BAB II

TINJAUAN PUSTAKA

2.1 Landasan Teori

2.1.1 Analisis Sentimen

Analisis sentimen merupakan proses mengidentifikasi, mengekstraksi, dan mengklasifikasikan opini atau emosi dalam teks untuk menentukan kecenderungan sikap seseorang terhadap suatu topik, produk, atau kebijakan (Irma Surya Kumala Idris dkk., 2023). Tujuan utama analisis sentimen adalah mengelompokkan teks menjadi kategori seperti positif, negatif dan netral (Mahira dkk., 2023). Sentimen positif ditandai oleh kata-kata yang mengandung apresiasi, dukungan, atau kepuasan, seperti *bagus*, *baik*, *hebat*, *mantap*, *memuaskan*, *berhasil*, *bermanfaat*, *efektif*, *unggul*, *layak*, *terbaik*, *luar biasa*, *senang*, *suka*, dan *setuju(Larasati dkk., 2025)*. Sebaliknya, sentimen negatif ditandai oleh kata-kata yang menunjukkan ketidakpuasan, kritik, atau penolakan, seperti buruk, jelek, kecewa, gagal, merugikan, tidak efektif, kurang baik, lemah, mengecewakan, marah, kesal, benci, tidak setuju, parah, dan bermasalah (Tenri Ribi Farhana & Sulistyowati, 2025). Sementara itu, sentimen netral merupakan pernyataan yang tidak menunjukkan kecenderungan positif maupun negatif secara jelas, melainkan hanya menyampaikan informasi atau fakta, contohnya adalah kata atau frasa seperti program, kebijakan, pelaksanaan, masyarakat, sekolah, pemerintah, data menunjukkan, berdasarkan laporan, telah dilaksanakan, dan akan dilakukan yang umumnya bersifat informatif tanpa mengandung penilaian tertentu (Prastyo dkk., 2024). Dalam konteks media sosial, analisis sentimen digunakan untuk memahami

respons masyarakat terhadap suatu isu atau kebijakan publik, termasuk dalam penelitian ini yang berfokus pada tanggapan masyarakat terhadap Program Makan Bergizi Gratis. Menurut (Azzahra' dkk., 2025). Analisis sentimen pada media sosial, khususnya X (twitter), menghadapi tantangan tersendiri karena pengguna sering mengekspresikan pendapat dengan bahasa informal, singkatan, emotikon, hingga unsur sarkasme.

2.1.2 Sarkasme

Sarkasme merupakan bentuk ekspresi bahasa yang menyampaikan makna berlawanan dengan kata-kata yang digunakan, sering kali dengan nada sindiran ejekan dan olok-olok (Sari dkk., 2023). Dalam konteks analisis sentimen, sarkasme menjadi tantangan karena teks yang tampak positif secara leksikal bisa memiliki makna negatif secara kontekstual Misalnya, kalimat "Mantap Program nya, bikin harga pangan naik" secara kata tampak positif, namun mengandung nada sinis. Deteksi sarkasme penting dilakukan agar model analisis sentimen tidak salah mengklasifikasikan pernyataan yang sebenarnya negatif menjadi positif. Untuk itu, model yang digunakan harus mampu memahami konteks, emosi tersembunyi, serta hubungan antar kata dalam satu kalimat. Klasifikasi sarkasme dalam penelitian ini dibagi menjadi dua kategori utama, yaitu sarkasme positif dan sarkasme negatif, sebagaimana ditunjukkan pada Tabel 2.1 Sarkasme Positif dan Negatif.

Tabel 2.1 Sarkasme Positif dan Negatif

Jenis Sarkasme	**Definisi**	**Karakteristik Utama**	**Contoh Kalimat**
Sarkasme Positif	Sarkasme yang secara tersirat	- Mengandung ironi halus	"Wah, program MBG ini benar-

Jenis Sarkasme	Definisi	Karakteristik Utama	Contoh Kalimat
	menyampaikan makna positif atau pujian, meskipun menggunakan kata atau struktur kalimat yang tampak negatif atau berlawanan. Sarkasme dapat digunakan untuk menyampaikan pujian secara tersirat (sarcastic praise), meskipun bentuk literalnya tampak negatif (Du dkk., 2024)	- Makna sebenarnya bersifat mendukung atau memuji - Sering digunakan untuk humor atau sindiran ringan	benar ‘gagal’, sampai semua anak kenyang.”
Sarkasme Negatif	Sarkasme yang menyampaikan kritik atau ketidakpuasan secara tidak langsung dengan menggunakan kata-kata yang tampak positif di permukaan (Mutiasari dkk., 2025)	- Mengandung ironi tajam - Makna sebenarnya bersifat mengejek atau mengkritik - Berpotensi menimbulkan kesalahan klasifikasi sentimen	“Hebat sekali program MBG ini, bikin harga pangan naik”

2.1.3 Media Sosial X (twitter)

X (twitter) merupakan salah satu media sosial paling populer untuk berbagi pendapat, berita, dan tanggapan terhadap isu sosial maupun kebijakan pemerintah. Format pesannya yang singkat (maksimal 280 karakter) menjadikan X (twitter) sebagai sumber data yang kaya untuk penelitian analisis sentimen (Pahtoni & Jati, 2024). Selain itu, X (Twitter) memungkinkan pengguna menyampaikan komentar secara lebih bebas, spontan, dan real-time dibandingkan beberapa media sosial lainnya, sehingga banyak ditemukan ekspresi opini yang beragam, termasuk penggunaan bahasa sindiran dan sarkasme (Syapriani dkk., 2024).

Namun, sifat informal bahasa di X (twitter) seperti penggunaan singkatan, hashtag, emoji, dan sarkasme membuat analisis menjadi kompleks. Oleh karena itu, dibutuhkan metode pemrosesan bahasa alami (*Natural Language Processing* – NLP) yang mampu menangani variasi bahasa dan memahami konteks percakapan pengguna (Zakharia dkk., 2025).

2.1.4 Program Makan Bergizi Gratis

Program Makan Bergizi Gratis (MBG) merupakan salah satu program prioritas pemerintah Indonesia yang bertujuan untuk meningkatkan kualitas gizi masyarakat, khususnya anak usia sekolah (Albaburrahim dkk., 2025). Program ini dirancang sebagai upaya preventif dalam menekan angka stunting, malnutrisi, dan ketimpangan akses terhadap makanan bergizi, yang masih menjadi permasalahan kesehatan nasional. Program Makan Bergizi Gratis (MBG) menjadi salah satu kebijakan pemerintah yang banyak diperbincangkan oleh masyarakat di media sosial X (twitter). Sebagai ruang publik digital, X (twitter) digunakan oleh

masyarakat untuk menyampaikan pendapat, dukungan, kritik, hingga sindiran terhadap kebijakan yang sedang berjalan (Syapriani dkk., 2025). Komentar-komentar yang muncul tidak hanya bersifat informatif, tetapi juga mengandung ekspresi emosional, opini subjektif, serta bentuk bahasa tidak langsung seperti ironi dan sarkasme.

