

## **BAB II**

### **TINJAUAN PUSTAKA**

#### **2.1 Cyberbullying (Ujaran Kebencian)**

Ujaran kebencian menjadi bukti merusak fungsi bahasa. Sekarang semakin banyak orang yang tidak tertarik dengan bahasa Indonesia yang baik (Syafyahya, 2018). *Cyberbullying* adalah tindakan negatif yang dilakukan seseorang secara terus menerus atau berulang kali (Candra & Rozana, 2020). *Cyberbullying* dapat didefinisikan sebagai tindakan pengucilan sosial atau intimidasi yang dilakukan secara online, seperti dengan sengaja mengirimkan pesan-pesan ofensif yang secara faktual tidak benar dan menyebarkan informasi pribadi tanpa izin dari pemiliknya (Radjavani & Sasongko, 2023). Hal ini adalah tindakan agresif di mana seseorang atau sekelompok orang dirugikan, dilecehkan, diancam, atau direndahkan melalui penggunaan komputer, ponsel, atau internet. Korban *cyberbullying* rentan terkena depresi dibandingkan dengan korban tindakan kekerasan verbal lainnya.

#### **2.2 Python**

Bahasa *Python* diciptakan pada tahun 1989 oleh Guido van Rossum, seorang ahli komputer Belanda, penciptaan (setup) bahasa pemrograman pada dasarnya adalah penetapan aturan penggunaan prosedur, dan kemudian program (pengaturan) dibuat oleh pemrogram (programmer) (Alfarizi et al., 2023). Bahasa pemrograman itu sendiri terdiri dari subbahasa, yaitu Bahasa yang dekat dengan bahasa *assembly* dan bahasa mesin. *Python* sendiri ditulis dalam Bahasa C karena Van Rossum tidak puas dengan kemampuan C dalam menulis program komputer. Untuk itu, saya mengusulkan Bahasa baru untuk menggantikan C, yang sulit dipelajari oleh pemula (Alfarizi et al., 2023).

##### **2.2.1 Pandas**

*Pandas* adalah *library Python* yang digunakan untuk manipulasi dan analisis data (Isa Albanna and R. Tri hadi laksono, 2022). *Library* ini sangat berguna untuk operasi-operasi pada tabel angka. Nama "*pandas*" berasal dari "*panel data*", sebuah istilah dalam statistik yang berkaitan dengan data *multi-dimensi* yang diukur terhadap waktu. Selain

itu, "*pandas*" juga merupakan singkatan dari frasa "*Python data analysis*" (Ghifari, 2021).

### **2.3 Seaborn**

*Seaborn* adalah *library Python* yang digunakan untuk membuat grafik dan visualisasi statistik (Mambang et al., 2022). *Seaborn* dibangun di atas *library matplotlib* dan terintegrasi dengan struktur data pada *Pandas*, sehingga memudahkan dalam visualisasi data yang disimpan dalam *DataFrame* (WIJOYO, 2021).

### **2.4 Matplotlib**

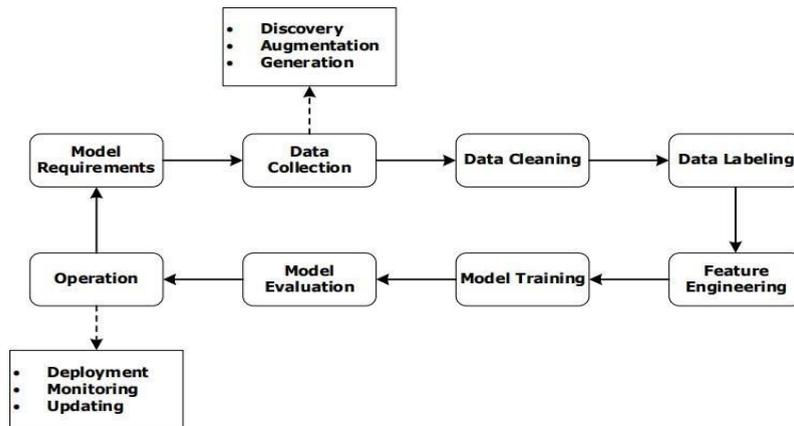
*Matplotlib* adalah *library Python* untuk menghasilkan plot 2D dengan kualitas tinggi dalam berbagai format dan di berbagai *platform* (Alfarizi et al., 2023). Jenis grafik yang dapat dibuat oleh *Matplotlib* meliputi grafik garis, batang, lingkaran, histogram, dan berbagai macam lainnya (WIJOYO, 2021).

### **2.5 Scikit-learn**

*Scikit-learn* adalah modul *Python* yang menyediakan berbagai algoritma *machine learning* untuk masalah berskala menengah, baik yang diawasi (*supervised*) maupun yang tidak diawasi (*unsupervised*) (Fahmi, 2023). Modul ini bertujuan untuk memudahkan "membawa *machine learning* ke *non-spesialis*" dengan menggunakan bahasa pemrograman tingkat tinggi yang umum. (Dainamang et al.,2022).

