BAB II
TINJAUAN PUSTAKA DAN DASAR TEORI

2.1 Tinjauan Pustaka

Dalam membuat penelitian klasifikasi masalah penggunaan aplikasi X berdasarkan ulasan google play store untuk peningkatan pengalaman pengguna menggunakan algoritma *Support Vector Machine* (SVM) mengacu dari penelitian sebelumnya diantaranya:

Oryza Habibie Rahman, Gunawan Abdillah, dan Agus Komarudin (2021) melakukan penelitian untuk mendeteksi ujaran kebencian pada tweet dengan metode SVM. Penelitian yang dilakukan mendapatkan hasil accuracy sebesar 93%, nilai *precision* sebesar 84%, nilai recall sebesar 86%, dan nilai F-measure sebesar 83% dengan menggunakan kernel RBF (Oryza Habibie Rahman et al., 2021).

Eskiyaturrofikoh dan Ryan Randy Suryono (2024) melakukan penelitian untuk menganalisa ulasan pengguna dari *Google Playstore* terhadap aplikasi X dengan 2 metode *Naïve Bayes* dan SVM serta penerapan SMOTE. Dari penelitian ini didapatkan akurasi sebesar 81% untuk metode SVM dan 75% untuk metode *Naïve bayes* (Eskiyaturrofikoh & Suryono, 2024).

Uro Abdurohim, Dedy Apriyadi, Arya Devi Listiani melakukan penelitian untuk mendeteksi spam pada komentar instagram dengan metode SVM. Penelitian ini mendapatkan akurasi sebesar 96,82% (Abdurohim et al., 2024).

Risa Wati, Siti Ernawati (2021) melakukan penelitian untuk menganalisa tweet mengenai PPKM dengan menggunakan metode SVM. Hasil dari penelitian ini mendapatkan akurasi 86%, *precision* sebesar 85%, recall sebesar 88% dan *F1-Score* sebesar 86% dengan kernel *linear* (Wati & Ernawati, 2021)

Raden Isnawan Argi Aryasatya (2023) melakukan penelitian untuk menganalisa tweet mengenai pernikahan di usia muda dengan metode SVM. Hasil penelitian ini medapatkan akurasi sebesar 87% (Aryasatya, 2023).

Ikram Maulana, Winda Apriandari, dan Agung Pambudi (2023) melakukan penelitian untuk menganalisa ulasan pada aplikasi MyPertamina. Hasil dari penelitian ini mendapatkan akurasi sebesar 92% (Maulana et al., 2023).

Tabel 2. 1 Tabel Penelitian Terdahulu yang Berkaitan dengan Penelitian

No	Nama Pengarang	Obyek	Metode	Hasil
1	Oryza Habibie Rahman, Gunawan Abdillah, Agus Komarudin (2021)	Sentimen untuk mendeteksi ujaran kebencian pada X	SVM	Hasil dari penelitian ini mendapatkan accuracy sebesar 93%, nilai *precision* sebesar 84%, nilai recall sebesar 86%, dan nilai F-measure sebesar 83% dengan kernel RBF.
2	Eskiyaturrofikoh, Ryan Randy Suryono (2024)	Sentiment ulasan terhadap aplikasi X dari Google Playstore	SVM, Naïve Bayes	Hasil dari penelitian ini mendapatkan 81% accuracy dengan penerapan SMOTE.
3	Uro Abdurohim, Dedy Apriyadi, Arya Devi Listiani (2024)	Sentimen untuk mendeteksi spam pada komentar Instagram.	SVM	Hasil dari penelitian mendapatkan hasil 96,82%
4	Risa Wati, Siti Ernawati (2021)	Tweet mengenai PPKM	SVM	Hasil dari penetian mendapatkan akurasi sebesar 86%, Precission sebesar 85%, Recall sebesar 88% dan *F1-Score* sebesar 86% dengan kernel linear.

