BAB II
LANDASAN TEORI

2.1 Tinjauan Pustaka

2.1.1 Data *mining*

> Data *mining* adalah proses mencari sumber informasi untuk sebuah data. Data informasi ini bisa didapatkan dengan menggunakan teknik yang kompleks seperti kecerdasan buatan (AI), metode statistik, pembelajaran mesin (*machine learning*), matematika, dan lainnya. Teknik yang disebutkan nantinya berguna untuk melihat informasi dari *database* dengan volume yang besar. Data *mining* berasal dari metode pembelajaran mesin dan statistik, tetapi bisa digunakan juga untuk bidang ilmu komputer dan disiplin ilmu lainnya (Iwandini et al., 2023).

2.1.2 *Text mining*

"*Text mining* merupakan cabang dari ilmu data *mining*. *Text mining* digunakan untuk memproses penambangan data teks mentah (tidak terstruktur) dalam jumlah banyak untuk menemukan informasi tertentu. Salah satu pengimplementasian yang popular yaitu analisis sentimen" (Putri et al., 2023).

"*Text mining* adalah proses mengekstraksi sebuah pola untuk diteliti yang datanya berasal dari teks. *Text mining* adalah disiplin ilmu yang didasarkan untuk *information retrival, machine learning,* data *mining,* statistik dan *linguistic* komputasi" (Kumala et al., 2023).

2.1.3 Ulasan Pengguna

"Ulasan atau *Review* merupakan kalimat atau teks berisikan penilaian dan komentar yang diberikan untuk hasil karya seseorang, ulasan bisa dimanfaatkan untuk dijadikan masukan untuk perbaikan" (Herlinawati et al., 2020).

2.1.4 *Sentiment Analysis*

"Analisis Sentimen yaitu proses menemukan sentimen (pendapat/pandangan) dari suatu kalimat atau teks apakah berperasaan negatif, positif atau netral. Pendekatan ini juga digunakan untuk mengetahui pandangan publik terhadap isu-isu, topik, kebijakan dan pelayanan berdasarkan data tekstual" (Aulia et al., 2023). Analisis sentimen memiliki tahapan dalam pengolahan datanya yaitu: (Aulia et al., 2023).

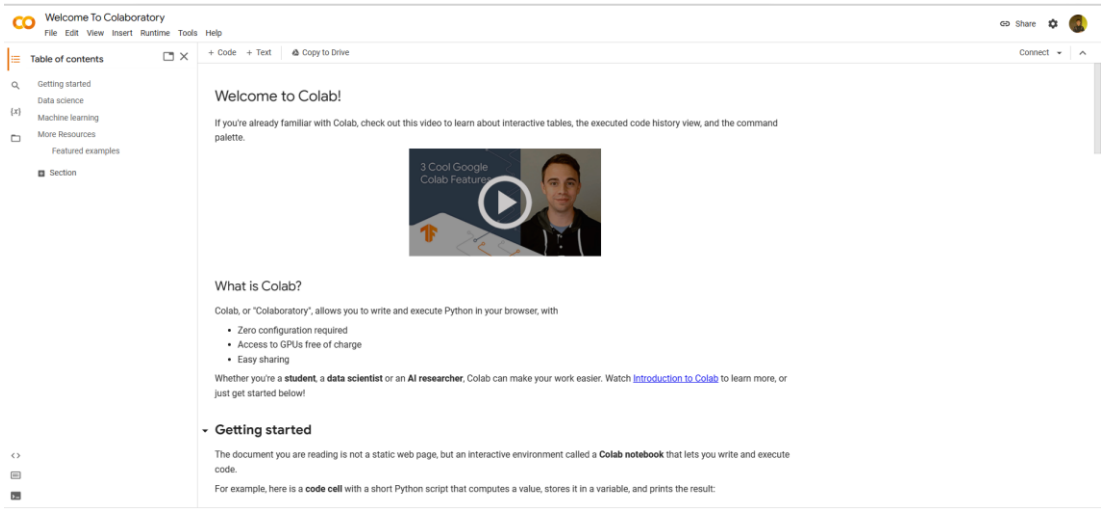
1. Pengumpulan data, proses awal untuk mendapatkan dan mengumpulkan data yang akan dianalisis.
2. *Preprocessing*, proses untuk memperoleh data yang sesuai dengan ketentuan klasifikasi. Seperti penghapusan data dari *noise*, menyamakan bentuk kata dan mengurangi ukuran data untuk memperoleh kumpulan data yang cocok dipakai saat pengklasifikasian.
3. Klasifikasi, proses untuk mengidentifikasi pola sentimen (pandangan atau pendapat) dari pesan teks yang sudah melalui proses *preprocessing*.
4. Evaluasi performa, proses untuk menghitung *accuracy*, presisi, *recall* dan f-*measure* atau *F1-score*.

