BAB II
KAJIAN LITERATUR

2.1 Artificial Intelligence

Artificial Intelligence merupakan sebuah simulasi terhadap kecerdasan dan pemikiran manusia yang dilakukan oleh mesin-mesin yang terhubung dengan lautan data dan informasi, sehingga mesin-mesin tersebut memiliki kemampuan yang hampir menyerupai kapasitas dari kecerdasan manusia itu sendiri[14].

AI juga didefinisikan sebagai teknologi yang memungkinkan komputer meniru proses berpikir dan pembelajaran manusia melalui sistem yang dirancang secara cerdas dan fleksibel[15]. Selain itu, AI dipandang sebagai sistem berbasis data yang mampu mengolah informasi untuk menyelesaikan berbagai tugas secara otomatis dengan mengenali pola dalam data, sehingga dapat menghasilkan prediksi maupun pengambilan keputusan[16].

AI mencakup beberapa cabang utama yaitu Machine *Learning* yang memungkinkan sistem belajar dari data tanpa diprogram secara eksplisit, *Deep Learning* sebagai pengembangan berbasis jaringan saraf tiruan, *Natural Language Processing* (NLP) yang berfokus pada pemrosesan dan pemahaman bahasa manusia, *Computer Vision* yang berkaitan dengan pengolahan serta interpretasi data visual, serta *Expert System* yang digunakan untuk mendukung pengambilan keputusan berbasis pengetahuan[17].

Berdasarkan uraian tersebut, dapat disimpulkan bahwa *Artificial Intelligence* merupakan teknologi yang memungkinkan komputer untuk meniru kecerdasan manusia melalui proses pembelajaran berbasis data dan algoritma, serta didukung oleh berbagai cabang yang saling terintegrasi dalam membangun sistem yang cerdas dan adaptif.

2.2 Machine Learning

Machine Learning merupakan salah satu cabang dari Artificial Intelligence yang memungkinkan sistem komputer untuk mempelajari pola dari data dan meningkatkan kinerjanya tanpa perlu diprogram secara eksplisit. Melalui proses ini, sistem dapat mengenali hubungan dalam data sehingga mampu menghasilkan prediksi maupun keputusan secara otomatis[18].

Secara umum, cara kerja Machine Learning dimulai dari penggunaan data sebagai input yang kemudian diolah menggunakan algoritma tertentu untuk membentuk sebuah model. Model yang dihasilkan selanjutnya dimanfaatkan untuk melakukan prediksi terhadap

data baru, di mana tingkat akurasi hasilnya sangat bergantung pada kualitas data yang digunakan serta metode pelatihan yang diterapkan[19].

Dalam praktiknya, Machine Learning memiliki beberapa pendekatan utama, yaitu supervised learning, unsupervised learning, dan reinforcement learning, yang digunakan dalam berbagai kebutuhan seperti klasifikasi, prediksi, serta pengenalan pola[20].

Dengan demikian, Machine Learning menjadi dasar penting dalam pengembangan sistem cerdas karena kemampuannya dalam mengolah data dan menghasilkan keputusan secara otomatis[21].

2.2.1 Algoritma Machine Learning

Algoritma *Machine Learning* merupakan serangkaian metode yang digunakan untuk mengidentifikasi pola dalam data serta membangun model yang mampu melakukan prediksi atau pengelompokan secara otomatis. Algoritma ini memiliki peran penting dalam proses pembelajaran sistem karena menentukan bagaimana data diolah, dianalisis, dan diinterpretasikan menjadi informasi yang bernilai.

Secara umum, algoritma *Machine Learning* dapat dibagi menjadi tiga pendekatan utama, yaitu *supervised learning*, *unsupervised learning*, dan *reinforcement learning*. Pengelompokan ini didasarkan pada cara model mempelajari data serta jenis informasi yang digunakan selama proses pelatihan[22].

Supervised learning merupakan pendekatan pembelajaran yang menggunakan data berlabel, di mana setiap data masukan telah memiliki pasangan keluaran yang diketahui. Melalui proses tersebut, model mempelajari hubungan antara masukan dan keluaran sehingga dapat digunakan untuk memperkirakan hasil pada data baru. Pendekatan ini banyak diterapkan pada permasalahan klasifikasi dan regresi karena mampu memberikan tingkat ketepatan yang baik apabila data latih tersedia secara memadai[16].

