BAB II
TINJAUAN PUSTAKA DAN DASAR TEORI

2.1 Penelitian Terdahulu

Dalam penelitian ini, algoritma C5.0, *Random Forest* (RF), dan *Support Vector Machine* (SVM) dipilih karena kemampuan mereka yang unggul dalam menganalisis data kompleks dan menangkap interaksi antar fitur, yang penting untuk analisis sentimen mengenai privasi data. C5.0, sebagai pengembangan dari C4.5, menawarkan efisiensi dan akurasi yang lebih baik. Sementara itu, RF meningkatkan kinerja klasifikasi melalui pendekatan *ensemble*, dan SVM efektif untuk data *non-linear*. Metode *Naive Bayes* tidak digunakan dalam penelitian ini karena alasan kebaruan, di mana metode ini sudah sering diterapkan dalam berbagai kasus analisis sentimen dan kombinasi metode analisis sentimen lainnya, sehingga tidak memberikan nilai tambah baru dalam konteks penelitian ini.

Penelitian Aufar dkk (2020), menjelaskan bahwa media sosial menjadi *platform* bagi pengguna untuk menyampaikan opini secara terbuka melalui komentar dan berbagi informasi tanpa batas waktu. YouTube adalah salah satu media sosial yang masih populer, dengan banyak konten video yang sering dimanfaatkan untuk promosi produk. Analisis dilakukan terhadap komentar publik tentang produk Nokia, yang sebelumnya merupakan merek global terkemuka sebelum mengalami penurunan pada tahun 2013. Analisis sentimen digunakan untuk mengklasifikasikan opini positif, negatif, dan netral terkait produk Nokia guna menilai kualitas produknya. Hasil menunjukkan bahwa sebagian besar komentar didominasi oleh sentimen netral. Tahapan mencakup persiapan data, pemrosesan data, dan evaluasi model. Kinerja model diuji menggunakan metrik akurasi, presisi, *recall*, dan *F1-measure*. Algoritma *Decision Tree* (metode C4.5) menghasilkan akurasi 89,4%, sedikit lebih tinggi dibandingkan *Random Forest* dengan akurasi 88,2%.

Shanmugavadivel dkk (2024), membahas pentingnya analisis sentimen dalam penelitian komunitas *daring*, khususnya terkait peningkatan penggunaan penulisan campuran *code* di media sosial multibahasa. Teks campuran Tamil-

Inggris dianalisis menggunakan pendekatan *supervised learning*, dengan algoritma *Decision Tree* sebagai metode utama untuk klasifikasi sentimen. Hasilnya menunjukkan bahwa *Decision Tree* mencapai akurasi 99% dan skor *F1-score* rata-rata makro 0,39, diikuti oleh *Random Forest* dengan akurasi 99% dan *F1-score* rata-rata makro 0,35. Algoritma lain yang diuji, seperti SVM (akurasi 78%, *F1-score* rata-rata makro 0,68), *Logistic Regression* (akurasi 75%, *F1-score* rata-rata makro 0,62), dan KNN (akurasi 73%, *F1-score* rata-rata makro 0,26), juga memberikan hasil yang signifikan.

Tosunoglu dan Kocak (2023), menjelaskan bahwa teknologi deteksi intrusi sangat penting untuk mendeteksi perilaku berbahaya yang mengancam keamanan jaringan. *Mobile Ad-Hoc Networks* (MANETs) dan *Vehicular Ad-Hoc Networks* (VANETs) adalah jaringan seluler yang dapat mentransmisikan data tanpa membutuhkan peralatan khusus. Dengan desain jaringan yang terfragmentasi dan sumber daya terbatas, perlindungannya menjadi tantangan besar. Diperlukan *Intrusion Detection System* (IDS) yang dapat beradaptasi dengan tantangan ini. Pencegahan intrusi di Internet menjadi tugas penting bagi operator jaringan, namun sistem manajemen kerentanannya yang tradisional masih bergantung pada pengidentifikasi ancaman sebelumnya atau arsip lalu lintas yang berlabel, yang mahal dan sulit untuk menganalisis profil pengguna. Teknik *machine learning* yang digunakan memungkinkan deteksi berdasarkan tanda tangan atau profil kerja. Pendekatan baru yang menggabungkan dua algoritma *machine learning*, *K-Means* dan C5.0, untuk deteksi anomali di VANETs, menunjukkan hasil yang memuaskan. *K-Means* mengelompokkan data berdasarkan kesamaan, sementara C5.0 mengklasifikasikan data menggunakan pohon keputusan. Dengan empat fitur utama yang teridentifikasi, sistem ini berhasil mengklasifikasikan intrusi dengan tingkat keberhasilan mencapai 96%.

Raza dkk (2022), menjelaskan bahwa atribusi karyawan merujuk pada penurunan jumlah karyawan dalam suatu organisasi karena faktor-faktor yang tak terhindarkan, yang dapat menyebabkan kerugian besar. Berdasarkan data dari *Society for Human Resource Management* (SHRM), biaya rata-rata untuk merekrut karyawan baru adalah USD 4129, dengan tingkat atribusi mencapai 57,3% pada

tahun 2021. Untuk menganalisis atribusi karyawan, digunakan empat teknik *machine learning*, yaitu *Extra Trees Classifier* (ETC), *Support Vector Machine* (SVM), *Logistic Regression* (LR), dan *Decision Tree Classifier* (DTC), yang menunjukkan hasil bervariasi. Hasil dari penerapan metode SVM memperoleh akurasi 87%, LR 72%, dan DTC 83%, sementara ETC yang dioptimalkan mencapai akurasi 93% dengan nilai *precision*, *recall*, *F1 score*, dan ROC yang sangat baik. Pendekatan yang diajukan mengungguli penelitian terkini dan dilakukan validasi dengan *k-fold* untuk menyeimbangkan dataset. Pada dataset *10-fold*, SVM memperoleh akurasi 88%, LR 74%, DTC 84%, dan ETC mencatatkan akurasi 93%. Melalui analisis *Employee Exploratory Data Analysis* (EEDA), ditemukan bahwa pendapatan bulanan, tarif per jam, tingkat pekerjaan, dan usia merupakan faktor-faktor utama yang menyebabkan atribusi karyawan.