Dalam konteks program MBG, komentar di X (twitter) mencerminkan beragam respons masyarakat. Sebagian pengguna menyampaikan sentimen positif yang menilai program ini bermanfaat dalam meningkatkan gizi anak dan membantu keluarga kurang mampu namun di sisi lain, terdapat komentar bernada negatif yang menyoroti aspek anggaran, mekanisme distribusi, efektivitas pelaksanaan, serta kekhawatiran terhadap potensi penyalahgunaan dana (Putriyekti dkk., 2025). Selain itu, tidak sedikit komentar yang disampaikan secara sarkastik, yaitu menggunakan kata-kata positif di permukaan tetapi bermakna kritik atau ketidakpercayaan terhadap program tersebut (Putri Salsabila & Markhamah, 2024).

2.1.5 *Natural Language Processing (NLP*)

Pemrosesan Bahasa Alami atau *Natural Language Processing* (NLP) adalah cabang ilmu kecerdasan buatan yang berfokus pada interaksi antara komputer dan bahasa manusia. NLP mencakup berbagai tahapan, mulai dari pra-pemrosesan teks (*teXt prepocessing*), ekstraksi fitur, hingga klasifikasi (Amien, 2023). Dalam penelitian ini, NLP digunakan untuk mengolah data komentar X sebelum dimasukkan ke model RoBERTa-BiLSTM guna mendeteksi sarkasme dan mengklasifikasikan sentimen.

2.1.6 Multi Kelas

Penelitian ini menggunakan pendekatan multi-kelas untuk memperluas cakupan deteksi sarkasme. Pendekatan ini mempertimbangkan hubungan antara polaritas sentimen dan keberadaan sarkasme, sehingga komentar diklasifikasikan menjadi empat kelas menurut (Alivia Rosita Wahyuni dkk., 2025) dan (Amalia dkk., 2024) :

1. Komentar positif - sarkasme positif yaitu komentar yang terlihat positif tetapi mengandung maksud sarkastik.
2. Komentar positif - sarkasme negatif yaitu komentar positif yang memang dimaksudkan sebagai pujian atau apresiasi
3. Komentar negatif - sarkasme positif yaitu komentar negative yang masih mengandung komentar positif yang mengekspresikan ketidakpuasan atau kritik dengan sarkasme.
4. Komentar negatif - sarkasme negatif yaitu komentar negatif yang langsung mengekspresikan ketidakpuasan atau kritik dengan sarkasme.

2.1.7 Pra-Pemrosesan Teks

Pra-Pemrosesan Teks adalah tahap awal pada pengolahan teks yang bertujuan mengubah data mentah menjadi format yang lebih terstruktur dan bermakna (Silaban & Gumay, 2025). Pada tahap ini bertujuan untuk membersihkan dan menyiapkan data teks mentah agar dapat diolah lebih lanjut. Data teks yang mentah umumnya masih mengandung berbagai unsur yang tidak relevan seperti tanda baca, simbol, angka, huruf kapital yang tidak konsisten, serta kata-kata yang tidak memiliki makna yang penting.

Beberapa tahap dalam Pra-Pemrosesan Teks:

1. *Case Folding*

Case Folding merupakan salah satu tahap awal dalam proses pra-pemrosesan teks pada pengolahan bahasa alami (*Natural Language Processing*) (Arifin & Mahdiana, 2024). Tahap ini bertujuan untuk menyeragamkan bentuk huruf dalam teks dengan cara mengubah seluruh karakter menjadi huruf kecil (*lowercase*). Proses ini dilakukan untuk menghindari perbedaan makna yang sebenarnya sama tetapi dianggap berbeda oleh sistem akibat perbedaan penggunaan huruf besar dan huruf kecil. Dalam konteks analisis sentimen dan deteksi sarkasme pada komentar media sosial X (twitter), *Case Folding* membantu menyederhanakan teks agar lebih konsisten dan mudah diproses oleh model (Hermawan dkk., 2025). Sebagai contoh, kata “Program”, “program”, dan “PROGRAM” memiliki makna yang sama, namun tanpa *Case Folding* kata-kata tersebut akan diperlakukan sebagai token yang berbeda. Dengan menerapkan *Case Folding,* seluruh kata tersebut diubah menjadi “program”, sehingga mengurangi jumlah variasi kata yang tidak perlu.

2. Tokenisasi

Dalam Konteks komentar pengguna di aplikasi X (twitter) proses memecah teks menjadi bagian kecil yang disebut token, yang bisa berupa kata, frasa, simbol, atau elemen lain (Agung & Bunyamin, 2024). Proses tokenisasi ini dilakukan dengan menggunakan *Roberta Tokenizer* untuk memudahkan pemrosesan teks lebih lanjut, seperti pengindeksan dokumen atau analisis teks. Contohnya, kalimat "Saya suka belajar" dipecah menjadi token ["Saya", "suka", "belajar"] (Prengki Putra, 2024).

3. Stopword Removal

Stopword mengacu pada kata-kata yang sering muncul tetapi tidak memberikan kontribusi signifikan terhadap pemaknaan kalimat, seperti kata hubung atau kata ganti umum (Lucky Hermanto dkk., 2025). Pada tahap pemrosesan teks, *stopword* biasanya dibuang agar analisis fokus pada kata-kata yang memiliki bobot informasi lebih kuat. *Stopword Removal* ini membantu mengurangi noise, terutama dalam komentar media sosial yang sering dipenuhi kata sambung tanpa makna sentimen yang jelas (Jessica Angelina dkk., 2023). Dengan menghilangkan *stopword*, model dapat bekerja lebih efisien dan memusatkan perhatian pada bagian teks yang benar-benar relevan (Suryaningrum dkk., 2024).

4. Lemmatization.

Lemmatization bertujuan untuk mengubah suatu kata ke dalam bentuk dasarnya (lemma) (Mutiasari dkk., 2025). Kata yang mengalami perubahan bentuk akibat imbuhan, waktu, atau jumlah akan dikembalikan ke bentuk kata dasar yang benar. Sebagai contoh, pada komentar terkait program MBG, kata “mendukung”, “didukung”, dan “dukungan” akan diubah menjadi kata dasar “dukung”.