### **2.6 Machine Learning**

*Machine Learning* adalah kecerdasan buatan yang belajar mengembangkan sistem, menghasilkan data, dan membuat algoritma yang dapat melakukan tugas sendiri tanpa bimbingan apa pun dari pengguna dan memungkinkan pemrogram untuk "belajar" (Putra, 2022). Berikut siklus metodologi *machine learning* oleh Gartner et al. Dapat di lihat pada gambar 2.1



**Gambar 2.1** Siklus *Metodologi Machine Learning* oleh Gartner

## 2.7 *Natural Language Processing*

*Natural Language Processing (NLP)* adalah suatu tindakan untuk melakukan ekstrak yang lebih bagus representasi yang dihasilkan dari suatu teks bebas (Fadilah, 2021). *Natural Language Processing (NLP)* adalah cabang kecerdasan buatan (AI) bagaimana komputer dan bahasa manusia berinteraksi dari pemrosesan bahasa alami (NLP). Tujuan utamanya adalah untuk memungkinkan komputer memahami, menafsirkan, dan mereplikasi bahasa manusia secara bermakna dan efektif.

## 2.8 *K-Nearest Neighbor*

*K-Nearest Neighbor (KNN)* merupakan algoritma untuk mengklasifikasikan objek berdasarkan data pembelajaran yang paling dekat dengan objek tersebut. (Candra & Rozana, 2020). Dengan menggunakan kasus-kasus yang sudah ada, algoritma *K-Nearest Neighbor (KNN)* sangat cocok untuk memperkirakan peluang apa yang akan terjadi selanjutnya (Chia dkk., 2021). Metode *K-Nearest Neighbor (KNN)* membuat pengambilan keputusan menjadi sangat mudah karena kemiripannya dengan kasus-kasus sebelumnya.

## 2.9 *Feature Engineering*

Proses *Feature Engineering* melibatkan pembagian data menjadi data training dan data testing (Agus et al., 2022).

## 2.10 **Pembobotan Kata TF-IDF**

Pembobotan kata atau *Term Weight* adalah metode pembobotan kata (*term*) yang memberikan bobot atau nilai pada kata yang terdapat dalam dokumen (Hendra Suputra, Prebiana and Gorianto, 2021). Ada beberapa metode pembobotan (*term*),

antara lain TF, TF-IDF, WIDF, dan TF-RF.

Data yang sudah menyelesaikan tahap preprocessing sebelumnya haruslah berbentuk numerik, untuk mengubah data yang berupa kata-kata atau string menjadi berbentuk numerik dengan menggunakan metode pembobotan kata atau *Term Weight* yaitu TF-IDF. Metode TF-IDF (*Term Frequency Invers Document Frequency*) adalah langkah yang dipakai dalam memastikan sejauh mana kemiripan suatu *term* atau kata mengenai suatu Kumpulan data melalui pembeian bobot pada setiap kata. Metode TF-IDF ini mengkombinasikan lebih dari satu konsep adalah frekuensi kemunculan sebuah kata dalam sebuah dokumen dan frekuensi kebalikan dari dokumen yang mengandung kata tersebut.

Rumus pembobotan kata TF-IDF adalah sebagai berikut:

$$Wt = \left(\log \frac{n}{df}\right) \cdot Tf$$

Keterangan:

Wt : Pembobotan TF-IDF

n : Jumlah total dokumen atau data

df : Jumlah kata yang muncul dalam seluruh dokumen

Tf : Jumlah kemunculan kata pada masing-masing dokumen

### **2.11 Confusion Matrix**

Confusion matrix adalah sebuah tabel yang digunakan untuk mengevaluasi kinerja suatu model klasifikasi dengan membandingkan nilai aktual atau nilai sebenarnya dari data dengan nilai hasil prediksi yang dihasilkan oleh model (Dainamang, Hayatin, and Chandranegara 2022). Hasil dari *confusion matrix* adalah sebuah ringkasan prediksi yang menyajikan jumlah nilai benar dan salah yang dikelompokkan berdasarkan masing-masing kategori labe, masing kategori label serta ditampilkan dalam format tabel (Farhan, 2023). Dengan rincian yang tercantum dalam penjelasan berikut:

#### *1. Accuracy*

Akurasi dihitung dengan membagi jumlah data yang diprediksikan benar (Trisnawati et al., 2021). baik positif maupun negatif, oleh model, dengan total jumlah data dalam dataset (Farhan, 2023). Berikut adalah rumus *accuracy* :

$$Accuracy = \frac{(TP+TN)}{(TP+TN+FP+FN)} \quad (2.1)$$

## 2. Precision

*Precision* adalah proporsi dari kasus yang diprediksi positif oleh model yang sebenarnya termasuk dalam kategori positif (Farhan, 2023). Berikut adalah rumus *precision* :

$$Precision = \frac{(TP)}{(TP+FP)} \quad (2.2)$$

## 3. Recall

*Recall* adalah proporsi dari kasus dengan kategori positif yang secara tepat diprediksi sebagai positif oleh model (Farhan, 2023). Berikut adalah rumus *recall* :

$$Recall = \frac{(TP)}{(TP+FN)} \quad (2.3)$$

## 4. F1-Score

*F1-Score* adalah nilai rata-rata harmonik dari *Precision* dan *Recall*, yang mengukur keseimbangan antara keduanya dalam memprediksi kategori hasil dengan kategori sebenarnya (Farhan, 2023). Berikut adalah rumus *f1-score* :

$$F1-Score = \frac{2x (Precision \times Recall)}{(Precision + Recall)} \quad (2.4)$$

Matrix ini memiliki empat sel utama yang mewakili empat kemungkinan kombinasi dari prediksi yang benar dan yang salah, yaitu *true positive* (TP), *trueneegative* (TN), *false positive* (FP), dan *false negative* (FN). Berikut adalah penjelasan dari setiap komponen dalam *confusion matrix*:

### 1. True Positives (TP)

Jumlah prediksi yang benar dan sesuai dengan label positif pada data yang diprediksi (Farhan, 2023).

### 2. True Negatives (TN)

Jumlah prediksi yang benar dan sesuai dengan label negatif pada data yang diprediksi (Farhan, 2023).

### 3. False Positives (FP)

Jumlah prediksi yang salah dan tidak sesuai dengan label positif pada data yang diprediksi, yang juga dikenal sebagai *Type I error* (Farhan, 2023).

### 4. False Negatives (FN)

Jumlah prediksi yang salah dan tidak sesuai dengan label negatif pada data yang diprediksi, merupakan *Type II error* (Farhan, 2023).

## 2.12 Kalimat yang Mengandung *Cyberbullying* atau Tidak

Untuk melakukan analisis terhadap kasus-kasus *cyberbullying* di media sosial seperti *twitter*, langkah pertama adalah mengumpulkan dan mengevaluasi kalimat-kalimat yang terindikasi mengandung unsur tersebut. Berikut adalah contoh dataset yang telah disiapkan untuk studi ini:

**Tabel 2.1 Contoh Kalimat yang Mengandung *Cyberbullying* atau Tidak**

Kalimat	Label
Tidak aneksasi invasi	Tidak
Tulis kota solo benar solo sala mas	Tidak
Masing-masing jalan penuh tanggung jawab terima kasih	Tidak
Nggih les masalah mohon adu kepala wilayah masing-masing kepala wilayah tingkat rendah tingkat tingkat teratas begitu makan gaji buta tupoksinya	Tidak
Sok cerdas ente paham tidak tugas walikota bank bumh luar ranah walikota	Iya
Hahaha kalau bawaslu tidak mengasih sangai bobby gibran ya meniru	Iya
Eh goblok kasih tau serah yang piknik jumat memaksakan biasanya sudah goblok kere	Iya
Wah pintar benar sih gibran monyet darah sudah didih	Iya
Laki sampah sih gibran monyet	Iya
Bandit oknum harus lekas disepiralkan biar viral mas	Tidak
Sial gibran dor ini saja tidak laki otaknya sudah hilang	Iya
Prabowo mundur	Iya

Tabel 2.1 menampilkan contoh kalimat-kalimat yang terkait dengan kasus *cyberbullying* ("iya") dan yang tidak terkait ("tidak"). Data ini dapat digunakan sebagai dasar untuk mengembangkan model deteksi *cyberbullying* menggunakan teknik machine learning.

## 2.13 Metode Pengembangan Perangkat Lunak *Agile Development*

*Agile software development methods* adalah sekumpulan metodologi perangkat lunak berdasarkan pengembangan berulang Dimana persyaratn dan Solusi dikembangkan dalam kolaborasi oleh tim yang terorganisir secara ahli. Langkah-langkah dalam metode pengembangan perangkat lunak *Agile*:

1. *Planning*
2. *Design*
3. *Coding*
4. *Testing*

## 2.14 Penelitian Sebelumnya

**Tabel 2.2 Peneliti Terdahulu**

No	Judul	Algoritma	Akurasi	Dataset
1	Analisis Sentimen pada Postingan <i>Hate Speech</i> di media sosial menggunakan metode <i>K-Nearest Neighbor</i> . (Putra, 2022)	<i>K-Nearest Neighbour &amp; NLP (Natural Language Processing)</i>	56,74%	704 data
2	Implementasi <i>K-Nearest Neighbor</i> untuk Klasifikasi Bunga dengan Ekstraksi Fitur Warna Rgb. (Farokhah, 2020)	<i>K-Nearest Neighbor</i>	K=1 57%, K=2 64%, dan K=5 70%	320 data
3	<i>Application Of Data Mining To Classify Hate Speech On Social Media By Using The K-Nearest Neighbor Algorithm</i> . (Purba dkk., 2019)	<i>K-Nearest Neighbor</i>	-	-
4	Penerapan Algoritma <i>K-Nearest Neighbour</i> Untuk Memprediksi Kelulusan Mahasiswa. (Fricles Ariwisanto Sianturi, 2021)	<i>K-Nearest Neighbor</i>	-	215 data
5	Penerapan Algoritma <i>K-Nearest Neighbor</i> untuk Mendeteksi Ujaran Kebencian dan Bahasa Kasar pada Twitter Bahasa Indonesia. (Fadilah, 2021)	<i>K-Nearest Neighbor</i>	hate speech 79,13%, pada abusive 83,54% dan pada kelas level 73,56%	13.127 <i>tweet</i>
6	Klasifikasi Komentar <i>Bullying</i> pada <i>Instagram</i> Menggunakan Metode <i>K-Nearest Neighbor</i> . (Candra & Rozana, 2020)	<i>K-Nearest Neighbor</i>	87,07%	1000 data
7	Deteksi <i>Cyberbullying</i> pada Cuitan Media Sosial <i>Twitter</i> . (Abdulloh & Hidayatullah, 2020)	<i>K-Nearest Neighbor (KNN), Support Vector Machine with linear</i>	0.9616 <i>Multinomial Naïve Bayes</i> , 0,9949 <i>Logistic Regression</i> , 0.9975 <i>Support Vector Machine with</i>	1971 data

		<i>kernel (SVM), Logistic Regression, dan Multinomial Naïve Bayes.</i>	<i>linear kernel, dan 0.9188 K-Nearest Neighbor.</i>	
8	Perbandingan Algoritma SVM dan KNN dalam Mengklasifikasi Kelulusan Mahasiswa pada Suatu Mata Kuliah. (Shedriko, 2021)	<i>Support Vektor Machine (SVM) &amp; K-Nearest Neighbor</i>	<i>Support Vektor Machine 95,4% dan K-Nearest Neighbor 96,5%</i>	175 data
9	Perbandingan Algoritma Support Vector Machine dan K-Nearest Neighbors Pada Sinyal Tubuh Perokok. (Majid Rahardi, 2023)	<i>Support Vector Machine dan K-Nearest Neighbors</i>	<i>Support Vector Machine 84% dan K-Nearest Neighbors 75%</i>	55.692 data.
10	Analisis Perbandingan Algoritma SVM, KNN, dan CNN untuk Klasifikasi Citra Cuaca. (Naufal, 2021)	<i>Algoritma Support Vector Machine, K-Nearest Neighbors, dan Convolutional Neural Network.</i>	<i>Support Vector Machine 86%, K-Nearest Neighbors 76,6%, dan Convolutional Neural Network 94,2%</i>	Cloudy 298 Rain 214 Shine 251 Sunrise 357
11	Penerapan Algoritma Support Vector Machine Untuk Model Prediksi Kelulusan Mahasiswa Tepat Waktu. (Sheila Pramita Hervianti, 2021)	<i>Support Vector Machine (SVM)</i>	<i>Support Vector Machine (SVM) 94,4%</i>	2181 data
12	Penerapan SVM untuk Klasifikasi Sentimen pada Review Comment Berbahasa Indonesia di Online Shop. (Refo dkk., 2022)	<i>Support Vector Machine (SVM)</i>	88%	-
13	Penerapan Metode Support Vector Machine Dalam Klasifikasi Bunga Iris. (Desiani dkk., 2023)	<i>Support Vector Machine (SVM)</i>	96,7%	150 data
14	Perbandingan KNN Dan SVM Untuk Klasifikasi Kualitas Udara di Jakarta. (Jayadi Dkk., 2023)	<i>Support Vector Machine dan K-Nearest Neighbors</i>	96%	4383 data
15	Deteksi Cyberbullying pada Cuitan Media Sosial	<i>SVM</i>	<i>SVM paling tinggi dengan</i>	4029 data

	Twitter. (Abdulloh and Hidayatullah 2021)		<i>nilai masing - masing 0.997; 1.00; 1.00; 1.00.</i>	
--	---	--	---	--

Berdasarkan penelitian sebelumnya, algoritma K-Nearest Neighbor (KNN) telah terbukti efektif dalam klasifikasi teks dan pengenalan pola. Oleh karena itu, dalam penelitian ini, KNN digunakan untuk mendeteksi dan menentukan apakah suatu postingan termasuk dalam tindakan cyberbullying atau tidak.