Berbeda dengan itu, *unsupervised learning* tidak memanfaatkan data berlabel dalam proses pelatihannya. Pendekatan ini bertujuan untuk mengidentifikasi pola atau struktur tertentu yang tersembunyi di dalam data, seperti pengelompokan (*clustering*) maupun reduksi dimensi. Oleh karena itu, metode ini umumnya digunakan pada tahap analisis awal, terutama ketika informasi mengenai data masih terbatas[16].

Sementara itu, *reinforcement learning* merupakan metode yang bekerja dengan mengumpulkan informasi dalam bentuk sinyal penguatan untuk menentukan langkah-

langkah yang mengarah pada hasil yang diinginkan secara optimal. Dalam pendekatan ini, suatu agen melakukan tindakan tertentu dan kemudian belajar dari konsekuensi atau hasil yang diperoleh, sehingga dapat memperbaiki keputusan yang diambil pada tahap berikutnya[16].

Berdasarkan ketiga pendekatan tersebut, penelitian ini menggunakan algoritma yang termasuk dalam kategori *supervised learning*, yaitu *Support Vector Machine* (SVM), karena metode ini memanfaatkan data berlabel untuk melakukan proses klasifikasi sentimen secara efektif.

2.2.2 Hyperparameter Tuning

Hyperparameter merupakan parameter yang krusial dalam hal *machine learning* dimana proses ini akan mengoptimalkan model, dan berbeda dengan parameter yang digunakan selama proses pelatihan dikarenakan *hyperparameter* akan berperan langsung dalam mengontrol kompleksitas, bias, dan kapasitas generalisasi model [23]. Oleh karena itu dengan pemilihan *hyperparameter* yang tepat menjadi faktor penting dalam menghasilkan model yang optimal dan stabil.

Metode tuning yang digunakan pada hyperparameter umumnya adalah *gridsearchCV* dan *randomizedsearchCV,* yang keduanya ada pada *scikit-learn.* Kedua metode tuning tersebut dijelaskan sebagai berikut [23]:

1. *gridsearchCV* bekerja dengan mengevaluasi seluruh kombinasi paramter dalam ruangan pencarian yang telah ditentukan secara eksplisit, yang akan menghasilkan solusi terbaik dalam paramter tersebut, namun akan memakan waktu komputasi yang tinggi.
2. *randomizedsearchCV* akan mengevaluasi kombinasi paramter yang dipilih secara acak dari ruang pencarian, pendekatan ini terbukti lebih efisien dan menghasilkan konfigurasi parameter yang kompetitif bahkan dalam kasus ruang paramter yang luas.

Hyperparmeter memiliki peran yang sangat penting dalam SVM untuk menetukan performa klasifikasi, pada SVM beberapa hyperparameter yang biasanya digunakan misalnya untuk prediksi *error* pada data hasilnya akan sangat dipengaruhi oleh parameter *regularisasi* C dan parameter *gamma* pada kernel tertentu [24]. Paramter C akan mengontrol *trade-off* antar margin maksimum dan kesalahan klasifikasi, parameter *gamma* akan berperan pentin dalam pembentukan keputusan [25].

2.3 Natural Language Processing

Natural Language Processing (NLP) merupakan salah satu cabang dari Artificial Intelligence (AI) yang berfokus pada kemampuan komputer dalam memahami, memproses, dan mengolah bahasa manusia. Dalam pengembangannya, NLP mengintegrasikan konsep dari ilmu linguistik, ilmu komputer, serta kecerdasan buatan agar sistem dapat memahami bahasa alami secara lebih efektif[26].

Ruang lingkup NLP meliputi berbagai tugas, seperti klasifikasi teks, analisis sentimen, hingga pengenalan entitas. Salah satu tugas yang paling umum adalah text classification, yaitu proses pengelompokan teks ke dalam kategori tertentu secara otomatis, yang didukung oleh teknik seperti tokenization serta analisis linguistik untuk memahami struktur dan makna bahasa[27].

Dalam implementasinya, NLP juga memiliki tahapan penting yang dikenal sebagai preprocessing teks, yaitu proses mengubah teks mentah menjadi bentuk yang lebih terstruktur agar dapat diolah oleh sistem. Tahapan ini mencakup case folding, tokenization, stopword removal, serta stemming atau lemmatization, yang berperan penting dalam meningkatkan kualitas hasil analisis[28].

