# BAB II
# KAJIAN LITERATUR

## 2.1 Kecerdasan Buatan (*Artificial Intelligence*)

Kecerdasan buatan merupakan bidang ilmu komputer yang mempelajari bagaimana sistem dapat meniru kemampuan kognitif manusia seperti penalaran, pembelajaran, dan pengambilan keputusan [13]. Pada era modern, kecerdasan buatan berkembang pesat karena penggunaan algoritma pembelajaran mendalam [14]. Perkembangan ini didukung oleh meningkatnya ketersediaan data digital serta kemampuan komputasi yang semakin tinggi, sehingga kecerdasan buatan dapat diterapkan sekcara luas dalam berbagai sistem digital [1].

Pertumbuhan kecerdasan buatan dalam beberapa tahun terakhir disertai dengan meningkatnya jumlah diskusi publik mengenai dampaknya terhadap pekerjaan, privasi, dan keamanan digital [13]. Opini publik ini tersebar luas melalui media sosial, platform video, dan sistem komunikasi daring lainnya, sehingga menghasilkan data opini dalam jumlah besar yang merepresentasikan persepsi masyarakat terhadap perkembangan teknologi AI [15].

Besarnya volume data opini di ruang digital menjadikan analisis sentimen sebagai pendekatan yang relevan untuk memahami kecenderungan persepsi publik terhadap perkembangan kecerdasan buatan secara sistematis [15], [16]. Oleh karena itu, pemahaman konseptual mengenai kecerdasan buatan menjadi landasan penting sebelum membahas pendekatan, jenis, dan dampaknya secara lebih rinci [14].

### 2.1.1 Definisi dan Konsep Dasar *Artificial Intelligence*

*Artificial Intelligence* (AI) secara umum didefinisikan sebagai cabang ilmu komputer yang berfokus pada pengembangan sistem yang mampu meniru kemampuan kognitif manusia, seperti penalaran, pembelajaran, dan pengambilan keputusan [13]. Definisi ini menempatkan AI bukan sekadar sebagai alat komputasi, tetapi sebagai disiplin ilmiah yang berupaya memodelkan proses berpikir manusia ke dalam sistem komputasi yang terstruktur [13]. Dalam konteks ini, AI dirancang untuk memproses informasi, mengevaluasi kondisi, serta menghasilkan keputusan berdasarkan pengetahuan atau data yang tersedia [13].

Secara konseptual, kecerdasan buatan dibedakan dari sistem komputasi konvensional karena kemampuannya untuk belajar dan beradaptasi. Sistem komputasi tradisional bekerja berdasarkan aturan tetap (*fixed rules*) yang dirancang secara eksplisit oleh manusia, sedangkan sistem AI modern memanfaatkan data sebagai sumber utama pembelajaran untuk

meningkatkan kinerjanya secara bertahap [1], [14]. Pendekatan ini memungkinkan sistem AI mengenali pola, menarik inferensi, dan menyesuaikan perilaku berdasarkan pengalaman sebelumnya, sehingga lebih fleksibel dalam menghadapi permasalahan yang kompleks dan dinamis [14].

Dalam pengembangan AI modern, data memegang peranan sentral sebagai fondasi pembelajaran sistem [1]. Kualitas, kuantitas, dan representasi data sangat mempengaruhi kemampuan sistem AI dalam menghasilkan keputusan yang akurat dan relevan [1], [14]. Oleh karena itu, kecerdasan buatan sering dipahami sebagai paradigma pengambilan keputusan berbasis data (*data-driven decision making*), di mana model dikembangkan melalui proses pembelajaran dari data historis untuk memprediksi atau mengklasifikasikan kondisi baru secara otomatis [13], [14].

### 2.1.2 Perkembangan *Artificial Intelligence*

Perkembangan kecerdasan buatan secara historis dapat dipahami melalui perubahan pendekatan dalam membangun sistem cerdas. Pada tahap awal, sistem AI banyak dikembangkan menggunakan pendekatan berbasis aturan (*rule-based systems*), di mana pengetahuan dan logika dirancang secara eksplisit oleh manusia [13]. Pendekatan ini efektif untuk permasalahan yang terstruktur, namun memiliki keterbatasan dalam menangani data yang kompleks, ambigu, dan berskala besar [13].

Seiring meningkatnya ketersediaan data digital, pendekatan berbasis aturan mulai bergeser menuju pendekatan *machine learning* [14]. Pada tahap ini, sistem AI tidak lagi sepenuhnya bergantung pada aturan yang telah ditentukan, melainkan belajar langsung dari data untuk membangun model prediktif atau klasifikasi [14]. Pendekatan *machine learning* memungkinkan sistem mengenali pola dari data historis dan melakukan generalisasi terhadap data baru, sehingga lebih adaptif dibandingkan sistem konvensional [1], [8].

Perkembangan selanjutnya ditandai dengan munculnya *deep learning*, yang memanfaatkan jaringan saraf tiruan berlapis banyak untuk mengekstraksi fitur secara otomatis dari data berskala besar [14]. *Deep learning* terbukti efektif dalam menangani data tidak terstruktur, seperti teks, gambar, dan video, serta menunjukkan peningkatan kinerja yang signifikan pada berbagai tugas pemrosesan bahasa alami dan analisis sentimen [12], [13]. Selain kemajuan algoritma, percepatan perkembangan AI juga didukung oleh kemajuan perangkat keras komputasi dan ketersediaan data dalam jumlah besar, yang menjadikan kecerdasan buatan sebagai teknologi inti dalam sistem digital Analisis sentimen [3].

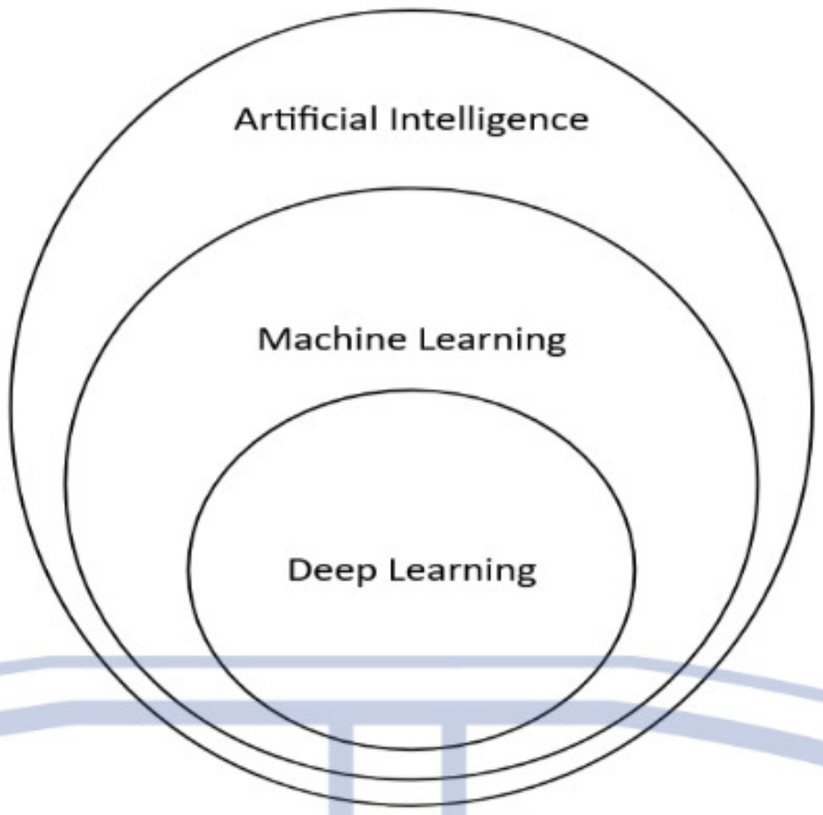
### 2.1.3 Jenis-Jenis *Artificial Intelligence*

*Artificial Intelligence* dapat diklasifikasikan berdasarkan tingkat kecerdasan dan cakupan kemampuannya [14]. Salah satu klasifikasi konseptual yang umum digunakan membedakan AI menjadi *Narrow Artificial Intelligence* (*Narrow* AI) *dan General Artificial Intelligence* (*General* AI) [13], [14]. *Narrow* AI merujuk pada sistem AI yang dirancang untuk menyelesaikan tugas tertentu dalam domain yang spesifik, seperti pengenalan suara, sistem rekomendasi, atau analisis sentimen [13]. Sebagian besar penerapan AI saat ini termasuk dalam kategori *Narrow* AI karena sistem tersebut bekerja secara optimal hanya pada fungsi yang telah ditentukan [14].

