

BAB II

LANDASAN TEORI

2.1 Cyberbullying

Cyberbullying adalah perlakuan yang ditujukan untuk mempermalukan, menakut-nakuti, melukai, atau menyebabkan kerugian bagi pihak yang lemah dengan menggunakan sarana komunikasi Teknologi Informasi (Rahayu, 2012). Dengan kalimat lain, *cyberbullying* adalah tindakan perundungan yang dilakukan melalui internet dan kemajuan teknologi komunikasi.

Ada tiga hal yang membedakan tradisional *bullying* dengan *cyberbullying* (Akbar & Utari, 2016). Pertama, tradisional *bullying* merupakan tindakan yang dilakukan secara langsung bertatap muka (face-to-face), namun *cyberbullying* tidak. Kedua, *cyberbullying* tidak memberi dampak secara fisik, melainkan dampak yang ditimbulkan menyerang psikis korban. Ketiga, tidak seperti tradisional *bullying*, *cyberbullying* dapat muncul kapan saja tidak dibatasi oleh ruang dan waktu.

Bentuk perilaku *cyberbullying* yang dilakukan pelaku kepada korban seperti (Akbar & Utari, 2016):

- a. Pelaku mengirimkan komentar atau pesan *cyberbullying* berulang kali
- b. Pelaku mengirim komentar atau pesan *cyberbullying* dengan kata kasar
- c. Pelaku ikut bereaksi menambahkan pesan *cyberbullying* dari pelaku lainnya.

2.2 Machine Learning

Machine Learning atau pembelajaran mesin adalah sebuah metode untuk membuat sistem dapat belajar sendiri tanpa harus diprogram oleh manusia berulang kali. *Machine Learning* juga didefinisikan sebagai pengembangan dan penerapan model matematika dan statistik dengan penekanan pada penggunaan data untuk menemukan struktur model yang baik (McCoy & Auret, 2018). *Machine Learning* merupakan salah satu cabang dalam ilmu kecerdasan buatan (*Artificial Intelligence*). Agar bisa belajar sendiri, *Machine Learning* membutuhkan data-data sebagai data pelatihan (data *training*) lalu kemudian diuji dengan data baru.

2.3 Pengolahan Bahasa Alami

Pengolahan Bahasa Alami atau *Natural Language Processing* (NLP) adalah sebuah sub bidang dari kecerdasan buatan (*Artificial Intelligence*) dan merupakan hal penting dalam *Text Mining*. Tujuan dilakukannya PBA pada sistem adalah agar sistem bisa mengerti bahasa

manusia dengan menuliskannya dalam bentuk teks atau dokumen kemudian menjadikannya kalimat yang baku dan terstruktur sehingga sistem bisa dengan mudah memanipulasi dan memberikan respon yang semestinya. Adapun komponen-komponen utama dari PBA sebagai berikut (Soyusiawaty & Haspiyan, 2009):