2.4 Analisis Sentimen

Analisis sentimen adalah salah satu alat yang digunakan dalam penambangan teks dan pemrosesan bahasa alami (NLP) untuk mengidentifikasi, menganalisis, dan mengklasifikasikan opini, sikap, atau emosi yang ditemukan dalam sebuah teks. Analisis sentimen sering digunakan untuk menganalisis data yang tidak terstruktur, seperti ulasan pengguna, komentar media sosial, dan opini tegas, karena dapat secara otomatis menggambarkan opini publik dari sejumlah besar data[29].

Secara umum, proses analisis sentimen melibatkan beberapa tahapan utama, yakni pra-pemrosesan teks untuk membersihkan data dari gangguan (*noise*), ekstraksi fitur untuk merepresentasikan teks ke dalam bentuk numerik, dan penerapan algoritma klasifikasi guna menentukan polaritas sentimen dari teks yang dianalisis. Pendekatan ini memungkinkan para peneliti untuk memahami tren opini pengguna terhadap suatu produk atau layanan secara lebih objektif dan sistematis[30].

Dalam proses tersebut, tahap klasifikasi sentimen berperan dalam mengelompokkan opini kedalam kategori tertentu berdasarkan karakteristik emosionalnya.

Proses ini penting karena menentukan bagaimana opini pengguna diinterpretasikan secara sistematis melalui pendekatan pemrosesan bahasa alami [31].

Secara umum, klasifikasi sentimen terbagi kedalam tiga kategori utama yaitu sentimen positif, negatif, dan netral[29]:

1. Sentimen Positif

Sentimen positif merupakan sebuah pendapat yang menggambarkan kelebihan layanan yang disediakan oleh pihak pengembang yang biasanya berisi kalimat dukugan atau pujian terhadap suatu aplikasi[32].

2. Sentimen Negatif

Sentimen negatif adalah pendapat yang berisikan kata atau kalimat yang menunjukkan ketidakpuasan pengguna atau ketidaksetujuan pengguna terhadap suatu hal, kalimat atau kata yang pada sentimen negatif bisa berisi kata penolakan, ejekan bahkan kata-kata kasar dari pengguna[33].

3. Sentimen Netral

Sentimen netral adalah sebuah pendapat atau argumen yang tidak mengarah ke arah positif atau negatif, bisanya berisi beberapa kalimat pujian namun juga memiliki kalimat pengecualian sehingga pendapat ini tidak dapat dimasukkan ke dalam kelas postif maupun negatif[32].

2.5 Algoritma SVM

Support Vector Machine merupakan sebuah metode *supervised learning* yang digunakan untuk menganalisis data serta pengenalan pola yang digunakan untuk melakukan klasifikasi maupun regresi, SVM bekerja dengan menentukan sebuah *hyperplane* terbaik yang mampu memishakan data untuk memaksimalkan jarak pada setiap kelas secara maksimal dengan jarak margin terbesar[34].

Hyperplane adalah sebuah fungsi atau batas keputusan yang dapat digunakan untuk memisahkan data ke dalam kelas-kelas yang berbeda. *Hyperplane* yang optimal ditentukan dengan memaksimalkan jarak antara *hyperplane* tersebut dengan titik terdekat dari masing-masing kelas, sehingga pemisahan kelas menjadi lebih jelas dan akurat [35].

Metode *Support Vector Machine* (SVM) dapat digunakan sebagai pendekatan dalam menentukan parameter terbaik melalui proses perbandingan sejumlah nilai diskrit

yang tergabung dalam himpunan kandidat. Dari proses tersebut akan dipilih parameter yang menghasilkan tingkat akurasi klasifikasi paling optimal.

Prinsip utama pada SVM terletak pada pencarian *hyperplane* atau garis pemisah antar kelas yang paling optimal. Penentuan *hyperplane* tersebut bertujuan untuk meningkatkan performa proses klasifikasi sehingga model mampu bekerja lebih efektif dibandingkan metode klasifikasi *konvensional*.

Tahapan yang dilakukan dalam metode tersebut adalah sebagai berikut[36].

