BAB II

LANDASAN TEORI

2.1 Penelitian Terdahulu

Penelitian terdahulu merupakan kumpulan studi atau riset yang dilakukan sebelumnya dalam suatu bidang ilmu yang disesuaikan dengan topik penelitian. Penelitian terdahulu digunakan sebagai pedoman dalam mengidentifikasi latar belakang masalah, mencegah duplikasi, dan membangun fondasi penelitian terbaru. Berikut adalah penelitian terdahulu yang digunakan dalam penelitian yang dapat dilihat pada Tabel 2.1 Penelitian Terdahulu

Tabel 2. 1 Penelitian Terdahulu

Judul	Nama Jurnal	Penulis	Metode	Hasil
Prediction of Indonesia Presidential Election Results for the 2019-2024 Period Using Twitter Sentiment Analysis [21]	2019 5th International Conference on New Media Studies Bali, Indonesia	Dinar Ajeng Kristiyanti, Normah, Akhmad Hairul Umam (2019)	Feature Selection: PSO, GA Machine Learning: SVM	PSO-SVM berkinjerja lebih optimal (Akurasi: 82.85% dan 86.20%)
Ant colony optimization for text feature selection in sentiment analysis [23]	*Intelligent Data Analysis*	Ahmad, S. R., Bakar, A. A., & Yaakub, M. R. (2019)	Feature Selection: ACO, Generic Algorithm (GA) Machine Learning: KNN, Information Gain (IG)	Hasil terbaiknya adalah ACO-KNN dengan rata-rata *f-score* 82.7%, mengungguli IG-RSAR dan IG-GA
Optimization of Sentiment Analysis for Indonesian Presidential Election using Naïve Bayes and Particle Swarm Optimization [24]	*Jurnal Online Informatika*	Hayatin, N., Marthasari, G. I., & Nuraini, L. (2020)	Feature Selection: PSO Machine Learning: Naïve Bayes	Terdapat peningkatkan akurasi sebesar 4.12% bila menggunakan PSO sebagai *feature selection* menjadi 90.74%.
Improved Support Vector Machine	Proceedings - 4th International	Windha Mega, P. D., &	Feature Selection:	PSO unggul dibandingkan

Judul	Nama Jurnal	Penulis	Metode	Hasil
(SVM) Performance on Go-Jek Service Review Classification Using Particle Swarm Optimization (PSO) [25]	Conference on Informatics, Multimedia, Cyber and Information System	Haryoko. (2022)	PSO Machine Learning: SVM	GA dalam meningkatkan kinerja model SVM dengan akurasi yang didapat sekitar 87%
Hyper-parameter tuning for support vector machine using an improved cat swarm optimization algorithm [26]	Journal of theNigerian Societyof PhysicalSciences	Silifat Adaramaja Abdulraheem, Salisu Aliyu, Fatimah Binta Abdullahi (2023)	Feature Selection: CSO Machine Learning: SVM	CSO-SVM menghasilkan akurasi yang lebih besar yakni 85,71%
A new machine learning-based method for android malware detection on imbalanced dataset [27]	Multimedia Tools and Applications	Diyana Tehrany Dehkordy, Abbas Rasoolzadegan (2021)	Oversampling: SMOTE Machine Learning: SVM, KNN, dan ID3	SMOTE + KNN menghasilkan akurasi yang signifikan sebesar 98,69%
A decision support system for heart disease prediction based upon machine learning [28]	Journal of Reliable Intelligent Environments	Pooja Rani, Rajneesh Kumar, Nada. M.O.Sid Ahmed, Anurag Jain	Oversampling: SMOTE Machine Learning: Naïve Bayes, GA, SVM, Random Foreset, Adaboost	SMOTE + Naïve Bayes menghasilkan tingkat akurasi yang signifikan sebesar 85,07%
Cryptocurrency Price Prediction Using Forecasting And Sentiment Analysis [29]	The Science and Information Organization	Shaimaa Alghamdi, Sara Alqethami, Tahani Alsubait, Hosam Alhakami	Machine Learning: Naïve Bayes, SVM	Algoritma SVM menunjukan peformasi yang lebih baik dibandingkan Naïve Bayes dengan akurasi sebesar 93,59% untuk dataset BTC dan 95,59% untuk ETH
Twitter Text Mining For Sentiment Analysis On Government's Response To Forest Fires With Vader Lexicon Polarity	6th International Conference on Mathematics, Science, and Education	T Mustaqim*, K Umam and M A Muslim	Machine Learning: KNN, Decision Tree, Naïve Bayes, Random Forest	Algoritma KNN menghasilkan akurasi sebesar 79,45%, lebih besar

Judul	Nama Jurnal	Penulis	Metode	Hasil
Detection And K-Nearest Neighbor Algorithm [30]				daripada algoritma lainnya
A Comparison Of Classification Algorithms For Hate Speech Detection [31]	IOP Conf. Series: Materials Science and Engineering 830 (2020)	T T A Putri, S Sriadhi, R D Sari, R Rahmadani, dan H D Hutahaean	Machine Learning: Naïve Bayes, MLP, SVM, Decision Tree Adaboost	Algoritma Naïve Bayes menghasilkan akurasi sebesar 71,2%