Sebaliknya, *General Artificial Intelligence* merupakan konsep AI yang memiliki kemampuan kognitif setara manusia dan mampu menyelesaikan berbagai jenis tugas secara fleksibel tanpa bergantung pada domain tertentu [13]. Hingga saat ini, *General* AI masih bersifat teoritis dan menjadi fokus penelitian jangka panjang karena kompleksitas teknis serta tantangan etis yang menyertainya [13], [14]. Oleh karena itu, pengembangan dan pemanfaatan AI saat ini lebih banyak diarahkan pada *Narrow* AI yang bersifat aplikatif dan relevan dengan kebutuhan praktis di berbagai sektor digital [14].

Selain berdasarkan tingkat kecerdasan, *Artificial Intelligence* juga dapat dibedakan berdasarkan pendekatan pengembangannya, yaitu pendekatan berbasis aturan (*rule-based*) dan pendekatan berbasis pembelajaran (*learning-based*) [13]. Pendekatan berbasis aturan mengandalkan seperangkat logika dan aturan yang dirancang secara eksplisit oleh manusia, sedangkan pendekatan berbasis pembelajaran memungkinkan sistem untuk belajar dari data dan menyesuaikan perilakunya berdasarkan pola yang ditemukan selama proses pelatihan [13], [14].

Dalam pengembangan AI modern, pendekatan berbasis pembelajaran menjadi dominan melalui pemanfaatan *Machine Learning* dan *Deep Learning* [14]. *Machine Learning* merupakan bagian dari *Artificial Intelligence* yang memungkinkan sistem belajar dari data tanpa diprogram secara eksplisit, sedangkan *Deep Learning* merupakan sub-bidang dari *Machine Learning* yang menggunakan jaringan saraf tiruan berlapis banyak untuk menangani permasalahan dengan tingkat kompleksitas yang lebih tinggi, khususnya pada data tidak terstruktur seperti teks [1], [14].




Gambar 2.1 Hierarki Kecerdasan Buatan, *Machine Learning, dan Deep Learning* [14]

Hierarki tersebut menunjukkan bahwa *Deep Learning* berada di dalam *Machine Learning*, dan *Machine Learning* merupakan bagian dari *Artificial Intelligence* secara keseluruhan [14]. Pemahaman mengenai klasifikasi dan hierarki AI ini penting sebagai dasar teoritis untuk memahami pendekatan yang digunakan dalam penelitian, khususnya dalam konteks analisis sentimen berbasis data teks [3].

### 2.1.4 Dampak *Artificial Intelligence* dalam Kehidupan Digital

Perkembangan kecerdasan buatan membawa dampak signifikan dalam kehidupan digital, baik dari sisi teknologi maupun sosial [13]. Penerapan sistem AI dalam berbagai layanan digital, seperti media sosial, platform berbagi video, dan sistem rekomendasi, memengaruhi cara informasi diproduksi, disebarkan, dan dikonsumsi oleh masyarakat [13], [15]. Dampak tersebut mendorong munculnya diskusi publik yang luas mengenai implikasi kecerdasan buatan terhadap pekerjaan, privasi data, serta keamanan dan transparansi sistem digital [3], [15].

Diskursus publik mengenai kecerdasan buatan berkembang secara intensif di ruang digital, khususnya melalui media sosial dan platform komunikasi daring [15]. Masyarakat secara aktif menyampaikan pandangan, kekhawatiran, serta harapan mereka terhadap penggunaan teknologi AI dalam kehidupan sehari-hari [15]. Opini-opini tersebut membentuk persepsi kolektif yang berperan penting dalam menentukan tingkat penerimaan masyarakat terhadap penerapan teknologi kecerdasan buatan [15].

Besarnya volume opini publik yang dihasilkan dari interaksi digital menciptakan tantangan dalam proses analisis secara manual [12]. Oleh karena itu, pendekatan analisis

berbasis komputasi diperlukan untuk mengekstraksi informasi dan pola persepsi masyarakat secara sistematis. Analisis sentimen menjadi salah satu pendekatan yang umum digunakan untuk mengelompokkan opini publik ke dalam kategori positif, negatif, atau netral secara otomatis, sehingga memungkinkan pemahaman yang lebih terstruktur terhadap sikap masyarakat terhadap kecerdasan buatan [14].

Pemahaman terhadap dampak dan persepsi publik mengenai kecerdasan buatan memiliki implikasi penting bagi pengembang teknologi, peneliti, serta pembuat kebijakan [15]. Informasi tersebut dapat digunakan sebagai dasar dalam merancang sistem AI yang lebih bertanggung jawab dan sesuai dengan kebutuhan masyarakat, serta sebagai bahan evaluasi terhadap kebijakan dan regulasi terkait teknologi digital [15]. Dalam konteks penelitian ini, platform YouTube menyediakan sumber data opini publik yang relevan melalui komentar pengguna, sehingga dapat dimanfaatkan untuk mengkaji persepsi masyarakat terhadap perkembangan kecerdasan buatan secara empiris [15] [17].

## 2.2 Analisis Sentimen

Analisis sentimen adalah proses mengidentifikasi dan mengklasifikasikan opini dalam teks ke dalam kategori polaritas seperti positif, negatif, atau netral [15]. Penelitian-penelitian tinjauan terbaru menegaskan bahwa data opini dari media sosial (misalnya Twitter) sangat bernilai untuk memahami persepsi publik, namun analisis sentimen pada data tersebut menghadapi tantangan khas seperti ambiguitas teks, sarkasme, noise, serta ketergantungan konteks dan domain, sehingga pemilihan pendekatan analisis menjadi faktor krusial untuk menjaga interpretasi tetap akurat [12]. Analisis sentimen pada media sosial tidak hanya berfokus pada pengukuran polaritas opini, tetapi juga berperan sebagai instrumen penting untuk memahami dinamika sosial, perilaku kolektif, serta respons publik terhadap isu-isu tertentu [12]. Tinjauan sistematis terbaru menunjukkan bahwa meskipun data media sosial menyediakan volume opini yang besar dan kaya, analisis sentimen pada domain ini menghadapi tantangan fundamental seperti ambiguitas linguistik, sarkasme, ironi, serta ketergantungan konteks dan domain, yang dapat menurunkan akurasi interpretasi apabila tidak ditangani secara tepat [12].

Metode analisis sentimen secara umum terbagi menjadi pendekatan berbasis aturan, pendekatan berbasis pembelajaran mesin, dan pendekatan berbasis pembelajaran mendalam [1]. Pendekatan berbasis aturan menggunakan kamus sentimen, sedangkan pendekatan *machine learning* menggunakan data berlabel untuk melatih model seperti *Support Vector Machine* (SVM) [1]. Pendekatan *deep learning* memanfaatkan jaringan saraf seperti

*Convolutional Neural Network* (CNN) atau *Long Short Term Memory* (LSTM) yang mampu mempelajari representasi teks secara otomatis [14]. Literatur ulasan juga menunjukkan spektrum metode analisis sentimen yang luas, mulai dari pendekatan leksikon hingga *machine learning*, *deep learning*, dan pendekatan *hybrid* [4]. Survei komprehensif menyoroti bahwa tantangan seperti sarkasme/ironi, data multibahasa, serta isu etika ikut memengaruhi efektivitas model, sehingga banyak penelitian bergerak ke arah pendekatan yang lebih adaptif dan kombinatif [18]. Perkembangan riset modern juga banyak meninjau kelebihan dan keterbatasan model ML, DL, hingga model pra-latih/LLM, serta menekankan pentingnya memahami algoritma, dataset, dan metrik evaluasi secara menyeluruh [19]. Pergeseran signifikan dari pendekatan berbasis leksikon menuju pendekatan berbasis pembelajaran mesin dan pembelajaran mendalam seiring meningkatnya kompleksitas data dan kebutuhan akan representasi semantik yang lebih kaya [18]. Namun demikian, tidak ada satu pendekatan pun yang secara universal unggul untuk semua konteks, karena kinerja model analisis sentimen sangat dipengaruhi oleh karakteristik dataset, bahasa, domain, serta strategi prapemrosesan yang digunakan [18], [19]. Oleh karena itu, pemahaman menyeluruh terhadap kelebihan dan keterbatasan masing-masing pendekatan menjadi aspek penting dalam penelitian analisis sentimen.

