# BAB 2
# LANDASAN TEORI

## 2.1 Ujaran Kebencian

Ujaran kebencian (*hate speech*) adalah segala bentuk komunikasi dalam ucapan, tulisan, dan juga perilaku yang bersifat menyerang atau merendahkan individu atau kelompok berdasarkan identitas mereka, seperti ras, agama, etnis, gender, kebangsaan, warna kulit, keturunan, atau faktor identitas lainnya [5, 25]. Dalam konteks penelitian deteksi otomatis, ujaran kebencian didefinisikan sebagai konten yang berpotensi mendorong perasaan benci, permusuhan, atau diskriminasi terhadap individu maupun kelompok berdasarkan atribut identitas tertentu [25].

Dalam kerangka hukum Indonesia, ujaran kebencian dijelaskan dalam Surat Edaran Kapolri No. SE/6/X/2015 sebagai keseluruhan perbuatan yang bersifat penghinaan, pencemaran nama baik, penistaan, provokasi, penghasutan, atau penyebaran berita bohong dan semua tindakan lain yang bertujuan atau berdampak pada tindak diskriminasi, kekerasan, penghilangan nyawa, dan/atau konflik sosial [26]. Selain itu, ujaran kebencian juga diatur dalam Undang-Undang Nomor 19 Tahun 2016 tentang Informasi dan Transaksi Elektronik (UU ITE), yang mengatur penyebaran konten kebencian melalui media elektronik termasuk platform media sosial [27].

Ujaran kebencian berdampak pada aspek psikologis, sosial, dan hukum. Dari perspektif psikologis, individu yang menjadi sasaran ujaran kebencian rentan mengalami tekanan psikologis, trauma berkepanjangan, serta perasaan terisolasi dari lingkungan sosialnya [25]. Paparan ujaran kebencian yang terus-menerus juga dapat menormalisasi kebencian dan memicu kemarahan ekstrem yang berpotensi berujung pada kekerasan [5].

Secara sosial, ujaran kebencian yang tidak ditangani dapat menciptakan polarisasi, diskriminasi sistemik, dan konflik antarkelompok, yang di platform media sosial menyebar jauh lebih cepat dibandingkan media konvensional [5]. Secara hukum di Indonesia, dampak tersebut direspons melalui SE Kapolri SE/6/X/2015 dan UU ITE Nomor 19 Tahun 2016, yang menegaskan bahwa ujaran kebencian yang tidak ditangani berpotensi menimbulkan konflik sosial dan dapat dikenai sanksi pidana [26, 27].

## 2.2 Media Sosial Platform X

Media sosial merupakan platform digital yang memungkinkan pengguna untuk membuat, berbagi, dan bertukar informasi, gagasan, serta konten kepada orang luas melalui jaringan virtual [5]. Kehadiran media sosial mengubah cara masyarakat berkomunikasi secara signifikan, juga membuka ruang bagi penyebaran konten negatif termasuk ujaran kebencian [25].

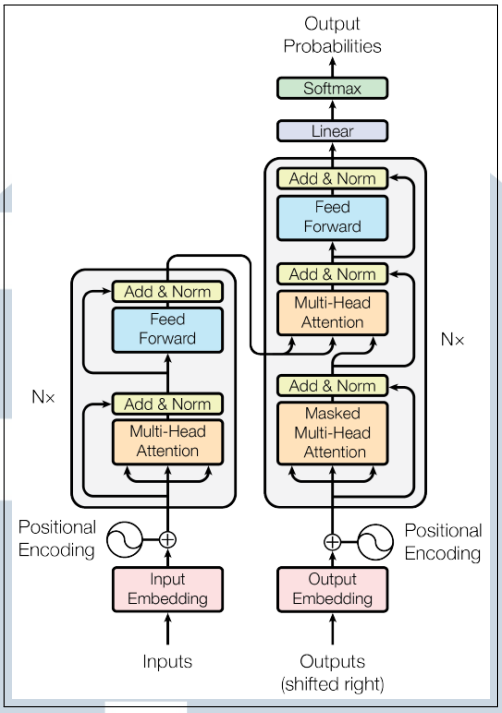
X yang sebelumnya dikenal sebagai Twitter merupakan platform media sosial berbasis *microblogging* yang memungkinkan pengguna menyampaikan pendapat secara terbuka dalam bentuk teks singkat, gambar, maupun video. Karakteristik komunikasi yang cepat, publik, dan terbuka menjadikan X sebagai salah satu platform paling aktif sekaligus rentan terhadap penyebaran ujaran kebencian [25].

