BAB 2
LANDASAN TEORI

2.1 Naturalisasi/Pewarganegaraan

Naturalisasi/Pewarganegaraan adalah proses yang memungkinkan orang asing untuk menjadi warga negara dari negara lain. Proses ini dapat dilakukan melalui berbagai cara, baik melalui tindakan hukum maupun melalui persyaratan lain yang ditetapkan oleh negara tersebut. Berdasarkan Undang-Undang Nomor 12 Tahun 2006, terdapat dua jenis proses naturalisasi yang dapat ditempuh oleh warga negara asing, yaitu naturalisasi biasa dan naturalisasi istimewa. Naturalisasi biasa umumnya diberikan kepada individu yang telah memenuhi persyaratan umum, seperti jangka waktu tinggal tertentu di wilayah Indonesia dan hubungan perkawinan dengan warga negara Indonesia. Sementara itu, naturalisasi istimewa merupakan bentuk pengecualian yang diberikan kepada individu yang memiliki kontribusi luar biasa atau keahlian khusus yang sangat dibutuhkan oleh negara. [9].

2.2 Analisa Sentimen

Analisis sentimen merupakan teknik untuk mengidentifikasi dan mengukur opini, perasaan, atau emosi yang diungkapkan dalam data teks. [10]. Fokus utama dari analisis sentimen adalah mempelajari pendapat, perasaan, penilaian, sikap, dan emosi orang-orang terhadap berbagai entitas. Entitas ini dapat meliputi produk, layanan, organisasi, individu, isu, topik, peristiwa, serta atribut yang terkait dengannya [11].

2.3 Text Preprocessing

Tahap pra-pemrosesan data merupakan langkah krusial yang perlu dilakukan. Pra-pemrosesan data memainkan peran penting dalam meningkatkan performa dan hasil akhir dari algoritma [11]. Berikut adalah beberapa proses pra-pemrosesan yang akan digunakan dalam penelitian ini:

1. *Cleaning*

   Tahap *cleaning* merupakan proses membersihkan karakter dari simbol selain huruf alfabet.

2. *Case Folding*

   Tahap *case folding* merupakan salah satu tahap penting dalam pengolahan data teks, dimana seluruh karakter dalam kalimat diubah menjadi huruf kapital ataupun huruf kecil.

3. *Normalization*

   Tahap *normalization* merupakan proses perubahan kata singkat yang tidak baku menjadi kata dasar yang baku.

4. *Tokenization*

   Tahap *tokenization* merupakan proses pemenggalan kata per kalimat.

5. *Stopward Removal*

   Tahap *stopward removal* merupakan penghapusan kata yang tidak penting dalam kalimat.

6. *Lemmatization*

   Tahap *lemmatization* adalah proses untuk mengubah kata berimbuhan menjadi bentuk kosakatanya.

7. *Labeling*

   Tahap *labelling* merupakan proses memberikan label positif, negatif ataupun netral pada suatu kalimat.

2.4 Data Splitting

Data splitting adalah proses pemecahan data menjadi dua bagian atau lebih. Dalam *data splitting*, pembagian data sering dilakukan dengan cara acak. Pembagian data pada umumnya dibagi menjadi *train* dan *test*. Rasio pembagian pada data dapat berbeda-beda. Pada penelitian ini, rasio 80:20 digunakan karena rasio tersebut umum digunakan. Selain itu, rasio tersebut berasal dari prinsip *Pareto* [23]. Contoh penelitian yang menggunakan rasio 80:20, yaitu analisis sentimen ulasan pada e-commerce Shopee [19].

2.5 TF-IDF

Term Frequency - Inverse Document Frequency (TF-IDF) merupakan metode ekstraksi fitur yang digunakan untuk memberikan bobot nilai pada setiap

kata dalam sebuah dokumen. Bobot ini mencerminkan tingkat relevansi kata tersebut terhadap dokumen. TF-IDF menghitung dua faktor yaitu *Term Frequency* (TF) yakni menghitung seberapa sering suatu kata muncul dalam dokumen dan *Inverse Document Frequency* (IDF) yakni mengukur seberapa penting suatu kata dalam dokumen dalam konteks koleksi yang lebih besar [24].