Dalam konteks data yang tidak terstruktur seperti komentar YouTube, analisis sentimen menghadapi berbagai tantangan seperti penggunaan bahasa informal, keberadaan emoji, serta variasi gaya bahasa [19]. Selain tantangan linguistik pada teks informal, ulasan terbaru menekankan persoalan generalisasi lintas domain dan keragaman bahasa sebagai hambatan utama, serta menunjukkan arah riset menuju model *deep learning* modern (termasuk *transformer*) dan arsitektur hybrid untuk meningkatkan ketahanan model terhadap variasi data [20]. Studi systematic *review* berbasis PRISMA juga menegaskan bahwa isu seperti ketidakseimbangan data, keterbatasan generalisasi lintas domain, serta perhatian pada aspek etika (misalnya bias algoritmik dan *explainability*) menjadi perhatian penting dalam penelitian analisis sentimen mutakhir [21]. Tantangan analisis sentimen bersifat lintas platform dan lintas domain, termasuk permasalahan ketidakseimbangan kelas, generalisasi model antar domain, serta isu bias dan interpretabilitas model. Ulasan sistematis berbasis pendekatan PRISMA menegaskan bahwa arah riset analisis sentimen modern bergerak menuju evaluasi komparatif antar pendekatan serta pengembangan model yang lebih adaptif dan kontekstual, khususnya untuk data media sosial yang bersifat dinamis dan heterogen [20], [21].

## 2.3 YouTube sebagai Sumber Data

YouTube merupakan platform video terbesar yang memungkinkan pengguna mengekspresikan pendapat melalui komentar [17]. Kolom komentar pada YouTube berisi opini publik yang bervariasi dan sering kali mencerminkan persepsi masyarakat terhadap suatu isu secara langsung [17]. Karakteristik komentar yang berupa teks bebas menjadikan YouTube sebagai sumber data menarik untuk analisis sentimen [17].

Penelitian berbasis komentar YouTube menunjukkan bahwa komentar dapat digunakan untuk memahami respons masyarakat terhadap isu-isu seperti kesehatan, teknologi, dan peristiwa sosial [15]. Komentar yang ditulis pengguna biasanya bersifat spontan sehingga dapat menggambarkan emosi dan sikap yang sebenarnya. Hal ini memberikan nilai tambah bagi analisis sentimen karena hasilnya mencerminkan opini publik secara organik [17].

Namun, komentar YouTube juga memiliki tantangan seperti keberadaan tautan spam, emoji, kesalahan pengetikan, dan penggunaan campuran bahasa (*code-mixing*) [17]. Tantangan ini memerlukan tahapan prapemrosesan yang komprehensif sebelum data dapat digunakan untuk pelatihan model. Dengan pengolahan yang tepat, komentar YouTube dapat memberikan wawasan berharga mengenai persepsi publik terhadap perkembangan kecerdasan buatan [22].

## 2.4 *Support Vector Machine* (SVM)

*Support Vector Machine* (SVM) merupakan salah satu algoritma *supervised machine learning* yang banyak digunakan untuk tugas klasifikasi, termasuk klasifikasi teks dan analisis sentimen [3], [23]. SVM bekerja dengan membangun sebuah fungsi pemisah (*classifier*) yang bertujuan untuk memisahkan data ke dalam kelas-kelas yang berbeda berdasarkan fitur yang dimilikinya [24]. Dalam konteks analisis sentimen, SVM digunakan untuk mengklasifikasikan teks ke dalam kategori sentimen seperti positif, negatif, atau netral berdasarkan representasi fitur teks [3], [17]. Penerapan *Support Vector Machine* pada analisis sentimen komentar YouTube telah diteliti sebelumnya dan menunjukkan bahwa kinerja SVM sangat dipengaruhi oleh karakteristik data serta teknik praproses yang digunakan, sehingga efektivitasnya bersifat kontekstual dan tidak selalu unggul secara universal [22].

TF-IDF merupakan metode yang digunakan pada *Natural Language Processing* (NLP) untuk merepresentasikan teks dalam bentuk numerik. Metode ini mengukur pentingnya suatu kata dalam dokumen terhadap keseluruhan dokumen dalam dataset [2].

Metode ini dibagi menjadi 2 bagian utama, yaitu :

- *Term Frequency* (TF) : Menghitung seberapa sering suatu kata muncul di dalam dokumen
- *Inverse Document Frequency* (IDF) : Menghitung seberapa jarang suatu kata muncul di dalam dokumen.

Secara konseptual, SVM memodelkan proses klasifikasi dengan mencari *hyperplane* optimal yang memisahkan data dari kelas yang berbeda di dalam ruang fitur (*feature space*) [24]. *Hyperplane* yang dianggap optimal adalah *hyperplane* yang memiliki *margin* terbesar, yaitu jarak maksimum antara *hyperplane* dengan titik data terdekat dari masing-masing kelas [24]. Titik-titik data yang berada paling dekat dengan *hyperplane* tersebut disebut sebagai *support vector*, dan titik-titik inilah yang menentukan posisi serta orientasi *hyperplane* [24].

*Support Vector Machine (SVM)* digunakan sebagai metode klasifikasi sentimen berbasis pembelajaran mesin dengan pendekatan supervised learning [22]. Diawali dengan prapemrosesan komentar YouTube yang meliputi normalisasi teks, penghapusan stopword, serta eliminasi duplikasi komentar. Teks yang telah dibersihkan kemudian direpresentasikan ke dalam bentuk vektor numerik menggunakan *Term Frequency–Inverse Document Frequency* (*TF-IDF*), sehingga setiap komentar direpresentasikan berdasarkan bobot kemunculan kata. Representasi teks menggunakan TF-IDF dilakukan untuk mengubah data teks menjadi bentuk numerik yang dapat diproses oleh model klasifikasi dan nilai *Inverse Document Frequency* (IDF) yang mengukur tingkat informatif suatu kata dihitung berdasarkan persamaan [2]:

$$TF(t,d) = \frac{f(t,d)}{\sum f(w,d)} \quad \text{.....(2.1)}$$

Rumus *Inverse Document Frequency (IDF) :*

$$IDF = \log_{10}\left(\frac{N}{df}\right) + 1 \quad \text{.....(2.2)}$$

Rumus TF-IDF *:*

$$TF - IDF\ (t,d) = TF(t,d) \times IDF(t) \quad \text{.....(2.3)}$$

Keterangan :

1. t = kata
2. d = dokumen

3. $f(t,d)$ = jumlah kemunculan kata t dalam d
4. $\sum f(w,d)$ = total seluruh kata dalam dokumen d
5. N = jumlah seluruh dokumen
6. df(t) = jumlah dokumen yang mengandung t

Setelah mendapatkan bobot TF-IDF, langkah selanjutnya adalah melakukan normalisasi terhadap vektor dokumen. Normalisasi (*L2 Norm*) bertujuan untuk menyamakan skala panjang setiap dokumen agar perbedaan panjang teks tidak memengaruhi performa model klasifikasi.

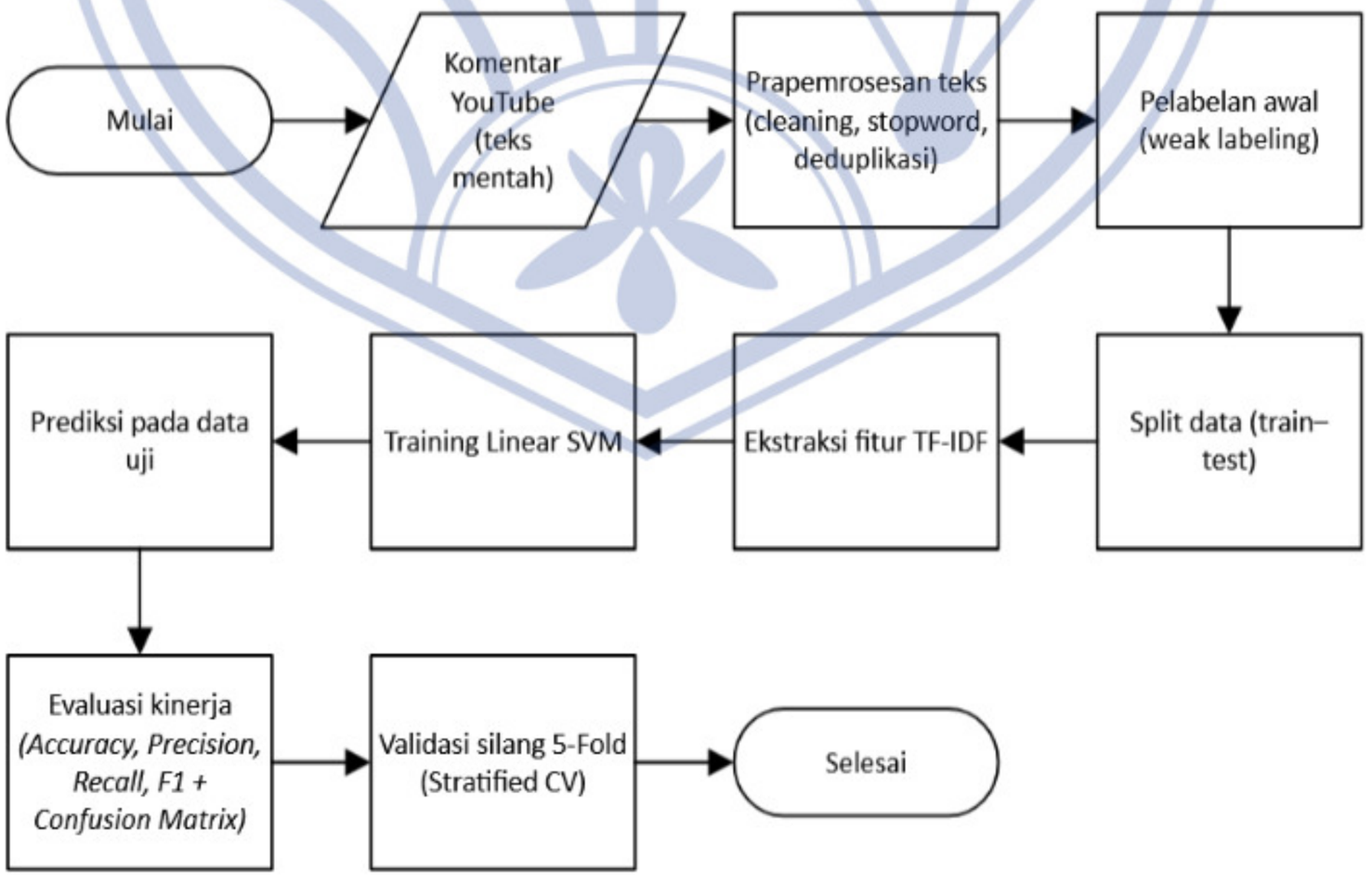
Rumus Normalisasi [25] :