Tabel 2. 1 (Lanjutan)

5	Raden Isnawan Argi Aryasatya (2023)	Tweet menegenai pernikahan di usia muda	SVM	Hasil dari penelitian mendapatkan akurasi 87%
6	Ikram Maulana1, Winda Apriandari, Agung Pambudi (2023)	Sentimen berbasis aspek mengenai ulasan aplikasi MyPertamina	SVM	Hasil dari penelitian ini mendapatkan akurasi sebesar 92%.

2.2 Dasar Teori

2.2.1 X

X adalah rebranding dari aplikasi twitter yang merupakan salah satu platform media sosial yang dikembangkan oleh Jack Dorsey, Noah Glass, Biz Stone, dan Evan Williams pada tahun 2006 (Paramastri & Gumilar, n.d.). Pada tahun 2023 X melakukan rebranding menjadi X. Aplikasi ini memungkinkan pengguna mengekspresikan emosi melalui postingan berupa gambar, video, bertukar pesan maupun melalui *tweets. Tweets* merupakan salah satu fitur pada aplikasi X berupa pesan status (Fikri et al., n.d.)

X telah menjadi salah satu platform media sosial paling populer di dunia dengan jutaan pengguna aktif setiap harinya. Selain sebagai media untuk berbagi informasi dan mengekspresikan diri, X juga sering digunakan untuk pemasaran, promosi, dan membangun jaringan profesional. Pada tahun 2024 jumlah pengguna X sekarang adalah 611,3 juta pengguna dimana lebih banyak 1,2% dibandingkan dengan kuartal sebelumnya (wearesocial.com, n.d.).

2.2.2 *Google Play Store*

Play Store merupakan layanan dari *Google* yang menyediakan aplikasi, permainan, buku digital, dan masih banyak lagi. *Play Store* juga berperan dalam mengembangkan atau meningkatkan kualitas dari aplikasi maupun game, karena *Play Store* memiliki fitur ulasan yang memungkinkan pengguna memberikan ulasan dan *rating* pada suatu aplikasi maupun game.

Banyak pengguna yang melihat ulasan maupun rating dari aplikasi sebelum melakukan unduhan pada suatu aplikasi. Banyak pengguna memeriksa ulasan dan rating sebelum mengunduh sebuah aplikasi. Hal ini menunjukkan bahwa ulasan dan rating dari pengguna dapat memberikan dampak signifikan terhadap sebuah aplikasi. Ulasan dan rating tersebut dapat berkontribusi pada peningkatan jumlah pengguna, serta menjadi salah satu faktor penentu apakah aplikasi tersebut banyak digunakan atau tidak (Aida Sapitri et al., 2023).

2.2.3 Python

Python adalah bahasa pemrograman tingkat tinggi yang dirancang untuk kemudahan penggunaan dan pembacaan. Diciptakan oleh Guido van Rossum dan dirilis pertama kali pada tahun 1991, *Python* telah menjadi salah satu bahasa pemrograman yang paling populer di dunia. Salah satu keunggulan *Python* adalah sintaksisnya yang sederhana dan mudah dipahami, membuatnya cocok untuk pemula maupun profesional. Selain itu, *Python* memiliki ekosistem yang kaya dengan berbagai pustaka yang mendukung berbagai aplikasi, mulai dari pengembangan web hingga analisis data (Riziq sirfatullah Alfarizi et al., 2023).

Dalam bidang bidang analisis data dan pembelajaran mesin, bahasa pemrograman memungkinkan pengguna untuk melakukan pemrosesan data, pemilihan algoritma, pelatihan model, serta evaluasi dan perbaikan model dengan lebih efisien. Dengan menggunakan pustaka-pustaka yang ada, pengguna dapat dengan mudah mengimplementasikan berbagai algoritma klasifikasi dan regresi untuk menganalisis dataset yang kompleks. Berkat pustaka-pustaka seperti *NumPy*, *Pandas*, dan *Matplotlib*, pengguna dapat dengan mudah melakukan analisis data dan visualisasi (Kencana Putri & Ichsanuddin Nur, 2023)

