BAB II
TINJAUAN PUSTAKA

2.1 Kajian Pustaka Relevan

Penelitian terdahulu yang relevan dijadikan sebagai rujukan untuk membandingkan serta memperkuat landasan penelitian ini. Melalui kajian tersebut, dapat diidentifikasi fokus kajian, metode yang digunakan, dan hasil yang telah dicapai oleh penelitian sebelumnya. Penelitian ini menitikberatkan pada penerapan *Explainable Artificial Intelligence* (XAI) untuk meningkatkan transparansi dan kepercayaan terhadap model deteksi hoaks politik berbahasa Indonesia yang dikembangkan menggunakan algoritma *Support Vector Machine* (SVM).

Penelitian pertama yang dilakukan oleh (Maulana et al., 2025) dengan judul *"Penerapan Metode Support Vector Machine dalam Mengklasifikasikan Berita Politik: Fakta vs Hoaks"* mengembangkan sistem klasifikasi otomatis menggunakan algoritma *Support Vector Machine* (SVM) dengan pendekatan LinearSVC dan pembobotan TF-IDF. Penelitian ini menggunakan 1.267 artikel berita politik berbahasa Inggris dan berhasil mencapai akurasi sebesar 94,24%. Hasil penelitian menunjukkan bahwa SVM sangat efektif dalam menangani data teks berdimensi tinggi dan bersifat *sparse* dengan efisiensi komputasi yang baik. Kesamaan penelitian tersebut dengan penelitian ini terletak pada fokus deteksi hoaks politik dan penggunaan SVM sebagai metode klasifikasi utama. Adapun perbedaannya terletak pada bahasa dataset yang digunakan serta belum diterapkannya pendekatan *Explainable Artificial Intelligence* (XAI), yang pada penelitian ini diintegrasikan melalui metode LIME untuk meningkatkan transparansi dan interpretabilitas model.

Penelitian kedua yang dilakukan oleh (DickiPrabowo et al., 2025) dengan judul *"Sistem Deteksi Berita Hoax Pemilu 2024 Indonesia Menggunakan Algoritma KNN dan SVM"* mengkaji kinerja algoritma *K-Nearest Neighbor* (KNN) dan *Support Vector Machine* (SVM) melalui pendekatan *ensemble stacking* pada 4.283 berita terkait Pemilu 2024. Hasil pengujian menunjukkan bahwa model gabungan menghasilkan akurasi

terbaik sebesar 93,31%, menegaskan keunggulan SVM dalam membentuk margin pemisah yang optimal pada data berdimensi besar. Persamaan penelitian ini dengan penelitian yang dilakukan terletak pada objek kajian berupa berita hoaks politik di Indonesia serta penggunaan SVM sebagai salah satu metode klasifikasi utama. Perbedaannya terletak pada pendekatan pengembangan model, di mana penelitian rujukan berfokus pada peningkatan performa melalui teknik ensemble, sedangkan penelitian ini menekankan aspek interpretabilitas dengan mengintegrasikan metode XAI LIME.

Penelitian ketiga yang dilakukan oleh(Khatib Sulaiman et al., n.d.) dengan judul *"Explainable Sentiment Analysis pada Ulasan Aplikasi Shopee Menggunakan Local Interpretable Model-agnostic Explanations"* menerapkan pendekatan *Explainable Artificial Intelligence* menggunakan metode LIME untuk menjelaskan hasil prediksi sentimen. Penelitian ini membandingkan beberapa algoritma machine learning dan menunjukkan bahwa *Support Vector Machine* (SVM) memberikan performa terbaik dengan akurasi sebesar 84,5% serta menghasilkan penjelasan LIME yang paling representatif dalam menunjukkan kontribusi kata positif dan negatif. Persamaan penelitian ini dengan penelitian yang dilakukan terletak pada penggunaan algoritma SVM dan metode LIME untuk meningkatkan interpretabilitas model. Perbedaannya terletak pada objek kajian, di mana penelitian rujukan berfokus pada analisis sentimen e-commerce, sedangkan penelitian ini berfokus pada deteksi berita hoaks politik.

Penelitian keempat yang dilakukan oleh (Wijaya et al., 2022)dengan judul *"Penerapan Text Mining untuk Klasifikasi Judul Berita Hoax Vaksinasi COVID-19 Menggunakan Algoritma Support Vector Machine"* mengevaluasi kinerja SVM dengan membandingkan empat jenis kernel, yaitu Linear, Polynomial, RBF, dan Sigmoid. Hasil penelitian menunjukkan bahwa kernel Linear memberikan performa terbaik dengan akurasi 89,43%, *precision* 94,93%, dan *F1-score* 90,90% pada pembagian data latih 80%. Kesamaan penelitian ini dengan penelitian yang dilakukan terletak pada penggunaan SVM untuk klasifikasi berita hoaks berbahasa Indonesia. Perbedaannya terletak pada domain topik, di mana penelitian rujukan berfokus pada isu kesehatan, sedangkan

penelitian ini berfokus pada hoaks politik serta mengintegrasikan metode XAI LIME untuk memberikan penjelasan atas keputusan model.

Penelitian kelima yang dilakukan oleh (Rakhmat Sani et al., 2022)dengan judul *"Analisis Perbandingan Algoritma Naive Bayes Classifier dan Support Vector Machine untuk Klasifikasi Hoax pada Berita Online Indonesia"* membandingkan performa Naïve Bayes Classifier dan *Support Vector Machine* dengan pembobotan TF-IDF pada 287 berita hoaks bertema kesehatan. Hasil penelitian menunjukkan bahwa SVM, khususnya dengan kernel Sigmoid, menghasilkan performa terbaik dengan akurasi 96,5% dan recall sebesar 100%. Persamaan penelitian ini dengan penelitian yang dilakukan terletak pada penggunaan SVM dan TF-IDF dalam klasifikasi berita hoaks berbahasa Indonesia. Perbedaannya terletak pada domain topik serta belum diterapkannya pendekatan XAI, yang pada penelitian ini digunakan untuk meningkatkan transparansi dan interpretasi fitur kata.