Rahmani dkk (2023), menyatakan bahwa prediksi spasial yang akurat mengenai kondisi drainase alami tanah sangat penting tidak hanya untuk pertanian dan pemodelan hidrologi, tetapi juga untuk pemasangan drainase bawah tanah dan sistem pembuangan limbah. Sebanyak 154 lokasi dipilih menggunakan metode *stratified random sampling*. Setiap lokasi dianalisis berdasarkan kelas drainase yang diidentifikasi melalui pemeriksaan visual inti tanah. Model elevasi digital yang dikembangkan dari data lidar digunakan untuk menghasilkan tujuh indeks medan, yang kemudian digunakan untuk memprediksi kelas drainase dengan empat model prediksi: regresi logistik multinomial dan tiga algoritma *machine learning* (*random forest*, C5.0, dan *artificial neural network*). Berdasarkan data *random validation* 30%, semua model *Digital Soil Mapping* (DSM) memberikan hasil yang serupa. *artificial neural network* memberikan akurasi keseluruhan yang lebih tinggi, yaitu 70%, dengan koefisien kappa 0,59 dan skor Brier 0,34. Model DSM lainnya, termasuk *Soil Survey Geographic Database* (SSURGO), memberikan akurasi antara 64% dan 66%, koefisien kappa antara 0,52 dan 0,54, serta skor Brier antara 0,41 dan 0,46.

Dalam penelitian Das dkk (2023), metode *Term Frequency-Inverse Document Frequency* (TF-IDF) dan *Natural Language Processing* (NLP) diterapkan untuk klasifikasi teks. Analisis difokuskan pada metode *feature*

weighting method untuk klasifikasi teks pada data tidak terstruktur, dengan menyoroti dua fitur utama, yaitu *N-Grams* dan TF-IDF, pada dataset ulasan film IMDB dan Amazon Alexa untuk analisis sentimen. Berbagai *advanced classifier algorithm*, termasuk *Support Vector Machine* (SVM), *Logistic Regression*, *Multinomial Naïve Bayes* (*Multinomial* NB), *Random Forest*, *Decision Tree*, dan *k-nearest neighbors* (KNN), digunakan untuk memvalidasi metode yang diajukan. Hasil yang diperoleh menunjukkan bahwa ekstraksi fitur dengan menggunakan TF-IDF menghasilkan peningkatan yang signifikan jika dibandingkan dengan *N-Gram*. TF-IDF memberikan akurasi tertinggi (93,81%), presisi (94,20%), *recall* (93,81%), dan *F1-score* (91,99%) saat menggunakan pengklasifikasi *Random Forest*.

Santoso dkk (2023), menyoroti bahwa Identifikasi berita *hoax* sangat penting untuk memastikan informasi yang beredar di masyarakat dapat dipercaya dan untuk membatasi penyebaran informasi palsu. Ketidakmampuan masyarakat membedakan fakta dan hoaks di media sosial dapat menyebabkan kesalahpahaman publik. Sistem deteksi hoaks dirancang menggunakan algoritma *Decision Tree* C5.0, yang merupakan pengembangan dari C4.5 dengan sejumlah keunggulan dalam proses data mining. Data sebanyak 1000 poin dikumpulkan melalui *web scraping* menggunakan kata kunci seperti "pemilu 2024", "politik", dan "checkfaktapilkadamafindo" dari situs Turnbackhoax.id dan Detik.com. Keunggulan penelitian ini terletak pada pengujian dengan skenario klasifikasi menggunakan pembobotan fitur TF-RF. Hasil uji *confusion matrix* pada algoritma C5.0 menunjukkan akurasi, presisi, dan *recall* yang bervariasi, yaitu pada data latih/uji (70/30) menghasilkan akurasi 79,33%, presisi 80,00%, *recall* 97,00%; data latih/uji (80/20) menghasilkan akurasi 79,50%, presisi 81,00%, *recall* 95,00%; dan data latih/uji (90/10) menghasilkan akurasi 72,00%, presisi 74,00%, *recall* 89,00%.

Ahakonye dkk (2023), melakukan penelitian dengan fokus pada pertumbuhan teknologi *Industrial Internet of Things* (IioT) dan *Supervisory Control and Data Acquisition* (SCADA), yang diikuti oleh peningkatan serangan yang tidak biasa dan mengancam keamanan. Meskipun ada *Intrusion Detection Systems* (IDS), sistem tersebut membutuhkan sumber daya komputasi yang besar. Pendekatan pengambilan keputusan dengan metode *feature selection* digunakan

untuk mendeteksi serangan dalam jaringan SCADA secara *real-time*. Pendekatan ini memiliki tiga tahap: persiapan data, *feature selection* dengan *Chi-Square*, dan penerapan pohon keputusan yang dimodifikasi untuk deteksi anomali. Hasil validasi menunjukkan keefektifan model yang diusulkan dalam mendeteksi ketidaknormalan dan mengurangi waktu komputasi dapat meningkatkan adaptabilitas dalam situasi *real-time*. model yang diusulkan juga menunjukan hasil yang lebih baik dibandingkan dengan alternatif dalam pengujian pada empat dataset publik. Validasi menggunakan *Cohen's kappa coefficient* (CKC) dan pengujian ketidakseimbangan data juga mengonfirmasi kehandalan model dalam situasi nyata.

