# BAB II LANDASAN TEORI

## Tinjauan Pustaka

Penelitian ini dilakukan tidak terlepas dari hasil penelitian-penelitian terdahulu yang digunakan sebagai bahan perbandingan dan referensi. Penelitiain ini bereferensi pada penelitian sebelumnya yang memiliki persamaan penelitian yaitu meneliti tentang tingkat sentimen ulasan pengguna aplikasi. Sedangkan perbedaan penelitian adalah dari studi kasus penelitian. Penelitian pertama adalah Sentimen Analisis Review Aplikasi Digital Korlantas Pada *Google Play Store* Menggunakan Metode *SVM* oleh Nanda Ressq Setiawan dan Emil R. Kaburuan, 2023. Penelitian tersebut memiliki kesamaan yaitu menggunakan metode *SVM* dalam melakukan analisis sentimen, namun perbedaanya terletak pada studi kasus yang diteliti yaitu ulasan aplikasi korlantas.

Penelitian kedua adalah Analisis Sentimen Ulasan Aplikasi Dana dengan Metode *Random Forest* oleh Fanka Angelina Larasat, Dian Eka Ratnawati, dan Buce Trias Hanggara, 2022. Penelitian tersebut memiliki kesamaan yaitu menggunakan metode *Random Forest* dalam melakukan analisis sentimen namun perbedaanya terletak pada studi kasus yang diteliti yaitu ulasan aplikasi Dana. Penelitian ketiga adalah Perbandingan Metode *Random Forest* dan Naive Bayes pada Analisis Sentimen Review Aplikasi BCA Mobile oleh Akhmad Miftahusalam, Hasih Pratiwi, dan Isnandar Slamet, 2023. Penelitian tersebut memiliki kesamaan yaitu meneliti tentang perbandingan metode *Random Forest* dalam melakukan analisis sentimen namun perbedaanya terletak pada metode pembanding nya dan studi kasus yang digunakan. Penelitian keempat adalah Perbandingan Algoritma Support Vector Machine dan *Random Forest* untuk Analisis Sentimen Terhadap Kebijakan Pemerintah Indonesia Terkait Kenaikan Harga BBM Tahun 2022 oleh Muhamad Samantri, dan Afiyati, 2023. Penelitian tersebut memiliki kesamaan yaitu meneliti tentang perbandingan metode *SVM* dalam melakukan analisis sentimen namun perbedaanya terletak pada metode pembanding nya dan studi kasus yang digunakan. Penelitian kelima adalah *Sentiment Analysis and Classification of Restaurant Reviews using Machine Learning* oleh Kanwal Zahoor, Narmeen Zakaria Bawany, dan Soomaiya Hamid, 2020. Penelitian tersebut memiliki kesamaan yaitu meneliti tentang perbandingan metode *SVM* dan *Random Forest* dalam melakukan analisis sentimen namun perbedaanya terletak pada studi kasus yang digunakan. Adapun uraian penelitian terdahulu dalam penelitian ini dijabarkan pada tabel berikut:

Tabel 2. 1 Peneliti Terdahulu

|  |  |  |  |
| --- | --- | --- | --- |
| Peneliti | Tahun | Judul | Hasil Penelitian |
| Nanda Ressq Setiawan, dkk. | 2023 | Sentimen Analisis  Review Aplikasi Digital Korlantas  Pada *Google Play Store*  Menggunakan Metode  *SVM*. | Model *SVM* lebih diunggulkan pada skenario rasio data 90:10. Dengan tingkat  nilai akurasi 82%[10]. |
| Fanka Angelina Larasat, dkk. | 2022 | Analisis Sentimen Ulasan Aplikasi Dana dengan Metode *Random Forest*. | Menggunakan metode *Random Forest* dengan rasio data 80:20 didapatkan nilai presisi 84%, *recall* 84%, *F1- Score* 84% dan akurasi  84% [11]. |
| Akhmad Miftahusalam, dkk. | 2023 | Perbandingan Metode *Random Forest* dan Naive Bayes pada Analisis Sentimen Review Aplikasi BCA Mobile. | Setelah dilakukan pengujian dengan skala perbandingan dataset 80:20, disimpulkan metode *Random Forest* menghasilkan prediksi yang lebih baik dengan perbandingan nilai akurasi sebesar 93,93%  untuk *Random Forest,* |

Berdasarkan dari penelitian diatas, maka penelitian ini akan melakukan perbandingan antara metode *SVM* dan *Random Forest* guna mencari metode mana yang lebih baik dan akurat dalam menentukan sentimen ulasan pada aplikasi M-Paspor.

## M-Paspor

Aplikasi M-Paspor adalah aplikasi yang dikembangkan oleh Direktorat Jenderal Imigrasi, yang sebelum nya dikenal dengan nama Aplikasi Pendaftaran Antrian Paspor Online (APAPO). Pada aplikasi tersebut, masyarakat dapat melakukan pendaftaran untuk permohonan paspor baru dan pendaftaran terkait permohonan penggantian paspor[7].