Penelitian keenam yang dilakukan oleh (Soenarya & Muslim, 2019) dengan judul*"Sentimen Analisis Politik Berita Media Online dalam Pemilihan Presiden 2019 Menggunakan Metode Support Vector Machine"* mengembangkan sistem klasifikasi sentimen berita politik menggunakan SVM dengan pembobotan TF-IDF. Penelitian ini membandingkan beberapa kernel SVM dan menunjukkan bahwa kernel Linear memberikan akurasi tertinggi sebesar 93,33%. Kesamaan penelitian ini dengan penelitian yang dilakukan terletak pada penggunaan SVM dan objek kajian di ranah politik. Perbedaannya terletak pada tujuan klasifikasi, di mana penelitian rujukan berfokus pada analisis sentimen, sedangkan penelitian ini berfokus pada deteksi hoaks serta mengintegrasikan metode XAI LIME.

Penelitian ketujuh yang dilakukan oleh (Abdurrazik, 2025)dengan judul *"Analisis Klasifikasi Tweet Berdasarkan Topik Sosial Menggunakan SVM"* mengklasifikasikan 1.478 tweet berbahasa Indonesia ke dalam beberapa kategori topik menggunakan TF-IDF dan SVM dengan pendekatan One-vs-Rest. Model yang dibangun mencapai akurasi sebesar 84% dan menunjukkan performa yang baik pada kategori politik. Persamaan penelitian ini dengan penelitian yang dilakukan terletak pada penggunaan SVM, TF-IDF, serta adanya irisan topik politik. Perbedaannya terletak pada sumber data dan tujuan

klasifikasi, di mana penelitian ini berfokus pada deteksi hoaks politik berbasis berita daring dengan pendekatan XAI LIME.

Penelitian kedelapan yang dilakukan oleh (Nanda et al., 2022) dengan judul *"Klasifikasi Berita Menggunakan Metode Support Vector Machine"* mengelompokkan 510 berita daring menggunakan SVM kernel Linear dan pembobotan TF-IDF, dengan akurasi tertinggi sebesar 88%. Persamaan penelitian ini dengan penelitian yang dilakukan terletak pada penggunaan SVM kernel Linear dan TF-IDF. Perbedaannya terletak pada tujuan klasifikasi, di mana penelitian rujukan berfokus pada pengelompokan topik, sedangkan penelitian ini berfokus pada deteksi hoaks politik dan interpretasi model menggunakan LIME.

Penelitian kesembilan yang dilakukan oleh oleh (Eka Juliana et al., 2021) dengan judul *"Klasifikasi Kategori Berita Menggunakan Algoritma Support Vector Machine"* menerapkan algoritma *Support Vector Machine* (SVM) untuk mengelompokkan 2.224 data berita ke dalam lima kategori topik, yaitu teknologi, bisnis, olahraga, hiburan, dan politik. Hasil penelitian menunjukkan bahwa SVM mampu menghasilkan performa yang sangat tinggi dengan akurasi sebesar 98,35%, menegaskan keandalannya dalam menangani klasifikasi teks yang kompleks. Persamaan penelitian tersebut dengan penelitian ini terletak pada penggunaan algoritma SVM sebagai metode utama klasifikasi teks. Adapun perbedaannya terdapat pada tujuan dan sumber data, di mana penelitian rujukan berfokus pada klasifikasi topik multi-kelas menggunakan dataset BBC News berbahasa Inggris, sedangkan penelitian ini menggunakan berita berbahasa Indonesia untuk deteksi hoaks politik serta mengintegrasikan pendekatan *Explainable Artificial Intelligence* (XAI) LIME guna meningkatkan transparansi model.

Penelitian kesepuluh yang dilakukan oleh (Suryana et al., 2020) dengan judul *"Klasifikasi Jenis Berita pada Sosial Media Twitter Menggunakan Algoritme Support Vector Machine (SVM)"* menerapkan algoritma SVM untuk mengklasifikasikan berita pada platform Twitter ke dalam beberapa kategori topik dan memperoleh akurasi rata-rata sebesar 85%. Persamaan penelitian tersebut dengan penelitian ini terletak pada penggunaan algoritma SVM dan metode TF-IDF pada data teks berbahasa Indonesia. Sementara itu, perbedaannya terletak pada sumber data dan fokus penelitian, di mana

penelitian rujukan menggunakan data media sosial untuk klasifikasi topik, sedangkan penelitian ini berfokus pada deteksi berita hoaks politik dari portal berita daring dengan integrasi metode XAI LIME.

Berdasarkan kajian terhadap penelitian-penelitian terdahulu, algoritma *Support Vector Machine* (SVM) terbukti efektif dan konsisten dalam klasifikasi teks, khususnya untuk deteksi berita hoaks dan isu politik, terutama ketika digunakan bersama pembobotan TF-IDF. Namun demikian, sebagian besar penelitian masih berfokus pada pencapaian performa model, seperti akurasi dan metrik evaluasi lainnya, tanpa memberikan penjelasan yang memadai terkait dasar pengambilan keputusan model. Kondisi ini menyebabkan rendahnya tingkat transparansi serta menyulitkan pemahaman terhadap peran fitur kata dalam proses klasifikasi, terutama pada konteks berita politik yang bersifat sensitif. Oleh karena itu, penelitian ini bertujuan mengisi celah tersebut dengan menerapkan pendekatan *Explainable Artificial Intelligence* (XAI) melalui metode *Local Interpretable Model-Agnostic Explanations* (LIME) guna meningkatkan transparansi dan interpretabilitas hasil klasifikasi berita hoaks politik berbahasa Indonesia.

2.2 Landasan Teori

2.2.1 Berita Hoaks

Berita hoaks merupakan informasi yang tidak benar atau menyesatkan yang disajikan seolah-olah sebagai fakta dan disebarkan kepada publik tanpa melalui proses verifikasi yang memadai. Penyebaran berita hoaks semakin meningkat seiring dengan perkembangan teknologi informasi dan media sosial, yang memungkinkan distribusi informasi berlangsung secara cepat dan luas. Kondisi ini menyebabkan masyarakat kesulitan membedakan antara informasi yang valid dan tidak valid, sehingga hoaks berpotensi menimbulkan kesalahpahaman, keresahan sosial, serta dampak negatif terhadap kehidupan bermasyarakat (Dewi et al., 2025b).