Dhalaria dan Gandotra (2020), menjelaskan penggunaan ponsel seluler telah menjadi faktor utama dalam meningkatnya serangan *malware*. *Malware* sering kali tersembunyi dalam aplikasi biasa, sehingga sulit untuk mengidentifikasinya. Teknik-teknik yang sudah ada hanya dapat mengenali *malware* yang sudah dikenal sebelumnya. Pendekatan baru digunakan untuk mendeteksi *malware* pada *platform Android* dengan menggunakan fitur-fitur statis dan dinamis. *Feature selection Chi-Square* digunakan untuk memilih fitur-fitur yang berkontribusi dalam mendeteksi *malware*. Teknik *ensemble stacking* diterapkan untuk meningkatkan tingkat deteksi. Hasil uji coba menunjukkan bahwa kombinasi teknik *ensemble stacking*, yaitu K-NN-RF (*K-nearest Neighbor-Random Forest*), memberikan tingkat akurasi deteksi yang sangat baik yaitu 98,02%.

Sarra dkk (2022), menjelaskan bahwa prediksi otomatis penyakit jantung merupakan isu penting dalam kesehatan global. Pengobatan yang efektif memerlukan diagnosis penyakit jantung yang akurat. *Algoritma Support Vector Machine* (SVM) diterapkan untuk mengembangkan model klasifikasi penyakit jantung yang lebih baik. Teknik pemilihan fitur optimal berbasis statistik $\chi 2$ digunakan untuk meningkatkan akurasi prediksi. Validasi kinerja model dilakukan dengan membandingkannya dengan metode konvensional menggunakan berbagai parameter evaluasi. Hasil menunjukkan bahwa model ini mampu meningkatkan akurasi dari 85,29% menjadi 89,7%, sekaligus mengurangi beban komputasi hingga setengahnya. Hal ini menunjukkan bahwa pendekatan ini lebih unggul

dibandingkan metode lain dalam mendeteksi penyakit jantung. Secara sederhana beberapa penelitian sebelumnya dapat diringkas seperti terlihat pada Tabel 2.1

Tabel 2. 1 Ringkasan penelitian terdahulu

No.	Judul dan Tahun Penelitian	Penulis	Hasil Penelitian
1.	"Sentiment Analysis on Youtube Social Media Using Decision Tree and Random Forest Algorithm: A Case Study". (2020)	Mohammad Aufar, Rachmadita Andreswari, dan Dita Pramesti	Hasil penelitian tentang analisis sentimen pada saluran YouTube Nokia Mobile menunjukkan bahwa algoritma *Decision Tree* (metode C4.5) mencapai akurasi 89,4%, sedangkan algoritma *Random Forest* mencatat akurasi sebesar 88,2%. Dengan demikian, algoritma *Decision Tree* (C4.5) menunjukkan performa yang sedikit lebih baik dalam mengklasifikasikan sentimen dibandingkan algoritma *Random Forest*.
2.	"Deteksi Malware Android menggunakan Pemilihan Fitur *Chi-Square* dan Metode Pembelajaran Ensemble". (2020)	Meghna Dhalaria dan Ekta Gandotra	Penelitian ini bertujuan untuk meningkatkan deteksi *malware* pada platform Android dengan mengusulkan pendekatan baru menggunakan fitur-fitur statis dan dinamis. *Feature selection Chi-Square* digunakan untuk memilih fitur yang berkontribusi, dan teknik *ensemble stacking*, khususnya K-NN_RF, diterapkan untuk meningkatkan tingkat akurasi deteksi. Hasil uji coba menunjukkan keberhasilan dengan tingkat akurasi mencapai 98,02%.
3.	"Enhanced Heart Disease Prediction Based on Machine Learning and $\chi 2$ Statistical Optimal Feature Selection Model". (2022)	Raniya R. Sarra, Ahmed M. Dinar, Mazin Abed Mohammed, dan Karrar Hameed Abdulkareem	Penelitian ini berfokus pada pengembangan model klasifikasi penyakit jantung berbasis algoritma *Support Vector Machine* (SVM) dengan menerapkan *feature selection Chi-Square*. Eksperimen dilakukan pada dua dataset terkemuka penyakit jantung, dan hasilnya menunjukkan peningkatan signifikan dalam akurasi, dengan kenaikan sebesar 84,21% menjadi 89,47% pada dataset Cleveland dan 85,29% menjadi 89,7% pada dataset Statlog. Selain itu, penggunaan fitur dalam sistem berhasil dikurangi dari 14 menjadi hanya 6 fitur, mengakibatkan pengurangan beban komputasi sekitar 42%.
4.	"Predicting Employee Attrition Using Machine Learning Approaches". (2022)	Ali Raza, Kashif Munir, Mubarak Almutairi, Faizan Younas, dan Mian Muhammad Sadiq Fareed	Penelitian ini bertujuan untuk menganalisis faktor-faktor dalam organisasi yang mempengaruhi pergantian karyawan dan memprediksinya menggunakan beberapa teknik *machine learning*. Empat teknik *machine learning* yang digunakan, yaitu *Extra Trees Classifier* (ETC), *Support Vector Machine* (SVM), *Logistic Regression* (LR), dan *Decision Tree Classifier* (DTC), Hasil pengujian menunjukan SVM mencapai akurasi 88%, LR 74%, DTC 84%, dan ETC mencapai akurasi tertinggi yaitu 93%.