2.1.8 *Deep Learning* dalam Analisis Teks

Pendekatan *deep learning* digunakan untuk memisahkan dua proses utama dalam penelitian, yaitu analisis sentimen dan deteksi sarkasme, sebelum keduanya digabungkan dalam tahap evaluasi silang, menurut (Jaya Kusoema dkk., 2025) proses ini dilakukan karena pola sentimen pada komentar sering berbeda dari pola bahasa sarkasme, sehingga diperlukan dua model dengan keunggulan masing-masing.

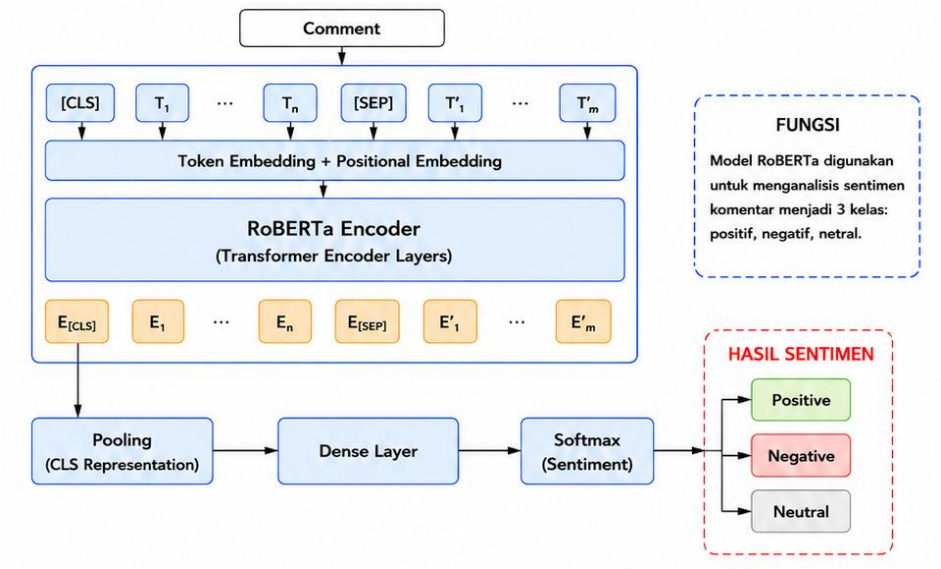
BERT *(Bidirectional Encoder Representations from Transformers)* merupakan model berbasis arsitektur Transformer yang mampu memahami konteks bahasa secara dua arah melalui mekanisme *self-attention* (Firmanto dkk., 2024). Model ini memproses seluruh token dalam kalimat secara bersamaan sehingga dapat menangkap hubungan antar kata secara kontekstual. Meskipun BERT telah menunjukkan kinerja yang baik dalam berbagai tugas pemrosesan bahasa alami, keterbatasan pada strategi pelatihannya membuat performanya masih dapat ditingkatkan.

RoBERTa digunakan untuk membaca komentar sebagai sebuah konteks penuh dan menghasilkan prediksi sentimen yang lebih stabil (Setiadi dkk., 2024). Menurut (Sihombing & Situmorang, 2024) kemampuan RoBERTa dalam memahami relasi antar kata membantu model ini mengenali nada positif maupun negatif, bahkan ketika pengguna menyampaikan pendapat secara tidak langsung. Model ini bekerja dengan memproses seluruh kalimat secara bersamaan sehingga mampu menangkap makna global dari sebuah komentar.

LSTM *(Long Short-Term Memory)* merupakan pengembangan dari *Recurrent Neural Network* (RNN) yang dirancang untuk memodelkan data berurutan dan mengatasi permasalahan *vanishing gradien*t (Hanafiah dkk., 2023) . LSTM mampu menangkap ketergantungan jangka panjang dalam teks dengan memanfaatkan mekanisme *gates* sehingga efektif dalam memahami konteks berdasarkan urutan kata. Namun, berdasarkan penelitian (Wangsajaya dkk., 2025) LSTM konvensional hanya memproses informasi secara satu arah, yaitu dari awal hingga akhir kalimat. Keterbatasan ini menyebabkan informasi dari kata-kata di bagian akhir kalimat

belum sepenuhnya dimanfaatkan ketika memaknai kata-kata di bagian awal, terutama pada teks dengan makna implisit. Sementara itu, BiLSTM difokuskan pada pendeteksian sarkasme. Pendekatan dua arah pada BiLSTM membantu model mengenali perubahan nada atau ketidaksesuaian makna yang biasanya menjadi ciri utama sarkasme (Kholida Zia Abidin dkk., 2023). Komentar sarkastik sering menunjukkan pergeseran makna antara bagian awal dan akhir kalimat.

2.1.9 Model RoBERTa *(Robustly Optimized BERT Approach)*



Gambar 2. 1 Model Robustly Optimized BERT Approach (RoBERTa) (Khusuma dkk., 2023)

Keterangan :

Sentence 1 / Sentence 2	: Kalimat teks yang menjadi input model.
T	: Token hasil pemotongan kata (subword) dari kalimat.
[CLS]	: Token khusus di awal input untuk merepresentasikan keseluruhan teks.
[SEP]	: Token pemisah antar kalimat atau penanda akhir kalimat.
E[CLS]	: Embedding numerik dari token [CLS].
$E_1 \dots E_n, E'_1 \dots E'_m$	: $E_1 \dots E_n, E'_1 \dots E'_m$

C	: *Output* akhir token [CLS] yang merangkum makna seluruh teks.
$T_1 \dots T_n, T'_1 \dots T'_m$	: *Output* RoBERTa untuk setiap token teks.
Class Label	: Hasil prediksi kelas (misalnya sarkasme atau non-sarkasme).