Hoaks politik merupakan bagian dari berita hoaks yang secara khusus berkaitan dengan isu politik, seperti aktor politik, kebijakan publik, maupun proses demokrasi. Jenis hoaks ini umumnya dibuat dengan tujuan memengaruhi opini publik, membentuk persepsi tertentu, atau menjatuhkan pihak tertentu melalui narasi yang provokatif dan emosional. Penyebaran hoaks politik yang masif di media sosial dapat berdampak serius

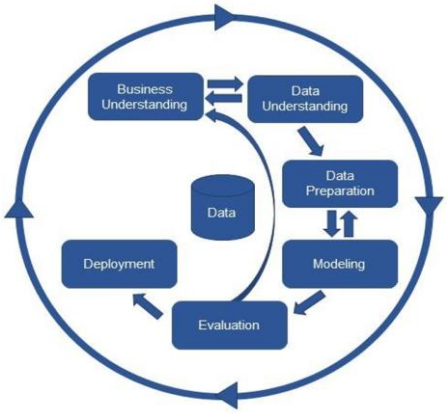
terhadap stabilitas politik dan kepercayaan masyarakat terhadap institusi publik, sehingga diperlukan pendekatan berbasis analisis teks dan pembelajaran mesin untuk mendeteksi serta mengklasifikasikan hoaks politik secara sistematis dan objektif.

2.2.2 *Data Mining*

Data mining merupakan suatu proses analitis yang bertujuan untuk menggali dan mengekstraksi pengetahuan yang bernilai dari kumpulan data berukuran besar dan kompleks melalui penerapan metode statistik, kecerdasan buatan, serta pembelajaran mesin. Proses ini dilakukan untuk menemukan pola, hubungan, atau informasi tersembunyi yang tidak dapat diidentifikasi secara langsung, sehingga dapat dimanfaatkan sebagai dasar pendukung pengambilan keputusan. Dalam berbagai bidang, data mining digunakan untuk meningkatkan efektivitas analisis data dengan mengidentifikasi karakteristik penting yang memengaruhi hasil prediksi. Oleh karena itu, pendekatan data mining relevan digunakan dalam penelitian ini sebagai kerangka pengolahan dan analisis data berita guna mengidentifikasi pola informasi secara sistematis (Fatah, 2025).

2.2.3 *Cross-Industry Standard Process for Data Mining* (CRISP-DM)

Cross-Industry Standard Process for Data Mining (CRISP-DM) merupakan suatu kerangka kerja baku yang digunakan untuk mengelola proses *data mining* secara sistematis dan terstruktur. Kerangka kerja ini dirancang untuk membantu peneliti maupun praktisi dalam mengolah data mentah menjadi informasi atau pengetahuan yang bernilai dengan mengaitkan tujuan bisnis atau permasalahan penelitian dengan proses analisis data dan pemodelan secara menyeluruh (Farid Rifai et al., 2019). Kerangka kerja CRISP-DM terdiri atas enam tahapan utama yang digambarkan sebagai berikut.



Gambar 2. 1 Alur CRISP-DM

Sumber: (Farid Rifai et al., 2019)

1. *Business Understanding*

   Tahap ini bertujuan untuk memahami permasalahan yang akan diselesaikan serta menetapkan tujuan penelitian atau analisis. Pada tahap ini dilakukan identifikasi kebutuhan, ruang lingkup, dan target yang ingin dicapai agar proses data mining selaras dengan tujuan yang telah ditetapkan.

2. *Data Understanding*

   Tahap *data understanding* berfokus pada pengumpulan data awal dan pemahaman karakteristik data. Proses ini mencakup eksplorasi data, identifikasi pola awal, serta pemeriksaan kualitas data sebelum dilakukan pengolahan lebih lanjut.

3. *Data Preparation*

   Pada tahap ini dilakukan penyiapan data agar siap digunakan dalam proses pemodelan. Kegiatan yang dilakukan meliputi pembersihan data (*data cleaning*), penghapusan data tidak relevan, normalisasi, transformasi data, serta pelabelan data apabila diperlukan.

4. *Modeling*

   Tahap *modeling* merupakan proses pembangunan model menggunakan metode atau algoritma tertentu berdasarkan data yang telah dipersiapkan. Pada tahap ini dilakukan pelatihan model serta penyesuaian parameter untuk memperoleh hasil yang optimal.

5. *Evaluation*

   Tahap evaluasi bertujuan untuk menilai kinerja model yang telah dibangun dan memastikan bahwa model tersebut telah memenuhi tujuan yang ditetapkan pada tahap pemahaman bisnis. Evaluasi umumnya dilakukan menggunakan metrik pengukuran tertentu sesuai dengan jenis permasalahan yang diteliti.

6. *Deployment*

   Tahap *deployment* merupakan tahap akhir, yaitu penerapan hasil analisis atau model yang telah dikembangkan. Hasil penerapan dapat berupa sistem, laporan analisis, atau rekomendasi yang dapat digunakan sebagai dasar pengambilan keputusan.