Tabel 2.1 memperlihatkan penelitian terdahulu dalam periode 5 tahun terakhir yang membahas mengenai penerapan *sentiment analysis* dengan menggunakan berbagai pendekatan *machine learning* dan *feature selection*. Penelitian ini dilakukan berdasarkan penelitian terdahulu dengan menggunakan metode dan objek yang berbeda dari penelitian sebelumnya. Beberapa penelitian sebelumnya mengungkapkan bahwa penerapan algoritma *machine learning*, seperti SVM [21], Naïve Bayes [32], dan KNN [30]dapat menghasilkan akurasi yang baik dalam analisis sentimen untuk dataset Twitter. Penerapan *feature selection* dan optimasi parameter dengan menggunakan algoritima berbasis *swarm intelligence*, seperti PSO [33] [25], ACO [23], dan CSO [26] menunjukan hasil model yang lebih baik dibandingkan melakukan pemodelan dengan algoritma *machine learning* tunggal. Selain itu, terdapat penelitian [27] [28] yang menggunakan teknik *oversampling* SMOTE untuk mengatasi permasalahan data tidak seimbang dan menunjukan hasil pemodelan yang lebih baik. Kebaruan dari penelitian ini adalah akan dilakukan optimasi *feature selection* dan parameter berbasis *swarm intelligence*, seperti PSO, ACO, dan CSO untuk model algoritma *machine learning* SVM, Naïve Bayes, dan KNN. Kemudian, akan dibandingkan model setelah dan sebelum dioptimasi berdasarkan tingkat akurasi dan waktu yang berhasil direduksi. Kemudian, diterapkan juga teknik *oversampling* SMOTE untuk mengatasi permasalahan data tidak seimbang.

2.2 Teori tentang Topik Skripsi

2.2.1 Cryptocurrency

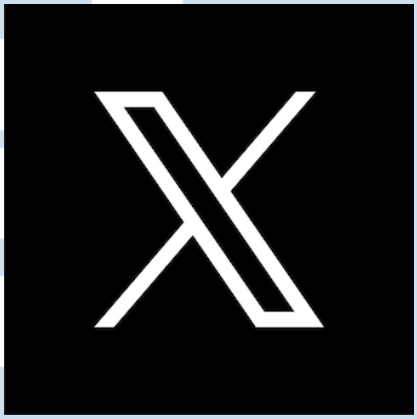
Cryptocurrency merupakan sebuah aset digital yang keamanan dalam transaksinya terjamin dalam kriptografi. Kriptografi adalah sebuah enkripsi

keamanan yang membuat aset digital tidak dapat dipalsukan dan dimanipulasi jumlahnya [34]. *Cryptocurrency* memiliki sistem pendistribusian yang khusus, dimana aset digital yang dimiliki tidak terikat oleh otoritas pusat, seperti perbankan. Melainkan menggunakan jaringan terpusat dari tekonologi *blockhain*, dimana yang memungkinkan para penggunanya untuk melakukan transaksi melalui komputer atau perangkat keras lainnya dimanapun dan kapanpun secara aman. *Blockchain* merupakan sistem keamanan pada *cryptocurrency* yang memungkinkan para penggunanya untuk melakukan transaksi secara online dan aman tanpa adanya campur tangan dari pihak ketiga. Secara garis besar, *block chain* dapat digambarkan sebagai sekelompok blok yang saling terikat dalam sebuah catatan besar. Setiap blok berisikan informasi transaksi yang sudah terverifikasi. Kemudian, jika terdapat transaksi yang sedang berlangsung maka transaksi tersebut akan dicatat sebagai sebuah blok untuk kemudian disebarkan kedalam jaringan yang terdesentralisasi. Setelah semua jaringan telah menyetujui transaksi tersebut maka transaksi menjadi valid dan akan dicatat kedalam catatan besar yang bersifat permanen dan transparan [35].

Cryptocurrency memiliki banyak bentuk yang dibeli dalam bentuk aset digital ataupun sebagai alat pembayaran, seperti bitcoin, etherium, dan litecoin dan lainnya. *Cryptocurrency* menjadi populer kalangan masyarakat dikarenakan harga pasarnya yang tidak dipengaruhi oleh inflasi dunia, sehingga aset digital tersebut sering digunakan oleh masyarakat sebagai sarana investasi alternatif. Namun, dikarenakan tidak adanya kepastian regulasi yang mengatur alur penyebaran *cryptocurrency* membuat banyak negara yang melarang *cryptocurrency* digunakan sebagai alat untuk bertransaksi, seperti China, Indonesia, Hong Kong, dan negara lainnya. Namun terdapat beberapa negara juga yang mendukung *cryptocurrency* sebagai alat untuk bertransaksi seperti Amerika Serikat, Korea Selatan, Jepang, dan Denmark.

2.2.2 X

X yang sebelumnya bernama Twitter merupakan salah satu media soial yang sangat populer digunakan oleh masyarakat dalam berkomunikasi dan bertukar informasi dengan sesama penggunanya [36].



Gambar 2. 1 Logo X

X pertama kali didirikan pada tahun 2006 oleh Jack Dorsey dengan nama “Twitter”. Ide awal media sosial ini terinspirasi dari aplikasi SMS, dimana semua orang dapat saling berkomunikasi dan bertukar informasi dimanapun dan kapanpun. Namun, yang menjadi pembedanya adalah visual aplikasinya dan alur penggunaanya yang dibuat lebih menarik dan *modern*. Twitter pun terus berkembang dan sampai akhirnya berhasil menjadi salah media sosial tersukses di dunia. Hingga akhirnya pada Juli 2023 Twitter berganti kepemilikan oleh Elon Musk, sekaligus mengganti namanya menjadi X dikarenakan kebijakan perusahaan tersebut [37].

2.2.3 Web Scraping

Web Scraping merupakan metode ekstrasi data atau informasi spesifik dalam skala besar. Metode ini dimanfaatkan untuk beragam tujuan, seperti riset, analisis sentimen, dan tujuan lainnya. Metode ini sering digunakan oleh para ahli karena dinilai lebih efisien dibandingkan denegan pengumpulan data secara manual, seperti survei. Web scraping dapat digunakan sebagai alat untuk dapat mengetahui berbagai jenis tren yang terjadi masyarakat [38]. Web scraping memiliki beberapa teknik dalam pengumpulan datanya, seperti parsing HTML, penggunaan XPath, CSS Selector, penggunaan API, dan

scraping dinamis. Selain untuk berbagai kepeluan yang telah disebutkan sebelumnya, web scraping juga digunakan untuk mengumpulkan posting dan komentar pada media sosial yang dituju. Kemudian, datanya dapat dianalisis untuk melakukan analisis sentimen publik, pendapat pelanggan, dan mendapatkan informasi mengenai preferensi konsumen [39].