1. Melakukan perhitungan matriks dengan kernel linear.

   $K(x, y) = x.y$ .....(1)

   Keterangan :

   K(x,y) = kernel linear

   x = data

   y = kelas data

2. Inisialisasi parameter yang digunakan, antara lain λ, γ, C dan iterasi maksimum.

3. Setelah itu, α=0 diinisialisasi dan *matriks Hessian* dihitung.

   $D_{ij} = y_i y_j \left(K(x_i . x_j) + \lambda^2\right.$ .....(2)

   Untuk i,j = 1,…,n

   Keterangan :

   x_i = data ke-i

   x_j = data ke-j

   y_i = kelas data ke-i

   y_j = kelas data ke-j

   n = jumlah data

   λ = lamda

   $\left(K(x_i . x_j)\right.$ = fungsi kernel

4. Membuat 3 perhitungan (sampai batas iterasi) seperti berikut

$E_i = \sum_{j=1}^{n} \alpha_i D_{ij}$ .....(3)

Keterangan:

α_i = alfa ke-i

D_{ij} = matriks hessian

E_i = *error rate*

$\delta\alpha_i = min\{max[\gamma(1 - E_i), -\alpha_i], \mathrm{C} - \alpha_i\}$ .....(4)

Keterangan:

α_i = alfa ke-i

γ = konstanta gamma

E_i = *error rate*

$\mathrm{C} =$ *variabel* slack

γ = *learning rate*

$max\{i\}D_{ij}$ = nilai max dari matriks *Hessian*

$\alpha_i = \alpha_i + \delta\alpha_i$ .....(5)

Keterangan:

$\alpha_i = alfa\ ke - i$

$\delta\alpha_i = delta\ alfa\ ke - i$

5. Kemudian melakukan perhitungan nilai bias yang ditunjukkan persamaan di bawah ini

$b = -\frac{1}{2}\left[\sum_{i=1}^{m} \alpha_i y_i K(x_i, x^+) + \sum_{i=1}^{m} \alpha_i y_i K(x_i, x^-)\right]$.....(6)

Keterangan :

$\alpha_i = alfa\ ke - i$

$y_i = kelas\ data\ ke - i$

$K(x_i, x^+) = kernel\ kelas\ positif$

$K(x_i, x^-) = kernel\ kelas\ negatif$

6. Melakukan perhitungan fungsi keputusan

$f(x) = w.x + b$ .....(7)

$f(x) = \sum_{i=1}^{m} \alpha_i y_i K(x_i, x) + b$ .....(8)

Keterangan :

$\sum_{i=1}^{m} \alpha_i y_i = jumlah\ nilai\ bobot\ w$

$K(x_i, x) = nilai\ kernel$

b = nilai bias

7. Penentuan kelas positif maupun negatif dilakukan berdasarkan ketentuan berikut untuk mengklasifikasikan data uji.

Data akan bernilai positif apabila :

$w.x + b = 1$ .....(9)

Data akan bernilai negatif apabila :

$w.x + b = -1$ .....(10)

Beberapa tipe kernel yang dapat digunakan:

1. Kernel Linear

Kernel linear diterapkan pada kondisi ketika pola hubungan antarfitur menunjukkan karakteristik yang linear, sehingga pemisahan antar kelas masih dapat dilakukan secara optimal menggunakan sebuah hyperplane tanpa memerlukan transformasi data ke dimensi yang lebih tinggi. Secara matematis, kernel ini dapat direpresentasikan sebagaimana ditunjukkan pada persamaan (5).

$K(x_i x_j) = x_i^T x_j$ .....(11)

Kernel linear memiliki persamaan

Keterangan;

$x_i^T x_j$ = hasil perkalian *dot product* antara dua vektor fitur xi dan xj.

Semakin besar nilai dari hasil perkalian titik ini, maka semakin tinggi juga tingkat kemiripan antara dua sampel data. Kernel linear pada umumnya digunakan ketika jumlah fitur lebih besar daripada jumlah sampel, serta akan memberikan hasil yang efisien dan mudah diinterpretasikan pada data dengan distribusi sederhana [37].

Berikut Kelebihan dan Kekurangan dari Kernel Linear:

Kelebihan Kernel Linear:

1. Mampu memisahkan kelas dengan baik secara keseluruhan, tercermin dari kemampuan menjaga kualitas pemisahan pada berbagai ambang keputusan.
2. Memberikan performa yang stabil dan seimbang antar kelas, sehingga tidak hanya menekankan akurasi tetapi juga konsistensi hasil klasifikasi.
3. Efektif digunakan pada data dengan hubungan antar fitur yang cenderung linier tanpa memerlukan pemodelan yang kompleks.
4. Memiliki efisiensi komputasi yang tinggi dan waktu pelatihan yang relatif cepat.
5. Hasil klasifikasi lebih mudah diinterpretasikan secara akademik dan praktis.

Kekurangan Kernel Linear:

1. Kurang mampu menangkap pola hubungan data yang bersifat nonlinier dan kompleks.
2. Tidak selalu menghasilkan nilai akurasi tertinggi dibandingkan kernel nonlinier.
3. Sangat bergantung pada asumsi bahwa struktur data dapat dipisahkan secara linier sehingga berpotensi mengalami underfitting.