2.1.5 Pemrograman *Python*

"*Python* adalah bahasa pemrograman interpretatif yang menggabungkan kapabilitas, kemampuan, dengan sintaksis yang jelas, dan dilengkapi fungsionalitas pustaka standar yang luas dan komprehensif, komunitas *python* juga sudah sangat banyak. Seperti pemrograman lain, *python* digunakan sebagai bahasa *script* tetapi tidak seperti bahasa *script* lainnya"(Kurniawan T. & Syahrudin A. N., 2018).

2.1.6 *Google Colab*

"*Google Colab* adalah IDE (*Integrated Development Environment*) untuk pemrograman *python*, dan proses berjalannya dilakukan oleh server *Google* yang difasilitasi perangkat lunak berupa pustaka program yang lumayan lengkap dan spesifikasi perangkat keras dengan performa yang tinggi" (Gelar Guntara, 2023).



Sumber: https://colab.research.google.com/

Gambar II. 1
Google Colaboratoty

2.1.7 Web *Scraping*

"Web *Scraping* merupakan teknik otomatisasi untuk mengekstrak data dari sebuah *website*, *database*, atau sistem *legacy* yang belum terstruktur, kemudian hasilnya disimpan berupa file dengan format *comma-separated values* (CSV)" (Diandra Audiansyah et al., 2022).

2.1.8 *Preprocessing*

"*Preprocessing* data adalah tahapan-tahapan untuk melakukan persiapan pemrosesan data yang akan dilakukan pemodelan. Pada tahapan ini mencakup beberapa kegiatan untuk membentuk data dan juga melakukan pembersihan data

sehingga data bisa diproses ke tahap selanjutnya" (Chohan et al., 2020). Tahapan-tahapan ketika melakukan *preprocessing* data (Putri et al., 2023), sebagai berikut:

1. *Case Folding*, proses mengganti semua huruf menjadi huruf kecil.
2. *Cleaning*, proses menghapus angka, tanda baca, emoji, dan sebagainya.
3. *Tokenization*, proses untuk memotong *string* berdasarkan dari setiap kata.
4. *Normalize*, proses memperbaiki kata yang belum sesuai ejahan yang disesuaikan (EYD).
5. *Stemming*, proses mengubah kata berimbuhan menjadi bawaan kata dasar.

2.1.9 TF-IDF

"*Term Frequency-Inverse Document Frequency* adalah teknik perhitungan atau pemberian nilai bobot pada setiap kata dalam *dataset*, berfungsi untuk menunjukkan pentingnya setiap kata dalam *dataset* tersebut setelah dilakukan proses *preprocessing* seperti *tokenize*, *stemming* dan frekuensi munculnya kata dalam dokumen" (Aulia et al., 2023).