2.1.9.1 Pseudocode RoBERTa

Algoritma RoBERTa

Input: text komentar atau teks yang akan dianalisis

Mulai
1. Input dataset teks $D = \{x_1, x_2, x_3, \dots, x_n\}$
2. Preprocessing teks
FOR setiap teks $x_i$dalam D
Ubah teks menjadi lowercase
Hapus URL, mention, hashtag, angka, dan tanda baca
END FOR
3. Tokenisasi menggunakan RoBERTa Tokenizer
Input: $token = \{CLS, w_1, w_2, \dots, w_n, SEP\}$
4. Embedding $E = TokenEmbedding + PositionalEmbedding$
5. Encoder RoBERTa (Transformer Layer)
5.1 Self-Attention

$$Q = E \times W_Q$$
$$K = E \times W_K$$
$$V = E \times W_V$$
$$Attention(Q, K, V) = Softmax\left(\frac{QK^T}{\sqrt{d_k}}\right)V$$
$$Softmax(z_i) = \frac{e^{z_i}}{\sum_{j=1}^{n} e^{z_j}}$$

6. Contextual Representation $H = RoBERTaEncoder(E)$
7. Pooling Layer (CLS Representation) $h_{cls} = H[CLS]$
8. Classification Layer $z = h_{cls}W + b$
9. Softmax Activation $Softmax(z_i) = \frac{e^{z_i}}{\sum_{j=1}^{n} e^{z_j}}$
10. Prediksi Sentimen $\hat{y} = argmax(P)$
IF argmax(P) = kelas_positif
RETURN "Positif"
ELSE IF argmax(P) = kelas_negatif
RETURN "Negatif"
ELSE
RETURN "Netral"
END IF
11. *Output* hasil klasifikasi sentimen
Selesai
Output : label kelas sentimen
("positif", "negatif", atau "netral")

Output: label kelas sarkasme
("sarkasme_positif" atau "sarkasme_negatif")

RoBERTa digunakan pada penelitian ini untuk membaca komentar secara menyeluruh dan menghasilkan prediksi sentimen yang stabil (Nainggolan & Handoko, 2025). Setiap kata dalam komentar diproses berdasarkan hubungan dengan kata lain, sehingga model dapat menangkap kecenderungan emosi yang tersirat dalam kalimat, meskipun disampaikan melalui bahasa yang tidak langsung. Menurut (Jaya Kusoema dkk., 2025) pendekatan berbasis konteks ini membantu RoBERTa mengidentifikasi apakah sebuah komentar condong pada sentimen positif atau negatif tanpa terpengaruh oleh variasi gaya bahasa pengguna di aplikasi X (twitter). Pada penelitian ini, keluaran RoBERTa diperlakukan sebagai hasil analisis sentimen utama yang nantinya dipadukan dengan informasi sarkasme dari model BiLSTM.

Mekanisme utama pada RoBERTa menggunakan self-attention untuk memahami hubungan antar kata dalam kalimat. Self-attention digunakan untuk menghitung tingkat perhatian setiap kata terhadap kata lainnya sehingga model dapat memahami konteks secara lebih mandalam (Basbeth Faishal, 2024). Perhitungan self-attention dilakukan menggunakan Persamaan 2.1.

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

Selanjutnya, hasil *Output* dari self-attention diproses menggunakan fungsi Softmax untuk menentukan probabilitas kelas hasil prediksi (Setia Budi dkk., 2024). Nilai probabilitas tersebut digunakan untuk menentukan kelas dengan kemungkinan tertinggi sebagai hasil klasifikasi sentimen maupun deteksi sarkasme. Semakin besar nilai probabilitas suatu kelas, maka semakin tinggi keyakinan model

terhadap kelas tersebut. Perhitungan Softmax dilakukan menggunakan Persamaan 2.2

$$P(y_i) = \frac{e^{z_i}}{\sum_{j=1}^{n} e^{z_j}} \quad 2.2$$

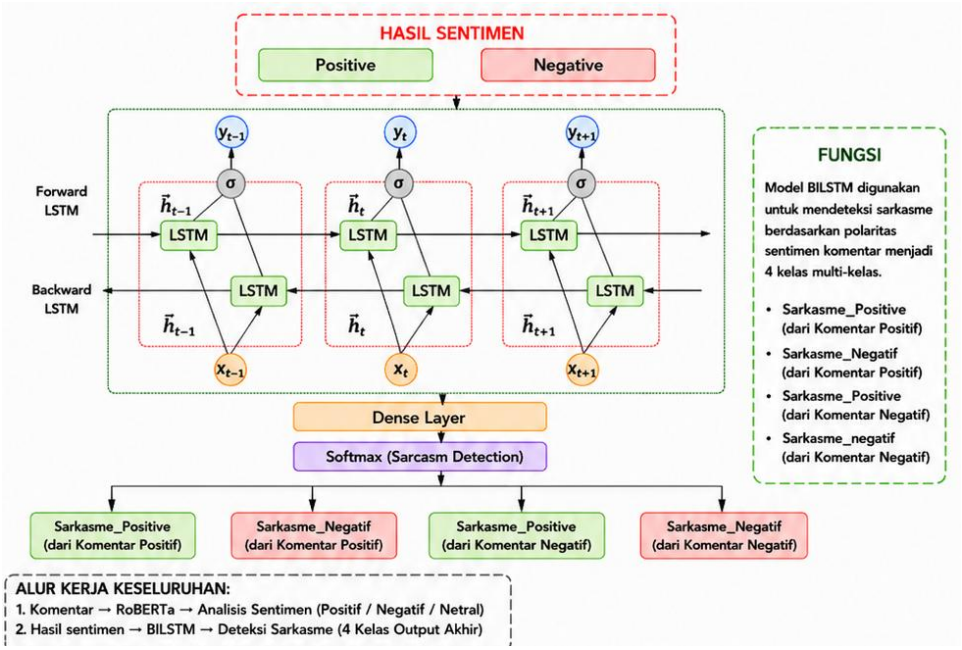
Setelah itu, classification layer digunakan untuk mengubah representasi kontekstual dari token [CLS] menjadi skor prediksi untuk setiap kelas. Pada tahap ini, *Output* hasil encoder RoBERTa diproses menggunakan operasi linear berupa perkalian bobot dan penambahan bias sebelum diteruskan ke fungsi Softmax. Perhitungan classification layer dilakukan menggunakan Persamaan 2.3

$$z = h_{cls}W + b \quad 2.3$$

Keterangan:

Q	: Representasi kata yang sedang mencari informasi
K	: Representasi kata yang dibandingkan dengan query
V	: Informasi atau nilai dari setiap kata
K^T	: Transpose dari key
d_k	: Dimensi vektor key
$Softmax$	: Fungsi untuk mengubah nilai menjadi probabilitas attention
$P(y_i)$	: Probabilitas kelas ke-i
e	: Bilangan eksponensial
z_i	: Nilai *Output* (logit) pada kelas ke-i
n	: Jumlah kelas
$\sum$	: Penjumlahan seluruh nilai eksponensial *Output* model
z	: *Output* skor klasifikasi sebelum softmax
h_{cls}	: Representasi vektor token [CLS] hasil encoder RoBERTa
W	: Bobot (weight) pada layer klasifikasi
b	: Bias pada layer klasifikasi

2.1.10 *Bidirectional Long Short-Term Memory* (BiLSTM)