2. Kernel RBF

Kernel rbf merupakan kernel yang paling populer di antara semua jenis kernel yang ada pada SVM, kernel ini meliputi kernel gaussian yang mengukur kemiripan antar sampel [38].

Kernel rbf dirumuskan sebagai:

$$K(x_i x_j) = exp\left(-\frac{\| x_i - x_j \|^2}{2\sigma^2} + 1\right) \quad (12)$$

keterangan:

σ = varian dan hyperparameter yang digunakan

$\|x_i x_j\|$ = jarak euclidean antara titik xi dan xj

3. Kernel Polynomial

Kernel polynomial didefinisikan sebagai kernel yang non-stasioner. Kernel ini sangat cocok untuk kuumpulan data pelatiahan yang di normalisasi, kernel ini digunakan ketika sampel lebih sedikit dengan parameter yang dapat di atur *slope alpha,* parameter konstanta dan derajat polynomial.

Kernel polynomial dirumuskan sebagai:

$$K(x_i, x_j) = [(\sigma x_i^T \times x_j) + c]^d \quad \text{.....(13)}$$

keterangan;

d = derajat fungsi kernel

c = konstanta bebas yang mengontrol pergerakan kernel

Setiap model SVM pada penelitian selalu dilatih menggunakan parameter dasar yang disebut dengan parameter penalti (parameter C) sehingga akan mengontrol keseimbangan antara memaksimalkan margin dan meminimalkan kesalahan klasifikasi pada data *training* dan akan menurunkan resiko overfitting jika C besar dan menstabilkan jika C kecil [37].

2.6 Confusion Matrix

Confusion matrix adalah sebuah tabel analisis yang menggambarkan jumlah prediksi yang benar atau tepat dan jumlah prediksi yang salah atau keliru untuk setiap kategori [39]. *Confusion matrix* akan menjadi dasar untuk evaluasi kinerja model seperti perhitungan *accuracy, precision, recall,* dan *F1-Score* [40]. Komponen utama dalam *confusion matrix* ada empat, yaitu *True Positif* (TP), *True Negatif* (TN), *False Positif* (FP), *False Negatif* (FN) [41]. Keempat komponen tersebut adalah sebagai berikut[42]:

1. *True Positive* merupakan data aktual "*positive*" dan diprediksi sebagai "*positive*".
2. *True Negative* merupakan data aktual "*negative*" dan diprediksi sebagai "*negative*".
3. *False Positive* merupakan data aktual "*negative*" tetapi diprediksi sebagai "*positive*".
4. *False Negative* merupakan data aktual "*positive*" tetapi di prediksi sebagai "*negative*".

Tabel 2. 1 *Confusion Matrix*

		Klasifikasi actual	
		Positive	Negatif
Klasifikasi Prediksi	Positif	True Positif (TP)	False Positif (FP)

	Negatif	False Negatif (FN)	True Negatif (TN)

Hal yang didapatkan dari evaluasi model menggunakan *Confusion Matriks* adalah sebagai berikut:

1. Accuracy

Accuracy merupakan metrik yang paling umum digunakan dalam hal klasifikasi, metrik ini digunakan untuk mengukur keakuratan kinerja model dalam melakukan klasifikasi dengan jumlah data yang diprediksi benar oleh model [43].

$$\frac{TP+TN}{TP+TN+FP+FN} \quad \text{.....(14)}$$

2. Precision

Precision adalah sebuah metrik yang berfungsi untuk mengukur kemampuan model dalam mengklasifikasikan data postif secara tepat[41].

$$Precision = \frac{TP}{TP+FP} \quad \text{.....(15)}$$

3. Recall

Recall adalah proporsi data postif yang benar dari seluruh data positif sebenarnya. Recall dapat dihitung dengan cara menjumlahkan jumlah prediksi benar dan membaginya dengan jumlah seluruh data positif sebenarnya[43].

$$\frac{TP}{TP+FN} \quad \text{.....(16)}$$

4. F1-Score

F1-*Score* merupakan rata-rata antara presisi dan recall, metrik ini akan berguna saat diperlukan keseimbangan antara kedua metrik tersebut, terutama ketika kedua metrik tersebut tidak seimbang[39].

$$F-1\ Score = 2 \times \frac{Precision \times Recall}{Precision+Recall} \quad \text{.....(17)}$$

2.7 Text Mining

Text mining merupakan sebuah proses untuk menemukan pengetahuan yang tidak dapat diketahui melalui ekstraksi informasi otomatis dari sejumlah besar teks yang tidak terstruktur seperti ulasan produk, dokumen, email, dan media sosial [44]. Berbeda dengan data mining dimana proses tersebut lebih berfokus pada data numerik, *text mining* lebih berfokus pada *preprocessing* teks, seperti tokenisasi, dan ekstraksi fitur yang merupakan langkah awal dalam analisis sentimen [45].