$$\omega(t,d) = \frac{\omega(t,d)}{\sqrt{\sum \omega^2(t,d)}} \quad \text{.....(2.4)}$$

Keterangan:

1. ω(t,d) = nilai bobot kata t pada dokumen d setelah dinormalisasi.
2. $\sum\omega^2$(t,d) = jumlah kuadrat dari seluruh bobot kata yang ada dalam dokumen d.

Algoritma SVM kemudian menentukan bobot dari setiap fitur untuk membentuk sebuah *hyperplane* yang mampu memisahkan data dari kelas yang berbeda. *Hyperplane* yang dipilih merupakan *hyperplane* optimal dengan margin terbesar sehingga model memiliki kemampuan generalisasi yang lebih baik terhadap data baru.




Gambar 2.2 Flowchart proses klasifikasi sentimen menggunakan metode *Support Vector Machine (*SVM*)*

Keunggulan utama SVM terletak pada kemampuannya dalam menangani data berdimensi tinggi dan bersifat *sparse*, yang merupakan karakteristik umum pada data teks hasil ekstraksi fitur seperti *Term Frequency–Inverse Document Frequency* (TF-IDF) [3], [17]. Dalam analisis sentimen, jumlah fitur dapat mencapai ribuan hingga puluhan ribu dimensi, sehingga diperlukan algoritma yang tetap stabil dan efektif pada kondisi tersebut [26]. SVM terbukti mampu mempertahankan performa yang baik meskipun bekerja pada ruang fitur berdimensi tinggi [26].

Selain itu, SVM dikenal memiliki kemampuan generalisasi yang baik karena prinsip *margin* maksimum yang digunakan dalam proses pembelajarannya [24]. Dengan memaksimalkan *margin*, SVM cenderung menghasilkan model yang lebih *robust* terhadap data baru yang belum pernah dilihat sebelumnya [24]. Oleh karena itu, SVM sering digunakan sebagai *baseline* model dalam berbagai penelitian analisis sentimen untuk mengevaluasi dan membandingkan kinerja algoritma klasifikasi lainnya [3], [26]. Setelah data direpresentasikan dalam bentuk vektor, SVM melakukan proses optimasi untuk menentukan hyperplane optimal melalui pendekatan matematis berbasis fungsi kernel dan optimasi Lagrange. SVM bekerja dengan mencari *hyperplane* atau bidang pemisah terbaik yang memaksimalkan *margin* (jarak) antar kelas sentimen [24]. Untuk menyelesaikan permasalahan optimasi *margin* pada data yang kompleks, dilakukan perhitungan *Matrix Hessian* [24]:

$$D_{ij} = Y_i Y_j (K(X_i X) + \lambda^2) \quad \text{.....(2.5)}$$

Keterangan :

1. $D_{ij}$ = elemen matriks ke *ij*
2. $Y_i$ = kelas ke - *i* (label)
3. $Y_j$ = kelas ke - *j* (label)
4. $\lambda^2$ = batas teoritis yang diturunkan

Selama proses pencarian bidang pemisah, tingkat kesalahan (*error*) pada setiap data dihitung menggunakan persamaan [24]:

$$E_i = \sum_{j=1}^{n} a_j D_{ij} \quad \text{.....(2.6)}$$

Keterangan :

1. $E_i$ = error data ke - *i*

Nilai *error* tersebut kemudian digunakan untuk memperbarui pengali Lagrange atau nilai *delta* $a_i$ pada setiap iterasi. Pembaruan ini bertujuan untuk menemukan titik *support vector* yang paling optimal, sesuai dengan persamaan [24]:

$$\delta a_i = \min\{\max[y(1 - E_i) - a_i], C - a_i\} \quad \text{.....(2.7)}$$

Keterangan :

1. $\delta a_i$ = *Delta* a ke – *i*
2. $y$ = *Gamma*
3. $C$ = *Complexity*

Rumus menghitung $a_i$ baru :

$$a_i baru = a_i + \delta a_i \quad \text{.....(2.8)}$$

Ulangi langkah 3 – 5 hingga $a_i$ baru konvergen atau tidak ada perubahan signifikan.

Setelah model berhasil dilatih secara teoritis dan *support vector* ditemukan, nilai bias $(b)$ dari *hyperplane* dapat dihitung. Selanjutnya, fungsi keputusan akhir $(f(x))$ untuk memprediksi kelas sentimen dari data komentar yang baru dirumuskan pada persamaan [24]:

$$w.x^+ = a_i Y_i K(w.x^+)$$

$$w.x^- = a_i Y_i K(w.x^-)$$

$$b = -\frac{1}{2}(w.x^+ + w.x^-) \quad \text{.....(2.9)}$$

Keterangan :

1. $w$ = parameter *hyperplane* yang dicari atau garis tegak lurus antara garis *hyperplane* titik *support vector*
2. $w.x^+$ = nilai kernel data $x$ dengan data $x$ kelas positif yang memiliki nilai $\alpha$ tertinggi
3. $w.x^-$ = nilai kernel data $x$ dengan data $x$ kelas negatif yang memiliki nilai $\alpha$ tertinggi

Rumus menghitung nilai keputusan :

$$f(x) = \sum_{i=1}^{m} sign(a_i y_i K(x, x_i) + b) \quad \text{.....(2.10)}$$

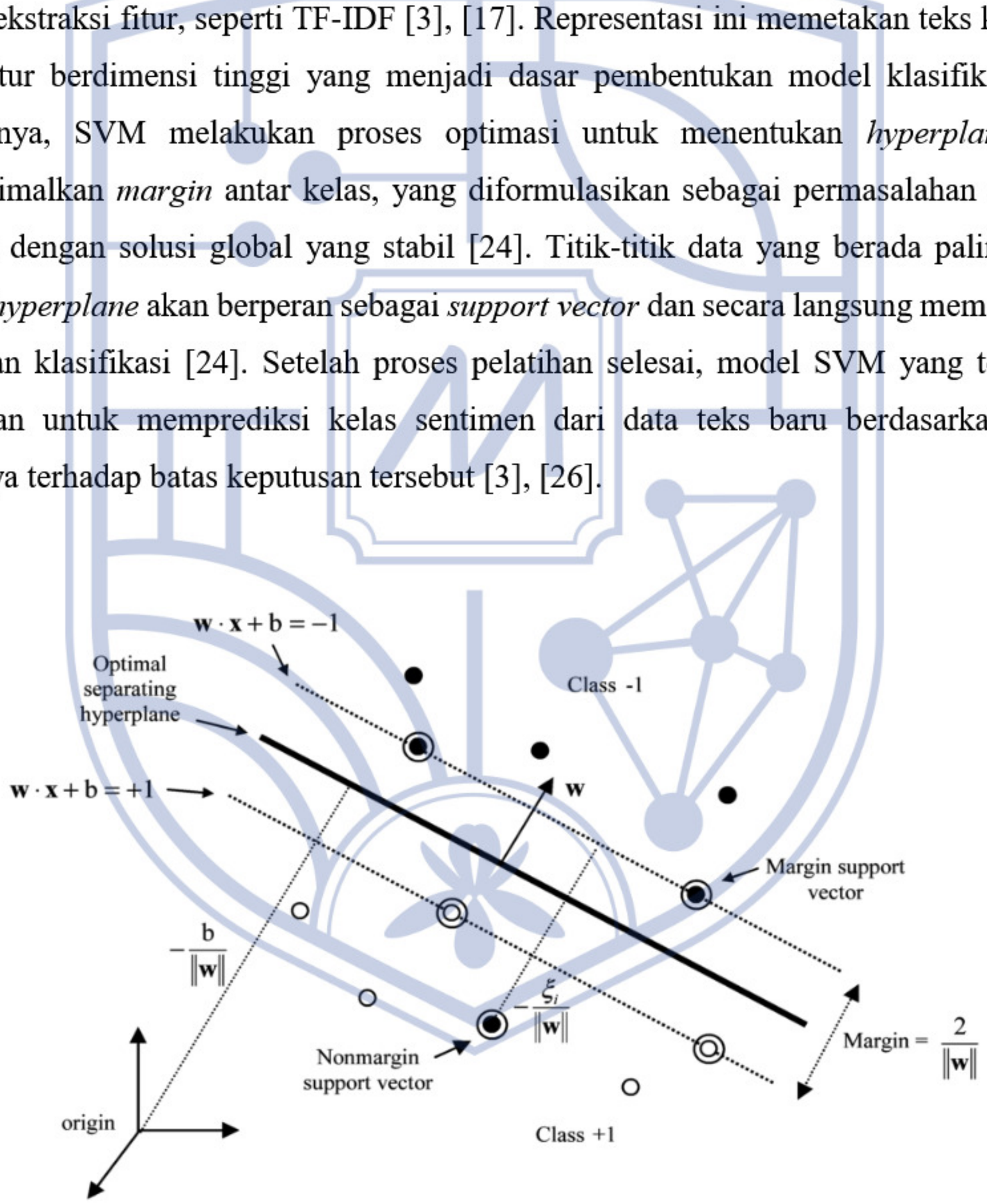
Keterangan :