a. *Parser*

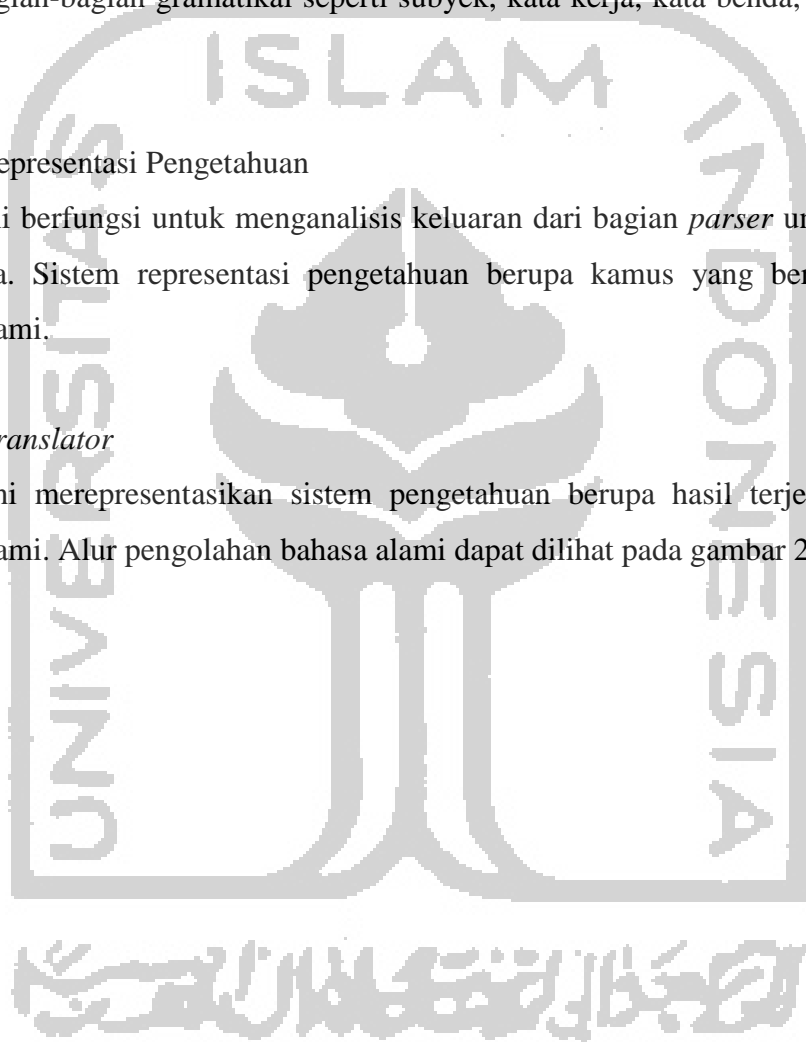
Bagian ini berfungsi untuk menerima masukan berupa teks kemudian menguraikannya dalam bagian-bagian gramatikal seperti subyek, kata kerja, kata benda, obyek, dan lain-lain.

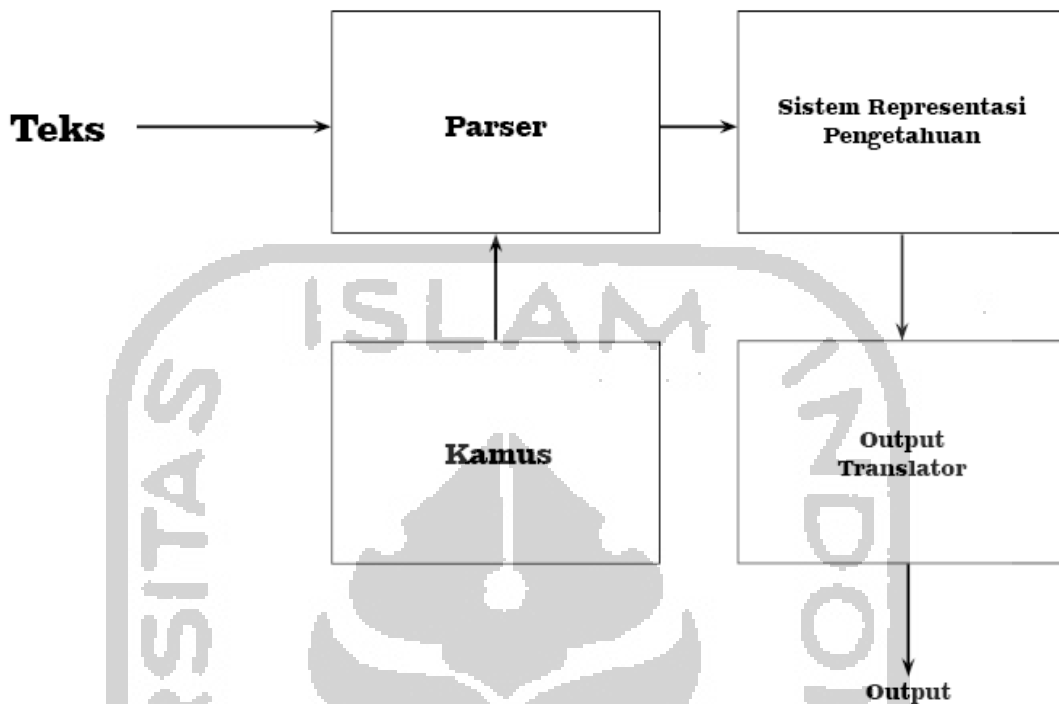
b. Sistem Representasi Pengetahuan

Bagian ini berfungsi untuk menganalisis keluaran dari bagian *parser* untuk menentukan maknanya. Sistem representasi pengetahuan berupa kamus yang berisikan kata-kata bahasa alami.