Tabel 2. 2 Lanjutan

No.	Judul dan Tahun Penelitian	Penulis	Hasil Penelitian
5.	"Estimating natural soil drainage classes in the Wisconsin till plain of the Midwestern U.S.A. based on lidar derived terrain indices: Evaluating prediction accuracy of multinomial logistic regression and machine learning algorithms" (2023)	Shams R. Rahmani, Zamir Libohova, Jason P. Ackerson, dan Darrell G. Schulze	Penelitian ini bertujuan untuk memprediksi drainase tanah dengan data lapangan dan topografi menggunakan model *Artificial Neural Network* (ANN). ANN sedikit lebih baik daripada model lain, tetapi *Multinomial Logistic Regression* (MNLR) dan C5.0 lebih diutamakan karena lebih mudah diinterpretasikan. Model *Digital Soil Mapping* (DSM) memberikan akurasi tertinggi sekitar 70%, validasi dengan metode *random hold-back* 30% menunjukkan hasil serupa untuk semua model DSM, dengan ANN mencapai akurasi tertinggi sekitar 70%.
6.	"A Comparative Study on TF-IDF Feature Weighting Method and its Analysis using Unstructured Dataset". (2023)	Mamata Das dan Selvakumar Kamalanathan and Pja Alphonse	Fokus penelitian ini adalah klasifikasi teks ulasan film IMDb dan produk Amazon Alexa menggunakan dua jenis fitur: N-Gram dan TF-IDF. Hasilnya menunjukkan bahwa TF-IDF lebih baik, dan algoritma terbaik adalah *Random Forest* dengan akurasi 93.81%, presisi 94.20%, *recall* 93.81%, dan *F1-Score* 91.99%. Jadi, penggunaan TF-IDF efektif, dan *Random Forest* efisien dalam klasifikasi teks.
7	"Identification of Hoax News in the Using Community TF-RF and C5.0 Tree Decision Algorithm". (2023)	Enrico Budi Santoso, Yulison Herry Chrisnanto, dan Gunawan Abdillah	Penelitian ini mengembangkan sistem identifikasi berita palsu menggunakan algoritma *Decision Tree* C5.0 untuk membedakan berita faktual dan palsu dalam media sosial. Pengujian sistem dengan berbagai skenario, termasuk pembobotan fitur seperti TF-RF, menunjukkan variasi hasil. Pengujian pada data latih (70/30) menghasilkan akurasi 79,33%, *precision* 80,00%, *recall* 97,00%. Kemudian pada data latih (80/20), menghasilkan akurasi sebesar 79,50%, *precision* 81,00%, *recall* 95,00%. Pada data latih (90/10), akurasi yang diperoleh sebesar 72,00%, *precision* 74,00%, dan *recall* 89,00%.
8	"SCADA intrusion detection scheme exploiting the fusion of modified decision tree and *Chi-Square* feature selection". (2023)	Love Allen Chijioke Ahakonye, Cosmas Ifeanyi Nwakanma, Jae-Min Lee, dan Dong-Seong Kim	Penelitian ini berfokus pada pertumbuhan IIoT dan SCADA, yang diikuti oleh peningkatan serangan yang mengancam keamanan. Untuk mengatasi hal ini, diusulkan pendekatan baru dengan *feature selection* untuk mendeteksi serangan dalam jaringan SCADA *real-time*. Berikut tahapan yang digunakan yaitu persiapan data, seleksi fitur, dan penerapan pohon keputusan yang dimodifikasi, model yang diusulkan juga menunjukan hasil yang lebih baik dibandingkan dengan alternatif dalam pengujian pada empat dataset publik. Validasi menggunakan *Cohen's kappa coefficient* (CKC) dan pengujian ketidakseimbangan data juga mengonfirmasi kehandalan model dalam situasi nyata.

Tabel 2. 3 Lanjutan

No.	Judul dan Tahun Penelitian	Penulis	Hasil Penelitian
9	"Feature Selection for Clustering and Classification Based Attack Detection Systems in Vehicular Ad-Hoc Networks". (2023)	B.A. Tosunoglu dan C. Kocak	Fokus penelitian ini adalah untuk mendeteksi intrusi dalam jaringan seluler, khususnya pada *Vehicular Ad-Hoc Networks* (VANETs) yang kompleks. Penggabungan algoritma K-Means dan C5.0 dalam penelitian ini berhasil mencapai akurasi hingga 99% dan tingkat *F1* mencapai 97,5% dalam simulasi dengan 500 kendaraan. Metode ini menawarkan solusi efektif untuk mendeteksi anomali dalam skenario VANET yang dinamis dan dapat membantu meningkatkan keamanan jaringan kendaraan.
10.	"Sentiment Analysis in *Code*-Mixed Tamil using Machine Learning Techniques". (2024)	Kogilavani Shanmugavadivel, Sowbharanika Janani J S, Navbila K, dan Malliga Subramanian	Penelitian ini menekankan pentingnya analisis sentimen dalam komunitas *online* yang menggunakan berbagai bahasa. Metode *supervised learning*, terutama dengan *Decision Tree*, menunjukkan hasil yang efisien dalam klasifikasi sentimen. *Decision Tree* mencapai akurasi 0,99 dengan *F1-Score* rata-rata makro sebesar 0,39. *Random Forest* juga memberikan hasil yang baik dengan akurasi 0,99 dan *F1-Score* rata-rata makro sebesar 0,35. Selain itu, *Support Vector Machine* mencapai akurasi 0,78 dengan *F1-Score* rata-rata makro sebesar 0,68, dan Regresi Logistik dengan akurasi 0,75 dan *F1-Score* rata-rata makro sebesar 0,62. KNN juga memberikan hasil memuaskan dengan akurasi 0,73 dan *F1-Score* rata-rata makro sebesar 0,26.

2.2 Dasar Teori

Dasar teori berisi teori-teori dari penelitian sebelumnya yang akan digunakan sebagai landasan dalam sebuah penelitian. Berikut teori-teori yang digunakan dalam penelitian ini yaitu:

2.2.1 Analisis Sentimen

Analisis sentimen adalah kegiatan yang bertujuan untuk mengelompokkan tweet ke dalam dua kategori utama, yaitu Positif dan Negatif. Hasil analisis sentimen ini nantinya dapat dimanfaatkan sebagai bagian dari proses pengambilan keputusan (Learning, 2020). Dalam analisis sentimen terdapat beberapa tahapan yang akan dilakukan yaitu, pengumpulan dataset berita, *pre-processing* meliputi (tahap *cleaning*, *case folding*, *tokenizing*, memperbaiki *slang words*, *stemming*, serta *stopword removal*), kemudian melakukan pembobotan kata menggunakan metode TF-IDF. melakukan seleksi fitur menggunakan metode *Chi-Square*. dan klasifikasi menggunakan algoritma C5.0 *Random Forest*, *Support Vector Machine* SVM. Kemudian, tahapan terakhir yang dilakukan yaitu evaluasi hasil menggunakan *confusion matrix* (Santoso, dkk, 2023).