Text mining memiliki hubungan yang erat dengan *Natural Language Processing* (NLP) karena keduanya sama-sama diterapkan untuk mengolah data berupa bahasa alami. NLP digunakan untuk membantu sistem memahami struktur bahasa serta makna yang terdapat dalam kalimat, sedangkan text mining berfokus pada proses pencarian pola dan informasi penting dari sekumpulan data teks. Dalam praktiknya, kedua metode tersebut banyak digunakan pada analisis sentimen untuk mengetahui kecenderungan opini pengguna terhadap produk, layanan, maupun aplikasi digital[46].

Metode *text mining* sangat penting dalam pengolahan ulasan dalam suatu produk atau aplikasi dikarenakan teks pada umunya pasti mengandung noise seperti simbol atau kata yang tidak baku sehingga akan menyebabkan proses klasifikasi tidak berjalan dengan baik, tahapan-tahapan yang disebutkan sebelumnya akan meningkatkan kualitas data serta akan berdampak signifikan terhadap akurasi dari model [47].

2.8 Preprocessing

Preprocessing merupakan tahap paling awal yang dapat dilakukan dalam klasifikasi teks untuk mempersiapkan data teks agar memiliki kualitas yang lebih baik sebelum digunakan untuk proses-proses selanjutnya [48]. Sebagian besar data yang ada di dunia ini tidak terstruktur dan dapat berbentuk gambar, teks, audio ataupun video. *Preprocessing* akan mengubah data mentah tersebut menjadi format yang lebih terstruktur dan siap digunakan untuk pembelajaran mesin agar meningkatkan akurasi dan keandalan model yang akan dibangun[49].

Preprocessing dilakukan dengan memperbaiki, mengubah, atau menghilangkan hal-hal yang tidak dibutuhkan dalam analisis sentimen, *preprocessing* meliputi penghilangan atau penghapusan tanda baca, emotikon serta karakter yang tidak dibutuhkan (*cleaning* data), lower*casing, standarizing text, tokenizing text, removing stopwords,* dan *stemming* [48]. Proses *preprocessing* ini sangat dibutuhkan dan penting untuk memastikan data yang digunakan baik dan konsisten. Terdapat beberapa proses dalam *preprocessing* yang dapat digunakan sesuai dengan kebutuhkannya, beberapa diantaranya adalah sebagai berikut:

1. Cleaning data

   Pembersihan data atau *cleaning* data merupakan proses penghilangan komponen tertentu yang terdapat dalam dataset yakni seperti *Uniform Resource Locator* (URL), *username*, karakter tertentu, kata lebih dari dua kali, emotikon dan juga hastag [50].

2. Lowercasing atau Case Folding

Case folding merupakan sebuah proses yang digunakan dalam klasifikasi sentimen untuk mengubah semua huruf yang ada pada dataset yang digunakan menjadi huruf kecil, agar seluruh dataset memiliki ukuran huruf yang sama dan tidak bercampur antar huruf besar dan kecil [49].

3. Standarizing text

Standarizing text merupakan proses untuk mengubah kata yang tidak formal atau singkatan menjadia kata yang formal dan tidak merupakan singkatan serta bukan merupakan bahasa bahasa gaul [51].

4. Stemming

Stemming merupakan sebuah metode yang digunakan untuk mendapatkan kata dasar dari sebuah kata dengan menghapus imbuhannya sehingga makna dari sebuah kata didapatkan[52].

5. Tokenizing

Tokenizing adalah proses memecah sebuah kalimat menjadi hanya sebuah kata sehingga setiap kata yang ada pada kalimat itu dapat diberikan bobot [53].

6. Removing stopwords (filtering)

Removing stopword atau *filtering* adalah tahapan yang dilakukan dengan membuang kata-kata yang tidak memiliki makna atau tidak terlalu berpengaruh pada klasifikasi [53].