2.2.4 Sentiment Analysis

Sentiment analysis adalah metode analisis yang menggunakan text analytics, dalam mengumpulkan dan mendapatkan berbagai jenis sumber data baik dari internet ataupun social media. Pada umumnya, analisis sentimen digunakan untuk mendeteksi kalimat-kalimat mengenai suatu pendapat atau komentar terkait produk. Pendapat yang dianalisis biasanya terjadi pada suatu postingan blog, media sosial, dan kolom komentar pada *e-commerce* [40]. Dalam analisis sentimen, pendapat atau komentar pada umumnya dibagi menjadi 2 jenis, yakni komentar positif dan komentar negatif. Positif atau negatif dari komentar dideteksi berdasarkan emosi dari pendapat yang dituliskan [41]. Analisis sentimen sering digunakan dalam berbagi macam jenis metode peneltian, salah satunya adalah metode *machine learning*. Algoritma yang biasa digunakan dalam *machine learning* untuk analisis sentimen adalah naïve bayes, J48, BFTree, dan OneR [42].

2.2.5 Text Preprocessing

Text preprocessing merupakan bagian dari tahapan proses *text mining*, dimana bertujuan untuk melakukan pengolahan data lebih lanjut. *Text preprocessing* berorientasi pada penyeleksian *data text* agar menjadi terstruktur. Penyeleksian data dilakukan untuk membersihkan data dari komponen-komponen yang tidak diperlukan, memperbaiki susunan tatanan bahasa, dan menghilangkan kata-kata yang tidak memiliki makna dalam analisis yang dilakukan. Hasil dari *text preprocessing* diharapkan dapat meningkatan kelengkapan teks data dan akurasi dari analisis data yang dilakukan [43]. Terdapat beberapa tahapan dalam melakukan *text preprocessing*, sebagai berikut:

a) *Remove URLs, Mention, Hashtags, Special Characters and Numbers*

Proses ini berfokus menghilangkan karakter-karakter khusus yang terdapat pada kalimat, seperti URL, hashtags (#), special characters (@!*), dan angka. Tujuan dari dilakukan penghapusan pada karakter tersebut adalah untuk menghilangkan elemen-elemen yang tidak memiliki arti dan kontribusi apapun terhadap nilai informasi yang terkandung pada teks.

b) *Remove Punctuation*

Tahapan ini merupakan proses menghilangkan tanda baca seperti, titik (.), koma (,), tanda seru (!), tanda tanya (?), dan tanda baca lainnya. Penghapusan ini dilakukan untuk mengurangi kompleksitas pada teks dan memudahkan pemrosesan pembersihan data lebih lanjut.

c) *Convert To Lower Case*

Proses ini berfokus pada membuat seluruh teks menjadi huruf kecil secara keseluruhan. Hal tersebut dilakukan agar dapat menyerdehanakan variasi dalam kalimat sehingga dapat meminimalisir kesalahan selama proses klafikasi berlangsung.

d) *Tokenization*

Proses ini merupakan pemecahan kalimat menjadi rententan kata atau unit-unit yang lebih kecil. *Tokenization* merupakan tahapan yang cukup penting dilakukan dalam *sentiment analysis* dikarenakan dapat membantu model *machine learning* untuk dapat melakukan klasifikasi teks dengan baik.

e) *Lemmatization*

Lemmatization merupakan proses mengubah kata-kata menjadi sebenarnya atau dasar, contohnya adalah seperti "*buying*" menjadi "*buy*". Adapun proses serupa yang bernama *stemming. Stemming* juga merupakan proses merubah kata-kata menjadi bentuk dasarnya. Pada penerapannya *stemming* memiliki proses yang lebih sederhana, namun penyerderhanaan kata yang dilakukan adalah

memotong akhiran kata tanpa memperhatikan konteks atau makna lingusitik. *Lemmatization* lebih cocok digunakan dikarenakan memperhitungkan konteks dan makna kata serta menggunakan kamus dan aturan linguistik yang lebih kompleks [44].

f) Remove Words with Length < 3

Proses ini berfokus pada penghapus kata-kata yang jumlah karakternya kurang dari 3 karakter. Hal tersebut dilakukan karena kata-kata yang terlalu pendek cenderung tidak memiliki informasi yang begitu penting. Penghapus kata tersebut juga dapat mengurangi kompleksitas pada saat menjalankan model algoritma[45].

2.2.6 VADER

VADER (*Valance Aware Dictionary And Sentiment Reasoner*) merupakan alat analisis sentiment berbasis lexicon yang digunakan untuk memberikan label pada ulasan agar dapat mengetahui sentimen di dalamnya. Label yang diberikan, pada umunya berupa positif dan negatif. VADER memberikan skor dan tingkat seberapa positif dan negatif sentimen pada suatu ulasan [46].

2.2.7 TF-IDF

TF-IDF (*Term Frequency – Inverse Document Frequency*) merupakan metode sentimen analisis yang digunakan untuk menghitung bobot setiap kata yang digunakan. Bobot setiap kata dihitung berdasarkan frekuensi kemunculan kata tersebut dalam sebuah kalimat. Pada TF (*Term Frequency*) setiap kata yang sering muncul dalam suatu kalimat, maka nilai pembobotan katanya akan semakin tinggi. Namun untuk IDF (*Inverse Document Frequency*), perhitungan dilakukan berdasarkan ditribusi kumpulan dokumen secara luas sehingga semakin sedikit kata yang muncul dalam sebuah kalimat atau dokumen, maka nilai pembobotannya akan semakin tinggi [47].