Teks yang dikumpulkan di X memiliki karakteristik linguistik yang khas, meliputi penggunaan tata bahasa informal, singkatan, *slang*, dan variasi ejaan yang tidak baku [16]. Karakteristik tersebut menjadi tantangan tersendiri dalam pemrosesan bahasa alami karena model yang dilatih pada korpus formal cenderung tidak mampu menangani variasi bahasa informal secara optimal [16, 25].

## 2.3 Transformer

*Transformer* adalah arsitektur *deep learning* yang diperkenalkan [28] dan menjadi fondasi bagi berbagai model bahasa modern. Berbeda dari arsitektur berbasis *recurrent neural network* (RNN) yang memproses token secara sekuensial, *transformer* memproses seluruh token secara paralel menggunakan mekanisme *attention*, sehingga lebih efisien secara komputasi dan mampu menangkap dependensi jangka panjang dalam teks [28].

Arsitektur *transformer* terdiri atas dua komponen utama, yaitu *encoder* dan *decoder*, seperti yang ditampilkan pada Gambar 2.1. *Encoder* bertugas untuk menghasilkan representasi kontekstual dari urutan input, sedangkan *decoder* bertugas untuk menghasilkan urutan output berdasarkan representasi tersebut [28]. Komponen inti dari kedua bagian tersebut adalah mekanisme *self-attention*.




Gambar 2.1. Arsitektur *transformer*

Sumber: [28]

Mekanisme *self-attention* memungkinkan setiap token dalam urutan memperhatikan seluruh token lain, sehingga representasi setiap token dipengaruhi oleh konteks keseluruhan kalimat [28]. Perhitungan *attention* dirumuskan sebagaimana ditunjukkan pada Persamaan 2.1.

$$\text{Attention}(Q,K,V) = \text{softmax}\left(\frac{QK^{\top}}{\sqrt{d_k}}\right)V \tag{2.1}$$

Pada Persamaan 2.1, $Q$, $K$, dan $V$ masing-masing merepresentasikan matriks *query*, *key*, dan *value* yang diturunkan dari representasi input, sedangkan $d_k$ adalah dimensi vektor *key* yang berfungsi sebagai faktor skala untuk menjaga kestabilan gradien pada nilai *softmax* [28].

Mekanisme ini dijalankan secara paralel sebanyak $h$ kepala (*heads*) dalam bentuk *multi-head attention*, sehingga setiap kepala dapat menangkap pola relasi token yang berbeda, sebagaimana dirumuskan pada Persamaan 2.2 [28].

$$\text{MultiHead}(Q,K,V) = \text{concat}(\text{head}_1,\ldots,\text{head}_h)\,W^{O} \tag{2.2}$$

Setiap $\text{head}_i$ pada Persamaan 2.2 dihitung sebagai $\text{head}_i = \text{Attention}(QW_i^Q, KW_i^K, VW_i^V)$, dengan $W_i^Q$, $W_i^K$, $W_i^V$, serta $W^O$ merupakan matriks bobot yang dapat dipelajari [28].

## 2.4 Bidirectional Encoder Representations from Transformers (BERT)

BERT (*Bidirectional Encoder Representations from Transformers*) merupakan model bahasa berbasis arsitektur *Transformer encoder* yang diperkenalkan oleh Devlin et al. [29]. Berbeda dengan model bahasa sekuensial sebelumnya yang memproses teks secara satu arah, BERT mempelajari representasi kontekstual secara dua arah (*bidirectional*) dengan mempertimbangkan hubungan antara token di sebelah kiri dan kanan secara bersamaan. Pendekatan ini memungkinkan model menghasilkan representasi yang lebih kaya terhadap makna kata dalam suatu kalimat.