## Analisis Sentimen

Analisis sentimen adalah cabang ilmu dari *natural language program*, yang berfokus pada mengekstrak informasi subjektif dan sentimen dari suatu teks. Dengan sentimen analisis, informasi yang tadinya tidak terstruktur dapat diubah menjadi data yang lebih terstruktur. Tugas utama analisis sentimen adalah mengklasifikasikan teks dalam dokumen, kalimat, atau opini menjadi sisi positif atau negatif [8].

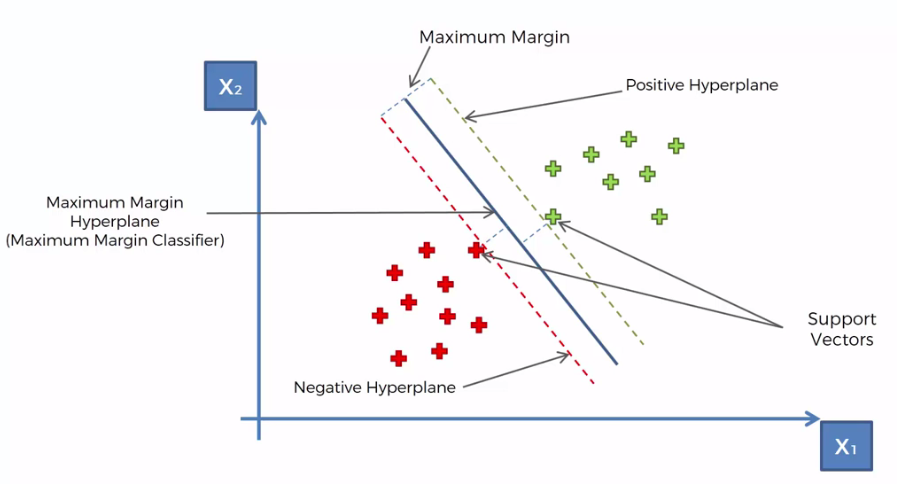
## Ulasan

Menurut Kamus Besar Bahasa Indonesia (KBBI), ulasan merupakan suatu pendapat atau komentar tentang suatu peristiwa. Ulasan adalah kalimat atau text yang memberikan penilaian atau komentar atas karya seseorang [9].

## *Support Vector Machine (SVM)*

Dalam metode *supervised learning*, terdapat metode *Support Vector Machine* (*SVM*) yang biasanya digunakan untuk klasifikasi. Algoritma *SVM* dipilih karena mampu menemukan *hyperplane* terbaik sebagai pemisah. Ini dapat digunakan untuk memisahkan dua kelas data, dalam hal ini kelas positif dan negatif[9]. Berikut adalah konsep dasar dari metode *SVM*:

1. ***Hyperplane***: Dalam konteks *SVM*, *hyperplane* adalah dimensi (n-1) yang digunakan untuk memisahkan data ke dalam dua kelas. Misalnya, dalam kasus dimensi dua, *hyperplane* adalah garis lurus; dalam dimensi tiga, *hyperplane* adalah bidang datar.
2. ***Margin***: *Margin* adalah jarak antara *hyperplane* dan titik-titik terdekat dari kedua kelas. *SVM* berusaha untuk menemukan *hyperplane* yang memiliki *margin* maksimum. Ini berarti *hyperplane* tersebut dapat memaksimalkan jarak antara titik-titik terdekat dari kedua kelas, sehingga meningkatkan kemampuan umum model untuk mengklasifikasikan data baru dengan benar.
3. ***Support Vectors***: *Support vectors* adalah titik-titik data yang berada paling dekat dengan *hyperplane*. Mereka merupakan titik-titik kunci yang mendefinisikan posisi dan orientasi *hyperplane*. Dalam *SVM*, hanya *support* *vectors* yang mempengaruhi penentuan *hyperplane*, sehingga model *SVM* dapat efisien bekerja dengan jumlah fitur atau dimensi yang besar.



Gambar 2. 1 SVM

Rumus perhitungan *hyperplane* dapat ditemukan di bawah ini:

|  |
| --- |
|  |

Dimana:

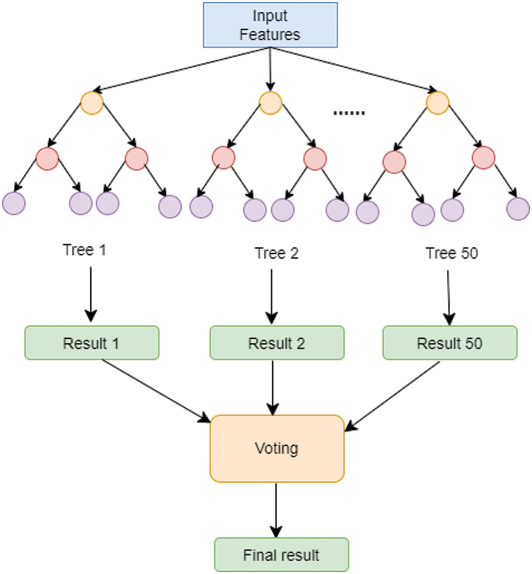
w : Parameter *hyperplane* yang dicari (Garis yang tegak lurus antara garis *hyperplane* dan titik *support vector*).

x : Data input *SVM* (x1 = index kata, x2 = bobot kata).

b : Parameter *hyperplane* yang dicari (Nilai bias).