2.2.8 SMOTE

SMOTE (*Synthetic Minority Oversampling Technique*) merupakan teknik yang digunakan untuk mengatasi masalah data tidak seimbang. Teknik

ini merupakan pengembangan dari metode *oversampling* untuk menyeimbangkan sebuah proporsi data agar menjadi seimbang. SMOTE menggunakan seluruh himpunan data sebagai input, sehingga dapat meningkatkan presentase dari analisis yang dilakukan. Data yang diinputkan untuk menggunakan SMOTE harus berupa kolom numerik dan memiliki label biner. Teknik ini berfokus pada menyeimbangkan data paling rendah atau *minority data* dengan data paling banyak. Secara teknis, teknik ini akan membuat duplikasi sampel data yang berasal dari kelas target dan tetangga terdekatnya. Kemudian, sampel data tersebut akan terus dibentuk atau diduplikasikan untuk membuat proporsi data menjadi lebih seimbang [48].

2.2.9 Feature Selection

Feature selection merupakan proses seleksi subset dari fitur-fitur yang terdapat dalam sebuah dataset yang digunakan pada *machine learning* untuk analisis data. *Feature selection* memiliki fungsi utama untuk menaikan angka akurasi dari sebuah pemodelan yang dibuat, seperti model pada *machine learning*. Peningkatan akurasi dilakukan dengan mengurangi julah fitur yang redundan dan tidak relevan. Terdapat 3 metode yang terdapat pada *feature selection*, yaitu metode *filter*, *wrapper*, dan *embedded*.

1) Metode Filter

   Metode ini memanfaatkan metrik statistik atau skor untuk menilai sejauh mana hubungan setiap fitur dengan target. Contoh dari metode ini mencakup analisis korelasi dan penggunaan uji statistik seperti *chi-square*.

2) MetodeWrapper

   Metode ini menggunakan subset fitur dan mengukur kinerja model dengan subset tersebut. Salah satu contohnya adalah pendekatan rekursif yang secara berulang menghapus atau menambahkan fitur untuk memaksimalkan kinerja model.

3) Metode Hybrid

   Proses seleksi fitur diintegrasikan sebagai bagian dari pelatihan model. Contoh metode ini mencakup penggunaan regularisasi seperti L1 (Lasso) regularization dalam konteks regresi logistik [49].

2.3 Teori tentang Framework / Algoritma yang digunakan

2.3.1 KDD

KDD (*Knowledge Discovery in Database Process*) merupakan sebuah metode yang digunakan dalam proses data mining. KDD sering digunakan untuk menggali informasi yang terdapat dalam data dan menemukan pola tertentu pada data tersebut. Proses pencarian informasi tersebut dapat melibatkan peenggunaan algoritma-algoritma pendukung untuk mengidentifikasi pola pada data [50]. Terdapat beberapa tahapan dalam proses penggalian informasi dengan menggunakan metode KDD, yakni:

1. *Data Cleansing*

   Merupakan tahapan dimana data diolah terlebih dahulu dengan menyeleksi data-data yang dianggap dapat digunakan. Proses pembersihan data dapat berupa menghapus, menyeleksi, dan memilih kolom data.

2. *Data Integration*

   Merupakan proses penggabungan data dari berbagai sumber dan kemudian dijadikan sebuah kesatuan data uang utuh.

3. *Selection*

   Merupakan tahapan proses seleksi data yang dianggap relevan untuk keperluan penelitian.

4. *Data Transformation*

   Merupakan proses data diubah kedalam bentuk model analitis dan membuat model data. Hal tersebut dilakukan agar data dapat dianalisis dan diterapkan pada model algoritma yang digunakan.

5. *Data Mining*

   Merupakan tahapan implementasi teknik yang digunakan untuk menemukan pola-pola yang terdapat dalam data. Pola-pola tersebut nantinya akan dianalisis kembali untuk melihat informasi yang potensial dalam data.

6. *Pattern Evolution*

   Merupakan proses identifikasi sejumlah pola yang ditemukan untuk kemudian dijadikan sebagai pengetahuan dalam penelitian

7. *Knowledge Presentation*

   Merupakan proses penerapan visualisasi dan hasil dari penelitian yang dihasilkan kepada para *user* [51].

2.3.2 Grid Search

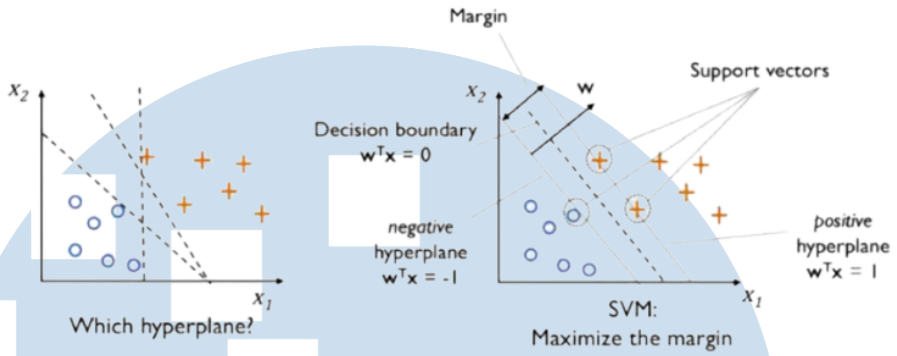
Grid Search merupakan suatu algoritma yang digunakan untuk menemukan parameter yang optimal bagi pada sebuah model [52]. Proses kerja grid search diawali dengan menentukan terlebih dahulu serangkaian nilai yang mungkin digunakan dalam model untuk setiap parameternya. Kemudian model tersebut dilatih dan divalidasi dengan menggunakan hasil dari parameter yang telah dilakukan optimasi. Hasil dari kinerja model dapat dinilai berdasarkan metrik tertentu, seperti akurasi ataupun tingkat kesalahan model dalam implementasinya. Penggunaan *grid search* dapat diimplementasikan pada semua model machine learning yang memiliki nilai parameter, sehingga model tersebut dapat dioptimasi untuk mendapatkan model terbaik [53].