2.2.4 *Data Mining*

Data mining adalah kombinasi dari berbagai disiplin ilmu komputer yang didefinisikan sebagai proses sistematis untuk menemukan pola-pola baru, hubungan, atau tren yang bermakna dalam kumpulan data yang sangat besar dan kompleks. Proses ini mencakup berbagai metode yang berasal dari bidang kecerdasan buatan, pembelajaran mesin, statistik, serta sistem basis data, yang

bekerja secara sinergis untuk menganalisis data dengan cara yang efisien. Melalui data mining, data mentah yang awalnya sulit dipahami dapat diolah menjadi informasi yang berguna dan relevan. Informasi ini sering kali digunakan untuk mendukung pengambilan keputusan, memprediksi hasil di masa depan, atau memahami fenomena tertentu secara lebih mendalam.

Tujuan utama dari data mining adalah untuk mengekstrak pengetahuan tersembunyi yang tersembunyi di balik tumpukan data, sehingga dapat dihasilkan struktur yang tidak hanya bermakna tetapi juga mudah dipahami dan diinterpretasikan oleh manusia. Dalam praktiknya, data mining merupakan salah satu komponen dari proses yang disebut *Knowledge Discovery in Databases* (KDD). Proses ini melibatkan ekstraksi informasi *non-trivial* dari data yang bersifat implisit dan sebelumnya tidak diketahui, namun memiliki potensi untuk menghasilkan informasi yang berguna dari data yang tersedia (Aryasatya, 2023).

2.2.5 Machine Learning

Machine learning adalah suatu mesin kecerdasan buatan yang memungkinkan komputer untuk belajar dari data tanpa harus diprogram secara khusus. Ini memungkinkan komputer untuk membuat keputusan sendiri berdasarkan pola-pola yang ditemukan dalam data. *Machine learning* membutuhkan data untuk melakukan pelatihan terhadap model, karena data berfungsi sebagai dasar bagi algoritma untuk belajar dan mengenali pola yang ada. Proses pelatihan melibatkan pembagian dataset menjadi dua bagian yaitu data pelatihan dan data pengujian. Data pelatihan digunakan untuk melatih model, sedangkan data pengujian digunakan untuk mengevaluasi kinerja model setelah dilatih. Pembagian ini penting untuk memastikan bahwa model tidak hanya menghafal data pelatihan tetapi juga mampu generalisasi pada data yang belum pernah dilihat sebelumnya.

Machine learning dibagi menjadi 3 yaitu *supervised learning*, *unsupervised learning*, *reinforcement learning*. *Supervised learning* adalah metode di mana model dilatih menggunakan dataset yang sudah dilabeli. Dalam konteks klasifikasi, tugas ini melibatkan pemetaan input ke dalam kategori tertentu

berdasarkan label yang ada. Contoh algoritma yang digunakan dalam supervised learning untuk klasifikasi termasuk *Decision Trees, Random Forest, dan Support Vector Machine*. Proses pembangunan model *machine learning* terdiri dari beberapa tahapan penting:

1. Pengumpulan Data: Mengumpulkan data yang relevan dari berbagai sumber, seperti *database*, API, atau pengukuran langsung. Kualitas dan kuantitas data yang dikumpulkan akan mempengaruhi hasil akhir model.
2. *Labeling* : Proses di mana setiap data dalam dataset diberikan label sesuai dengan kategori yang telah ditentukan sebelumnya. Label ini digunakan untuk membantu algoritma pembelajaran mesin memahami dan mengenali pola dalam data berdasarkan kategori tertentu. Kata kunci khusus digunakan untuk menentukan mengkategorikan data dalam suatu kelas. Dari beberapa penelitian yang sudah dilakukan untuk kasus analisis penggunaan aplikasi, label yang digunakan adalah label yang mencerminkan aspek-aspek dari ulasan pengguna. Label ini mencakup kinerja aplikasi, *bug* dan *error*, fungsionalitas fitur, serta koneksi dan server yang dapat dilihat pada tabel 2.2 (Maulana et al., 2023; Rosetya Wardhana et al., 2016).