1. $x$ = titik data masukan Support Vector Machine
2. $a_i$ = nilai bobot setiap titik data
3. $K(x, x_i)$ = fungsi kernel

4. $b$ = parameter *hyperplane* yang dicari atau nilai bias

Dalam berbagai penelitian terdahulu, SVM menunjukkan kinerja yang kompetitif pada analisis sentimen data media sosial, termasuk komentar YouTube [23]. Beberapa studi melaporkan bahwa SVM mampu mencapai performa yang tinggi pada dataset tertentu, meskipun pada kondisi lain kinerjanya dapat dipengaruhi oleh karakteristik data, metode *preprocessing*, serta distribusi kelas yang digunakan dalam penelitian [3], [17], [26].

Secara algoritmik, proses kerja *Support Vector Machine* dalam klasifikasi sentimen diawali dengan merepresentasikan data teks ke dalam bentuk vektor numerik menggunakan metode ekstraksi fitur, seperti TF-IDF [3], [17]. Representasi ini memetakan teks ke dalam ruang fitur berdimensi tinggi yang menjadi dasar pembentukan model klasifikasi [26]. Selanjutnya, SVM melakukan proses optimasi untuk menentukan *hyperplane* yang memaksimalkan *margin* antar kelas, yang diformulasikan sebagai permasalahan optimasi konveks dengan solusi global yang stabil [24]. Titik-titik data yang berada paling dekat dengan *hyperplane* akan berperan sebagai *support vector* dan secara langsung memengaruhi keputusan klasifikasi [24]. Setelah proses pelatihan selesai, model SVM yang terbentuk digunakan untuk memprediksi kelas sentimen dari data teks baru berdasarkan posisi vektornya terhadap batas keputusan tersebut [3], [26].




Gambar 2.3 Ilustrasi *Hyperplane, Margin,* dan *Support Vector* pada SVM [24]

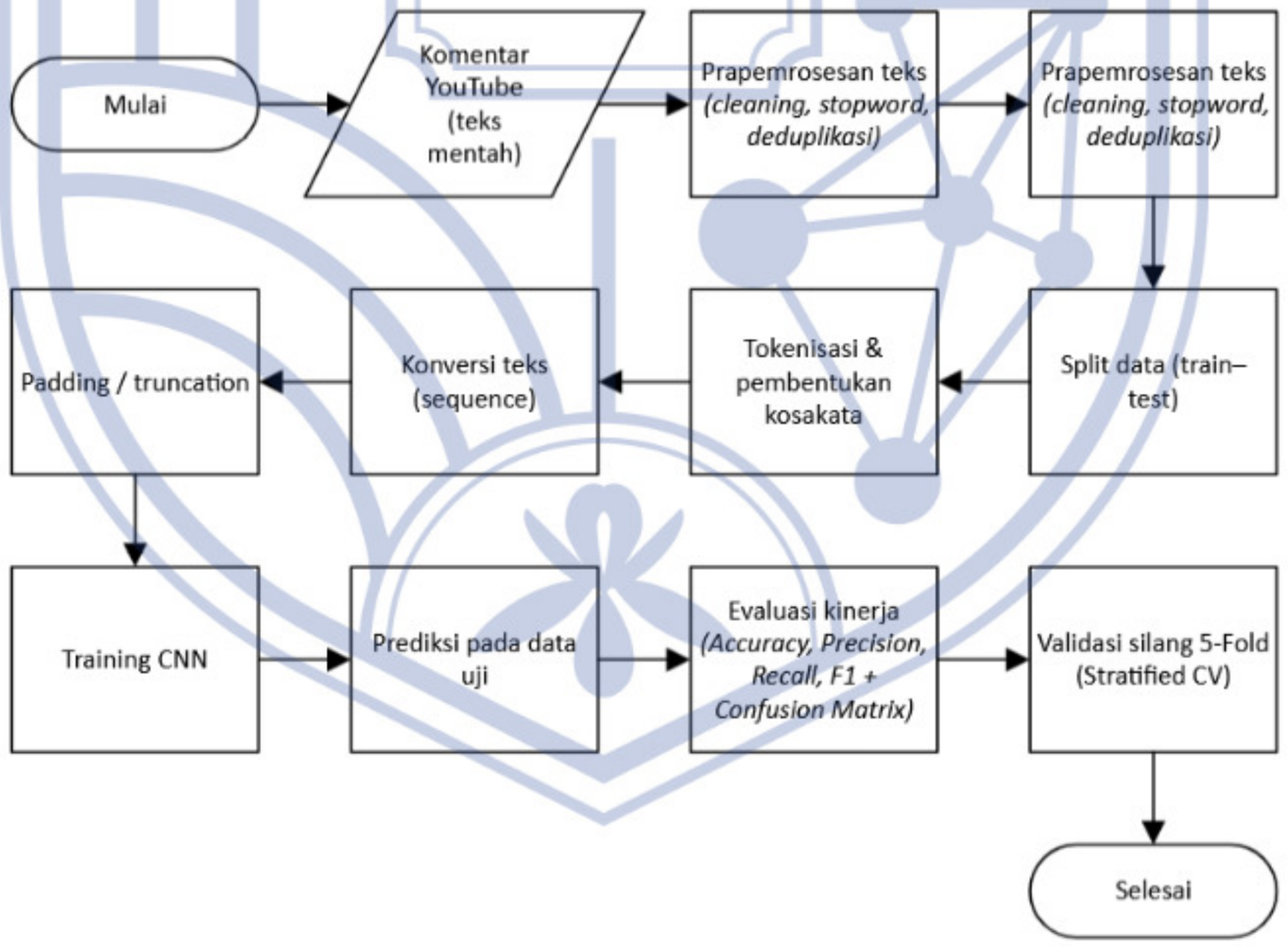
Dimana:

1. ***Hyperplane*** = Garis pemisah dua kelas.

2. **Optimal *Hyperplane*** = Garis pemisah terbaik dengan *margin* terbesar.
3. ***Margin*** = Jarak antara dua batas kelas.
4. ***Support Vector*** = Titik data terdekat yang menentukan posisi *hyperplane*.
5. ***Non-Margin Support Vector*** = Titik data yang tidak berada di garis *margin*.
6. **w** = Vektor yang menentukan arah *hyperplane.*
7. **b** = Parameter penggeser *hyperplane.*
8. $\xi_i$ = Nilai pelanggaran *margin* (*soft-margin*).
9. **Class +1 / −1** = Dua kelompok data yang dipisahkan SVM.
10. **Origin** = Titik koordinat awal (0,0).

## 2.5 *Convolutional Neural Network* (CNN)

*Convolutional Neural Network* (CNN) adalah arsitektur *deep learning* (pembelajaran mendalam) yang paling terkenal [26]. Meskipun awalnya dikembangkan dan meraih sukses besar di bidang *computer vision* (pemrosesan gambar), arsitektur CNN telah berhasil diadaptasi untuk tugas *Natural Language Processing* (NLP) dan terbukti sangat efektif untuk klasifikasi teks, termasuk analisis sentimen [3].