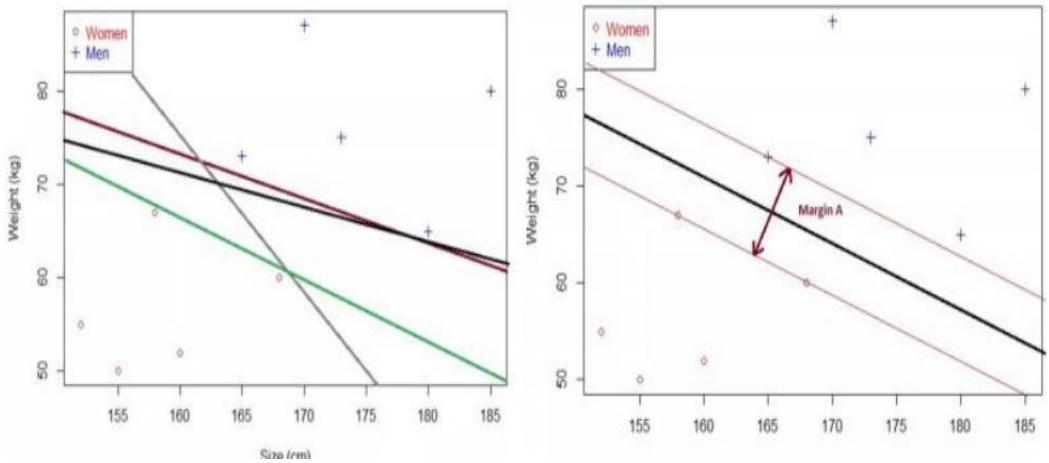
2.1.10 *Machine Learning*

"*Machine learning* adalah teknologi kecerdasan buatan (AI) yang memungkinkan sistem untuk belajar secara otomatis dari pengalaman dan menjadi lebih baik tanpa perlu dirancang secara eksplisit. *Machine learning* memberikan perspektif baru dan membantu dalam pemecahan masalah bagi organisasi penelitian" (Anwar, 2022).

2.1.11 *Support Vector Machine*

"*Support Vector Machine* (SVM) adalah teknik yang digunakan untuk melakukan prediksi, baik dalam konteks regresi maupun klasifikasi. Untuk membedakan antara dua kelas yang berbeda, prinsip kerja SVM adalah memilih *hyperplane* terbaik dengan margin maksimum yang diizinkan" (Muttaqin & Kharisudin, 2021).

> Tujuan SVM yaitu mencari *hyperplane* yang optimal. *Hyperplane* yang bisa membagi kedua class dengan jarak margin terjauh antar kelas. Margin adalah jarak antara *hyperplane* dan pola masing-masing class. *Instance* terdekat disebut *support vector*. Di bawah adalah contoh *hyperplane* dengan keterangan garis merah yang di atas garis hitam tebal dengan tanda "+" sebagai *support vector* kelas *men*, dan garis merah di bawah garis hitam tebal dengan tanda "o" sebagai *support vector* kelas *women* (Pamungkas et al., 2019).



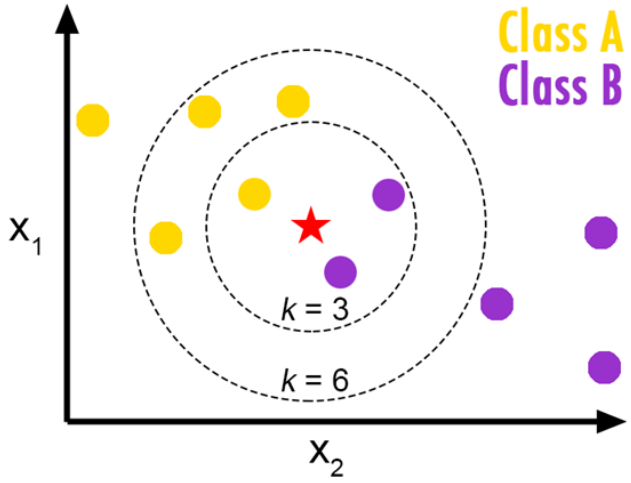
Sumber: (Pamungkas et al., 2019)

Gambar II. 2
Visualisasi *hyperplane* tidak maksimal dan maksimal

2.1.12 *K-Nearest Neighbor*

"K-*Nearest Neighbor* (k-NN/KNN) adalah sebuah teknik atau metode dalam melakukan klasifikasi terhadap objek yang berdasar pada data pembelajaran (*neighbor*) yang jaraknya paling dekat dengan objek tersebut. Dekat atau jauhnya *neighbor* dihitung berdasarkan jarak *Euclidean*" (Angreni et al., 2019).

Metode KNN bekerja dengan mencari pola K yang terdekat kemudian menentukan kelas keputusan yang berdasar pada jumlah pola terbanyak. Prosesnya menghasilkan K dengan akurasi tertinggi dalam menggeneralisasi data yang akan datang. Masalahnya, k ini tidak dapat ditentukan secara matematik, sehingga proses pelatihannya dengan observasi terhadap sebanyak k sampai mendapatkan hasil k yang paling optimum (Pamungkas et al., 2019).