Tabel 2. 2 Tabel Label dan Kata Kunci

Label	**Kata kunci**
kinerja aplikasi	lambat, berat, lemot, macet, lelet, tidak responsif, tidak stabil, buruk, *loading*, ngelag, berhenti, *force close*, *crash*, performa, kinerja, *overheating*.
bug dan *error*	*error*, rusak, tidak stabil, tidak bekerja, *bug*, kesalahan, pesan *error*, fitur, sistem, *crash*, gagal
fungsionalitas fitur	tidak bisa, terkunci, hilang, tidak sesuai, sulit, *login*, *logout*, memposting, mengirim, menggulir, memperbarui, notifikasi, unggah, fitur, pesan, gambar, video, *feed*, *tweet*, fungsi

Tabel 2.2 (Lanjutan)

koneksi dan server	gagal, buruk, sibuk, tidak stabil, sinyal, koneksi, jaringan, terhubung, sambung, putus, memutuskan, memblokir, koneksi, *server*, jaringan, internet, sambungan.

3. *Preprocessing*: Menyiapkan dan membersihkan data yang akan digunakan. Ini termasuk menangani nilai yang hilang, menghapus duplikasi, dan mengubah format data agar sesuai untuk analisis.

4. *Feature Extraction*: Memilih fitur-fitur penting dari data yang akan digunakan dalam model. Proses ini bertujuan untuk mengurangi dimensi data dan meningkatkan efisiensi serta akurasi model.

5. *Splitting Data*: Membagi dataset menjadi dua bagian, yaitu data pelatihan dan data pengujian. Ini penting untuk mengevaluasi kinerja model secara objektif.

6. *Training*: Melatih model menggunakan data pelatihan untuk mengenali pola dan membuat prediksi. Pada tahap ini, algoritma machine learning diterapkan untuk menemukan hubungan antara input dan output.

7. *Evaluation*: Menguji kinerja model menggunakan data pengujian untu(Kencana Putri & Ichsanuddin Nur, 2023) memastikan akurasi dan efektivitasnya. Metode evaluasi yang umum digunakan termasuk confusion matrix, *precision*, recall, dan *F1-Score* (Dharma & Tambunan, 2021; Wijoyo et al., 2024).

2.2.6 *Web Scrapping*

Web scraping adalah proses pengambilan data dari situs web yang memungkinkan pengguna untuk mengekstrak informasi tertentu dari halaman web. Proses ini biasanya dilakukan dengan menggunakan teknik pemrograman untuk mengakses dan menganalisis dokumen semi-terstruktur, seperti halaman web, guna mendapatkan data yang diinginkan. teknik ini digunakan untuk mempercepat pengambilan data yang sebelumnya dilakukan secara manual. Sehingga memungkinkan dalam mengambil data dalam jumlah besar (Ulfah &

Najiah, 2023). Salah satu alat untuk melakukan web *scrapping* adalah *Google Play Scraper*. *Google Play Scrapper* digunakan untuk mengambil data dari *Google Play Store* secara otomatis. Alat ini memanfaatkan teknik scraping untuk mengumpulkan informasi terkait aplikasi seperti nama, deskripsi, rating, dan ulasan pengguna (Fadhillah, 2024)

2.2.7 *Text Preprocessing*

Text Preprocessing merupakan suatu proses yang bertujuan untuk mengubah teks agar terstruktur yang nantinya digunakan untuk kalasifikasi. Text preprocessing memiliki beberapa tahap yaitu *cleansing*, *case folding*, *tokenizing filtering*, normalisasi, *stemming* (Harnelia, 2024)

1. *Cleansing*

   Cleansing merupakan tahapan proses untuk membersihkan karakater yang tidak diperlukan seperti angka, *link, mention,* operator, spasi, dan lain-lain. Selain itu juga penghapusan dupilkasi data juga dilakukan pada tahapan ini. Hal ini bertujuan untuk mengurangi *noise* pada data.