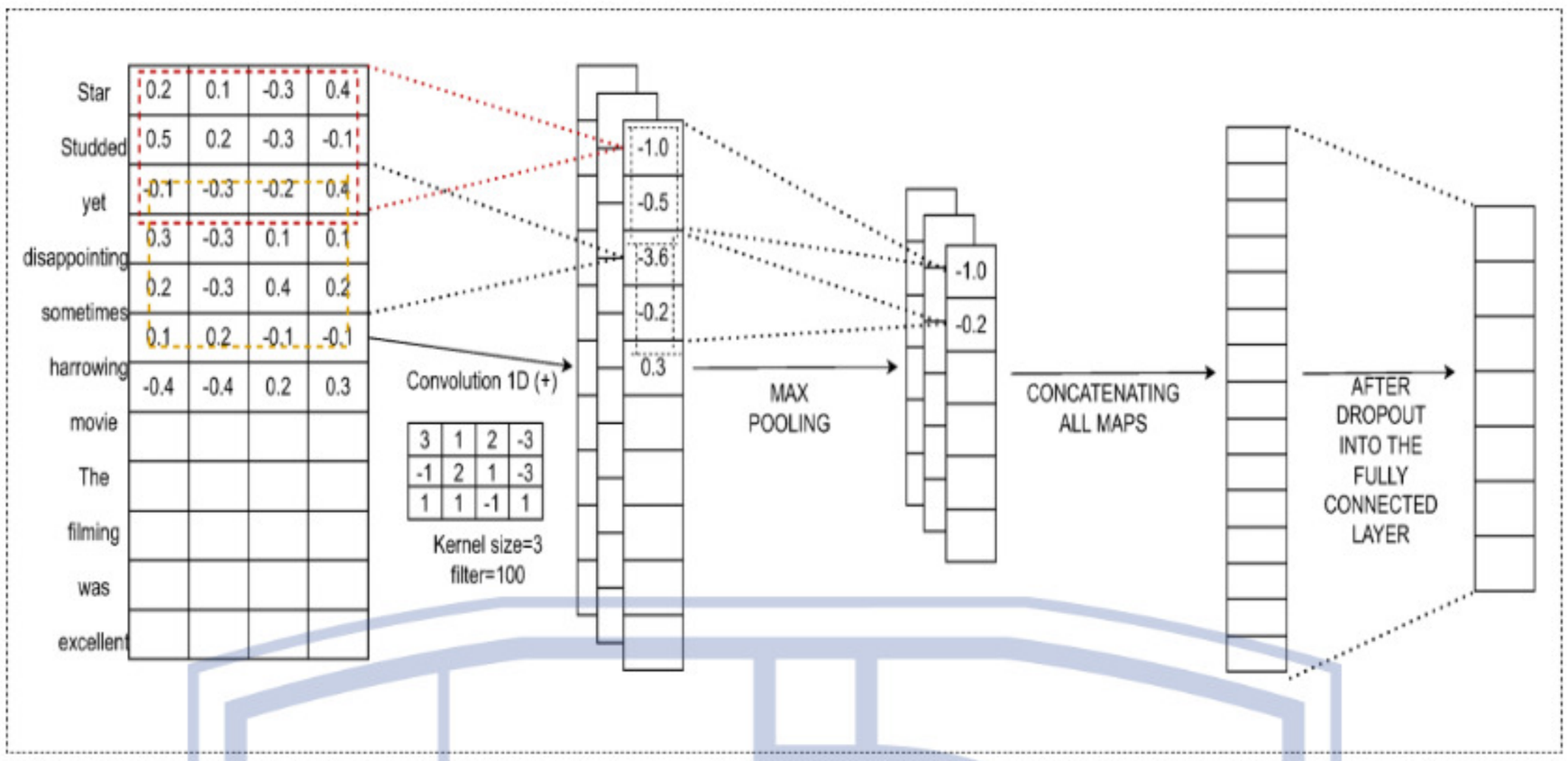



Gambar 2.4 Flowchart proses klasifikasi sentimen menggunakan *Convolutional Neural Network (*CNN*)*

Berbeda dengan SVM yang mengandalkan representasi fitur yang direkayasa secara eksplisit (seperti TF-IDF), keunggulan utama CNN adalah kemampuannya untuk

mempelajari representasi fitur secara otomatis (*automatic feature learning*) langsung dari data teks mentah [14]. Dalam konteks analisis sentimen, arsitektur CNN dirancang untuk mengidentifikasi pola-pola lokal yang signifikan dalam teks. Arsitektur ini umumnya bekerja melalui lapisan-lapisan berikut [3]:

1. Lapisan *Embedding*: Ini adalah lapisan input di mana kata-kata dalam kalimat masukan diubah menjadi representasi vektor numerik padat (*word embeddings*) [3], [14]. Lapisan ini memetakan kata-kata dengan makna semantik yang serupa ke lokasi yang berdekatan dalam ruang vektor, sehingga makna kata dapat ditangkap oleh model.
2. Lapisan Konvolusi (*Convolutional Layer*): Lapisan ini adalah inti dari CNN. Lapisan ini menerapkan beberapa "filter" (juga disebut *kernel*) dengan ukuran yang berbeda-beda (misalnya, filter yang mencakup 2 kata, 3 kata, 4 kata sekaligus) yang "menyapu" (*slide*) seluruh matriks *embedding* [3], [14]. Tujuan dari proses ini adalah agar model dapat mendeteksi fitur-fitur lokal yang informatif, seperti frasa atau n-gram (contoh: "sangat bagus" atau "tidak suka"), yang merupakan indikator kuat sentimen, terlepas dari di mana posisi frasa itu muncul dalam kalimat.
3. Lapisan *Pooling*: Segera setelah lapisan konvolusi, lapisan *pooling*—biasanya *Max-Pooling* diterapkan [3] [14]. Lapisan ini mengambil nilai maksimum (fitur yang paling "aktif" atau paling penting) dari hasil setiap filter. Ini berfungsi ganda: (1) mengurangi dimensionalitas data (meringkas informasi) dan (2) mempertahankan sinyal sentimen yang paling kuat yang telah dideteksi oleh filter.
4. Lapisan *Fully-Connected*: Akhirnya, fitur-fitur yang telah diringkas dari lapisan *pooling* diratakan (*flattened*) menjadi satu vektor panjang. Vektor ini kemudian dimasukkan ke lapisan *fully-connected* (jaringan saraf tiruan biasa) yang berfungsi sebagai *classifier* akhir untuk menentukan probabilitas setiap kelas sentimen (positif, negatif, atau netral), seringkali menggunakan fungsi aktivasi *Softmax* [3] [12].




Gambar 2.5 Arsitektur *Convolutional Neural Network* (CNN) untuk Klasifikasi Teks [27]

Secara algoritmik, proses kerja *Convolutional Neural Network* (CNN) dalam klasifikasi sentimen diawali dengan mengubah teks mentah menjadi representasi numerik melalui embedding kata [27]. Representasi ini berfungsi sebagai input awal yang menyimpan informasi semantik kata-kata dalam kalimat. Selanjutnya, vektor *embedding* tersebut diproses oleh lapisan konvolusi untuk mengekstraksi fitur-fitur lokal yang merepresentasikan pola hubungan antar kata, seperti frasa atau n-gram yang memiliki keterkaitan kuat dengan sentimen tertentu [3], [12], [27].

Fitur-fitur hasil konvolusi kemudian diringkas melalui lapisan *pooling* untuk mengurangi dimensi data sekaligus mempertahankan informasi yang paling relevan. Hasil *pooling* selanjutnya diteruskan ke lapisan *fully connected* yang berfungsi sebagai *classifier* akhir untuk menentukan kelas sentimen, seperti positif, negatif, atau netral. Alur algoritmik ini memungkinkan CNN mempelajari fitur sentimen secara otomatis langsung dari data teks tanpa memerlukan perancangan fitur manual, sehingga menjadikannya efektif untuk analisis sentimen pada data teks tidak terstruktur dalam skala besar [14], [27]. Proses ekstraksi fitur lokal dari representasi *word embedding* dilakukan melalui operasi *convolution* dengan nilai fitur ($C_i$) yang dihasilkan oleh sebuah filter pada jendela kata tertentu dihitung menggunakan persamaan [27]:

$$C_i = f(k.X_{i:i+h-1} + b) \quad \text{.....(2.11)}$$

Keterangan :

1. $C_i$ = *feature maps*
2. $k$ = matriks kernel atau filter

3. $h$ = ukuran kernel

4. $X_{i:i+h-1}$ = merepresentasikan dokumen yang telah dilakukan *embedding*

5. $b$ = bias

Untuk mereduksi dimensi dari *feature map* sekaligus mengambil sinyal sentimen terkuat, diterapkan lapisan *Max Pooling* dimana proses *pooling* ini mengambil nilai maksimal dari kumpulan fitur yang dihasilkan, yang dirumuskan pada persamaan [27]:

$$S_i = \max\{C_1, C_2, C_3, \dots, C_{n-h+1}\} \quad (2.12)$$

Keterangan :

1. $S_i$ = hasil pooling
2.