Gambar 2. 2 Model Bidirectional Long Short-Term Memory (BiLSTM) (Puteri, 2023)

Keterangan :

X_t	: Input pada waktu ke-t, yaitu vektor kata ke-t dalam sebuah kalimat.
h_{t-1} (hidden state sebelumnya)	: Informasi yang dibawa dari kata sebelumnya. Nilai ini menyimpan konteks yang telah dipelajari LSTM hingga waktu ke-(t−1).
h_t (hidden state saat ini)	: Representasi konteks yang dihasilkan setelah LSTM memproses input X_t dan informasi dari h_{t-1}.
Forward LSTM	: Menghasilkan h_t dengan membaca urutan kata dari kiri ke kanan, sehingga menangkap konteks masa lalu.
Backward LSTM	: Menghasilkan h_t dengan membaca urutan kata dari kanan ke kiri, sehingga menangkap konteks masa depan.
Penggabungan (BiLSTM)	: Hidden state dari arah forward dan *backward* digabungkan untuk membentuk representasi

konteks yang lebih lengkap pada setiap waktu ke-t.

y_t : *Output* pada waktu ke-t yang dihasilkan dari hidden state dan digunakan untuk proses klasifikasi.

2.1.11.1 Pseudocode BiLSTM

Algoritma BiLSTM
Input : - Dataset komentar - Daftar kata sarkasme positif (kata_positif) - Daftar pola sarkasme positif (pola_positif) - Daftar kata sarkasme negatif (kata_negatif) - Daftar pola sarkasme negatif (pola_negatif)
Mulai 1. Input dataset teks $D = \{x_1, x_2, x_3, \ldots, x_n\}$ 2. Pra-pemrosesan teks FOR setiap komentar $x_i$dalam D Ubah teks menjadi lowercase Hapus URL, mention, hashtag, angka, dan tanda baca END FOR 3. Tokenisasi teks FOR setiap komentar x_i $tokens = tokenize(x_i)$ END FOR 4. Konversi token menjadi indeks numerik FOR setiap tokens $x_{index} = mapping(tokens \rightarrow vocabulary\ index)$ END FOR 5. Padding sequence $x_{pad} = pad(x_{index}, max\_length)$ 6. Input Layer $X = x_{pad}$ 7. Embedding Layer $x_t = Embedding(X)$ 8. BiLSTM Layer Forward LSTM: $h_f(t) = LSTM(x_t, h_f(t-1))$ *Backward* LSTM: $h_b(t) = LSTM(x_t, h_b(t+1))$ Gabungan hidden state: $h(t) = [h_f(t); h_b(t)]$ 9. Dense Layer $z = h(t)W + b$

10. Aktivasi Softmax $P(y_i) = \frac{e^{z_i}}{\sum_{j=1}^{n} e^{z_j}}$

11.Prediksi kelas $\hat{y} = argmax(P)$

```
IF argmax(P) = kelas_1
RETURN "sarkasme_positif"
ELSE IF argmax(P) = kelas_2
RETURN "sarkasme_negatif"
ELSE
RETURN "netral"
END IF
```

12.*Output* hasil prediksi

Selesai

Output **:**

- Kelas sarkasme komentar ("sarkasme_positif" atau "sarkasme_negatif")
- Probabilitas prediksi dari model BiLSTM

BiLSTM digunakan untuk mendeteksi sarkasme dalam komentar dengan membaca urutan kata dari dua arah. Pendekatan dua arah ini membantu model menangkap kontras makna yang sering muncul dalam kalimat bernada sindiran (Alghifari dkk., 2022), misalnya ketika awal kalimat bernada positif tetapi diikuti penegasan yang berlawanan. Pola semacam ini sering menjadi tanda sarkasme dalam percakapan di media sosial. Penggunaan BiLSTM pada penelitian ini berfokus pada sensitivitas model terhadap dinamika perubahan makna di dalam satu kalimat (Hilmawan, 2022), sehingga setiap komentar memperoleh penilaian apakah mengandung sarkasme atau tidak sebelum digabungkan dengan hasil sentimen dari RoBERTa.

Pada prosesnya, forward LSTM digunakan untuk membaca urutan kata dari awal hingga akhir kalimat, sedangkan *backward* LSTM digunakan untuk membaca urutan kata dari akhir menuju awal kalimat. Hasil dari kedua arah tersebut

kemudian digabungkan untuk menghasilkan representasi konteks yang lebih lengkap. Selanjutnya, hasil hidden state diproses pada dense layer untuk menghasilkan skor klasifikasi sebelum diteruskan ke fungsi Softmax guna menentukan probabilitas kelas prediksi. Perhitungan forward LSTM, *backward* LSTM, gabungan hidden state, dense layer, dan Softmax dilakukan menggunakan Persamaan 2.4 sampai Persamaan 2.8 sebagai berikut.

$$h_f(t) = LSTM(x_t, h_f(t-1)) \quad 2.4$$

$$h_b(t) = LSTM(x_t, h_b(t+1)) \quad 2.5$$

$$h(t) = [h_f(t); h_b(t)] \quad 2.6$$

$$z = h(t)W + b \quad 2.7$$

$$Softmax(z_i) = \frac{e^{z_i}}{\sum_{j=1}^{n} e^{z_j}} \quad 2.8$$

Keterangan:

$h_f(t)$: Hidden State Forward pada waktu ke-t

$h_b(t)$: Hidden State *Backward* pada waktu ke-t

x_t : Input pada waktu ke-t

$h_f(t-1)$: Hidden State Forward sebelumnya

$h_b(t+1)$: Hidden State *Backward* setelahnya

$h(t)$: Gabungan Hidden State BiLSTM

z : *Output* skor klasifikasi sebelum softmax

W : Bobot (Weight) pada Dense Layer

b : Bias pada Dense Layer

$Softmax(z_i)$: Probabilitas kelas ke-i

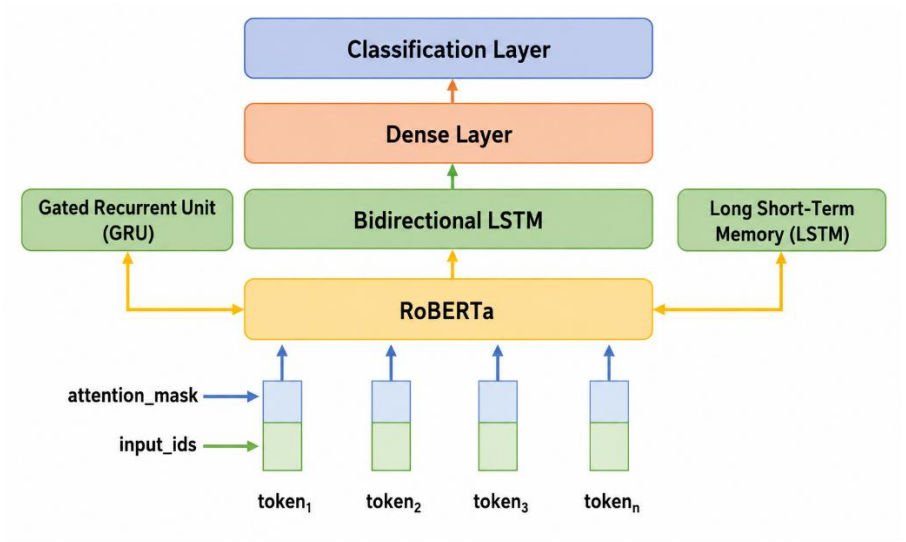
e : Bilangan eksponensial

z_i : Nilai *Output* (Logit) pada kelas ke-i

n : Jumlah kelas prediksi

$\sum$: Penjumlahan seluruh nilai eksponensial *Output* model

2.1.11 Model Hibrida RoBERTa-BiLSTM



Gambar 2. 3 Model Hibrida RoBERTa-BiLSTM (Yuliyanti dkk., 2026)

RoBERTa-BiLSTM merupakan model *hybrid deep learning* yang mengombinasikan arsitektur RoBERTa (*Robustly Optimized BERT Approach*) dengan *Bidirectional Long Short-Term Memory* (BiLSTM) untuk meningkatkan performa pemodelan teks berdasarkan penelitian yang dilakukan oleh (Yuliyanti dkk., 2026) menunjukan bahwa RoBERTa-BiLSTM lebih baik untuk mendeteksi Sarkasme. Pada penelitian tersebut, deteksi sarkasme dilakukan menggunakan pendekatan klasifikasi satu label (single-label classification), di mana setiap komentar hanya diklasifikasikan ke dalam satu kelas keluaran, yaitu sarkasme atau

non-sarkasme dan pendekatan *hybrid* diterapkan dengan menggunakan model RoBERTa untuk analisis sentimen karena mampu memahami konteks bahasa secara mendalam, sedangkan model BiLSTM digunakan pada deteksi sarkasme untuk menangkap hubungan dan pola urutan kata dalam teks sehingga dapat mengenali makna sarkastik dengan lebih baik.

2.1.12 TF IDF

Sebelum proses klasifikasi dilakukan, teks terlebih dahulu melalui tahap pembentukan n-gram untuk menghasilkan kombinasi kata yang dapat merepresentasikan konteks kalimat secara lebih spesifik. Melalui proses tersebut, kata tunggal (unigram) maupun gabungan dua kata (bigram) seperti "bergizi gratis" dapat terbentuk sehingga model dapat menangkap pola frasa yang sering muncul pada komentar. Setelah pembentukan n-gram, dilakukan perhitungan TF *(Term Frequency)* untuk mengetahui seberapa sering suatu kata atau n-gram muncul dalam sebuah dokumen. Selanjutnya dilakukan perhitungan IDF *(Inverse Document Frequency)* untuk mengetahui tingkat keunikan suatu kata dibandingkan seluruh dokumen dalam dataset. Semakin sedikit kata muncul pada dokumen lain, maka nilai IDF akan semakin tinggi. Nilai TF dan IDF kemudian dikalikan untuk menghasilkan bobot TF-IDF yang menunjukkan tingkat kepentingan suatu kata atau n-gram dalam dataset. Setelah seluruh nilai TF-IDF diperoleh, dilakukan perhitungan rata-rata bobot TF-IDF untuk mengetahui kata atau n-gram yang paling dominan pada keseluruhan data. Pembentukan n-gram, perhitungan TF, IDF, TF-IDF, dan rata-rata TF-IDF dilakukan menggunakan Persamaan 2.9 sampai Persamaan 2.13

$$G_n = (w_i, w_{i+1}, \dots, w_{i+n-1}) \quad 2.9$$

$$TF(t,d) = \frac{f_{t,d}}{\sum_k f_{k,d}} \quad 2.10$$

$$IDF(t) = \log\left(\frac{N}{df(t)}\right) \quad 2.11$$

$$F\text{-}IDF(t,d) = TF(t,d) \times IDF(t) \quad 2.12$$

$$TF\text{-}\bar{I}DF = \frac{\sum_{i=1}^{n} TF\text{-}ID\ F_i}{n} \quad 2.13$$

Keterangan

G_n : N-Gram

w_i : Kata ke-i

n : Jumlah kata dalam satu kombinasi

$TF(t,d)$: Nilai Term Frequency suatu term pada dokumen

$f_{t,d}$: Jumlah kemunculan term $t$pada dokumen d

$\sum_k f_{k,d}$: Total seluruh term pada dokumen

$IDF(t)$: Nilai Inverse Document Frequency suatu term

N : Jumlah seluruh dokumen

$df(t)$: Jumlah dokumen yang mengandung term t

$TF\text{-}IDF(t,d)$: Bobot TF-IDF suatu term pada dokumen

$TF\text{-}\bar{I}DF$: Rata-rata bobot TF-IDF

$TF\text{-}IDF_i$: Nilai TF-IDF tiap term atau n-gram

2.1.13 Akurasi (*Accuracy*)

Akurasi menunjukkan tingkat ketepatan model dalam mengklasifikasikan seluruh data uji secara benar (Agustina dkk., 2022) dan Mengukur proporsi total prediksi yang benar dari semua prediksi yang dibuat dengan Persamaan 2.14 (Pambudi dkk., 2025)

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad 2.14$$

Keterangan :

TP (True Positive) : Jumlah data yang diprediksi positif oleh model dan benar-benar positif berdasarkan label asli.

TN (True Negatif) : Jumlah data yang diprediksi negatif oleh model dan benar-benar negatif berdasarkan label asli.

FP (False Positive) : Jumlah data yang diprediksi positif oleh model, tetapi sebenarnya negatif.

FN (False Negatif) : Jumlah data yang diprediksi negatif oleh model, tetapi sebenarnya positif.

2.1.14 *Precision*

Precision mengukur tingkat ketepatan prediksi model terhadap suatu kelas tertentu (Pambudi dkk., 2025). Metrik ini menunjukkan seberapa banyak prediksi positif yang benar-benar sesuai dengan kelas sebenarnya menggunakan Persamaan 2.15

$$Precision = \frac{TP}{TP + FP} \quad 2.15$$

Keterangan :

TP (True Positive) : jumlah data yang diprediksi sebagai kelas tertentu (misalnya sarkastik) dan hasil prediksi tersebut benar.