## *Random Forest*

*Random Forest* merupakan metode machine learning yang berisi gabungan *Decision Tree* untuk dilakukan klasifikasi, dimana untuk membuat keputusan akhir perlu diterapkan *voting mayority* [10]. Metode *Random Forest* dapat bekerja dengan optimal pada ukuran dataset yang sangat besar untuk menyelesaikan tugas klasifikasi, namun memiliki kekurangan yaitu waktu pelatihan yang cukup lama dibanding model lainnya.



Gambar 2. 2 Random Forest

Proses pembentukan setiap pohon keputusan dalam *Random Forest* adalah sebagai berikut:

1. Pemilihan Sampel Acak: Dari dataset pelatihan yang tersedia, sampel data diambil secara acak dengan penggantian (*bootstrap*) untuk membentuk dataset yang berukuran sama dengan dataset pelatihan asli.
2. Pemilihan Fitur Acak: Dari jumlah fitur yang tersedia, subset fitur diambil secara acak untuk digunakan dalam membangun pohon keputusan. Biasanya, jumlah fitur yang diambil setiap kali jauh lebih kecil dari jumlah total fitur yang tersedia.
3. Membangun Pohon Keputusan: Dengan menggunakan sampel data dan subset fitur yang telah dipilih, pohon keputusan dibangun dengan menggunakan algoritma seperti ID3, C4.5, atau CART. Pohon ini dibentuk dengan melakukan pemisahan data berdasarkan fitur-fitur yang paling informatif, dengan tujuan untuk meminimalkan keragaman (*variance*) dalam setiap simpul pohon.
4. Pembentukan Ensambel: Langkah-langkah 1 hingga 3 diulang beberapa kali untuk membentuk kumpulan pohon keputusan. Setiap pohon keputusan dalam *Random Forest* memberikan suara atau memberikan prediksi terhadap kelas atau nilai regresi yang diinginkan.
5. Keputusan Majoritas: Prediksi akhir dalam *Random Forest* diperoleh dengan mengambil mayoritas suara atau rata-rata prediksi dari semua pohon keputusan dalam ensambel.

## *Preprocessing*

Langkah pertama dalam proses pelatihan ialah dengan melakukan preprocessing. Tujuannya adalah untuk memperbaiki data yang tidak konsisten, mengatasi masalah pada data, dan memperbaiki data bermasalah [11]. *Cleaning, case folding, filtering, tokenizing, slangword conversion, stopword removal*, dan *stemming* adalah bagian dari langkah *preprocessing* data.

## Jupyter Notebook

*Jupyter notebook* adalah aplikasi pengembangan interaktif berbasis web, *Jupyter notebook* yang memiliki singkatan *julia* (JU) *python* (PY) dan R sesuai dengan bahasa yang biasa digunakan pada *Jupyter notebook*[12]. *Jupyter notebook* ini bersifat gratis dan yang dapat digunakan untuk visualisasi hasil analisis, dan penentuan keputusan dari pengolahan data.

## Text Mining

Text Mining adalah proses untuk mengekstrak informasi dari data sumber untuk dilakukan analisis, mengelompokan informasi berdasarkan kata-kata, agar dapat mengetahui hubungan dari data sumber lainnya [13].

## *Web Scraping*

Web Scraping merupakan cara untuk mengambil sekumpulan data yang tidak terstruktur dari suatu web secara otomatis dengan perintah program. Hasil dari web scraping biasanya disimpan dalam bentuk spreadsheet atau comma-separated values [14].

## *Confusion Matrix*

Confusion matrix atau matriks kebingungan adalah alat yang digunakan untuk mengukur kinerja model klasifikasi pada data uji yang sudah diketahui hasil sebenarnya. Dalam *confusion matrix* terdapat 4 bagian penting, yaitu:

|  |  |
| --- | --- |
| True Positive(TP) | False Positive(FP) |
| True Negative(TN) | False Negative(FN) |

Dari penggunaan *confusion matrix,* dapat dihitung nilai *precision, accuracy, recall, F1- score*, dengan rumus perhitungan sebagai berikut [15] :

1. Akurasi adalah metrik evaluasi yang mengukur seberapa baik model memprediksi yang benar dari semua pengamatan. Rumus untuk mencari nilai akurasi ialah:

𝐴𝑐𝑐𝑢𝑟𝑎𝑐𝑦 = x100% =

1. Presisi adalah metrik evaluasi yang mengukur prediksi benar positif dibandingkan dengan keseluruhan hasil yang diprediksi positf. Rumus untuk mencari nilai presisi ialah:

*Precision* = x100% =

1. *Recall* adalah metrik evaluasi yang mengukur prediksi benar positif dibandingkan dengan keseluruhan data yang benar positif. Rumus untuk mencari nilai *recall* ialah:

*Recall* = x100% =

1. *F1-Score* merupakan metrik evaluasi yang mencerminkan keseimbangan antara Presisi (*Precision*) dan Sensitifitas (*Recall*). Rumus untuk mencari nilai *f1-score* ialah:

*F1-Score* = =