2.6 ADASYN

ADASYN (Adaptive Synthetic) adalah teknik resampling data yang dirancang khusus untuk menangani dataset tidak seimbang. Ini bertujuan untuk menghasilkan contoh kelas minoritas sintetis yang berfokus pada area ruang fitur tempat kelas minoritas kurang terwakili. ADASYN akan mengambil lebih banyak sampel kelas minoritas di dalam area *k-nearest neighborhood*. ADASYN menggunakan distribusi tertimbang untuk setiap sampel kelas minoritas berdasarkan kesulitan pembelajaran setiap kelas [12]. Berikut Rumus 2.1 yaitu perhitungan ADASYN:

$$X_{new} = X_i + \text{rand}(0,1) * (X_i - X_1) \tag{2.1}$$

Dengan keterangan sebagai berikut:

X new : Data sintetis baru yang dihasilkan.

Xi : Sampel data kelas minoritas sebagai titik dasar untuk menghasilkan data sintetis.

X1 : Salah satu dari k-tetangga terdekat dari Xi (juga dari kelas minoritas).

rand(0,1) : Angka acak antara 0 dan 1, digunakan untuk menambahkan elemen acak dalam proses pembuatan data sintetis.

(Xi-X1) : Vektor selisih antara Xi dan X1, yang menentukan arah di mana data sintetis akan dihasilkan.

2.7 Naive Bayes

Naïve Bayes merupakan metode dalam *machine learning* yang menggunakan perhitungan probabilitas dan statistik untuk memprediksi kemungkinan kejadian di masa depan berdasarkan pengalaman di masa lalu,

yang dikenal dengan Teorema Bayes. Dalam penerapannya, *Naive Bayes* mengasumsikan bahwa setiap atribut saling independen. Klasifikasi dengan *Naive Bayes* berasumsi bahwa kehadiran atau ketiadaan suatu fitur dalam sebuah kelas tidak berhubungan dengan fitur lainnya dalam kelas tersebut[25].

Naïve Bayes banyak digunakan dalam analisis sentimen. Algoritma ini memanfaatkan probabilitas dan statistik untuk memprediksi kemungkinan suatu teks termasuk dalam kategori sentimen tertentu [16]. Berikut Rumus 2.2 yaitu perhitungan algoritma *Naive Bayes*:

$$P(Y \mid X) = \left(\frac{P(X \mid Y)P(Y)}{P(B)} \right) \tag{2.2}$$

Dengan keterangan sebagai berikut:

Y : Kelas atau label yang ingin diprediksi dari suatu data.

X : Fitur atau atribut dari data yang akan digunakan untuk prediksi.

P(Y—X): Probabilitas posterior, yaitu probabilitas kelas Y diberikan fitur X.

P(Y) : Probabilitas prior, yaitu probabilitas kelas Y tanpa memperhitungkan fitur X.

P(X—Y): *Likelihood*, yaitu probabilitas mengamati fitur X jika data tersebut berasal dari kelas Y.

P(X) : Probabilitas marginal dari fitur X.

Algoritma ini memprediksi kelas data baru berdasarkan probabilitasnya terhadap kelas-kelas yang sudah ada, dengan menggunakan teorema *Bayes*. Berikut teorema *Bayes* yang digunakan dalam penelitian:

(a) *Multinomial Naive Bayes*

Algoritma *Multinomial Naive Bayes* berasumsi bahwa setiap kata dalam sebuah dokumen tidak saling terkait atau independen. Klasifikasi dokumen tidak hanya bergantung pada ada atau tidaknya kata, tetapi juga pada frekuensi kemunculan kata tersebut.