BERT dibangun hanya menggunakan tumpukan lapisan *Transformer encoder*. Setiap lapisan encoder terdiri atas mekanisme *multi-head self-attention* dan *feed-forward neural network*. Melalui mekanisme *self-attention*, setiap token dapat memberikan perhatian (*attention*) kepada seluruh token lain dalam urutan masukan, sehingga model mampu menangkap hubungan kontekstual baik jarak dekat maupun jarak jauh secara lebih efektif dibandingkan model sekuensial tradisional.

### 2.4.1 Tokenisasi WordPiece

BERT menggunakan metode tokenisasi berbasis *WordPiece*, yaitu teknik *subword tokenization* yang memecah kata menjadi unit-unit yang lebih kecil apabila kata tersebut tidak ditemukan secara utuh dalam kosakata model [29]. Pendekatan ini dirancang untuk mengatasi permasalahan kata di luar kosakata (*out-of-vocabulary*) dan memungkinkan model memahami variasi morfologi suatu bahasa.

Sebagai contoh, kata:

```
bermainlah
```

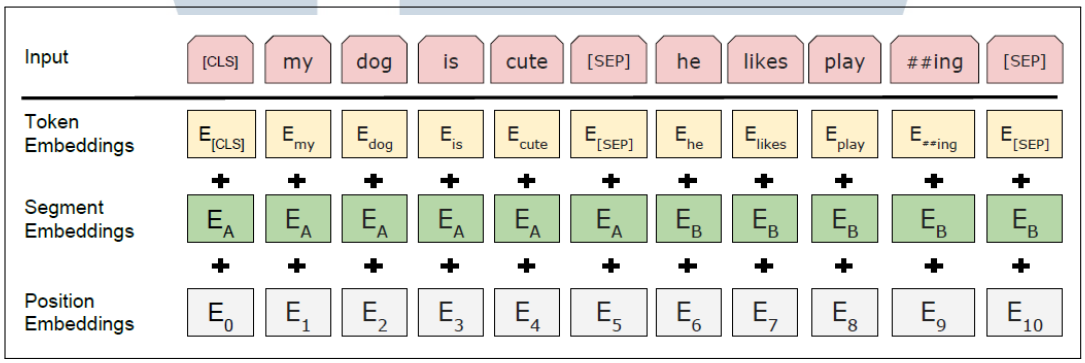
dapat dipecah menjadi:

```
bermain + ##lah
```

Awalan ## menunjukkan bahwa token tersebut merupakan lanjutan dari token sebelumnya dan bukan awal kata baru. Dengan memanfaatkan unit *subword*, model dapat mengenali pola morfologis dan memperoleh representasi yang lebih baik terhadap kata-kata yang jarang muncul maupun kata baru yang tidak ditemukan secara utuh pada korpus pelatihan.

### 2.4.2 Representasi Masukan BERT

Sebelum diproses oleh lapisan encoder, setiap token masukan direpresentasikan sebagai penjumlahan dari tiga komponen embedding, yaitu *token embedding*, *segment embedding*, dan *position embedding* [29]. Representasi masukan BERT ditunjukkan pada Gambar 2.2.




Gambar 2.2. Representasi masukan pada BERT

Sumber: [29]

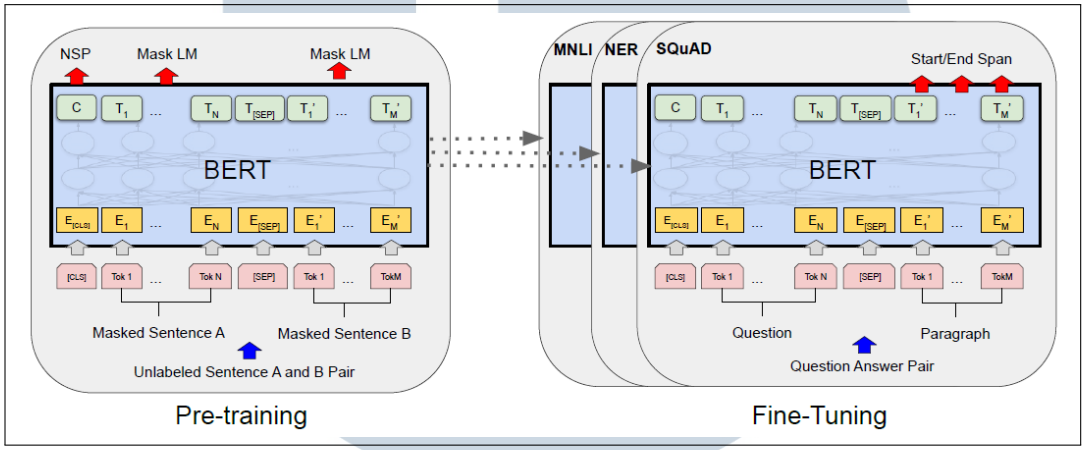
Ketiga komponen tersebut dijelaskan sebagai berikut.