2.3.3 Machine Learning

Machine learning merupakan sebuah teknologi yang dikembangkan agar dapat belajar secara mandiri tanap arahan dari penggunanya. Pembelajaran ini digunakan dalam data mining sehingga model dapat belajar tanpa perlu perintah ulang. *Machine learning* memiliki 3 cabang pembelajaran, yaitu *deep learning, supervised learning, unsupervised learning*. Pada penelitian ini menggunakan supervised learning karena cocok diterapkan pada *sentiment analysis* [54].

2.3.4 SVM

Support Vector Machine merupakan algoritma klasifikasi yang termasuk ke dalam jenis supervised learning yang digunakan untuk mencari hyperpalne yang dapat mencari jarak maksimal dari dua kelas data. *Hyperplane* pada SVM merupakan sebuah cara atau fungsi yang digunakan sebagai pembeda antar kelas dalam dimensi yang lebih tinggi.



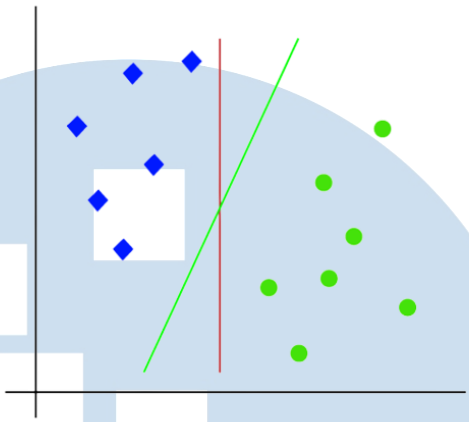
Gambar 2. 2 Proses Support Vector Machine

Gambar 2.1 diatas merupakan representasi dari proses penggunaan algoritma SVM. Cara kerja dari algoritma tersebut dengan meimplementasikan konsep kernel ke dalam dimensi yang tinggi untuk menemukan *hyperplane* dengan jarak pemisah yang maksimal. *Hyperplane* digambarkan diantara dua kelas, dimana jarak *hyperplane* dengan objek-objek data berbeda dengan kelas-kelas yang berada diluar dan dilambangkan dengan simbol lingkaran dan postif. Kemudian, terdapat beberapa objek yang berdekatan dengan *hyperplane* yang sulit untuk diklasifikasikan karena posisisnya yang terlalu berdekatan dengan objek serupa lainnya, objek tersebut disebut sebagai *support vector*. Dalam perhitungan dalam SVM, *support vector* tersebutlah yang menjadi indikator dalam menemukan *hyperplane* paling optimal [55].

Pada implementasi dari algoritma *support vector machine* (SVM), terdapat 4 tipe dari algoritma tersebut, yaitu *linear*, *polinomial*, *radial basis funcition* (RBF), dan *sigmoid*.

1. *Linear* SVM

   Linear merupakan salah tipe dari algoritma SVM yang digunakan pada dataset dengan jenis linear. Data *linear* berarti data yang dapat dikelompokan mejadi dua kelas dan dipisahkan dengan satu garis lurus tunggal [56].



Gambar 2. 3 Linear Support Vector Machine

2. *Polinomial* SVM

   Polinomial juga merupakan salah satu dari tipe kernel algoritma SVM yang digunakan pada dataset yang tidak linear. Dataset tersebut akan dibuat sebuah data latih (*training data*) yang akan dinormalisasi [57].

3. RBF-SVM

   RBF (*Radial Basis Function*) adalah salah satu tipe dari algoritma SVM yang digunakan pada dataset yang tidak dapat dipisahkan secara linear. Tipe RBF memiliki dua parameter yang digunakan dalam implementasinya, yaitu X1 dan X2 untuk menghitung kesamaan atau seberapa dekat kedua parameter tersebut satu sama lain [58].

4. Sigmoid

   Sigmoid merupakan tipe terakhir dari algoritma SVM yang digunakan dalam klasifikasi dalam pengembangan jaringan saraf tiruan [59].

2.3.5 Naïve Bayes

Naïve bayes merupakan salah satu dari algoritma klasifikasi machine learning yang cukup populer untuk digunakan. Teknik klasifikasi ini memanfaatkan probalitas sederhana berdasarkan pada teorema Bayes dengan menggunakan asumsi independensi yang kuat. *Naïve bayes* akan menganggap semua data adalah data yang tidak memiliki ketegantungan atau independen. Faktor dependenya akan dihasilkan melalui analisa dari

fitur data pelatihan (*data training*). Berikut adalah rumus perhitungan *naïve bayes* yang dapat dilihat pada Rumus 2.1

Rumus:

$$P(A|B) = \frac{P\,(B|A).P\,(A)}{P\,(B)} \quad (2.\ 1)$$

Rumus 2. 1 Naive Bayes

Pada penerapannya, algoritma *naïve bayes* terdiri dari 3 tipe, yaitu

1. *Multinomial Naïve Bayes*

   Multinomial merupakan salah satu tipe algoritma naïve bayes yang bertujuan untuk membuat klasifikasi data pada kelas tertentu. Klasifikasi dilakukan berdasarkan banyaknya kata yang muncul dalam sebuah teks atau kalimat yang dianalisis [60].
2. *Gaussian Naïve Bayes*

   Gaussian adalah salah satu jenis dari algoritma klasfikasi *naïve bayes* yang memungkinkan untuk membuat klasifikasi dalam bentuk yang sederhana. Pendistribusian data akan mengikuti distribusi dari normal atau atau Gaussian [61].
3. *Bernouli Naïve Bayes*

   Bernouli merupakan tipe dari algoritma *naïve bayes* yang hampir serupa dengan implementasi dari algoritma multinomial *Naive Bayes*. Namun, yang menjadi pembeda adalah klasifikasi yang dilakukan dengan cara membuat fitur yang diasumsikan menjadi biner, yaitu 0 dan 1 atau "ya" dan "tidak". Naïve bayes dengan tipe bernouli ini baik digunakan untuk memprediksi sebuah kata yang akan muncul dalam sebuah teks [62].