2. *Case Folding*

   Case folding merupakan tahapan proses untuk mengubah huruf menjadi huruf kecil. Dengan mengubah semua huruf menjadi huruf kecil, kita menciptakan konsistensi dalam data teks. Hal ini sangat penting ketika melakukan pencarian atau perbandingan antara kata-kata.

3. *Tokenizing*

   Tokenizing adalah proses memecah teks menjadi token-token tertentu atau dibuat menjadi perkata berdasarkan batasan seperti tanda baca dan spasi. Untuk melakukan proses ini digunakan library nltk.(Ramira Putra et al., 2022).

4. *Filtering*

   Filtering merupakan proses untuk menyaring dan memilih kata-kata yang relevan dari hasil tokenisasi, sehingga hanya kata-kata yang memiliki makna signifikan yang dipertahankan untuk analisis lebih lanjut. *Filtering* biasanya dilakukan dengan menggunakan teknik seperti penghapusan

stopwords, yaitu kata-kata umum yang tidak memberikan informasi penting, seperti "dan", "atau", "adalah", dan lain-lain (Yulistiani & Styawati, 2024).

5. Normalisasi

   Normalisasi merupakan proses untuk menyamakan format penulisan kata-kata yang tidak baku atau singkatan menjadi bentuk yang lebih standar. Dengan mengubah suatu kata menjadi bentuk standar dapatmembantu mengurangi variasi dalam data.

6. Stemming

   Stemming adalah untuk mengubah kata-kata yang memiliki imbuhan atau variasi bentuk ke dalam bentuk dasarnya (*root word*). Dengan melakukan stemming, variasi kata yang memiliki makna sama dapat disamakan, sehingga membantu mengurangi kompleksitas data dan meningkatkan efisiensi dalam analisis (Harnelia, 2024).

2.2.8 *Term Frequency – Inverse Document Frequency*

Term Frequency – Inverse Document Frequency (TF-IDF) merupakan salah satu metode untuk ekstraksi fitur yang digunakan dalam proses pembobotan yang memungkinkan untuk menghasilkan vektor di mana setiap kata yang dihitung berfungsi sebagai satu fitur. Dengan cara ini, setiap kata dalam dokumen dapat diwakili secara numerik, sehingga memudahkan algoritma machine learning untuk mengenali dan menganalisis pola yang ada. Vektor ini mencerminkan pentingnya setiap kata dalam konteks keseluruhan dokumen dan membantu dalam meningkatkan akurasi model saat melakukan klasifikasi atau pengolahan teks lainnya. Selain itu, representasi fitur ini juga memungkinkan pemrosesan data teks dalam dimensi tinggi, yang sangat berguna untuk algoritma seperti *Support Vector Machine* (SVM) dalam mengidentifikasi kategori atau tema tertentu dari dokumen (Darwis et al., 2020).

Proses ini dimulai dengan *term frequency* (TF), yang akan menghitung jumlah kemunculan kata pada dataset. Untuk menentukan bobot dari masing-masing term/kata dalam sebuah dokumen yang ada pada dataset dengan persamaan:

$$tf = 0{,}5 + 0{,}5 \frac{tf}{\max(tf)}$$ (Darwis et al., 2020a)

Di mana:

a. *tf* = banyaknya kata yang muncul pada sebuah dokumen
b. *max(t f)* = panjang kata dari sebuah dokumen itu sendiri.

Proses selanjutnya adalah *inverse document frequency* yang berfungsi untuk menilai seberapa penting suatu istilah muncul dalam seluruh kumpulan dokumen dengan persamaan :

$$Idf = ln + \frac{N}{df} + 1$$ (Darwis et al., 2020a)

Di mana:

a. ln : Logarima Natural
b. N : Jumlah semua dokumen
c. df :Jumlah *term*/ kata pada dokumen

Pada proses *TF-IDF* cara menghitung adalah dengan dengan mengalikan nilai *Term Frequency* (TF) dan Inverse Document Frequency (IDF) (Darwis et al., 2020b; Syafa Fahreza & Setiawan, 2024). Untuk rumusnya adalah :

$$W_t = TF_t \, x \, IDF_t$$

TF dan IDF dihitung berdasarkan frekuensi kemunculan kata dan jumlah dokumen yang mengandung kata tersebut. Hasil dari perhitungan TF-IDF memberikan bobot yang lebih tinggi pada kata-kata yang lebih spesifik dan relevan, sehingga meningkatkan efektivitas dalam aplikasi seperti pencarian informasi dan klasifikasi teks.