Sumber: (Pamungkas et al., 2019)

Gambar II. 3
Visualisasi K-Nearest Neighbor

2.1.13 *K-Fold Cross Validation*

"*K-Fold Cross Validation* adalah salah satu teknik pengujian data yang melipat gandakan data menjadi bagian-bagian n dan membaginya menjadi bagian-bagian n dengan jumlah yang sama (misalnya, bagian K, L, M, dan N), masing-masing memiliki data yang berbeda, kemudian diterapkan pada data"(Aulia et al., 2023).

"*K-Fold Cross Validation* merupakan suatu metode dari *Cross Validation* yang banyak digunakan, penggunaannya metodenya dengan cara melipat data sebanyak k dan mengulangi (iterasi) eksperimennya sebanyak k" (Fajar et al., 2018).

Sumber: (Rangga et al., 2019)

Gambar II. 4
Ilustrasi K-Fold Cross Validation

2.1.14 *Wordcloud*

"*Wordcloud* merupakan visualisasi untuk mengetahui kata-kata yang sering muncul dalam suatu dokumen, dan bisa mengklasifikasikannya berdasarkan kategori sentimennya. Semakin besar ukuran *font* pada *wordcloud*, semakin sering juga kata itu digunakan" (Muttaqin & Kharisudin, 2021).

2.1.15 *Confusion Matrix*

"Metode yang digunakan untuk mengevaluasi seberapa akurat metode klasifikasi dalam memprediksi kelas data di mana teknik ini menghasilkan perbandingan nilai kelas aslinya dengan nilai prediksi. Pada *confusion matrix* dilakukan perhitungan yaitu, Akurasi, Presisi, *Recall*, dan F-*measure*" (Aulia et al., 2023).

2.2 Penelitian Terkait

Skripsi ini memiliki konteks dasar metode klasifikasi sentimen ulasan pengguna di *Google Play Store*. Dalam pengerjaan skripsi ini digunakan beberapa penelitian terdahulu sebagai pedoman dan referensi pengerjaan skripsi. Penelitian terkait bidang analisis sentimen ini sudah banyak dilakukan oleh peneliti-peneliti sebelumnya karena topik ini sangat menarik untuk dibahas.

Penelitian yang dilakukan Siti Masturoh dkk., dengan judul *Application Of* The *K-Nearest Neighbor* (K-NN) *Algorithm In Sentiment Analysis Of The Ovo E-Wallet Application*, penelitian ini membahas penggunaan metode KNN dalam analisis sentimen aplikasi OVO E-*Wallet*, dari 5000 data yang diambil, kemudian dipecah menjadi 90% data *training* dan 10% data *test*, peneliti meng-kategorikan data menjadi 3 kelas (kelas pertama bintang 1 sampai bintang 5, kelas ke dua bintang 1 dan bintang 5, kelas ke tiga memberikan pengelompokan bintang 1 dan 2 label negatif, bintang 3 label netral dan bintang 4 dan 5 label positif. Pengujian dilakukan dengan mencari nilai k dari nilai k 1-10, dan hasilnya didapatkan nilai akurasi tertinggi 84.86% pada pengujian kelas ke 2, berarti ovo sudah memberikan kesan yang baik kepada pengguna. (Masturoh et al., 2023).

Penelitian yang dilakukan Alfio Kusuma, dkk., dengan judul Analisis Sentimen Pada Ulasan Aplikasi Indodax di *Google Play Store* Menggunakan Metode *Support Vector Machine*, penelitian ini membandingkan *kernel-kernel* yang ada dalam metode SVM. Data yang digunakan sebanyak 1138 data ulasan, yang didapatkan dari halaman *review Google Play Store*, sebelum dilakukan *preprocessing* data ulasan yang didapatkan diberi label berdasarkan *Rating* atau *Score* yang diberikan pengguna, ketentuannya label negatif atau ‘0’ diberikan jika *score* atau *rating* bernilai 1, 2, 3 dan label positif atau ‘1’ diberikan jika *score* atau *rating* bernilai 4, 5. Didapatkan 565 sentimen positif dan 573 sentimen negatif. Setelah pelabelan dilanjutkan dengan *preprocessing*, TF-IDF dan implementasi dengan berbeda *kernal* (*Linear, RBF, Polynomial, Sigmoid*) dan berbeda data yang di split menjadi (80:20, 70:30, 60:40). Hasilnya, akurasi tertinggi didapatkan dari pembagian rasio data tes 80:20 dengan 910 data latih dan 228 data uji, dan akurasi didapatkan 85%. (Kusuma & Nurramdhani Irmanda, 2022).