2.2.2 YouTube

YouTube adalah jaringan media sosial terbesar di dunia yang berfokus pada berbagi video, menarik hingga satu miliar pemirsa setiap bulan. *Platform* ini tidak hanya menampilkan video yang diunggah oleh penggunanya, tetapi juga menyajikan program-program khusus yang diproduksi di studio-studionya (Kuyucu, 2019). Selain itu, YouTube menyediakan berbagai jenis konten, termasuk berita, pendidikan, hiburan, dan banyak lagi. Pengguna juga memiliki kesempatan untuk memberikan pendapat mereka melalui layanan komentar, yang memungkinkan interaksi dan diskusi di antara pengguna mengenai konten yang ditampilkan. Sehingga Dalam penelitian ini, YouTube digunakan sebagai sumber pengambilan data komentar berbasis teks yang terkait dengan berita "Ancaman Privasi Data". Data komentar tersebut diambil menggunakan teknik *scraping* dengan bantuan ekstensi *Data Miner*.

2.2.3 Bahasa Pemrograman R

R merupakan perangkat lunak statistik yang dapat diprogram secara gratis, yang dirancang khusus untuk analisis data (Ariel de Lima, dkk, 2022). RStudio adalah salah satu IDE (*Integrated Development Environment*) yang populer. Ini menyediakan konsol, editor dengan penyorotan sintaks, dan alat untuk *plotting*, riwayat, *debugging*, serta manajemen ruang kerja. RStudio tersedia dalam versi gratis dan berbayar, dan dapat digunakan di desktop (*Windows*, *Mac*, dan *Linux*) atau di *browser* yang terhubung ke RStudio Server. RStudio didirikan pada tahun 2008 dan rilis pertamanya adalah pada tahun 2011 (Niu, dkk, 2021).

2.2.4 *Text Mining*

Text mining merupakan teknologi kecerdasan buatan yang bertujuan untuk mengekstrak, menganalisis, dan memproses sejumlah besar data teks tidak terstruktur, sehingga dapat menghasilkan wawasan bisnis yang berharga bagi perusahaan. Contoh sumber tulisan yang menghasilkan banyak teks tidak terstruktur setiap harinya adalah internet, yang perlu diubah menjadi informasi yang dapat dibaca dan dipahami oleh mesin (Khan, dkk, 2020).

2.2.5 Pra-pengolahan

Pra-pengolahan teks merupakan tahap awal dalam mempersiapkan data mentah sebelum dilakukan analisis. Dalam tahapan ini akan dilakukan pembersihan data untuk menggabungkan variasi bentuk kata dan mengurangi jumlah kata dalam koleksi dokumen (Santoso, dkk, 2023). Tahapan pra-pengolahan yang akan digunakan dalam penelitian ini meliputi tahap *cleaning*, *case folding*, *tokenizing*, memperbaiki *slang words*, *stemming*, dan *stopword removal*.

1) *Cleaning*

   Tahap *cleaning* dilakukan untuk menghilangkan angka, url, emoji, dan karakter selain huruf dari komentar. Penghapusan elemen tersebut tidak akan berdampak signifikan pada analisis sentimen, sehingga menghapusnya tidak akan mempengaruhi hasil akurasi klasifikasi yang dilakukan (Learning, 2020). Berikut tahap *cleaning* dapat dilihat pada Tabel 2.4.

Tabel 2. 4 Tahap *cleaning*

Sebelum *Cleaning*	Setelah *Cleaning*
Jawabannya cuman " Jangan Bodoh !!!!	Jawabannya cuman Jangan Bodoh
Meningkatkan kewaspadaan.||Bkrja dengan ikhlas||Semangat||Pasti ada jalan keluar.||Tuhan berserta kita	Meningkatkan kewaspadaan Bkrja dengan ikhlas Semangat Pasti ada jalan keluar Tuhan berserta kita

2) *Case Folding*

Case folding adalah proses mengubah semua huruf dalam teks menjadi huruf kecil. Tujuannya untuk menjaga konsistensi dan menghindari kesalahan penulisan, sehingga kata-kata dianggap sama baik dalam huruf besar maupun kecil (Rosid, dkk, 2020). Proses *case folding* dapat dilihat pada Tabel 2.5.

Tabel 2. 5 Proses *case folding*

Sebelum *Case Folding*	Setelah *Case Folding*
Jawabannya cuman Jangan Bodoh	jawabannya cuman jangan bodoh
Meningkatkan kewaspadaan Bkrja dengan ikhlas Semangat Pasti ada jalan keluar Tuhan berserta kita	meningkatkan kewaspadaan bkrja dengan ikhlas semangat pasti ada jalan keluar tuhan berserta kita

3) *Tokenizing*

Tokenisasi adalah proses menguraikan kalimat yang lebih panjang menjadi frasa atau kata tunggal dengan tujuan membentuk *Bag of Words* (BoW) (Learning, 2020). Tabel 2.6 menunjukan tahapan dari proses *tokenizing*.

Tabel 2. 6 Proses *tokenizing*

Sebelum *Tokenizing*	Setelah *Tokenizing*
jawabannya cuman jangan bodoh	"jawabannya" "cuman" "jangan" "bodoh"
meningkatkan kewaspadaan bkrja dengan ikhlas semangat pasti ada jalan keluar tuhan berserta kita	"meningkatkan" "kewaspadaan" "bkrja" "dengan" "ikhlas" "semangat" "pasti" "ada" "jalan" "keluar" "tuhan" "berserta" "kita"

4) Memperbaiki *slang words*

Pada media sosial, pengguna seringkali menggunakan singkatan dan bahasa sehari-hari karena batasan jumlah karakter. Masalahnya timbul ketika setiap pengguna memiliki gaya penulisan dan singkatan yang berbeda. Akronim dan frasa tertentu dapat terkait dengan konteks khusus atau kelompok tertentu. Untuk mengatasi ketidaksempurnaan bahasa ini, penting untuk menggantikan singkatan dan slang dengan makna yang sesuai sehingga mesin dapat memahaminya dengan mudah (Naseem, dkk, 2021). Tabel 2.7 menunjukan tahapan proses memperbaiki *slang words*.