2.2.4 *Natural Language Processing* (NLP)

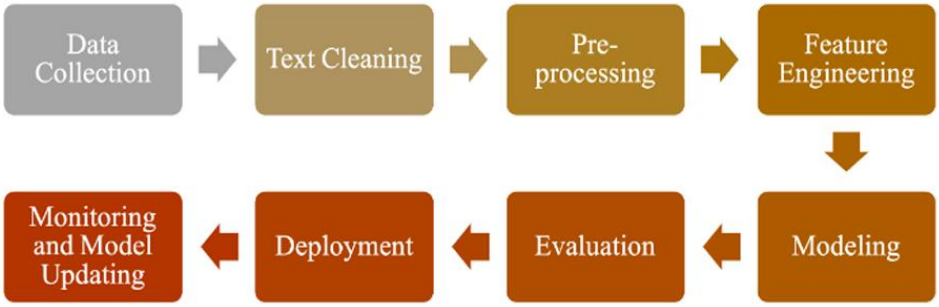
Natural Language Processing (NLP) merupakan cabang kecerdasan buatan yang berfokus pada kemampuan sistem komputer dalam memahami, mengolah, dan menganalisis bahasa alami yang digunakan oleh manusia. Dengan NLP, interaksi antara manusia dan mesin dapat berlangsung secara alami, menyerupai komunikasi antarmanusia, baik melalui teks maupun ujaran (Rizky Suherlan & Pambudi, 2023).

Seiring dengan perkembangan metode dalam NLP, muncul pendekatan berbasis deep learning yang mampu memahami teks secara lebih kontekstual. Salah satu model yang banyak digunakan adalah *Bidirectional Encoder Representations from Transformers* (BERT). BERT merupakan model deep learning yang bekerja dengan membaca teks secara dua arah (*bidirectional*), sehingga dapat memahami makna kata berdasarkan keseluruhan konteks dalam kalimat. Kemampuannya dalam menangkap hubungan semantik antar kata membuat BERT unggul dalam berbagai tugas NLP, meskipun penggunaannya memerlukan sumber daya komputasi yang relatif besar (Maulidya Prastita Syah et al., 2025)

Untuk pemrosesan bahasa Indonesia, dikembangkan varian khusus yaitu IndoBERT. IndoBERT merupakan versi BERT yang dibangun menggunakan kosakata bahasa Indonesia dalam jumlah besar, dengan lebih dari 220 juta kata yang bersumber dari berbagai teks berbahasa Indonesia seperti koran daring dan korpus web Indonesia. Model ini dilatih menggunakan teknik *transfer learning* pada korpus teks berskala besar

sehingga mampu mempelajari pola bahasa Indonesia secara mendalam dan menghasilkan representasi bahasa yang sesuai dengan karakteristik linguistik Indonesia (Nuryadi et al., 2025).

Adapun tahapan dari NLP sebagai berikut.



Gambar 2. 2 Tahapan NLP

Sumber: (Rizky Suherlan & Pambudi, 2023)

1. *Data Collection*

   Tahap ini merupakan proses pengumpulan data teks yang akan digunakan dalam analisis NLP. Data dapat diperoleh dari berbagai sumber, seperti dokumen tertulis, interaksi pengguna, basis data, maupun sumber lain yang relevan dengan tujuan sistem yang dikembangkan.

2. *Text Cleaning*

   Text cleaning bertujuan untuk membersihkan dan menyiapkan data teks agar memiliki struktur yang lebih terorganisasi. Proses ini meliputi penghapusan tanda baca, pengubahan seluruh huruf menjadi huruf kecil (*case folding*), normalisasi kata tidak baku, penghapusan stopword, proses stemming untuk mengubah kata berimbuhan menjadi kata dasar, serta tokenisasi guna memecah teks menjadi unit-unit kata atau frasa.

3. *Pre-processing*

   Tahap *pre-processing* merupakan lanjutan dari *text cleaning* yang mempersiapkan data teks secara lebih mendalam agar siap digunakan dalam proses

ekstraksi fitur. Pada tahap ini dilakukan transformasi dan normalisasi data teks sehingga menghasilkan representasi yang konsisten dan terstandarisasi.

4. *Feature Engineering*

   Pada tahap ini, data teks yang telah melalui proses pengolahan diubah ke dalam bentuk representasi numerik agar dapat dipahami oleh model *machine learning*. Tahap *feature engineering* bertujuan untuk mengekstraksi fitur-fitur yang relevan dari data teks sehingga informasi semantik yang terkandung di dalamnya dapat dimanfaatkan secara optimal dalam proses pembelajaran model.

5. *Modelling*

   Tahap *modeling* merupakan proses pelatihan model dengan memanfaatkan data yang telah direpresentasikan dalam bentuk fitur numerik. Pada tahap ini, model mempelajari pola-pola bahasa yang terdapat dalam data untuk menjalankan tugas tertentu, seperti klasifikasi, prediksi, atau pencocokan *intent*. Hasil dari proses ini berupa model yang siap digunakan untuk mengolah data baru.

6. *Evaluation*

   Tahap evaluasi dilakukan untuk menilai kinerja serta kualitas sistem NLP. Evaluasi ini bertujuan untuk memastikan bahwa sistem telah berfungsi sesuai dengan kebutuhan yang ditetapkan, baik dari aspek ketepatan hasil, fungsionalitas, maupun kualitas respons yang dihasilkan.

7. *Deployment*

   Tahap *deployment* merupakan proses penerapan sistem NLP ke dalam lingkungan pengguna. Sistem yang telah dikembangkan diintegrasikan ke dalam aplikasi atau platform tertentu agar dapat digunakan secara langsung, misalnya dalam bentuk *chatbot* berbasis web maupun aplikasi.

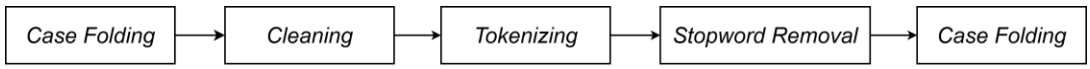
8. *Monitoring and Model Updating*

   Tahap ini merupakan proses pemantauan kinerja model secara berkelanjutan setelah diterapkan. Apabila ditemukan penurunan performa atau perubahan karakteristik data, model diperbarui agar tetap relevan dan akurat dalam menghasilkan prediksi.

2.2.5 *Text Processing*

Text preprocessing merupakan tahap awal yang sangat penting dalam pengolahan data teks karena berfungsi mengubah data teks yang tidak terstruktur menjadi data yang lebih terorganisasi dan siap diolah. Proses ini mencakup pembersihan data dari *noise* serta transformasi teks ke dalam bentuk terstruktur, yang umumnya direpresentasikan sebagai nilai numerik agar dapat diproses lebih lanjut dalam tahapan data mining dan *machine learning* (Ulfah Siregar et al., 2019).