2.3.6 KNN (*K-Nearest Neighbour*)

KNN merupakan algoritma klasifikasi yang digunakan untuk mengelompokan data dengan menentukan nilai K dari data tetangga terdekat (*neighbor*) untuk dapat menentukan kelas baru dari data yang

digunakan. Algoritma ini melakukan klasifikasi dengan cara proyeksi data pengkajian dengan kapasitas dimensi yang banyak [63]. KNN tidak memerlukan besaran parameter yang pasti dalam melakukan klasifikasi. Tidak ada besaran pasti jumlah parameter yang tetap dalam sebuah model pada klasifikasi KNN. Algoritma KNN termasuk dalam kategori *lazy learning*, dimana proses klasifikasi dilakukan hanya menggunakan titik data training dalam pembuatan model. Asumsi pada algoritma klasifikasi ini adalah bahwa data-data serupa akan cenderung saling berdekatan satu sama lain atau bertetanggaan [64]. Terdapat beberapa cara dalam melakukan pencari data terdekat, seperti *euclidean distance, hamming distance, manhattan distance,* dan *minkowski distance.*

1. *Euclidean Distance*

   Merupakan teknik pencarian data terdekat dengan menentukan jarak antara dua titik dalan ruang dua dimensi. Perhitungan rumus pencarian jaraknya, terinspirasi dari perhitungan *phytagoras*. Berikut adalah rumus perhitungan *euclidean distance* yang dapat dilihat pada Rumus 2.2

$$d(x,y) = \sqrt{\sum_{i=1}^{m} (xi - yi)^2} \quad (2.\ 2)$$

Rumus 2. 2 Euclidean Distance

2. *Hamming Distance*

   Merupakan salah satu teknik dalam algoritma KNN yang digunakan untuk mencari jarak dari kedua titik yang sudah dideklarasikan. Pencarian jarak dilakukan dalam perhitungan jarak antara 2 titik dengan panjang biner yang membentuk *block code biner* dari kedua titik tersebut. Berikut adalah rumus perhitungan *hamming distance* yang dapat dilihat pada Rumus 2.3.

   Rumus perhitungan:

$$dH = \sum_{i=1}^{k} |x_i - yi| \quad (2.\ 3)$$

Rumus 2. 3 Hamming Distance

3. *Manhattan Distance*

Merupakan salah satu dari formula teknik klasifikasi KNN yang mempunyai kemiripan dengan teknik *euclidean distance*. Perbedaannya terletak pada perhitungan pencarian jarak antara kedua titik variabelnya, yaitu dengan menambahkan semua pengurangan dari jarak *xi* dan *yi*. Berikut adalah rumus perhitungan *manhattan distance* yang dapat dilihat pada Rumus 2.4

$$d(xy) = \sum_{i=1}^{m} |x_i - yi| \quad (2.\ 4)$$

Rumus 2. 4 Manhattan Distance

4. *Minkowski Distance*

Minkowski Distance adalah salah satu teknik perhitungan jarak pada algoritma klasifikais KNN. Perumusan formulanya terinspirasi dari perhitungan aljabar dengan dideklarasikan vektor dimensi *n*. Tujuan dari perhitungan ini adalah untuk mencari jarak maksimum dari variabel *xi* dan *yi*. Berikut adalah rumus perhitungan *minkowski distance* yang dapat dilihat pada Rumus 2.5 [65]:

$$d(x, y) = \left(\sum_{i=1}^{m} |xi - y_i|^r \right)^{\frac{1}{r}} \quad (2.\ 5)$$

Rumus 2. 5 Minkowski Distance

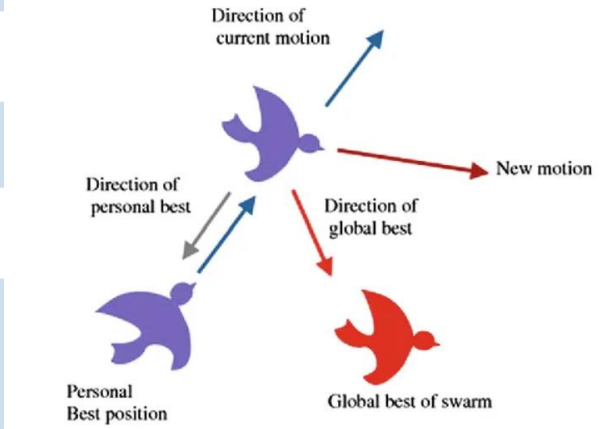
2.3.7 Swarm Intelligence

Swarm intelligence adalah saah bentuk dari feature selection yang memanfaatkan penggunaan *artificial intelligence* dengan mengadopsi perilaku sosial dari hewan yang hidup dalam kelompok. Swarm intelligence umumnya digunakan untuk mendukung optimasi dari sebuah bentuk pemodelan. Salah satu nya optimasi yang digunakan pada machine learning untuk *sentiment*

analysis [66]. Pada penelitian ini algoritma swarm intelligence yang digunakan adalah PSO, ACO, *dan CSO*.

2.3.8 *PSO (*Particle Swarm Intelligence)

PSO (*Particle Swarm Optimization*) adalah salah satu bentuk dari algoritm *swarm intelligence* yang digunakan sebagai optimasi sebuah pemodelan yang berbasis populasi dengan menggunakan populasi dari partikel itu sendiri.