Fitur-fitur dominan yang telah diringkas dari lapisan *pooling* kemudian diratakan (*flattened*) dan dimasukkan ke dalam lapisan *fully-connected* atau *Dense Neural Network* (DNN) yang akan memproses input fitur menggunakan bobot (*weight*) dan bias untuk menghasilkan keluaran matriks atau nilai logit melalui persamaan [28]:

$$z = \sigma(W.x) + b \quad (2.13)$$

Keterangan :

1. $z$ = output layer
2. $W$ = bobot
3. $x$ = input

Pada tahap klasifikasi akhir untuk studi multikelas, keluaran nilai logit dari *dense layer* belum berada dalam rentang probabilitas [27]. Oleh karena itu, digunakan fungsi aktivasi *Softmax* yang akan digunakan untuk mengonversi nilai keluaran menjadi probabilitas distribusi antar kelas sentimen (seperti positif, negatif, atau netral) menggunakan persamaan [27] :

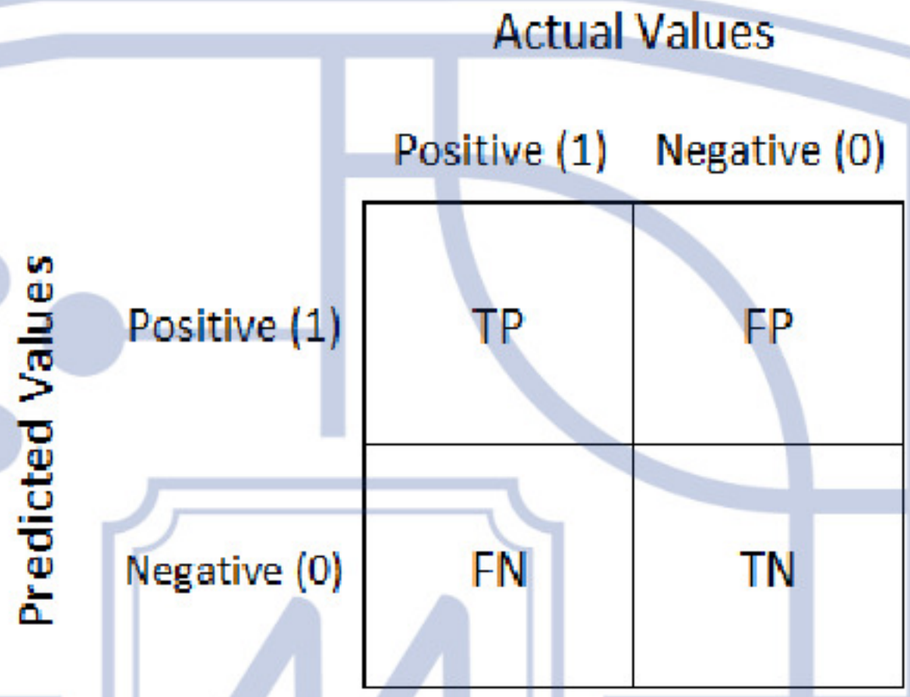
$$\sigma(z)_i = \frac{e^{z_i}}{(\sum_{j=1}^{K} e^{z_j})} \quad (2.14)$$

Keterangan :

1. $z_i$ = output neuron
2. $K$ = jumlah kelas

## 2.6 Metrik Evaluasi

Untuk mengevaluasi dan membandingkan kinerja model SVM dan CNN secara adil, objektif, dan kuantitatif, diperlukan serangkaian metrik evaluasi yang standar. Dasar dari semua metrik ini adalah *Confusion Matrix* [27]. *Confusion Matrix* adalah sebuah tabel yang dirancang untuk memvisualisasikan kinerja algoritma klasifikasi dengan menyajikan hasil prediksi model dibandingkan dengan label data yang sebenarnya (data aktual) [8].




Gambar 2.6 Diagram Matriks Kebingungan (*Confusion Matrix*) [8], [9]

*Confusion Matrix* membagi hasil prediksi ke dalam empat kategori: *True Positive* (TP) yaitu data positif yang diprediksi benar sebagai positif, *True Negative* (TN) yaitu data negatif yang diprediksi benar sebagai negatif, *False Positive* (FP) yaitu data negatif yang salah diprediksi sebagai positif, dan *False Negative* (FN) yaitu data positif yang salah diprediksi sebagai negatif [26].

Dari empat komponen fundamental ini, empat metrik utama dihitung untuk menilai kinerja model:

1. Akurasi (*Accuracy*): Mengukur proporsi total prediksi yang benar (TP + TN) terhadap keseluruhan jumlah data [25]. Ini adalah metrik paling umum untuk kinerja model secara keseluruhan, namun bisa menyesatkan jika jumlah data antar kelas tidak seimbang [25].

$$Accuracy = \frac{(TP + TN)}{(TP + TN + FP + FN)} \quad (2.15)$$

2. Presisi (*Precision*): Dihitung dengan rumus, Metrik ini mengukur seberapa banyak dari sentimen yang diprediksi positif oleh model yang benar-benar positif dimana presisi yang tinggi menunjukkan tingkat *false positive* yang rendah [25].

$$Precision = \frac{TP}{(TP + FP)} \quad (2.16)$$

3. *Recall*: Dihitung dengan rumus, Metrik ini mengukur seberapa banyak dari sentimen yang sebenarnya positif yang berhasil diidentifikasi oleh model dimana *recall* yang tinggi menunjukkan tingkat *false negative* yang rendah [25].

$$Recall = \frac{TP}{(TP + FN)} \quad \text{.....(2.17)}$$

4. F1-Skor (*F1-Score*): Merupakan rata-rata harmonik (rerata harmonis) dari Presisi dan *Recall*, dengan rumus [25]. Metrik ini sangat penting karena memberikan keseimbangan antara Presisi dan *Recall*, dan dianggap sebagai metrik yang lebih *robust* (andal) untuk mengevaluasi model, terutama jika terjadi ketidakseimbangan kelas data [3].

$$F1 = 2 \times \frac{(Precision \times Recall)}{(Precision + Recall)} \quad \text{.....(2.18)}$$

## 2.7 Reliabilitas Antar-Pelabel (*Inter-Annotator Agreement*)

Kualitas dataset terlabeli sangat menentukan validitas model klasifikasi yang dihasilkan yang dimana ketika pelabelan dilakukan oleh lebih dari satu penilai, perlu dilakukan pengukuran objektif terhadap tingkat kesepakatan antar-penilai untuk memastikan bahwa label yang diberikan bersifat konsisten dan tidak hanya mencerminkan persepsi subjektif satu penilai tertentu [29], [30]. Indikator yang umum digunakan untuk tujuan ini adalah koefisien *Cohen's Kappa* untuk dua penilai dan *Fleiss' Kappa* untuk tiga atau lebih penilai [29], [30].

### 2.7.1 *Cohen's Kappa*

*Cohen's Kappa* merupakan metode statistik klasik yang umum digunakan untuk menilai tingkat reliabilitas antar-penilai pada data nominal atau kategorikal, khususnya pada kasus yang melibatkan dua penilai [30]. Berbeda dengan persentase kesepakatan mentah yang hanya menghitung proporsi label yang sama, *Cohen's Kappa* memperhitungkan kemungkinan kesepakatan yang terjadi secara kebetulan (*chance agreement*) sehingga memberikan ukuran kesepakatan yang lebih akurat [30]. Nilai *Cohen's Kappa* dihitung berdasarkan persamaan berikut [31]:

$$\kappa = \frac{P_o - P_e}{1 - P_e} \quad \text{.....(2.19)}$$

Keterangan:

$\kappa$ = nilai koefisien *Cohen's Kappa*

$P_o$ = proporsi kesepakatan teramati (*observed agreement*) antara dua penilai

$P_e$ = proporsi kesepakatan yang diharapkan terjadi secara kebetulan (*expected agreement)*

Nilai $\kappa$ berkisar antara −1 hingga 1, di mana nilai 1 menunjukkan kesepakatan sempurna, nilai 0 menunjukkan kesepakatan setara dengan kebetulan, dan nilai negatif menunjukkan kesepakatan lebih buruk daripada kebetulan [30]. *Cohen's Kappa* memiliki beberapa keterbatasan yang perlu diperhatikan, antara lain sensitivitas terhadap prevalensi kelas yang tidak seimbang dan asumsi bahwa kedua penilai bersifat independen. Meskipun demikian, kelebihan utama metode ini adalah kemampuannya memberikan ukuran kesepakatan yang lebih akurat dibandingkan persentase kesepakatan biasa dengan memperhitungkan kesepakatan kebetulan [30].