Melalui proses prapemrosesan, kinerja algoritma klasifikasi dapat ditingkatkan karena data yang diolah menjadi lebih konsisten dan memiliki makna semantik yang lebih jelas (Tri Widodo & Setyawan, n.d.). Tahapan dari *text processing* sebagai berikut.



Gambar 2. 3 Tahapan *Text Processing*

Sumber: (Tri Widodo & Setyawan, n.d.)

1. *Case Folding*

   Tahap *case folding* bertujuan untuk menyeragamkan seluruh huruf dalam dokumen menjadi huruf kecil (*lowercase*). Proses ini dilakukan agar kata-kata yang memiliki perbedaan penulisan huruf, seperti “Hoaks” dan “hoaks”, diperlakukan sebagai satu entitas yang sama oleh sistem.

2. *Cleaning*

   Tahap *cleaning* merupakan proses penghapusan elemen-elemen yang tidak relevan dalam analisis teks, seperti angka, tanda baca, simbol khusus, serta tautan URL. Elemen-elemen tersebut dihilangkan karena umumnya tidak memberikan kontribusi informasi yang signifikan dalam penentuan kategori teks.

3. *Tokenizing*

   Tokenisasi adalah proses pemecahan teks yang awalnya berupa kalimat atau dokumen menjadi unit-unit kata tunggal yang disebut sebagai token. Melalui tahap ini, teks yang tidak terstruktur diubah menjadi kumpulan kata sehingga lebih mudah diproses pada tahap selanjutnya.

4. *Stopword Removal*

   Tahap *stopword removal* bertujuan untuk menghilangkan kata-kata umum yang sering muncul namun memiliki nilai informasi yang rendah, seperti "yang", "dan", "di", dan "dari". Penghapusan kata-kata tersebut membantu mengurangi kompleksitas komputasi serta memfokuskan model pada kata-kata yang lebih bermakna.

5. *Stemming*

   Tahap *stemming* bertujuan untuk mengubah kata berimbuhan menjadi bentuk kata dasar. Dalam pengolahan teks berbahasa Indonesia, proses ini umumnya dilakukan menggunakan library khusus seperti Sastrawi, yang berfungsi untuk menghapus awalan, sisipan, dan akhiran. Dengan demikian, kata-kata seperti "menyebarkan" dan "disebar" dapat direduksi menjadi kata dasar yang sama, yaitu "sebar".

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

Term Frequency-Inverse Document Frequency (TF-IDF) adalah metode yang digunakan untuk memberikan bobot pada setiap kata (*term*) dalam suatu dokumen berdasarkan frekuensi kemunculannya. Metode ini menggabungkan dua konsep utama, yaitu *Term Frequency* (TF) dan *Inverse Document Frequency* (IDF).

Term Frequency (TF) mengukur seberapa sering suatu kata muncul dalam dokumen tertentu, sehingga mencerminkan tingkat kepentingan kata tersebut di dalam dokumen.

Inverse Document Frequency (IDF) mengukur seberapa jarang kata tersebut muncul dalam seluruh koleksi dokumen, sehingga kata yang muncul di banyak dokumen memiliki bobot lebih rendah.

Dengan penggabungan kedua konsep ini, TF-IDF mampu menilai tingkat kepentingan suatu kata dalam dokumen tertentu dibandingkan dengan keseluruhan dataset (Siswipraptini, 2023). Secara matematis, TF-IDF dapat dinyatakan sebagai berikut.

$$IDF = log\left(\frac{D}{DF}\right) \qquad (2.1)$$

$$Weight = TF \times IDF \qquad (2.2)$$

Dimana:

D = jumlah seluruh dokumen dalam dataset

DF = jumlah dokumen yang mengandung kata tertentu

TF = frekuensi kemunculan kata dalam dokumen

IDF = *Inverse Document Frequency*

W = bobot akhir kata dalam dokumen

Dengan TF-IDF, kata-kata yang sering muncul dalam dokumen tertentu tetapi jarang muncul pada dokumen lain akan memiliki bobot lebih tinggi. Hal ini membuat metode ini efektif untuk menentukan kata kunci dan merepresentasikan dokumen, terutama dalam proses klasifikasi dan pengolahan teks.

2.2.7 Klasifikasi Teks

Klasifikasi teks merupakan bidang mendasar dalam *Natural Language Processing* (NLP) yang berfokus pada pengelompokan otomatis data tekstual ke dalam kelas-kelas yang telah ditetapkan berdasarkan isi dan makna semantiknya. Proses klasifikasi umumnya dilakukan dengan memanfaatkan teknik *machine learning* melalui pemanfaatan data historis yang dibagi ke dalam *data training* dan *data testing* untuk membangun model prediktif. Model yang dihasilkan kemudian digunakan untuk melakukan pengambilan keputusan atau memberikan rekomendasi secara otomatis pada data baru di masa mendatang. Seiring dengan meningkatnya volume data yang berasal dari berbagai sumber, seperti media sosial, konten web, dan dokumen organisasi, teknik ini menjadi sangat penting untuk mengelola serta mengekstraksi informasi secara efisien. Secara umum, sistem klasifikasi teks melibatkan serangkaian tahapan terstruktur yang mencakup prapemrosesan data, ekstraksi dan seleksi fitur, hingga proses klasifikasi menggunakan pendekatan *machine learning*, baik melalui metode *supervised* maupun *unsupervised learning* (Firizkiansah et al., 2025).