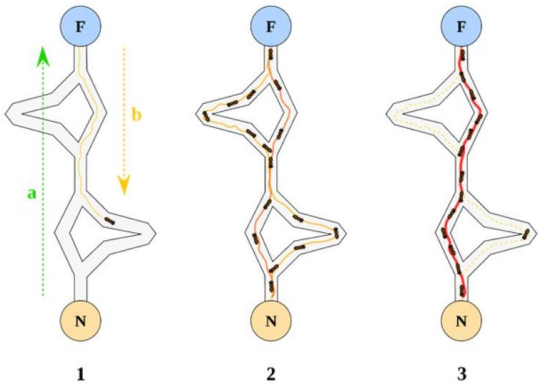


Gambar 2. 4 Particle Swarm Optimization

Pada Gambar 2.2 memvisualisasikan inspirasi dari terbentuk dan cara kerja dari algoritma *feature selection* PSO. PSO terinspirasi dari sekelompok kawanan burung yang dimana memiliki posisi yang berbeda-beda. Namun, akhirnya akan menemukan tujuan atau titik yang sama setelah kesepakatan telah terjadi [67].

2.3.9 ACO (*Ant Colony Optimization*)

ACO adalah salah satu jenis algoritma swarm intelligence yang cara pemecahan masalahnya terinspirasi dari perilaku sekelompok serangga atau semut. ACO pada umumnya cocok digunakan dalam menyelesaikan permasalahannya yang memiliki banyak variabel. Hal tersebut sangat cocok diterapkan dalam menganalisis sentimen pada teks yang memiliki berbagai macam variabel didalamnya.

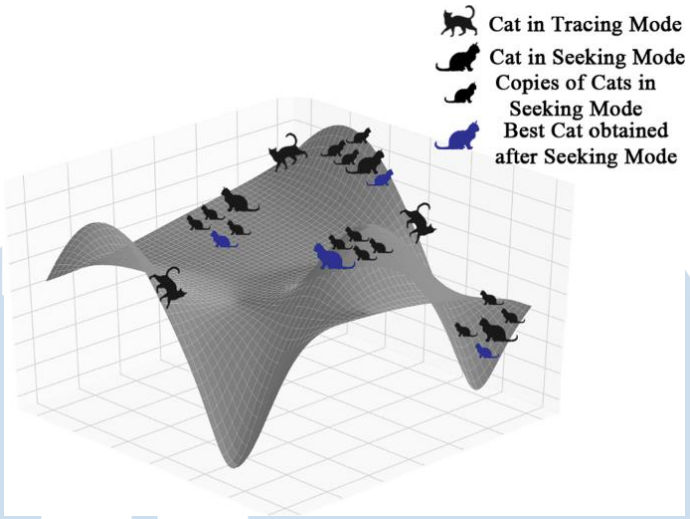


Gambar 2. 5 Ant Colony Optimization

Pada Gambar 2.3 memvisualisasikan cara kerja dari algoritma ACO. Algoritma *feature selection* tesebut menggunakan perilaku dari semut untuk melakukan optimasi. Semut memiliki kecenderungan untuk menemukan jarak terpendek untuk titik makanan berada dari sarang mereka. Dari ketiga jalur tersebut, jalur terpendek akan memberikan sinyal yang kuat pada semut yang telah menjelajah berbagai cabang dari jalan tersebut. Semut lainnya hanya tinggal mengikuti jalur terpendek yang telah ditemukan tersebut [23].

2.3.10 CSO (*Cat Swarm Optimization*)

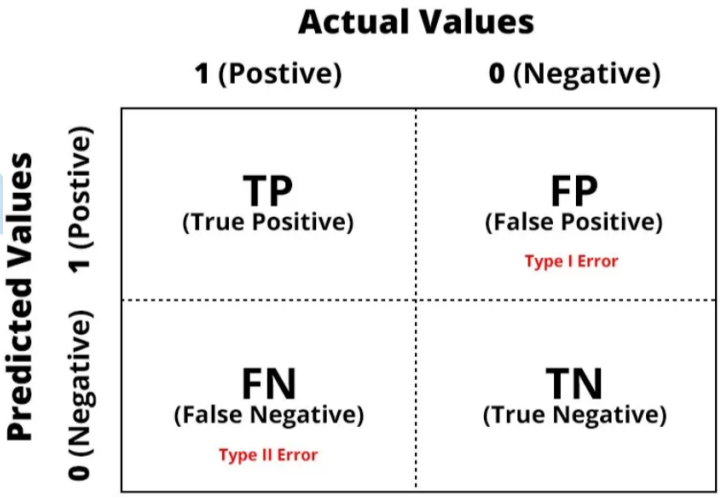
CSO adalah algoritma yang dibangun berdasarkan perilaku sekelompok kucing. Kucing yang dimaksud disini adalah keluarga kucing, seperti harimau, macan, dan kucing. Kucing merupakan hewan yang menghabiskan sebagian besar kehidupannya untuk beristirahat, namun sisi menarik dari kucing ialah mereka selalu terjaga saat sedang beristirahat. Hal tersebut dinamakan metode *seeking mode*. Seeking mode memiliki 4 faktor lainnya, yaitu *seeking memory pool* (SMP), *seeking range of the selected dimension* (SRD), *counts of dimension to change* (CDC) dan *self position consideration* (SPC) Kemudian, terdapat satu metode lagi yang dinamakan *seeking mode*, metode memiliki kemiripan pada perumusan yang digunakan dalam PSO [68]. Berikut adalah gambaran dari penjelasan diatas.