Tabel 2. 7 Proses memperbaiki *slang words*

Sebelum Normalisasi	Sesudah Normalisasi
"jawabannya" "cuman" "jangan" "bodoh"	"jawabannya" "cuman" "jangan" "bodoh"
"meningkatkan" "kewaspadaan" "bkrja" "dengan" "ikhlas" "semangat" "pasti" "ada" "jalan" "keluar" "tuhan" "berserta" "kita"	"meningkatkan" "kewaspadaan" "bekerja" "dengan" "ikhlas" "semangat" "pasti" "ada" "jalan" "keluar" "tuhan" "berserta" "kita"

5) *Stemming*

Stemming merupakan proses mengganti kata-kata dalam teks ke bentuk dasarnya, sehingga membantu mengurangi variasi kata dalam *Bag of Words* (BoW) (Learning, 2020). Berikut Proses *stemming* dapat dilihat pada Tabel 2.8.

Tabel 2. 8 Proses *stemming*

Sebelum *Stemming*	Setelah *Stemming*
"jawabannya" "cuman" "jangan" "bodoh"	"jawab" "cuman" "jangan" "bodoh"
"meningkatkan" "kewaspadaan" "bekerja" "dengan" "ikhlas" "semangat" "pasti" "ada" "jalan" "keluar" "tuhan" "berserta" "kita"	"tingkat" "waspada" "kerja" "dengan" "ikhlas" "semangat" "pasti" "ada" "jalan" "keluar" "tuhan" "serta" "kita"

6) *Stopword Removal*

Stopword removal adalah proses untuk menghilangkan kata-kata yang kurang penting serta hanya berfungsi sebagai penghubung dalam kalimat tanpa mempengaruhi sentimen dalam *tweet*. Hal ini membantu dalam menyederhanakan penggunaan *Bag of Words* (BoW) (Learning, 2020). Berikut proses *stopword removal* dapat dilihat pada Tabel 2.9.

Tabel 2. 9 Proses *stopword removal*

Sebelum *Stopword Removal*	Sesudah *Stopword Removal*
"jawab" "cuman" "jangan" "bodoh"	"jawab" "jangan" "bodoh"
"tingkat" "waspada" "kerja" "dengan" "ikhlas" "semangat" "pasti" "ada" "jalan" "keluar" "tuhan" "serta" "kita"	"tingkat" "waspada" "kerja" "ikhlas" "semangat" "pasti" "jalan" "keluar" "tuhan" "serta" "kita"

2.2.6 *Stemming* Nazief dan Adriani

Algoritma Nazief & Adriani menghasilkan hasil *stemming* yang lebih akurat dibandingkan dengan Algoritma Porter karena pemanfaatan kamus yang lebih lengkap. Meskipun proses ini mungkin memerlukan waktu sedikit lebih lama, tingkat akurasinya yang lebih tinggi menjadikannya lebih sesuai untuk aplikasi yang memerlukan ketelitian. Semua langkah ini dirancang untuk mempersiapkan data dengan efektif sebelum melanjutkan ke tahap analisis berikutnya (Setiabudi, dkk, 2021).

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

TF-IDF adalah metode statistik untuk mengukur pentingnya kata-kata dalam kumpulan dokumen. Ini melibatkan penggunaan dua matriks: matriks TF (*Term Frequency*) adalah matriks dua dimensi, dan IDF (*Inverse Document Frequency*) adalah matriks satu dimensi (Das, dkk, 2023). Berikut tahapan dari metode TF-IDF (Kabra & Nagar, 2023);

1. *Term frequency* (TF) adalah pengukuran statistik yang menunjukkan seberapa sering sebuah kata muncul dalam sebuah dokumen atau teks. Berikut rumus dari metode *Term frequency* (TF).

$$tf(t,d) = \frac{f(t,d)}{Total\ kemunculan\ kata\ dalam\ suatu\ dokumen} \quad (2.1)$$

Keterangan:

$tf(t,d)$: Frekuensi kemunculan kata *t* pada dokumen *d*

$f(t,d)$: Jumlah frekuensi kata *t* pada dokumen *d*

2. *Inverse document frequency* (IDF) adalah metrik yang digunakan untuk menilai pentingnya suatu kata dalam sekumpulan dokumen. Nilai IDF yang lebih kecil menunjukkan bahwa kata tersebut muncul di banyak dokumen, sedangkan nilai IDF yang lebih tinggi menunjukkan bahwa kata tersebut hanya muncul di sedikit dokumen. Berikut rumus *Inverse document frequency* (IDF) menurut (Eom & Kim, 2023).

$$idf\ (t,D) = log\ \frac{N}{jumlah\ dokumen\ yang\ mengandung\ kata\ t} \quad (2.2)$$

Keterangan:

$idf\ (t,D)$: *Inverse document frequency* kata t dalam dokumen *D*

D : Jumlah keseluruhan dokumen

N : Jumlah total dokumen dalam kumpulan *D*

3. Setelah menghitung nilai TF dan IDF, langkah berikutnya adalah menghitung nilai TF-IDF. Berikut rumus dari TF-IDF menurut (Zhuohao, dkk, 2021).