Keberhasilan klasifikasi teks sangat dipengaruhi oleh tahap *preprocessing* yang dilakukan secara cermat untuk mentransformasikan teks mentah ke dalam bentuk yang lebih terstruktur melalui proses tokenisasi, *stopword removal*, dan *stemming*. Tahapan-tahapan tersebut bertujuan untuk mengurangi gangguan (*noise*) serta menormalkan variasi kata sehingga unit informasi dasar menjadi lebih bermakna sebelum memasuki tahap ekstraksi fitur. Mengingat algoritma *machine learning* memerlukan representasi numerik untuk mengenali pola semantik, setiap kosakata yang ada pada dokumen akan dilakukan pembobotan kata dengan perhitungan token menggunakan metode *Term*

Frequency–Inverse Document Frequency (TF-IDF) pada setiap dokumen (Asri et al., 2022). Melalui TF-IDF, kata-kata yang memiliki tingkat kepentingan tinggi dalam suatu dokumen tetapi jarang muncul dalam keseluruhan kumpulan data akan memperoleh bobot yang lebih besar, sehingga model mampu mengidentifikasi karakteristik yang relevan secara lebih akurat dalam proses klasifikasi.

2.2.8 *Support Vector Machine* (SVM)

Support Vector Machine (SVM) merupakan salah satu algoritma klasifikasi teks yang banyak digunakan dalam bidang data mining dan machine learning. Algoritma SVM pertama kali diperkenalkan pada tahun 1992 oleh Valdimir Vapnik bersama Bernard Boser dan Isabelle Guyon. *Support Vector Machine* (SVM) merupakan salah satu algoritma klasifikasi yang bekerja dengan membangun sebuah *hyperplane*, yaitu garis pemisah pada ruang dua dimensi, bidang pada ruang tiga dimensi, atau *hyperplane* pada ruang berdimensi lebih tinggi, yang berfungsi sebagai batas pemisah antar kelas data(Milal et al., 2023).

Keunggulan utama SVM terletak pada kemampuannya dalam menangani data berdimensi tinggi serta penggunaan fungsi kernel. Dengan adanya fungsi kernel, SVM dapat memetakan data input yang tidak terpisah secara linear ke dalam ruang berdimensi lebih tinggi sehingga memungkinkan pemisahan kelas secara optimal. Beberapa jenis kernel yang umum digunakan pada SVM antara lain kernel Linear, Polynomial, dan Radial Basis Function (RBF). *Hyperplane* sendiri merupakan fungsi yang digunakan sebagai batas pemisah antar kelas dalam proses klasifikasi.

2.2.8.1 Konsep *Hyperplane* pada SVM

Dalam SVM, *hyperplane* digunakan untuk memisahkan data ke dalam kelas-kelas yang berbeda. Secara matematis, *hyperplane* dapat dirumuskan sebagai berikut:

$$w \cdot x + b = 0 \rightarrow w_0 + w_1 x_1 + w_2 x_2 \quad (2.3)$$

dengan keterangan sebagai berikut:

$W = \{w_1, w_2, \dots, w_n\}$ adalah vektor bobot

$n =$ adalah jumlah atribut

b = skalar yang digunakan sebagai bias

x = nilai atribut

2.2.8.2 *Kernel Trick*

Pada beberapa kasus klasifikasi, data tidak dapat dipisahkan secara linear pada ruang input sehingga SVM kesulitan membentuk *hyperplane* pemisah yang optimal. Kondisi ini berdampak pada menurunnya performa klasifikasi dan kemampuan generalisasi model. Untuk mengatasinya, SVM menerapkan *kernel trick*, yaitu pendekatan yang memetakan data ke ruang fitur berdimensi lebih tinggi agar pola data dapat dipisahkan secara linear (Romadoni et al., 2020).

Pada dasarnya, proses pembelajaran SVM untuk menentukan *support vector* hanya bergantung pada operasi perkalian titik (*dot product*) antar data di ruang fitur. Namun, fungsi pemetaan ke ruang fitur, yang dilambangkan dengan Φ, umumnya tidak diketahui secara eksplisit dan sulit untuk dihitung secara langsung. Oleh karena itu, perhitungan *dot product* pada ruang fitur digantikan dengan fungsi kernel yang merepresentasikan pemetaan tersebut secara implisit. Pendekatan inilah yang dikenal sebagai *kernel trick*, yang memungkinkan SVM tetap efisien secara komputasi meskipun bekerja pada ruang berdimensi tinggi.

Berdasarkan (Romadoni et al., 2020), beberapa fungsi kernel yang umum digunakan dalam SVM antara lain:

1. Kernel Linear

$$K(x, x_k) = x^T x_k \qquad (2.5)$$

   Kernel ini sesuai untuk data dengan hubungan linear dan kompleksitas yang relatif rendah.

2. Kernel Polynomial

$$K(x, x_k) = (x^T x_k + 1)^d \qquad (2.6)$$

   Kernel polynomial digunakan untuk memodelkan hubungan non-linear, dengan tingkat kompleksitas yang dikontrol oleh parameter derajat d.

3. Kernel Radial Basis Function (RBF)

$$K(x, x_k) = \exp\left(-\frac{\| x - x_k \|^2}{2\sigma^2}\right) \qquad (2.7)$$

Kernel RBF mengukur kedekatan antar data berdasarkan jarak Euclidean dan efektif untuk menangani pola non-linear yang kompleks.

4. Kernel Sigmoid

$$K(x, x_k) = \tanh\left(k\, x^T x_k + \theta\right) \qquad (2.8)$$

Kernel sigmoid menggunakan fungsi *hyperbolic tangent* yang menyerupai fungsi aktivasi pada jaringan saraf tiruan. Parameter $k$berperan sebagai faktor skala, sedangkan θ berfungsi sebagai bias.