Gambar 2. 6 Cat Swarm Optimization

2.3.11 Evaluation Metrics

Evaluation metrics merupakan sebuah bentuk evaluasi nilai yang dapat digunakan untuk merepresentasikan peforma model yang dihasilkan. *Metrics* yang dihasilkan dapat dijadikan sebagai bahan penilaian untuk kelayakan model yang telah dibuat. *Evaluation metrics* yang digunakan pada penelitian ini merupakan *metrics* pada klasifikasi yang menggunakan *confusion matrix* dan menunjukan 4 komponen utama, yaitu *True Positive*, *False Positive*, *False Negative*, dan *True Negative*. Berikut adalah *confusion matrix* yang ditunjukan pada Gambar 2.7 *Confusion Matrix.*



Gambar 2. 7 *Confusion Matrix*

Sumber:

Berdasarkan Gambar 2.1 *Confusion Matrix*, menunjukan terdapat beberapa komponen yang terdapat dalam *matrix* tersebut. *True Positive* memiliki arti bahwa data yang benar diklasifikasikan sebagai data positif, dan begitupun dengan *True Negative* yang digunakan untuk nilai negatif yang memang diklasifikasikan sebagai nilai negatif. Kemudian, false positive merupakan tingkat salah klafikasi atas nilai positif, dimana semakin rendah nilainya atau mendekati nol, maka dianggap baik. Berdasarkan hasil dari perhitungan pada *confusion matrix*, dapat memberikan suatu turunan dari evaluation metrics, seperti *accuracy, precision, recall*, dan *f1-score*.

2.3.12 Accuracy

Accuracy adalah rasio antara *instance* yang diklasifikasikan dengan benar terhadap jumlah total *instance*. Ini memberikan gambaran keseluruhan tentang kinerja model. Berikut adalah rumus perhitungan akurasi yang terlihat pada Rumus 2.6.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (2.\ 6)$$

Rumus 2. 6 Rumus *Accuracy*

2.3.13 Precision

Precision adalah rasio antara jumlah instance positif yang benar (*true positive*) dengan jumlah instance yang diprediksi positif. Berikut adalah rumus perhitungan *precision* yang dapat dilihat pada Rumus 2.7.

$$Precision = \frac{TP}{TP + FP} \quad (2.\ 7)$$

Rumus 2. 7 Rumus *Precision*

2.3.14 Recall

Recall adalah rasio antara jumlah instance positif yang benar (true positive) dengan jumlah instance yang seharusnya positif dalam dataset. Berikut adalah rumus perhitungan *recall* yang dapat dilihat pada Rumus 2.8.

$$Recall = \frac{TP}{TP + FN} \quad (2.\ 8)$$

Rumus 2. 8 Rumus *Recall*

2.3.15 F1-score

F1-score adalah rata-rata harmonik dari *precision* dan *recall*. Ini memberikan indikasi yang seimbang antara kedua metrik tersebut. Berikut adalah rumus perhitungan *F1-score* yang dapat dilihat pada Rumus 2.9 [69].

$$F1 = \frac{TP}{TP + \frac{1}{2}(FP + FN)} \quad (2.\ 9)$$

Rumus 2. 9 Rumus *F1-Score*

2.4 Teori tentang *Tools / Software* yang digunakan

2.4.1 Python

Pyhton merupakan bahasa pemograman yang diciptakan oleh Guido van Rossum untuk memudahkan dalam pembuatan aplikasi, analisis data, dan keperluan pemograman lainnya. *Python* termasuk dalam ketegori bahasa pemograman tingkat tinggi, namun disatu sisi mudah untuk dipelajari. *Python* pertama kali diciptakan pada era 1990-an dan dalam waktu yang singkat sudah mulai banyak digunakan oleh perusahaan industri dalam mengembangkan aplikasi yang diciptakannya. Alasan utama *Python* menjadi bahasa pemgraman yang paling banyak diminati adalah karena *syntax*-nya yang mudah untuk dipahami dan fleksibel. *Python* dapat digunakan pada *multi-platform* dan tentunya gratis untuk publik [70].

2.4.2 Google Collab

Google Collaboratory atau biasa yang dikenal denagn istilah google colab merupakan sebuah platform *online* yang digunakan untuk menjalankan berbagai bentuk kode pemograman komputer dengan bahasa pemograman Python. Google Colab dikembangkan oleh Google untuk dapat digunakan oleh para developer, analyst, dan peneliti yang bekerja pada bidang *machine learning* atau *data science* tanpa dipungut biaya sepeser pun. Google Colab mempuyai kesamaan fitur dengan Google Document, yakni dapat digunakan untuk berkolaborasi secara real-time dan data akan disimpan pada Google

Drive. Google Colab juga menyediakan akses gratis untuk penggunaan GPU dan TPU. Google Colab dapat langsung digunakan utnuk menjalankan kode pemograman Python dan dapat mengakses *library machine learning* secara *online* dan secara bawaan telah dilengkapi dengan sistem TensorFlow dan PyTorch [71].

Gambar 2. 8 Logo Google Colab

2.4.3 Jupyter Lab

Jupyter lab merupakan sebuah aplikasi web terkemuka yang digunakan untuk membuat dan membangun sebuah lembar kerja yang berisikan kode, hasil perhitungan, visualisasi, dan teks. Jupyter lab menawarkan berbagai macam fitur yang berguna bagi *user*, seperti server Jupyter, debugger visual, dan dukungan untuk berbagai macam bahasa pemograman. Aplikasi web ini juga dilengkapi dengan kernel yang membuat *user* dapat melakukan *restart*, *reconnect*, dan *stop* pada lembar kerja yang sedang menjalankan sebuah kode pemograman. Terdapat beberapa fitur yang disediakan oleh Jupyter Lab untuk mendukung *user* dalam bekerja, seperti:

a) Konsol kode, fitur ini menyediakan sebuah *scrathpads* untuk mengeksekusi kode secara interaktif
b) Lembar kerja yang mendukung berbagai macam bahasa pemograman, seperti Python, LaTex, R, Julia, dan sebagainya.
c) Memfasilitasi berbagai bentuk visual dokumen yang dilengkapi dengan fitur pengeditan dokumen kerja dan dapat dilakukan secara *real-time*