2.2.9 *Support Vector Machine (SVM)*

Support Vector Machine (SVM) merupakan metode klasifikasi dalam pembelajaran terawasi (*supervised learning*) yang membutuhkan target atau tujuan tertentu selama proses pelatihannya. SVM bekerja dengan memanfaatkan pemetaan nonlinier untuk mentransformasikan data pelatihan ke dalam dimensi yang lebih tinggi. Pada dimensi ini, SVM berusaha menemukan hyperplane yang dapat memisahkan data secara linier, dan dengan pemetaan nonlinier yang sesuai, dapat

mencapai dimensi yang lebih kompleks (Mahmud Nawawi et al., 2024).

Awalnya, SVM dirancang untuk mengklasifikasikan data menjadi dua kelas yang dapat dipisahkan secara linier. Namun, banyak permasalahan di dunia nyata bersifat nonlinier. Untuk mengatasi hal ini, SVM menggunakan teknik kernel trick, yang berfungsi memetakan data dari ruang input ke ruang vektor berdimensi lebih tinggi sehingga data dapat dikelompokkan dengan lebih baik. Beberapa jenis fungsi kernel yang umum digunakan adalah *linear, polynomial,* dan *radial basis function* (RBF)

Dalam proses pembentukan model, SVM hanya menggunakan sejumlah data tertentu yang disebut *support vectors*. Data ini adalah titik-titik terdekat dengan *hyperplane* yang berperan penting dalam menentukan posisi dan orientasi *hyperplane* tersebut. Persamaan SVM untuk menggunaka rumus :

$$w \cdot x + b = 0$$ (Maulana et al., 2023)

Di mana :

a. w adalah parameter hyperplane yang dicari (garis yang tegak lurus antara garis *hyperplane* dan titik support vector)
b. x adalah titik data masukan *Support Vector Machine*
c. b = parameter hyperlane yang dicari (nilai bias)

2.2.10 Evaluasi

Evaluasi model bertujuan untuk menilai seberapa baik model tersebut dalam melakukan prediksi. Beberapa metode evaluasi yang umum digunakan meliputi confusion matrix, akurasi, *precision*, *recall*, dan *F1-Score* . *Confusion Matrix* adalah tabel yang digunakan untuk menggambarkan kinerja model klasifikasi dengan menunjukkan jumlah prediksi yang benar dan salah untuk setiap kelas. Tabel ini memberikan informasi tentang *true positives* (TP), *true negatives* (TN), *false positives* (FP), dan *false negatives* (FN), yang kemudian dapat digunakan untuk menghitung metrik evaluasi lainnya (Harnelia, 2024)

a. Akurasi mengukur seberapa sering model menghasilkan prediksi yang benar, dihitung dengan rumus:

$$\text{Akurasi} = \frac{TP+TN}{TP+TN+FP+FN}$$

b. *Precision* menunjukkan proporsi prediksi positif yang benar dari semua prediksi positif yang dihasilkan oleh model:

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

c. *Recall* atau sensitivitas mengukur seberapa banyak dari semua kasus positif yang berhasil dideteksi oleh model:

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

d. *F1-Score* adalah ukuran yang menggabungkan *precision* dan *recall*, memberikan gambaran yang lebih lengkap tentang kinerja model terutama pada dataset yang tidak seimbang dengan rumus:

$$F1\text{-}Score = 2 \text{ x } \frac{Precision \times Recall}{Precision+Recall}$$