Keterangan:

x, x_k: vektor data atau vektor fitur

x^T: transpose dari vektor x

d: derajat polinomial pada kernel polynomial

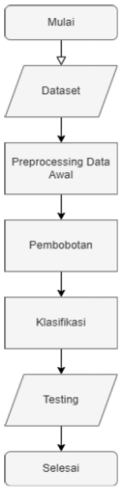
σ: parameter lebar kernel pada RBF

k: parameter skala pada kernel sigmoid

θ: parameter bias pada kernel sigmoid

2.2.8.3 Tahapan Penelitian SVM

Penerapan metode *Support Vector Machine* (SVM) dalam klasifikasi teks memerlukan tahapan penelitian yang sistematis untuk menghasilkan model dengan kinerja yang optimal dan akurat. Secara umum, tahapan penelitian menggunakan SVM dijelaskan sebagai berikut:



Gambar 2. 4 Tahapan Penelitian SVM

Sumber: (Milal et al., 2023)

Adapun tahapan penelitian menggunakan algoritma Support Vector Machine (SVM) secara umum dijelaskan sebagai berikut:

1. Dataset

   Tahap awal penelitian adalah penyusunan dataset yang akan digunakan sebagai data penelitian. Dataset berupa kumpulan berita politik berbahasa Indonesia yang telah diberi label ke dalam dua kelas, yaitu berita hoaks dan berita valid. Dataset ini menjadi dasar utama dalam proses pelatihan dan pengujian model klasifikasi.

2. *Preprocessing* Data Awal

   Pada tahap preprocessing, data teks dilakukan pembersihan untuk menghilangkan noise yang dapat memengaruhi kinerja model. Proses ini meliputi penghapusan karakter yang tidak relevan, normalisasi teks, serta tahapan lain yang bertujuan untuk menghasilkan data teks yang lebih bersih dan terstruktur sehingga siap untuk diproses lebih lanjut.

3. Pembobotan

   Tahap pembobotan bertujuan untuk mengubah data teks menjadi bentuk numerik agar dapat diproses oleh algoritma SVM. Pada tahap ini, setiap kata dalam dokumen diberikan bobot tertentu berdasarkan tingkat kemunculannya. Hasil dari proses pembobotan berupa vektor fitur yang merepresentasikan dokumen teks secara matematis.

4. Klasifikasi

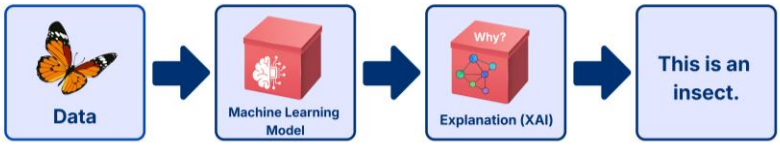
   Tahap klasifikasi merupakan proses pelatihan model *Support Vector Machine* menggunakan data yang telah melalui tahap *preprocessing* dan pembobotan. Pada tahap ini, SVM membangun *hyperplane* untuk memisahkan data ke dalam kelas hoaks dan valid berdasarkan pola yang dipelajari dari data latih.

5. Testing

   Tahap testing dilakukan untuk menguji kinerja model SVM yang telah dibangun. Model diuji menggunakan data uji untuk mengetahui kemampuan klasifikasi dalam mengelompokkan berita politik ke dalam kelas hoaks dan valid. Hasil pengujian selanjutnya dievaluasi menggunakan metrik evaluasi seperti akurasi, *precision*, *recall*, dan *f1-score*.

2.2.9 *Explainable Artificial Intelligence* (XAI)

Explainable Artificial Intelligence (XAI) merupakan pendekatan dalam pengembangan kecerdasan buatan yang berfokus pada peningkatan transparansi, keterpahaman, dan akuntabilitas model pembelajaran mesin, terutama pada model kompleks yang bersifat *black box*. Berbagai model seperti *Deep Neural Networks*, *Ensemble Learning*, dan *Support Vector Machines* mampu menghasilkan prediksi dengan akurasi tinggi, namun mekanisme pengambilan keputusannya sering kali sulit untuk diinterpretasikan. Kondisi tersebut menimbulkan tantangan dalam penerapan AI pada bidang-bidang yang memiliki dampak signifikan terhadap manusia, seperti layanan kesehatan, sektor keuangan, dan penegakan hukum. Oleh karena itu, XAI dikembangkan untuk menjelaskan proses serta alasan di balik keputusan yang dihasilkan model AI, sehingga dapat meningkatkan kepercayaan pengguna, mendukung kepatuhan terhadap regulasi, serta mendorong penerapan AI yang etis dan bertanggung jawab (Article & Michael, 2025).



Gambar 2. 5 Alur *Explainability*

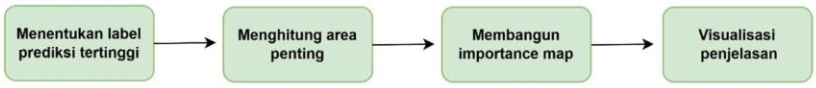
Sumber: (Article & Michael, 2025)

Pada Gambar 2.6 ditampilkan alur *explainability* dalam *Explainable Artificial Intelligence* (XAI) yang menunjukkan bagaimana model pembelajaran mesin memproses data masukan hingga menghasilkan keputusan yang dapat dipahami oleh manusia. Alur ini diawali dari data sebagai *input* yang diproses oleh model yang umumnya bersifat *black box*, kemudian dilanjutkan dengan tahap *explanation* untuk menguraikan alasan (*why*) di balik prediksi yang dihasilkan sebelum diperoleh keluaran akhir. Melalui proses tersebut, pengguna dapat memahami faktor-faktor yang memengaruhi keputusan model, sehingga transparansi dan tingkat kepercayaan terhadap sistem dapat meningkat. Lebih lanjut, explainability dalam XAI dibedakan menjadi *global explainability* yang memberikan pemahaman menyeluruh terhadap perilaku serta keterbatasan model, dan *local explainability* yang menitikberatkan pada penjelasan keputusan individual pada setiap data tertentu, yang sangat relevan dalam penerapan klasifikasi teks (Hamida et al., 2024)

2.2.10 *Local Interpretable Model-Agnostic Explanations* (LIME)

Local Interpretable Model-Agnostic Explanations (LIME) merupakan salah satu metode dalam *Explainable Artificial Intelligence* (XAI) yang bertujuan memberikan penjelasan terhadap prediksi model kompleks. LIME bersifat model-agnostik, artinya dapat diterapkan pada berbagai jenis model, termasuk model *black box* yang strukturnya tidak diketahui. Pendekatan ini menekankan penjelasan secara lokal, yaitu interpretasi terhadap prediksi pada instance tertentu saja, sehingga hasilnya dapat divisualisasikan secara intuitif dan mudah dipahami (Ananda et al., 2025).

