performa, dan dukungan berkelanjutan.