FP (False Positive) : jumlah data yang diprediksi sebagai kelas tertentu, tetapi sebenarnya termasuk kelas lain.

2.1.15 *Recall*

Recall menunjukkan kemampuan model dalam menemukan seluruh data yang benar-benar termasuk ke dalam suatu kelas (Pambudi dkk., 2025). Metrik ini penting untuk melihat seberapa baik model mendeteksi komentar sarkastik menggunakan Persamaan 2.16.

$$Recall = \frac{TP}{TP + FN} \qquad 2.16$$

Keterangan :

TP (True Positive) : Jumlah data yang diprediksi sebagai suatu kelas dan benar sesuai dengan label aslinya.

FN (False Negatif) : Jumlah data yang sebenarnya termasuk dalam suatu kelas, tetapi gagal dideteksi atau diprediksi sebagai kelas lain oleh model.

2.1.16 *F1-Score*

F1-Score merupakan rata-rata harmonis antara *precision* dan *recal (Pambudi dkk., 2025)*. Metrik ini digunakan untuk memberikan gambaran keseimbangan antara ketepatan dan kelengkapan prediksi model, terutama pada data dengan distribusi kelas tidak seimbang.

$$F1\text{-}Score = 2 \times \frac{Precision \times Recall}{Precision + Recall} \qquad 2.17$$

Keterangan :

Precision : Tingkat ketepatan model dalam memprediksi suatu kelas.

Recall : Kemampuan model dalam mendeteksi seluruh data yang termasuk dalam suatu kelas.

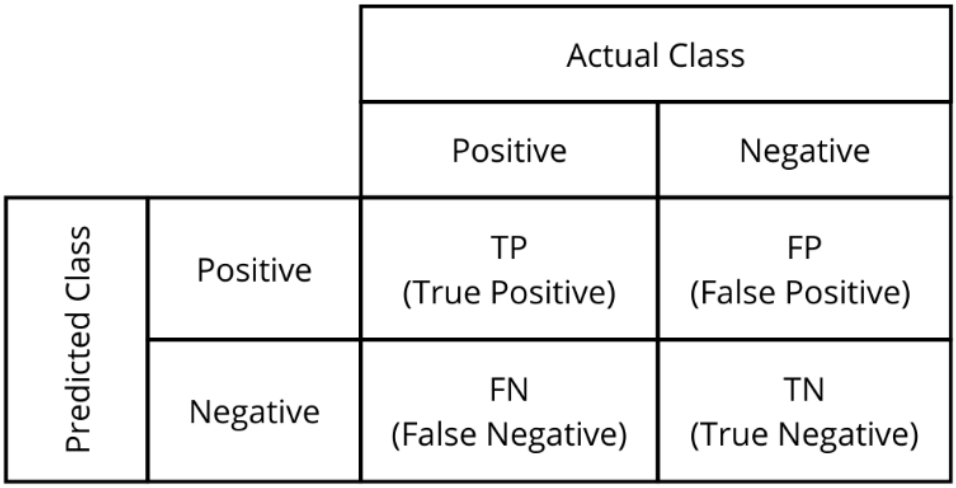
Selanjutnya *macro average* digunakan untuk menghitung rata-rata performa model pada seluruh kelas tanpa mempertimbangkan jumlah data pada masing-masing kelas. Perhitungan ini dilakukan dengan menghitung rata-rata nilai *F1-Score* dari setiap kelas sehingga setiap kelas memiliki kontribusi yang sama dalam evaluasi model. Perhitungan *macro average* dilakukan menggunakan Persamaan 2.18.

$$Macro\ F1 = \frac{F1_{negatif} + F1_{positif}}{Jumlah\ Kelas} \qquad 2.\ 18$$

Sedangkan weighted average digunakan untuk menghitung rata-rata performa model dengan mempertimbangkan jumlah data (support) pada setiap kelas. Pada perhitungan ini, kelas dengan jumlah data lebih besar akan memberikan pengaruh lebih besar terhadap hasil akhir evaluasi model. Perhitungan weighted average dilakukan menggunakan Persamaan 2.19.

$$Weighted\ F1 = \frac{(F1_{negatif} \times Support_{negatif}) + (F1_{positif} \times Support_{positif})}{Total\ Support} \qquad 2.\ 19$$

2.1.17 *Confusion Matrix*



Gambar 2. 4 *Confusion Matrix*

Pada gambar 2.4 menampilkan *Confusion Matrix* yang terdiri dari empat istilah yang merepresentasikan hasil dari proses klasifikasi, yaitu *True Positive* (TP), *True* Negatif (TN), *False Positive* (FP), dan *False* Negatif (FN). Tabel ini menunjukkan jumlah prediksi yang benar maupun salah untuk setiap kelas, sehingga memberikan gambaran yang lebih rinci dibandingkan hanya menggunakan nilai akurasi.

2.1.18 Probabilitas

Dalam tugas klasifikasi teks, model tidak hanya menghasilkan label kelas, tetapi juga nilai probabilitas yang menunjukkan tingkat keyakinan model terhadap setiap kelas. Probabilitas merupakan ukuran peluang suatu data termasuk ke dalam kelas tertentu berdasarkan pola yang telah dipelajari selama proses pelatihan. Semakin tinggi nilai probabilitas suatu kelas, semakin besar keyakinan model

bahwa data tersebut termasuk ke dalam kelas tersebut (Susana & Suarna, 2022)> Perhitungan probabilitas menggunakan Persamaan 2.20.

$$P(Positif) = 1 - P(Negatif) \qquad 2.20$$

Keterangan:

P(Positif) : Probabilitas suatu data atau komentar termasuk ke dalam kelas positif.

P(Negatif) : Probabilitas suatu data atau komentar termasuk ke dalam kelas negatif.

Nilai 1 : Menunjukkan total probabilitas seluruh kemungkinan kelas yang berjumlah 100%.