Adapun tahapan dari LIME di jelaskan pada gambar berikut.



Gambar 2. 6 Tahapan LIME

Sumber: (Ananda et al., 2025)

1. Menentukan label prediksi tertinggi: Model mengidentifikasi prediksi dengan probabilitas tertinggi pada input tertentu.

2. Menghitung area penting: LIME menilai bagian-bagian input yang paling relevan terhadap prediksi dengan membandingkan pengaruh setiap segmen input terhadap output model.
3. Membangun peta kepentingan (*importance map*): Hasil estimasi digunakan untuk membangun model linier sederhana yang meniru perilaku model kompleks di sekitar data yang dianalisis.
4. Visualisasi penjelasan: Peta kepentingan divisualisasikan sebagai *overlay* pada input asli, sehingga area input yang paling berpengaruh terhadap prediksi dapat dikenali secara intuitif.

2.2.11 Evaluasi Model

Evaluasi model merupakan tahap penting dalam penelitian klasifikasi untuk menilai kinerja dan akurasi prediksi yang dihasilkan oleh model. Salah satu metode evaluasi yang umum digunakan adalah *Confusion Matrix*, yang menggambarkan performa model klasifikasi dengan membandingkan prediksi model terhadap data aktual (Hokijuliandy et al., n.d.)

Confusion Matrix untuk klasifikasi biner dapat disajikan dalam bentuk tabel sebagai berikut:

	Prediksi: Positif	Prediksi: Negatif
Aktual: Positif	TP	FN
Aktual: Negatif	FP	TN

Gambar 2. 7 *Confusion Matrix*

Sumber: (Hokijuliandy et al., n.d.)

Keterangan:

1. *True Positives* (TP): jumlah data kelas positif yang berhasil diprediksi sebagai positif.
2. *True Negatives* (TN): jumlah data kelas negatif yang berhasil diprediksi sebagai negatif.
3. *False Positives* (FP): jumlah data kelas negatif yang salah diprediksi sebagai positif.

4. *False Negatives* (FN): jumlah data kelas positif yang salah diprediksi sebagai negatif.

Berdasarkan komponen tersebut, kinerja model dapat diukur menggunakan beberapa metrik evaluasi (Arvio et al., 2024)

1. Akurasi (*Accuracy*)

$$Accuracy = \frac{(TP + TN)}{(TP + TN + FP + FN)} \quad (2.6)$$

Akurasi menunjukkan proporsi prediksi yang benar terhadap keseluruhan data.

2. *Precision*

$$Precision = \frac{(TP)}{(TP + FP)} \quad (2.7)$$

Precision mengukur ketepatan model dalam memprediksi kelas positif.

3. *Recall*

$$Recall = \frac{TP}{(TP + FN)} \quad (2.8)$$

Recall menilai kemampuan model dalam mendeteksi semua data positif yang sebenarnya.

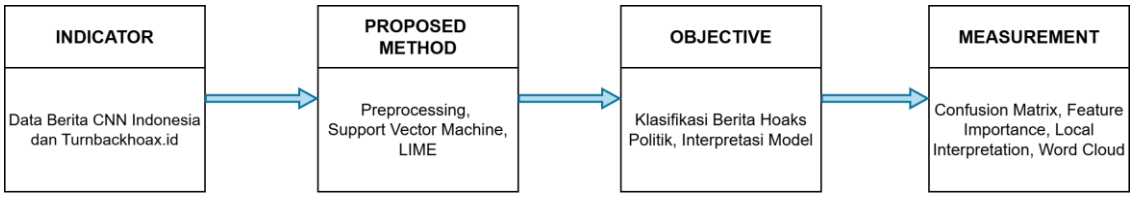
4. *F1-Score*

$$F1\text{-}Score = \frac{(2 * Recall * Precision)}{(Recall + Precision)} \quad (2.9)$$

F1-Score merupakan rata-rata harmonik dari *precision* dan *recall*, yang mencerminkan keseimbangan antara kedua metrik.

Dalam konteks evaluasi, TP menunjukkan prediksi positif yang benar, TN prediksi negatif yang benar, FP merupakan kesalahan tipe I (positif yang salah), dan FN kesalahan tipe II (negatif yang salah), yang umumnya lebih berisiko karena prediksi positif tidak terdeteksi (Arvio et al., 2024)

2.3 Kerangka Pemikiran



Gambar 2. 8 Kerangka Pemikiran

Gambar di atas menggambarkan kerangka kerja penelitian dengan alur metodologi yang sistematis. Penjelasan tiap tahap adalah sebagai berikut:

1. *Indicator*

   Data yang digunakan dalam penelitian ini adalah berita dari CNN Indonesia dan TurnBackHoax.id. Data tersebut diperoleh melalui *web scraping* pada tahun 2024, sehingga mencakup berita faktual dari CNN dan berita hoaks yang telah diverifikasi oleh TurnBackHoax.

2. *Proposed Method*

   Metode yang diusulkan dalam penelitian ini digunakan untuk mengolah data berita yang telah dikumpulkan. Tahapan yang diterapkan meliputi *preprocessing*, klasifikasi menggunakan *Support Vector Machine* (SVM), serta interpretasi hasil prediksi menggunakan LIME.

3. *Objective*

   Objective adalah tujuan yang ingin dicapai dalam penelitian ini. Tujuan yang diharapkan adalah melakukan klasifikasi berita hoaks politik dan menyediakan interpretasi terhadap prediksi model, sehingga peneliti dapat mengetahui fitur atau kata yang paling berpengaruh dalam hasil klasifikasi.

4. *Measurement*

   Pada tahap *measurement*, kinerja model dievaluasi menggunakan *Confusion Matrix* untuk mengukur akurasi, *precision*, *recall*, dan *F1-score*. Selain itu, hasil interpretasi model dari LIME divisualisasikan melalui *feature importance*, *local interpretation*, dan *word cloud* untuk mempermudah pemahaman terhadap prediksi yang dihasilkan.