2.9 Data Balancing

Data *imbalance* merupakan sebuah kondisi dimana suatu kelompok kelas memiliki jumlah data yang dominan dan jauh berbeda dibandingkan kelas lainnya, kelompok kelas ini dapat disebut sebagai kelas mayoritas dan yang memiliki data lebih sedikit disebut sebagai kelas minoritas [54].

Teknik yang biasanya digunakan dalam data *balancing* adalah:

1. Class weight

Class weight merupakan teknik yang digunakan dalam *machine learning* (pembelajaran mesin) untuk mengatasi dataset yang tidak seimbang dengan cara memberikan bobot yang lebih tinggi pada sampel kelas minoritas sehingga bobot antara kelas mayoritas dan minoritas sama atau seimbang [55]. Nilai dari parameter *class weight* untuk setiap kelasnya dihitung dari banyak sampel dibagi dengan jumlah kelas yang dikalikan dengan banyaknya jumlah kelasnya [55].

Perhitungan bobot kelas pada metode class weight menggunakan pendekatan balanced yang menghitung bobot berdasarkan distribusi jumlah data pada masing-masing kelas. Menurut dokumentasi scikit-learn, bobot kelas dirumuskan sebagai[56]:

$$w_i = \frac{N_{samples}}{N_{classes} \times N_i} \quad \text{..... (18)}$$

Keterangan:

Wi = bobot kelas ke-i

Nsamples = Jumlah seluruh data training

Nclasses = jumlah kelas

Ni = jumlah data pada kelas ke-i

2. Synthetic Minority Over-sampling Technique (SMOTE)

Smote merupakan sebuah teknik *oversampling* data yang menghasilkan sampel sintetik dari kelas minoritas dengan melakukan *oversampling* pada setiap titik data dengan mempertimbangkan kombinasi linear dari tetangga kelas minoritas yang ada [57].

Pada penelitian ini akan menggunakan *class weight* sebagai teknik dalam data balancing yang secara otomatis menghitung bobot masing-masing kelas berdasarkan frekuensi kemunculannya dalam dataset, sehingga kelas negatif mendapatkan bobot yang lebih tinggi dibandingkan dengan positif, penggunaan *class weght* dikaranakan lebih stabil pada algortima SVM [58].

2.10 Ekstraksi fitur (feature extraction)

Ektraksi fitur merupakan proses yang penting dalam klasifikasi teks untuk mengubah format tekstual yang tidak terstruktur menjadi terstruktur sehingga dapat diproses oleh algoritma machine learning untuk mengklasifikasikan ke kelas yang sudah ditentukan [59]. Tahapan ekstraksi fitur termasuk ke dalam pemrosesan *word embending* yang bertujuan untuk mengubah informasi teks menjadi vector kata untuk setiap kalimat. Dalam klasifikasi teks, terdapat beberapa teknik ekstraksi fitur yang umum digunakan, seperti TF-IDF, Bag of Words (BoW), dan Word2Vec[60].

Pada penelitian ini digunakan metode Term Frequency–Inverse Document Frequency (TF-IDF) sebagai teknik ekstraksi fitur. TF-IDF merupakan metode pembobotan kata yang menentukan tingkat kepentingan suatu kata dalam dokumen dengan

mempertimbangkan frekuensi kemunculan serta distribusinya dalam kumpulan dokumen [61]. Dibandingkan pendekatan sederhana seperti Bag of Words, TF-IDF mampu menghasilkan representasi fitur yang lebih relevan karena menurunkan bobot kata yang kurang informatif dan meningkatkan bobot kata yang lebih unik [62].

TF-IDF terdiri dari dua komponen utama, yaitu Term Frequency (TF) yang mengukur frekuensi kemunculan kata dalam dokumen, dan Inverse Document Frequency (IDF) yang mengukur tingkat kepentingan kata dalam keseluruhan dokumen. Nilai bobot TF-IDF diperoleh dari hasil perkalian antara TF dan IDF, yang dirumuskan sebagai berikut:

$$TFIDF(w,d) = TF(w,d) * IDF(w) \quad (19)$$

Keterangan:

w: suatu kata

d : suatu dokumen

Rumus TF = Pembobotan atau weight setiap kata (*term*) pada suatu dokumen berdasarkan jumlah kemunculannya dalam dokumen.

IDF dirumuskan sebagai berikut:

$$IDF(w) = log\left(\frac{N}{DF(w)}\right) \quad (20)$$

Nilai DF(w) pada merupakan jumlah dokumen yang mengandung kata w.

keterangan:

N : jumlah keseluruhan dokumen [63].