1. *Token Embedding* mengubah token hasil tokenisasi *WordPiece* menjadi representasi vektor berdasarkan kosakata model [29].
2. *Segment Embedding* memberikan informasi mengenai segmen atau kalimat asal setiap token [29].
3. *Position Embedding* menambahkan informasi posisi token dalam urutan teks [29].

### 2.4.3 Pre-training BERT

BERT dilatih menggunakan dua objektif pelatihan utama. Pertama, *Masked Language Modeling* (MLM), di mana sebagian token dipilih secara acak lalu

disembunyikan, memaksa model untuk menebak kata tersebut menggunakan konteks dua arah. Kedua, *Next Sentence Prediction* (NSP), model menerima pasangan kalimat dan menentukan apakah kalimat kedua adalah kelanjutan dari kalimat pertama secara logis. Seperti yang divisualisasikan pada bagian bawah kiri Gambar 2.3 dengan *Masked Sentence A* dan *Masked Sentence B*. Kombinasi keduanya memungkinkan BERT mempelajari representasi kontekstual pada level kata sekaligus relasi semantik antar kalimat.




Gambar 2.3. Gambaran *pre-training* dan *fine-tuning* BERT
Sumber: [29]

Pada *Masked Language Modeling*, sebagian token pada masukan diganti dengan token khusus `[MASK]`. Tugas model adalah memprediksi token asli berdasarkan konteks di kedua sisi token yang disembunyikan. Sementara itu, *Next Sentence Prediction* bertujuan mempelajari hubungan antar kalimat dengan menentukan apakah kalimat kedua benar-benar mengikuti kalimat pertama pada dokumen asli [29].

Melalui kombinasi MLM dan NSP, BERT memperoleh representasi bahasa yang kaya secara semantik dan kontekstual sehingga dapat digunakan sebagai model dasar untuk berbagai tugas pemrosesan bahasa alami.

### 2.4.4 Fine-tuning BERT

Setelah *pre-training*, BERT dapat disesuaikan untuk berbagai tugas NLP melalui proses *fine-tuning*. Pada tahap ini ditambahkan *task-specific layer* pada bagian keluaran model, sementara bobot hasil *pre-training* digunakan sebagai titik awal pelatihan. Seluruh parameter model kemudian diperbarui menggunakan

dataset yang sesuai dengan tugas yang akan diselesaikan [29].

Dalam tugas klasifikasi teks, representasi token khusus `[CLS]` digunakan sebagai representasi keseluruhan urutan masukan. Representasi tersebut diteruskan ke lapisan klasifikasi untuk menghasilkan prediksi kelas. Ilustrasi proses *fine-tuning* untuk berbagai tugas ditunjukkan pada bagian kanan Gambar 2.3. Pada penelitian ini, proses *fine-tuning* dilakukan untuk tugas klasifikasi ujaran kebencian dengan dua kelas, yaitu `HS` dan `Non-HS`.

## 2.5 IndoBERT

IndoBERT merupakan model bahasa berbasis BERT yang dilatih khusus pada korpus besar berbahasa Indonesia [15]. Model ini menggunakan arsitektur BERT-Base dengan 12 lapisan *transformer* dan 110 juta parameter, serta dilatih pada korpus Indo4B yang berisi sekitar 4 miliar kata (±250 juta kalimat) dari berbagai sumber publik seperti berita, media sosial, Wikipedia, dan situs web berbahasa Indonesia [15]. Proses pra-pelatihan dilakukan menggunakan objektif *masked language modeling*, sehingga model mampu menghasilkan representasi kata yang kontekstual dengan mempertimbangkan kata pada kedua arah dalam sebuah kalimat. Dalam penggunaannya, IndoBERT dimanfaatkan melalui mekanisme *transfer learning*, yaitu pengetahuan dari proses pra-pelatihan disesuaikan kembali (*fine-tuning*) untuk tugas hilir tertentu, termasuk klasifikasi teks.