### 2.7.2 *Fleiss' Kappa*

Keterbatasan *Cohen's Kappa* hanya memungkinkan pengukuran kesepakatan antara dua penilai independen, oleh karena itu Fleiss memperkenalkan *Fleiss' Kappa* pada tahun 1971 yang memungkinkan jumlah penilai tetap sebanyak dua atau lebih, di mana para penilai mengkategorikan subjek ke dalam tepat satu kategori dari kategori yang tersedia [31]. Misalkan I menyatakan jumlah total subjek, J menyatakan jumlah penilai yang tetap, dan C menyatakan jumlah kategori. Misalkan $X_{ic}$ menyatakan jumlah penilai yang mengklasifikasikan subjek ke-i ke dalam kategori ke-c [31].

Karena setiap subjek dinilai oleh tepat J penilai, proporsi pasangan penilai yang sepakat pada subjek ke-i, yang dinotasikan sebagai $P_i$, dihitung sebagai perbandingan antara pasangan penilai yang sepakat dengan seluruh kemungkinan pasangan penilai, dan dapat dinyatakan dalam persamaan berikut [31]:

$$P_i = \frac{\sum_c X_{ic}^2 - J}{\mathrm{J}(\mathrm{J}-1)} \quad \text{.....(2.20)}$$

Proporsi kesepakatan teramati keseluruhan $P_o$ diperoleh dari rata-rata seluruh $P_i$ pada I subjek, sebagaimana persamaan berikut [31]:

$$P_o = \frac{1}{\mathrm{I}} \sum_i P_i \quad \text{.....(2.21)}$$

Sedangkan proporsi kesepakatan yang diharapkan terjadi secara kebetulan $P_e$ dihitung dari penjumlahan kuadrat proporsi keseluruhan tiap kategori, mengikuti persamaan berikut [31]:

$$P_e = \sum_c (\frac{\sum_i X_{ic}}{I\,J})^2 \quad \text{.....(2.22)}$$

Dengan demikian, koefisien *Fleiss' Kappa* diperoleh dari persamaan [31]:

$$\kappa = \frac{\frac{\sum_i \sum_c X_{ic}^2 - I\,J}{\mathrm{I\,J\,(J-1)}} - \sum_c (\frac{\sum_i X_{ic}}{I\,J})^2}{1 - \sum_c (\frac{\sum_i X_{ic}}{I\,J})^2} \quad \text{.....(2.23)}$$

Dengan keterangan:

$\kappa$ = nilai koefisien *Fleiss' Kappa*

$I$ = jumlah total subjek yang dinilai

$J$ = jumlah penilai yang tetap

$C$ = jumlah kategori klasifikasi

$X_{ic}$ = jumlah penilai yang mengklasifikasikan subjek ke-i ke kategori ke-c

$P_i$ = proporsi kesepakatan pasangan penilai untuk subjek ke-i

$P_o$ = proporsi kesepakatan teramati keseluruhan

$P_e$ = proporsi kesepakatan yang diharapkan terjadi secara kebetulan

### 2.7.3 Interpretasi Nilai *Kappa*

Untuk menginterpretasikan nilai koefisien Kappa baik *Cohen's Kappa* maupun *Fleiss' Kappa*, digunakan kategori benchmark yang telah menjadi standar acuan dalam berbagai bidang penelitian, termasuk pemrosesan bahasa alami dan analisis sentimen dengan klasifikasi tingkat kesepatakan tersebut dibagi menjadi [29], [30]:

Tabel 2.1 Kategori Interpretasi Nilai *Kappa*

| **Nilai *Kappa* (κ)** | **Tingkat Kesepakatan** |
|---|---|
| < 0,00 | *Poor* |
| 0,00 – 0,20 | *Slight* |
| 0,21 – 0,40 | *Fair* |
| 0,41 – 0,60 | *Moderate* |
| 0,61 – 0,80 | *Substantial* |
| 0,81 – 1,00 | *Almost Perfect* |

Pada penelitian analisis sentimen teks media sosial Bahasa Indonesia, kelas netral cenderung lebih ambigu karena seringkali mengandung campur kode, sarkasme, dan ekspresi yang sulit diinterpretasikan secara objektif. Akibatnya, nilai *Kappa* pada rentang

*Moderate* (0,41 – 0,60) hingga *Substantial* (0,61 – 0,80) umumnya dianggap memadai untuk landasan pelatihan model klasifikasi sentimen [29]. Hal ini sejalan dengan praktik yang diterapkan yang memperoleh nilai *Fleiss' Kappa* sebesar 0,62187 pada pelabelan sentimen Twitter Bahasa Indonesia dan mengkategorikannya sebagai *Substantial Agreement*, serta dianggap cukup memadai untuk memastikan konsistensi dan reliabilitas dataset [29].

## 2.8 Penelitian Terkait dan Kerangka Berpikir

Tinjauan terhadap penelitian-penelitian akademis sebelumnya menunjukkan adanya hasil yang beragam dan bertentangan mengenai algoritma mana yang paling unggul untuk analisis sentimen, khususnya pada data media sosial berbahasa Indonesia.

Tabel 2.2 Hasil Penelitian Terdahulu

| No | Peneliti | Dataset / Domain | Metode yang Dibandingkan | Hasil yang Dikutip (Rinci) |
|---|---|---|---|---|
| 1 | Ifriza & Sam'an [23] | Komentar spam YouTube | SVM vs *Gaussian Naive Bayes* | SVM mencapai *F1-score* 88.00%, lebih tinggi dibandingkan *Gaussian Naive Bayes* dengan *F1-score* 84.38%. Hasil menunjukkan SVM lebih efektif dalam menangani klasifikasi spam berbasis teks pendek dan tidak seimbang pada komentar YouTube. |
| 2 | Mohamad Jamil et al [22]. | Komentar simulasi manajemen bencana YouTube (Indonesia) | *Naive Bayes* vs SVM | *Naive Bayes* memperoleh akurasi 80.4%, sedangkan SVM hanya mencapai 72.3%. Studi ini menunjukkan bahwa pada konteks data simulasi bencana dengan distribusi kata tertentu, model probabilistik sederhana dapat mengungguli SVM. |
| 3 | Shinta Nilam Sari Muslim et al. [32] | Ulasan aplikasi CapCut | *Naive Bayes* vs *K-Nearest Neighbor* (KNN) | *Naive Bayes* menghasilkan akurasi 79.41%, lebih tinggi dibandingkan KNN dengan akurasi 75.63%. Temuan ini menunjukkan bahwa algoritma berbasis probabilitas lebih stabil untuk teks ulasan pendek dibandingkan metode berbasis jarak. |
| 4 | Guido Tamara & Kemas | Komentar YouTube Indonesia | CNN vs LSTM | CNN mencapai akurasi 0.92, sedikit lebih tinggi dibandingkan LSTM dengan |

| | | | | |
|---|---|---|---|---|
| | Muslim [19] | (kasus obat sirup / *Acute Kidney Syrup*) | | akurasi 0.90, terutama setelah dilakukan penanganan data tidak seimbang. Hal ini menunjukkan efektivitas CNN dalam mengekstraksi pola lokal pada teks komentar YouTube. |
| 5 | Rajesh Kumar Das et al [26]. | Ulasan e-commerce multibahasa (termasuk Bangla) | SVM vs Conv1D (CNN) | SVM memperoleh akurasi 86.43%, sedangkan Conv1D (CNN) mencapai 81.78%. Hasil ini menunjukkan bahwa pada dataset tertentu dan bahasa tertentu, model *machine learning* tradisional masih dapat mengungguli model *deep learning*. |

Berdasarkan Tabel 2.2, terlihat bahwa hasil penelitian sebelumnya menunjukkan temuan yang beragam dan tidak konsisten terkait metode yang paling unggul untuk analisis sentimen. Perbedaan kinerja model dipengaruhi oleh domain data, karakteristik teks, bahasa, serta metode yang digunakan, sehingga belum terdapat konsensus apakah pendekatan *machine learning* tradisional seperti *Support Vector Machine* atau pendekatan *deep learning* seperti *Convolutional Neural Network* yang secara definitif lebih unggul. Oleh karena itu, tabel ini digunakan sebagai acuan untuk mengidentifikasi celah penelitian yang menjadi dasar dilakukannya penelitian ini, yaitu melakukan perbandingan kinerja SVM dan CNN pada analisis sentimen komentar YouTube berbahasa Indonesia dengan topik perkembangan Kecerdasan Buatan.