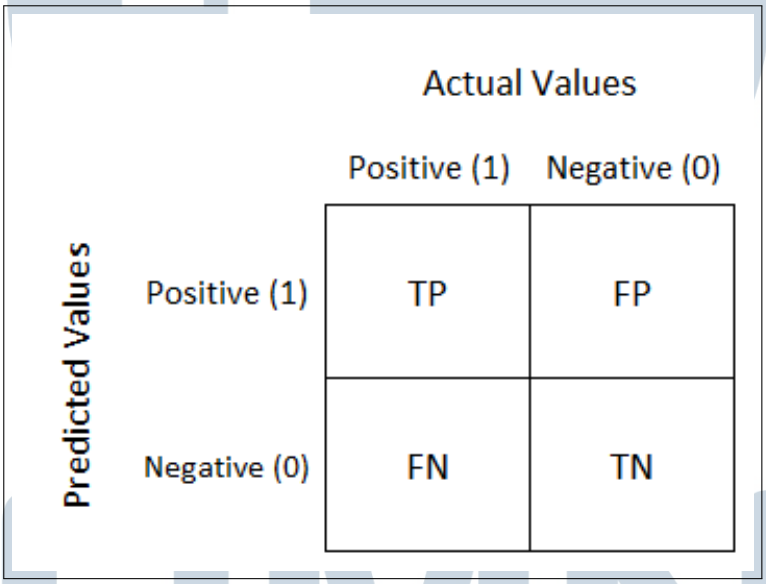
(b) *Complement Naive Bayes*

Algoritma *Complement Naive Bayes* (CNB) bekerja dengan cara yang berbeda dari *Multinomial Naive Bayes*. CNB tidak berasumsi bahwa setiap

kata independen. Algoritma ini justru memfokuskan perhitungannya dengan mengabaikan kata yang sama. Hal ini membuat CNB cocok dengan data yang tidak seimbang.

2.8 Confusion Matrix

Confusion matrix merupakan alat evaluasi yang digunakan untuk membandingkan hasil prediksi model dengan nilai kebenaran dasar dari data. Tabel ini menyajikan perbandingan antara output model klasifikasi dengan label kelas aktual. [26].



Gambar 2.1. *Confusion Matrix* (sumber: Medium)

Gambar 2.1 merupakan tabel *confusion matrix* yang memiliki empat kombinasi berbeda dari hasil nilai prediksi dan nilai aktual.

- TP (*True* Positif): Total sampel yang sebenarnya positif dan diprediksi positif.
- TN (*True* Negatif): Total sampel yang sebenarnya negatif dan diprediksi negatif.
- FP (*False* Positif): Total sampel yang sebenarnya negatif tetapi diprediksi positif.
- FN (*False* Negatif): Total sampel yang sebenarnya negatif tetapi diprediksi positif.

Perhitungan *accuracy*, *precision*, *recall*, dan *F1-score* didasarkan pada informasi yang terkandung dalam *confusion matrix*. Metrik tersebut dapat memberikan gambaran dengan lebih detail mengenai kinerja model klasifikasi.

2.8.1 Accuracy

Confusion matrix digunakan untuk menghitung akurasi model. Akurasi menunjukkan seberapa benar model dalam memprediksi data. Rumus dalam menentukan akurasi bisa dilihat pada Rumus 2.3.

$$Accuracy = \frac{TP + TN}{\text{TotalData}} \tag{2.3}$$

2.8.2 Precision

Confusion matrix digunakan untuk menghitung presisi model. Presisi menunjukkan seberapa tepat model memprediksi kelas positif dengan benar. Rumus lengkapnya bisa dilihat di Rumus 2.4.

$$Precision = \frac{TP}{TP + FP} \tag{2.4}$$

2.8.3 Recall

Confusion matrix digunakan untuk menghitung recall model. Recall menunjukkan seberapa banyak contoh positif yang berhasil ditemukan oleh model. Rumus lengkapnya bisa dilihat di Rumus 2.5.

$$Recall = \frac{TP}{TP + FN} \tag{2.5}$$

2.8.4 F1-Score

Confusion matrix digunakan untuk menghitung *F1-score* model. *F1-Score* menunjukan perbandingan rata-rata dari hasil *precision* dan *recall*. Rumus dalam menentukan akurasi bisa dilihat pada Rumus 2.6.

$$F1 - Score = 2 \times \frac{precision \times recall}{precision + recall} \tag{2.6}$$