## 2.6 IndoBERTweet

IndoBERTweet merupakan model *transformer* dengan arsitektur yang sama seperti IndoBERT, tetapi diadaptasi secara khusus untuk domain Twitter (sekarang X) berbahasa Indonesia [16]. Model ini dikembangkan dengan memperluas IndoBERT melalui penambahan kosakata khusus domain, kemudian dilatih kembali pada korpus berisi sekitar 409 juta token kata yang bersumber dari tweet berbahasa Indonesia [16]. Kosakata baru tersebut diinisialisasi menggunakan rata-rata *embedding subword* BERT, sehingga proses pra-pelatihan menjadi lebih efisien dibandingkan pelatihan dari awal. Adaptasi ini membuat IndoBERTweet lebih mampu menangani karakteristik teks media sosial seperti singkatan, *slang*, dan tata bahasa informal yang jarang ditemukan dalam korpus formal. IndoBERTweet menunjukkan peningkatan performa dibandingkan IndoBERT pada berbagai tugas

NLP berbasis data Twitter berbahasa Indonesia [16].

## 2.7 Dataset

Dataset yang digunakan dalam penelitian ini adalah *Multi-label Hate Speech and Abusive Language Detection in Indonesian Twitter* yang dikembangkan oleh Ibrohim dan Budi [9]. Dataset ini terdiri dari 13.169 *tweet* berbahasa Indonesia yang telah dianotasi secara manual [9]. Dataset tersedia secara publik melalui repositori GitHub milik Ibrohim dan Budi.

Proses anotasi dataset dirancang untuk menjaga validitas label. Pedoman anotasi disusun berdasarkan hasil *Focus Group Discussion* dengan Direktorat Tindak Pidana Siber Bareskrim Polri selaku lembaga yang menangani penyidikan kejahatan siber di Indonesia, kemudian divalidasi melalui diskusi dengan ahli linguistik agar pedoman tersebut valid dan mudah dipahami oleh anotator [9]. Anotasi melibatkan 30 anotator dengan latar belakang agama, etnis, usia, dan pekerjaan yang beragam untuk mengurangi bias, mengingat anotasi ujaran kebencian bersifat subjektif [9, 25]. Setiap tweet dianotasi oleh tiga anotator, dengan label tahap pertama ditentukan melalui teknik *100% agreement* dan tahap berikutnya melalui *majority voting*, dan tingkat kesepakatan yang diperoleh menunjukkan hasil anotasi memiliki tingkat persetujuan yang baik [9].

Setiap *tweet* dalam dataset dilabeli dengan kolom `HS` sebagai label biner ujaran kebencian (1 untuk ujaran kebencian, 0 untuk bukan ujaran kebencian), kolom `Abusive` untuk konten kasar, kolom target yang mencakup `HS_Individual`, `HS_Group`, `HS_Religion`, `HS_Race`, `HS_Physical`, `HS_Gender`, dan `HS_Other`, serta kolom tingkat keparahan yang terdiri dari `HS_Weak`, `HS_Moderate`, dan `HS_Strong` [9]. Distribusi anotasi pada setiap kolom label ditunjukkan pada Tabel 2.1.

Tabel 2.1. Distribusi Anotasi Bernilai 1 pada Setiap Kolom Label

| Kolom Label | Jumlah Tweet |
|---|---|
| HS | 5.561 |
| Abusive | 5.043 |
| HS_Individual | 3.575 |
| HS_Group | 1.986 |
| HS_Other | 3.740 |
| HS_Religion | 793 |
| HS_Race | 566 |
| HS_Physical | 323 |
| HS_Gender | 306 |
| HS_Weak | 3.383 |
| HS_Moderate | 1.705 |
| HS_Strong | 473 |