Penelitian yang dilakukan Muhammad Rangga Aziz Nasution, dkk., dengan judul Perbandingan Akurasi Dan Waktu Proses Algoritma KNN Dan SVM Dalam Analisis Sentimen Twitter, penelitian ini membandingkan dua algoritma yaitu *Support Vector Machine* dan *K-Nearest Neighbor* dalam segi akurasi dan waktu proses. Menggunakan 1113 data dan diproses melalui tahapan *preprocessing*, selanjutnya data dibagi menjadi rasio 80:20 data *training*:data *test*. Hasil perhitungan yang didapatkan dari segi akurasi, SVM hasilnya 88,76% dan K-NN hasilnya 88,1%. Dari segi perhitungan proses, SVM hasilnya 0.4532s pada waktu prosesnya, dan K-NN 0.1505s lebih cepat dari metode SVM, disimpulkan bahwa metode *K-Nearest Neighbor* memiliki performa yang lebih baik dari *Support Vector Machine.* (Rangga et al., 2019).

Penelitian lainnya dilakukan Atang Saepudin, dkk., dengan judul Optimasi Algoritma SVM Dan K-NN Berbasis *Particle Swarm Optimization* Pada Analisis *Sentiment* Fenomena Tagar #2019GantiPresiden, penelitian ini menggunakan algoritma SVM dan KNN, lalu ditambahkan optimasi yaitu *Particle Swarm Optimization*. Data yang digunakan sebanyak 400 *review* terdiri dari 200 *review* positif dan 200 *review* negatif. Hasilnya SVM menghasilkan akurasi 88,00% dengan AUC 0,964 (*Area Under Curve*) dan KNN menghasilkan akurasi 88,50% dengan AUC 0,948. Jika kedua perbandingan ditambahkan optimasi *Particle Swarm Optimization* (PSO) maka hasilnya akurasi SVM+PSO 92.75% dengan AUC 0.973 dan akurasi KNN+PSO 75.25% dengan AUC 0.768. Dari data tersebut akurasi SVM menggunakan optimasi terbukti lebih baik dalam pengklasifikasian data *tweet.* (Saepudin et al., 2020).

Penelitian yang dilakukan oleh M. Nurul Muttaqin dkk., dengan judul Analisis Sentimen Aplikasi Gojek Menggunakan Support Vector Machine dan K-Nearest Neighbor. Tujuan penelitian ini adalah melakukan klasifikasi pada ulasan aplikasi

Gojek menggunakan SVM dan KNN. Sumber data yang digunakan diambil dari Google Play Store, karena data yang didapatkan murni opini pengguna tanpa ada promosi dan iklan. Hasilnya metode KNN dengan K=22 memperoleh nilai akurasi, presisi, dan recall dengan 82,14%, 82,28%, dan 95,43% dan SVM memperoleh nilai akurasi, presisi, dan recall berturut-turut sebesar 87,98%, 88,55%, dan 95,43%. Disimpulkan bahwa metode SVM melakukan klasifikasi dengan lebih baik dari metode KNN. (Muttaqin & Kharisudin, 2021).

2.3 Objek Penelitian

2.3.1 Aplikasi *Home Credit*

My *Home Credit* merupakan aplikasi buatan PT. *Home Credit* Indonesia atau *Home Credit*. PT *Home Credit* Indonesia adalah perusahaan yang telah beroperasi sejak 2013, bergerak dalam bidang usaha pembiayaan multiguna multinasional. *Home Credit* Indonesia adalah salah satu anak perusahaan *Home Credit* B.V.(HCBV). HCBV didirikan tahun 1997 di dirikan di Ceko, HCBV merupakan perusahaan *Home Credit* secara global, HCBV fokus dalam mengembangkan strategi, teknologi, risiko produk dan juga pendanaan yang menyesuaikan dengan kebutuhan pasar di tiap negara. Agar bisnis pembiayaan ini terus berkembang, dibuatlah aplikasi *Home Credit* agar pelanggan lebih mudah mengajukan persyaratan peminjaman, *Home Credit* bisa di *download* melalui *apps store* (iOS) dan *Play Store* (Android). Cukup siapkan dokumen pendukung seperti Kartu Tanda Penduduk, Kartu Keluarga, dan nomor *handphone* yang aktif sudah bisa mengajukan pinjaman.