c. *Output Translator*

Bagian ini merepresentasikan sistem pengetahuan berupa hasil terjemahan masukan bahasa alami. Alur pengolahan bahasa alami dapat dilihat pada gambar 2.1.





Gambar 2.1 Alur pengolahan bahasa alami

2.4 *Sentiment Analysis*

Sentiment Analysis adalah proses mengolah sebuah data yang berbentuk tekstual secara otomatis dan memperoleh makna yang terkandung di dalamnya. *Sentiment Analysis* juga dimaknai sebagai proses yang bertujuan untuk menentukan isi dari *dataset* yang berbentuk teks bersifat positif, negatif, atau netral (Chandani, Wahono, & Purwanto, 2015).

Berdasarkan level sumber datanya *Sentiment Analysis* terbagi menjadi 2 kelompok besar yaitu :

a. *Coarse-grained Sentiment Analysis*

Pada *Sentiment Analysis* jenis ini, *Sentiment Analysis* yang dilakukan adalah pada level dokumen. Secara garis besar fokus utama dari *Sentiment Analysis* jenis ini adalah menganggap seluruh isi dokumen sebagai sebuah sentiment positif atau sentiment negatif (Clayton, 2011).

b. *Fined-grained Sentiment Analysis*

Fined-grained Sentiment Analysis adalah *Sentiment Analysis* pada level kalimat. Fokus utama *fined-grained Sentiment Analysis* adalah menentukan sentiment pada setiap kalimat pada suatu dokumen, dimana kemungkinan yang terjadi adalah terdapat sentiment pada level kalimat yang berbeda pada suatu dokumen (Clayton, 2011).

2.5 Text Mining

Text mining adalah salah satu teknik yang dapat digunakan untuk melakukan klasifikasi dimana, *text mining* merupakan variasi dari data mining yang berusaha menemukan pola yang menarik dari sekumpulan data tekstual yang berjumlah besar (Kurniawan, Effendi, & Sitompul, 2012). *Text mining* pada saat ini diterapkan pada berbagai bidang seperti :

a. *Information Extraction*

Identifikasi terhadap hubungan dan frase-frase kunci dalam teks dengan mencari urutan yang sudah ditetapkan dalam text menggunakan pencocokan pola.

b. *Topic Tracking*

Memprediksi dokumen-dokumen yang menjadi minat pengguna, berdasarkan dokumen yang menjadi minat pengguna.

c. *Summarization*

Meringkas sebuah dokumen menjadi dokumen yang ringkas dan padat.

d. *Clustering*

Melakukan pengelompokan terhadap dokumen-dokumen sejenis.

e. *Concept Linking*

Menghubungkan berbagai dokumen terkait dengan mengidentifikasi konsep yang digunakan bersama dan dengan demikian membantu para user untuk menemukan informasi yang barangkali mereka tidak akan temukan dengan menggunakan metode-metode pencarian tradisional.