Berdasarkan tabel tersebut, dari 5.561 tweet ujaran kebencian, sebagian besar menargetkan individu (3.575 tweet) dibandingkan kelompok (1.986 tweet). Berdasarkan kategorinya, kategori terbesar adalah makian atau fitnah umum (`HS_Other`) sebanyak 3.740 tweet yang tidak berbasis identitas tertentu, sedangkan kategori berbasis identitas didominasi agama/kepercayaan (793 tweet), diikuti ras/etnis (566 tweet), fisik/disabilitas (323 tweet), dan gender/orientasi seksual (306 tweet) [9]. Kategori makian atau fitnah umum dimasukkan sebagai ujaran kebencian mengikuti pedoman anotasi hasil *Focus Group Discussion* karena berpotensi memicu diskriminasi dan konflik sosial, meskipun sebagian peneliti mengategorikan makian yang tidak berbasis identitas sebagai ujaran ofensif biasa dan bukan ujaran kebencian [25].

## 2.8 Metrik Evaluasi

Evaluasi performa model dilakukan menggunakan *Confusion matrix* memetakan hasil prediksi terhadap label sebenarnya ke dalam empat kategori sebagaimana ditunjukkan pada Tabel 2.2, yaitu *true positive* (TP) untuk ujaran kebencian yang diprediksi dengan benar, *true negative* (TN) untuk bukan ujaran kebencian yang diprediksi dengan benar, *false positive* (FP) untuk bukan ujaran kebencian yang keliru diprediksi sebagai ujaran kebencian, dan *false negative* (FN)

untuk ujaran kebencian yang keliru diprediksi sebagai bukan ujaran kebencian [30].

Dari keempat nilai tersebut diturunkan empat metrik evaluasi. *Accuracy* mengukur proporsi prediksi yang benar terhadap seluruh data, sebagaimana ditunjukkan pada Persamaan 2.3.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{2.3}$$

*Precision* mengukur proporsi prediksi ujaran kebencian yang benar terhadap seluruh prediksi ujaran kebencian, sebagaimana ditunjukkan pada Persamaan 2.4.

$$Precision = \frac{TP}{TP + FP} \tag{2.4}$$

*Recall* mengukur proporsi ujaran kebencian yang berhasil dideteksi terhadap seluruh ujaran kebencian yang sebenarnya, sebagaimana ditunjukkan pada Persamaan 2.5.

$$Recall = \frac{TP}{TP + FN} \tag{2.5}$$

*F1-score* merupakan rata-rata harmonis antara *precision* dan *recall*, sebagaimana ditunjukkan pada Persamaan 2.6.

$$F1\text{-}score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \tag{2.6}$$

*F1-score* digunakan sebagai metrik utama karena memberikan gambaran yang seimbang antara *precision* dan *recall*, sehingga lebih sesuai untuk data dengan distribusi kelas yang tidak seimbang [30].

Tabel 2.2. Confusion matrix untuk klasifikasi ujaran kebencian

| | | **Prediksi** | |
|---|---|---|---|
| | | **HS** | **Non-HS** |
| **Aktual** | **HS** | TP | FN |
| | **Non-HS** | FP | TN |

Seluruh metrik yang diturunkan dari *confusion matrix* dihitung pada satu titik ambang keputusan (*threshold*). Pada klasifikasi biner, model menghasilkan

probabilitas kelas melalui fungsi *softmax*, dan prediksi ditetapkan pada kelas dengan probabilitas tertinggi, yang setara dengan penggunaan *threshold* 0,5. Perubahan nilai *threshold* mengubah komposisi TP, TN, FP, dan FN sehingga seluruh metrik turut berubah [31]. Oleh karena itu, evaluasi dilengkapi dengan kurva *Receiver Operating Characteristic* (ROC) yang memetakan *true positive rate* (TPR) terhadap *false positive rate* (FPR) pada seluruh nilai *threshold* [31]. TPR dapat dihitung menggunakan Persamaan 2.7 dan FPR dihitung sebagaimana ditunjukkan pada Persamaan 2.8.

$$TPR = \frac{TP}{TP + FN} \tag{2.7}$$

$$FPR = \frac{FP}{FP + TN} \tag{2.8}$$

*Area Under the Curve* (AUC) merupakan luas di bawah kurva ROC yang merangkum kemampuan model membedakan kedua kelas tanpa bergantung pada satu titik *threshold*, dengan nilai 0,5 setara dengan tebakan acak dan nilai 1 menunjukkan pemisahan sempurna [31].