$$tf - Idf(t,d,D) = Tf(t,d) \times Idf(t,D) \quad (2.3)$$

Keterangan:

t : term (kata) yang akan dihitung dalam TF-IDF

d : dokumen tertentu di dalam kumpulan dokumen *D*

D : kumpulan dokumen

$tf\ (t,d)$: Frekuensi kemunculan sebuah kata *t* dalam dokumen *d*

$idf(t,D)$: *Inverse document frequency* kata *t* dalam dokumen *D*

2.2.8 *Chi-Square*

Chi-Square (χ^2) adalah alat untuk mengukur tingkat hubungan antara variabel-variabel tersebut secara statistik (Ali, dkk, 2020). Statistik *Chi-Square* yang tradisional sering disebut sebagai metode statistik uji kecocokan kuadrat (χ^2), berikut rumus dari uji *Chi-Square* (Li, dkk, 2022):

$$\chi 2(t,c) = \frac{N(AD-CB)^2}{(A+C)(B+D)(A+B)(C+D)} \quad (2.4)$$

t : Kata

c : Kelas/Kategori

N : Jumlah dokumen latih

A : Total dokumen pada kategori *c* yang memuat *t*

B : Total dokumen bukan kategori *c* yang memuat *t*

C : Total dokumen pada kategori *c* yang tidak memuat *t*

D : Total dokumen bukan kategori *c* yang tidak memuat *t*

Dalam seleksi fitur, perlu dihitung nilai *Chi-Square* tunggal untuk setiap kata terhadap kategori yang relevan, lalu mengurutkannya secara menurun. Semakin tinggi nilai *Chi-Square*, semakin penting fitur tersebut dalam proses klasifikasi (Putra, 2023). Persamaan 2.5 merupakan rumus untuk mencari nilai *Chi-Square* maksimum.

$$X^2\ max(t) = \max_{i=1,\dots,m} \{X^2(t, c_i)\} \quad (2.5)$$

X^2: Statistik yang menunjukkan nilai X^2 untuk kata tertentu *t* dalam kumpulan dokumen

t: Fitur yang dievaluasi untuk mengetahui pentingnya dalam membedakan kelas-kelas dalam klasifikasi teks

m: Total kelas dalam kumpulan kelas yang digunakan dalam analisis

$|c_i|$: Jumlah kelas spesifik dalam kumpulan kelas yang digunakan dalam analisis klasifikasi teks

$X^2\ max(t)$: Nilai maksimum dari X^2untuk kata tertentu *t* dalam kumpulan dokumen,

$\{X^2(t, c_i)\}$: Nilai statistik X^2 antara kata *t* dan kelas tertentu c_i.

2.2.9 Algoritma C5.0

Menurut Izzah dan Pitri (2023), Algoritma C5.0 adalah sebuah algoritma yang merupakan pengembangan dari algoritma C4.5. Algoritma ini menggunakan representasi berbentuk pohon, di mana setiap *node* merepresentasikan suatu atribut, sementara cabang-cabangnya menunjukkan nilai dari atribut tersebut dan memiliki daun yang berfungsi sebagai kelas. Proses pengambilan keputusan didasarkan pada nilai keuntungan tertinggi yang dihasilkan dari perhitungan semua atribut. Selain itu, algoritma C5.0 mampu mengidentifikasi atribut yang relevan dalam klasifikasi, terutama dalam data berdimensi tinggi. C4.5 sendiri adalah penyempurnaan dari *Iterative Dichotomizer* 3 (ID3) yang ditemukan oleh Ross Quinlan. ID3 menggunakan konsep *entropy* dan *information gain* dalam pendekatan iteratif untuk membuat pohon keputusan dengan maksimal dua kelas (Mankar dan Bhoite, 2023). Adapun rumus untuk mencari entropy menurut Pratiwi dkk (2020), yaitu:

$$Entropy\ (S) \quad = \sum_{j=1}^{k} p_j\ log_2(p_j) \quad (2.6)$$

Keterangan:

S : Himpunan kasus

K : Jumlah kelas dalam data

j : Nomor indeks yang menunjukan setiap nilai yang mungkin dari variabel yang digunakan

$|p_j|$: Proporsi dari *Pj* terhadap *S*

Setelah mendapatkan nilai *entropy S*, tahap selanjutnya yaitu mencari nilai gain dengan menggunakan persamaan berikut:

$$Gain\ (S, A) = Entropy\ (S) \quad - \sum_{i=1}^{m} \frac{|p_j|}{|S|}\ Entropy(S_i) \tag{2.7}$$

Keterangan:

A : Variabel yang digunakan

m : Jumlah kategori pada variabel *A*

$|S|$: Jumlah kasus dalam *S*

$|S_i|$: Himpunan kasus pada kategori ke-*i*

Selanjutnya adalah menghitung nilai *gain ratio*. Adapun rumus dasar yang dapat digunakan untuk perhitungan adalah sebagai berikut:

$$Gain\ Ratio = \frac{Gain\ (S, A)}{\sum_{i=1}^{m} Entropy(S_i)} \tag{2.8}$$

Gain (*S*, *A*) : Nilai gain dari suatu variabel

$\sum_{i=1}^{m} Entropy\ (S_i)$: Jumlah nilai *entropy* dalam suatu variabel.

2.2.10 Algoritma *Random Forest*

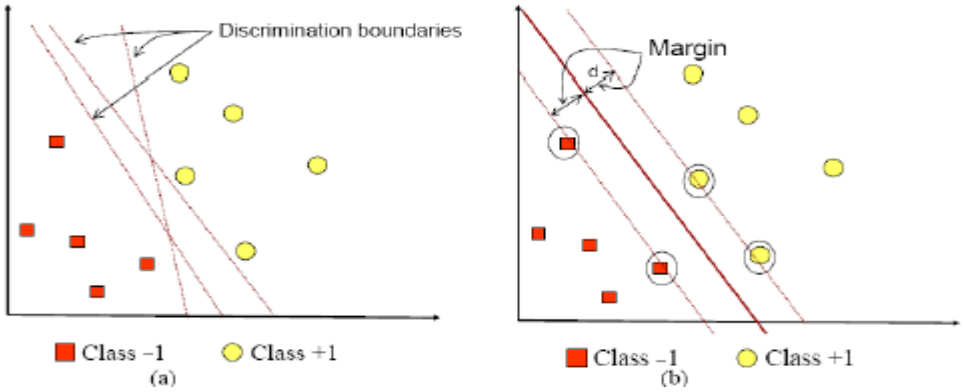
Algoritma *Random Forest* (RF) adalah model yang menggabungkan beberapa pohon keputusan menjadi satu "hutan". Dalam memahami *Random Forest*, ada dua hal penting: pengambilan sampel secara acak dan pemungutan suara mayoritas. Setiap pohon keputusan dilatih menggunakan data yang dipilih secara acak dari seluruh kumpulan data. Penelitian ini menggunakan klasifikasi pohon keputusan dengan parameter teknis yang terdapat dalam matriks fitur sebagai panduan. Setiap pohon membuat prediksi secara independen, dan *Random Forest*