f. *Question Answering*

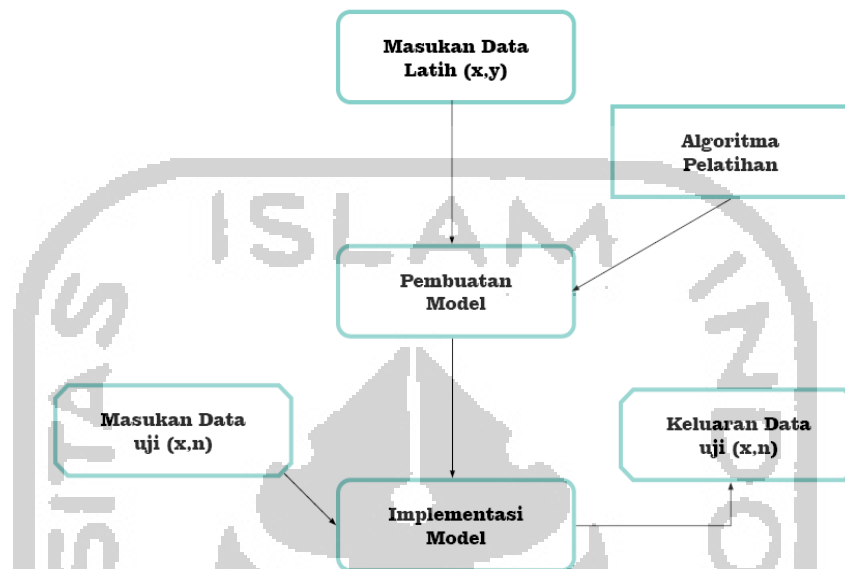
Menemukan jawaban terbaik pada pertanyaan yang diberikan melalui pencocokan pola berbasis pengetahuan.

2.6 Klasifikasi

Klasifikasi adalah proses pengkategorian terhadap sekumpulan dokumen menjadi beberapa kelas (Indriani, 2014). Dalam pembelajaran mesin, terdapat beberapa algoritma untuk

mengklasifikasikan data di antaranya *Naïve Bayes Classifier* (NBC), *Support Vector Machine* (SVM), *k-Nearest Neighbor* (KNN), dan lain-lain.

Secara umum proses klasifikasi dapat digambarkan pada gambar 2.2



Gambar 2.2 Alur klasifikasi

Pada gambar 2.2 proses yang terjadi adalah proses induksi dan deduksi. Induksi adalah proses pembuatan model berdasarkan data pelatihan. Sedangkan deduksi atau prediksi adalah proses penerapan model pada data uji untuk mengetahui kelasnya.

2.7 Multinomial Naïve Bayes

Model multinomial mengambil jumlah kata yang muncul pada sebuah dokumen, dalam model multinomial sebuah dokumen terdiri dari beberapa kejadian kata dan diasumsikan panjang dokumen tidak bergantung pada kelasnya. Dengan menggunakan asumsi Bayes yang sama bahwa kemungkinan tiap kejadian kata dalam sebuah dokumen adalah bebas tidak terpengaruh dengan konteks kata dan posisi kata dalam dokumen (Destuardi & Sumpeno, 2009).

Dalam *Multinomial Naïve Bayes*, dokumen urutan kejadian munculnya kata dalam dokumen tidak dipedulikan, jadi dokumen dianggap seperti “*bag of words*”, sehingga setiap kata diolah menggunakan distribusi multinomial (Fanissa, Fauzi, & Adinugroho, 2018).

Persamaan *Naïve Bayes* secara umum dapat dilihat pada persamaan berikut:

$$P(H|X) = \frac{P(H|X)P(H)}{P(X)} \quad (2.1)$$

Keterangan:

- X = Data dengan class yang belum diketahui
 H = Hipotesis data X merupakan suatu class spesifik
 P(H|X) = Probabilitas hipotesis H berdasarkan kondisi x (posteriori prob.)
 P(H) = Probabilitas hipotesis H (prior prob.)
 P(X|H) = Probabilitas X berdasarkan kondisi tersebut
 P(X) = Probabilitas dari X.

2.8 Performance Evaluation Measure

Performance Evaluation Measure (PEM) adalah tahapan yang digunakan untuk mengetahui performa model. Ada banyak persamaan untuk menghitung nilai PEM. Beberapa persamaan dalam PEM antara lain:

a. *Precision*

Adalah ketepatan nilai antara permintaan pengguna dengan respon sistem.