## 2.9 Explainable AI (XAI) dan LIME

*Explainable Artificial Intelligence* (XAI) merupakan pendekatan dalam pengembangan sistem kecerdasan buatan yang bertujuan untuk membuat keputusan model dapat dipahami oleh manusia [22]. Kebutuhan akan XAI muncul karena model *machine learning* modern, khususnya model berbasis *deep learning*, bersifat *black box*, sehingga model tersebut menghasilkan prediksi yang akurat namun sulit dijelaskan proses pengambilan keputusannya [22].

Arrieta et al. [22] membedakan pendekatan XAI berdasarkan dua dimensi utama. Dimensi pertama adalah transparansi model, yang membedakan model yang secara inheren dapat dipahami (*transparent by design*) dari model yang memerlukan teknik tambahan untuk dijelaskan. Dimensi kedua adalah cakupan penjelasan, yang terbagi menjadi penjelasan *global* yang mendeskripsikan perilaku model secara keseluruhan, dan penjelasan *local* yang menjelaskan prediksi pada satu instance tertentu [22]. Teknik XAI yang diterapkan setelah model selesai dilatih disebut teknik *post-hoc*, karena penjelasan dihasilkan secara terpisah dari proses pelatihan itu sendiri [22].

LIME merupakan teknik XAI *post-hoc* yang dikembangkan oleh Ribeiro et al. [21] untuk menjelaskan prediksi model secara lokal. LIME bekerja dengan menghasilkan sampel perturbasi di sekitar *instance* yang ingin dijelaskan, kemudian melatih model aproksimasi linier sederhana pada sampel tersebut untuk mengidentifikasi fitur-fitur yang paling berpengaruh terhadap prediksi [21].

Secara formal, penjelasan yang dihasilkan LIME untuk suatu *instance* $x$ diperoleh dengan menyelesaikan persamaan optimasi pada Persamaan 2.9 [21].

$$\xi(x) = \arg\min_{g \in G} \mathscr{L}(f, g, \pi_x) + \Omega(g) \tag{2.9}$$

Pada Persamaan 2.9, $f$ adalah model *black box* yang ingin dijelaskan, $G$ adalah keluarga model penjelasan (model linier), $\mathscr{L}(f, g, \pi_x)$ adalah fungsi kehilangan yang mengukur ketidaksetiaan $g$ dalam mengaproksimasi $f$ pada lokalitas yang didefinisikan oleh $\pi_x$, dan $\Omega(g)$ adalah ukuran kompleksitas model penjelasan $g$ [21]. Fungsi kedekatan $\pi_x$ didefinisikan sebagai kernel eksponensial pada Persamaan 2.10.

$$\pi_x(z) = \exp\left(\frac{-D(x,z)^2}{\sigma^2}\right) \tag{2.10}$$

Pada Persamaan 2.10, $D(x,z)$ adalah fungsi jarak antara *instance* asli $x$ dan sampel perturbasi $z$ (misalnya jarak kosinus pada teks), dan $\sigma$ adalah lebar *kernel* [21].

Pada klasifikasi teks, LIME bekerja dengan menghilangkan kata-kata secara acak dari teks input untuk membentuk sampel perturbasi. Setiap sampel dimasukkan ke model $f$ untuk mendapatkan prediksi. Bobot setiap sampel ditentukan berdasarkan kedekatannya dengan *instance* menggunakan $\pi_x$. Model linier sederhana kemudian dilatih pada sampel berbobot tersebut, sehingga koefisien yang dihasilkan merepresentasikan kontribusi setiap kata terhadap prediksi model [21].

LIME bersifat *model-agnostic*, artinya dapat diterapkan pada model apapun tanpa memerlukan akses ke struktur internal model tersebut [21]. Sifat ini menjadikan LIME sangat fleksibel dan dapat digunakan untuk menjelaskan model-model kompleks seperti *deep learning* yang tidak transparan secara inheren [22]. Ibrahim et al. membuktikan bahwa LIME efektif digunakan untuk mengevaluasi kualitas keputusan model deteksi ujaran kebencian berbahasa Indonesia [13].