2.11 Cryptocurrency Exchange

Cryptocurrency exchange adalah platform digital yang berfungsi sebagai tempat bagi pengguna untuk menukar mata uang kripto menggunakan mekanisme jual-beli. Secara teknis, bursa memfasilitasi transaksi kripto dengan memungkinkan pengguna untuk melakukan pesanan beli dan jual berdasarkan harga pasar. Bursa ini merupakan bagian krusial dari ekosistem aset kripto karena memberikan kemudahan dan akses pasar bagi para investor serta pedagang yang ingin terlibat dalam perdagangan digital[64].

Platform exchange tidak hanya berfungsi sebagai alat perdagangan, tetapi juga berperan sebagai dompet digital (*wallet*) serta pasar digital bagi para pengguna. Beberapa

penelitian mendefinisikan *exchange* sebagai infrastruktur yang memfasilitasi aktivitas jual dan beli dalam satu sistem perdagangan yang efisien, di mana harga ditentukan oleh interaksi pasar dan permintaan pengguna. Melalui *exchange* tersebut, investor dapat memantau harga secara *real-time*, melihat grafik pergerakan nilai, dan mengambil keputusan investasi berdasarkan data pasar yang tersedia[65].

Exchange memiliki peran krusial dalam perkembangan aset kripto, terutama karena fungsinya yang memungkinkan transaksi diselesaikan secara cepat, aman, dan transparan. Tanpa adanya *exchange*, para investor akan kesulitan menemukan investor lain yang dapat membeli atau menjual aset digital dengan mudah. Dalam konteks penelitian ini, pemahaman mengenai *Cryptocurrency Exchange* berfungsi sebagai fondasi untuk menjelaskan alasan pemilihan objek penelitian, termasuk platform bursa di indonesia seperti triv, indodax, tokocrypto, dan floq.

1. Triv

Triv merupakan salah satu *platform cryptocurrency exchange* yang beroperasi di Indonesia dengan menawarkan cara daring untuk membeli mata uang kripto. Sebagai *platform exchange*, Triv memfasilitasi jual-beli aset kripto melalui sistem pembayaran digital yang memudahkan pengguna dalam melakukan aktivitas investasi kripto. Keberadaan Triv memungkinkan *exchange* kripto untuk meningkatkan akses masyarakat terhadap pasar digital di Indonesia.

2. Indodax

Indodax merupakan salah satu platform *Cryptocurrency Exchange* terbesar yang beroperasi di Indonesia yang memfasilitasi transaksi secara daring. Sebagai sebuah bursa, Indodax berfungsi sebagai pasar tidak terorganisir yang mendorong pembeli dan penjual untuk terlibat dalam transaksi di pasar digital, seperti Bitcoin dan pasar mata uang kripto lainnya, sekaligus membantu memberikan kemudahan penggunaan dan transparansi bagi para pengguna[66].

3. Tokocrypto

Tokocrypto merupakan salah satu platform perdagangan mata uang kripto yang beroperasi di Indonesia. Platform ini telah terdaftar dan diawasi oleh Badan Pengawas Perdagangan Berjangka Komoditi (Bappebti). Tokocrypto memudahkan masyarakat Indonesia untuk melakukan transaksi aset yang legal dan aman dengan menawarkan berbagai layanan perdagangan aset kripto menggunakan mata uang Rupiah. Berdasarkan penelitian,

Tokocrypto tidak hanya populer sebagai aplikasi perdagangan, tetapi juga menjadi objek studi dalam literatur kripto di Indonesia karena ulasan dan tanggapan pengguna mengenai fitur-fiturnya yang terkadang menghambat pengguna untuk menggunakannya secara langsung (*straightforward*)[67].

4. Floq

Floq merupakan salah satu platform exchange asset kripto yang beroperasi di Indonesia dan bertujuan untuk menyediakan layanan pemasaran digital yang mudah digunakan serta selaras dengan kebutuhan pengguna lokal. Platform ini mendorong para investor untuk berinvestasi dalam mata uang kripto dengan fokus pada edukasi dan pertumbuhan komunitas digital di Indonesia [68]. Sebagai salah satu exchange lokal, kehadiran Floq memfasilitasi pengembangan ekosistem aset kripto di Indonesia dan meningkatkan minat masyarakat terhadap investasi digital. Floq menyediakan berbagai fitur dan layanan yang mendukung aktivitas perdagangan kripto, sehingga platform ini relevan untuk diteliti, terutama dalam hal pengalaman pengguna (*user experience*) dan sentimen aplikasi.