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

b. *Accuracy*

Adalah perbandingan antara informasi benar yang dijawab sistem dengan keseluruhan data.

$$Accuracy = \frac{TN + TP}{TP + TN + FP + FN} \quad (2.3)$$

c. *Recall*

Adalah ketepatan antara informasi yang sama dengan informasi yang pernah dipanggil sebelumnya.

$$Recall = \frac{TP}{TP + FN} \quad (2.4)$$

d. *F1-Score*

Adalah perbandingan rata-rata presisi dan recall yang dibobotkan

$$F1 - Score = \frac{2 \times Precision \times Recall}{Precision + Recall} \quad (2.5)$$

PEM biasanya direpresentasikan dengan sebuah *confusion matrix*, yaitu matriks yang sangat intuitif dalam memberi informasi performa model.

		Predicted Values	
		Positive (0)	Negative (1)
Actual Values	Positive (0)	TP	FP
	Negative (1)	FN	TN

Gambar 2.3 *Confusion Matrix*

Keterangan:

- True Positive (TP) = Jumlah data kelas positif diprediksi benar sebagai kelas positif
- False Negative (FN) = Jumlah data kelas positif diprediksi salah sebagai kelas negatif
- True Negative (TN) = Jumlah data kelas negatif diprediksi benar sebagai kelas negatif
- False Positive (FP) = Jumlah data kelas negatif diprediksi salah sebagai kelas positif

2.9 Django

Django adalah sebuah *framework* atau kerangka kerja yang digunakan untuk pengembangan *website* yang dibangun menggunakan bahasa *Python*. Dalam *Django* model pemrograman yang digunakan adalah MVT (*Model, View, Template*). Seperti halnya pada *framework Php*, *model* berperan untuk berinteraksi dengan basis data. Perbedaan *Django* dengan *framework Php* seperti *Codeigniter* dan *Laravel* adalah pada layer *View* dan *Template*. *View* pada *Codeigniter* atau *Laravel* berhubungan dengan desain antarmuka, sedangkan pada *Django* layer *view*

mengakomodasi logika pemrograman yang menghubungkan *layer Model* dan *layer Template*. *Layer Template* memiliki peran yang sama dengan *layer view* pada *framework Php* yaitu interaksi sistem dengan pengguna melalui antarmuka sistem.

2.10 Penelitian serupa

Penelitian tentang klasifikasi teks secara umum sudah banyak dilakukan oleh peneliti-peneliti sebelumnya. Xiang et al. melakukan penelitian tentang ujaran kebencian (*Hate Speech*) pada media sosial *Twitter* dengan membagi *dataset* menjadi dua kelas, yaitu kelas *hate speech* dan kelas *non-hatespeech* kemudian menguji data dengan mengklasifikasikannya dengan menggunakan algoritma *Logistic Regression* dengan persentase nilai *True Positive* (TP) 75.1% dari total 4029 baris data cuitan (Xiang, Fan, Wang, Hong, & Rose, 2012).

Klasifikasi konten kasar pada cuitan media sosial *Twitter* dengan korpus berbahasa Indonesia juga pernah dilakukan. Hidayatullah mengklasifikasi cuitan menjadi dua kelas dan melakukan komparasi terhadap performa algoritma *Multinomial Naïve Bayes* dan *SVM* dalam melakukan klasifikasi (Hidayatullah, Yusuf, Juwairi, & Nayoan, 2019).

Penelitian tentang deteksi *cyberbullying* pada media sosial sudah pernah dilakukan sebelumnya. Homa Hosseinmardi melakukan penelitian tentang *cyberbullying* pada media sosial Instagram dengan memanfaatkan foto dan komentarnya, kemudian diklasifikasikan menggunakan NBC dan Linear SVM (Hosseinmardi, et al., 2015). Penelitian tentang deteksi *cyberbullying* yang dipadukan dengan pendekatan psikologi juga pernah dilakukan sebelumnya. Model dibangun dengan menggunakan model Big Five and Triad dan algoritma *Random Forest* (Balakrishnan, Khan, Fernandez, & Arabnia, 2019).

Hal yang menjadi pembeda dari penelitian – penelitian serupa sebelumnya adalah di samping membuat model klasifikasi, penelitian ini juga memiliki luaran berupa sebuah aplikasi deteksi *cyberbullying*.