menggabungkan prediksi tersebut melalui pemungutan suara, di mana prediksi dengan jumlah suara terbanyak menjadi hasil akhir. Pendekatan ini mencegah *overfitting* pada satu pohon keputusan. Kemudian, hasil klasifikasi untuk sampel ditentukan melalui pemungutan suara mayoritas. Terakhir, *rasio* antara jumlah kesalahan prediksi dan total jumlah data digunakan untuk menilai tingkat akurasi model *Random Forest* (Zheng, dkk, 2024). Rumus dari algoritma *Random Forest* dapat dilihat pada persamaan 2.6. dan 2.7.

2.2.11 Algoritma *Support Vector Machine* (SVM)

Support Vector Machine yang dikembangkan oleh Vladimir Vapnik, merupakan metode klasifikasi yang mampu menangani dataset berukuran besar secara efektif. SVM memanfaatkan kernel untuk memproyeksikan data ke ruang berdimensi lebih tinggi, sehingga memungkinkan klasifikasi data yang tidak dapat dipisahkan secara *linear*. Metode ini terkenal karena kinerjanya yang tinggi dan hanya memerlukan sedikit penyesuaian pada *hyper parameters* (Isnan, dkk, 2023).

SVM bekerja dengan mencari *hyperplane* yang memisahkan dua kelas dalam ruang fitur. *Hyperplane* ini adalah garis (dalam dua dimensi) atau permukaan (dalam dimensi lebih tinggi) yang memisahkan data dengan *margin* maksimum. Dalam penelitian ini proses klasifikasi SVM menggunakan kernel *linear*. SVM dengan kernel linier adalah algoritma pembelajaran mesin yang efektif untuk klasifikasi data, dengan tujuan utama menemukan *hyperplane* optimal yang dapat memisahkan dua kelas data dengan jelas di ruang fitur. *Hyperplane* ini memiliki *margin* maksimum, yaitu jarak terpendek antara *hyperplane* dan titik-titik terdekat dari masing-masing kelas data. Algoritma ini sangat berguna ketika data dapat dipisahkan secara langsung oleh garis, bidang, atau *hyperplane*. Proses pembelajaran pada SVM dengan kernel linier melibatkan optimasi fungsi objektif yang bertujuan memaksimalkan *margin* sambil meminimalkan norma atau norma kuadratik dari vektor bobot model. Setelah menemukan *hyperplane* yang optimal, model tersebut dapat digunakan untuk memprediksi kelas data baru dengan memproyeksikan data ke dalam ruang fitur dan mengukur jaraknya dari *hyperplane* (Abdul Fadlil, dkk, 2024). Gambar 2.1 merupakan ilustrasi dari metode SVM.



Gambar 2. 1 *Support Vector Machine* (Sumber*: https://www.researchgate.net*)

2.2.12 *K-Fold Cross-Validation*

K-fold cross-validation adalah cara membagi sampel ke dalam *k* kelompok dan memperlakukan masing-masing kelompok sebagai sampel validasi (atau data pengujian) untuk mengevaluasi kinerja model. Kemudian dilanjutkan dengan, menentukan bobot model dengan cara mengurangi jumlah kesalahan prediksi kuadrat dari semua kelompok tersebut. Berbeda dengan metode lainnya yang rumit, *k-fold cross-validation* mudah digunakan dan tidak bergantung pada struktur model (Zhang dan Liu, 2023).

2.2.13 *Confusion Matrix*

Saat menilai model klasifikasi untuk analisis sentimen, penting untuk memahami cara kerja model dalam membuat prediksi, bukan hanya mengevaluasi akurasinya. Salah satu alat yang berguna untuk ini adalah *Confusion Matrix*. Penelitian ini menggunakan empat metrik kinerja untuk mengevaluasi hasilnya. Secara umum, *Confusion Matrix* adalah tabel dengan dua baris dan dua kolom yang menampilkan jumlah *false positives* (FP), *false negatives* (FN), *true positives* (TP), dan *true negatives* (TN) (Singh, dkk, 2022). Contoh *confusion matrix* 2 x 2 dapat dilihat pada Tabel 2.10.

Tabel 2. 10 *Confusion matrix* 2 x 2

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

Untuk perhitungan akurasi, presisi, dan *recall* yang digunakan dapat dilihat pada persamaan Persamaan 2.9, 2.10, 2.11 (Mutinda, dkk, 2023).

$$Akurasi \quad == \frac{TP + TN}{(TP + FN + FP + TN)} \tag{2.9}$$

Selain akurasi, nilai *recall* dan presisi juga dapat diukur dari sebuah model klasifikasi. Presisi adalah probabilitas yang menunjukkan bahwa item yang dipilih relevan. Nilai presisi dapat dihitung menggunakan persamaan 2.10.

$$Presisi \quad = \frac{TP}{TP + FP} \tag{2.10}$$

Sementara itu, *recall* merupakan perbandingan antara jumlah item yang relevan yang dipilih dengan total keseluruhan item yang relevan. Nilai *recall* dapat dihitung menggunakan persamaan 2.11.

$$Recall \quad = \frac{TP}{TP + FN} \tag{2.11}$$

Keterangan:

True Positive (TP) = Kasus yang sebenarnya positif dan diprediksi positif.

True Negative (TN) = Kasus yang sebenarnya negatif dan diprediksi negatif.

False Positive (FP) = Kasus yang sebenarnya negatif tetapi diprediksi positif (*error tipe I*).

False Negative (FN) = Kasus yang sebenarnya positif tetapi diprediksi negatif (*error tipe II*).

Nilai *recall* dan presisi berkisar antara 0 hingga 1. Semakin tinggi nilai tersebut, semakin baik hasilnya.