2.2 State Of The Art (SOTA)

Penelitian-penelitian terkait deteksi sarkasme dan analisis sentimen multi-kelas pada media sosial telah berkembang pesat dalam beberapa tahun terakhir. Berbagai metode dan algoritma pembelajaran mesin serta *deep learning* telah diterapkan untuk meningkatkan akurasi dalam mengidentifikasi sentimen serta makna tersirat yang terkandung dalam teks sarkastik. *State Of The Art (SOTA)* pada Tabel 2.3 yang disajikan menunjukkan beragam pendekatan yang telah dikaji, termasuk pemanfaatan model berbasis *transformer* dan arsitektur *hybrid*, guna menghasilkan sistem deteksi sarkasme multi-kelas yang lebih akurat dan andal dalam menganalisis sentimen media sosial. Dengan demikian, penelitian ini memperluas kontribusi dari penelitian (Kumar dkk., 2020) dan (Tan dkk., 2022) dengan melakukan deteksi sarkasme multi-kelas berbasis konteks menggunakan model *hybrid* RoBERTa-BiLSTM yang terintegrasi dengan analisis sentimen.

Tabel 2.2 *State Of The Art* (SOTA)

No	Judul	Peneliti	Riset	Metode	Hasil Penelitian	Kelebihan	Keterbatasan
1	RoBERTa-LSTM: A *Hybrid* Model for Sentiment Analysis with Transformer and Recurrent Neural Network	(Tan dkk., 2022)	Analisis sentimen pada teks media sosial dan ulasan (IMDb, Twitter US Airline, Sentiment140)	Model *hybrid* RoBERTa (Transformer) + LSTM (RNN), dilengkapi data augmentation berbasis GloVe	Mencapai *F1-Score* 93% (IMDb), 91% (Twitter US Airline), dan 90% (Sentiment140); mengungguli metode state-of-the-art sebelumnya	Menggabungkan keunggulan Transformer (embedding kontekstual & paralel) dan LSTM (dependensi jangka panjang	Waktu pelatihan lebih lama dibanding model tunggal seperti LSTM atau CNN. Belum
2	Sentiment Analysis with Ensemble *Hybrid* Deep Learning Model	(Long Tan dkk., 2022)	Analisis Sentimen pada dataset Sentiment 140	RoBERTa-BiLSTM	RoBERTa-BiLSTM 89,62%	Penelitian Kian Long Tan menggunakan model *hybrid* RoBERTa-BiLSTM yang mampu memahami konteks kalimat dan hubungan	Penelitian masih berfokus pada analisis sentimen dan memerlukan komputasi serta waktu pelatihan yang

No	Judul	Peneliti	Riset	Metode	Hasil Penelitian	Kelebihan	Keterbatasan
						antar kata dengan baik sehingga memperoleh *accuracy* sebesar 89,62%.	cukup besar karena menggunakan model *hybrid* deep learning.
3	Sarcasm Detection Using Multi-Head Attention Based Bidirectional LSTM	(Kumar dkk., 2020)	Deteksi sarkasme pada komentar media sosial	Multi-Head Attention + BiLSTM + fitur semantik & pragmatik	Model MHA-BiLSTM mengungguli BiLSTM dan SVM berbasis fitur dalam *F1-Score*	Mampu menangkap konteks implisit dan bagian kalimat yang penting, efektif untuk bahasa sarkastik	Penelitian ini masih memfokuskan pada deteksi sarkasme secara umum tanpa membedakan polaritas sentimen sarkasme, yaitu sarkasme positif dan sarkasme negatif.
4	Sarcasm Detection over Social Media Platforms Using *Hybrid* Auto-	(Sharma dkk., 2022)	Deteksi sarkasme pada data media sosial	RoBERTa BiLSTM	RoBERTa 79%, BiLSTM 79%	Mampu menangkap fitur lokal dan konteks sekuensial secara bersamaan	Kompleksitas model tinggi dan membutuhkan waktu *Training* lebih lama

No	Judul	Peneliti	Riset	Metode	Hasil Penelitian	Kelebihan	Keterbatasan
	Encoder-Based Model						
5	Detection of Sarcasm in Urdu Tweets Using Deep Learning and Transformer Based *Hybrid* Approaches	(Hassan dkk., 2024)	Deteksi sarkasme pada tweet berbahasa Urdu	*Hybrid* (Deep Learning + Transformer)	Model *hybrid* menunjukkan performa lebih baik dibanding model tunggal dalam mendeteksi sarkasme. Mbert-BiLSTM 79,51%	Menggabungkan pemahaman konteks (Transformer) dan pola sekuensial (Deep Learning)	Membutuhkan dataset besar dan komputasi tinggi
6	Sarcasm Detection in Twitter – Performance Impact while using Data Augmentation:	(Handoyo & Suhartono, 2021)	Deteksi sarkasme pada Twitter dan pengaruh data augmentation terhadap performa model	RoBERTa + Data Augmentation berbasis GloVe word embeddings	Data augmentation meningkatkan *F1-Score* hingga 3,2% pada dataset iSarcasm (*F1-Score* dari 37,2%	Memanfaatkan konteks dengan model transformer, mengatasi ketidakseimbangan data	Pengujian hanya pada empat dataset; peningkatan performa masih terbatas dan perlu diuji pada konteks lebih luas, masih

No	Judul	Peneliti	Riset	Metode	Hasil Penelitian	Kelebihan	Keterbatasan
	Word Embeddings				menjadi 40,4%)		terbatas pada klasifikasi biner sarkasme
7	A Term Weighted Neural Language Model and Stacked Bidirectional LSTM Based Framework for Sarcasm Identification	(Onan & Tocoglu, 2021)	Mengembangkan model deteksi sarkasme berbasis pembobotan kata dan deep learning	Term-weighted word embedding (IGM + trigram) + Stacked BiLSTM	Akurasi tertinggi 95.30% pada dataset Twitter	Mempertimbangkan bobot kata penting dan urutan kata	Penelitian tersebut berfokus pada pendeteksian sarkasme secara umum dengan memanfaatkan pembobotan kata dan arsitektur BiLSTM. penelitian ini belum mengkaji sarkasme berdasarkan polaritas sentimen
8	Analisis sentimen program makan	(Modianus Laia dkk., 2025)	Analisis sentimen publik terhadap	TF-IDF + Naïve Bayes, validasi 10-	Akurasi 86,46%, *F1-Score* 90,77%,	Metode sederhana dan efisien, performa cukup tinggi pada teks pendek	Hanya klasifikasi biner, belum menangani sarkasme

No	Judul	Peneliti	Riset	Metode	Hasil Penelitian	Kelebihan	Keterbatasan
	gratis pada platform X menggunakan agoritma naïve bayes		program “Makan Bergizi Gratis” di platform X	Fold Cross Validation	stabil pada cross-validation		dan konteks mendalam
9	ConteXt-Based Feature Technique for Sarcasm Identification in Benchmark Datasets Using Deep Learning and BERT Model	(Eke dkk., 2021)	Identifikasi sarkasme pada data media sosial dan forum	Bi-LSTM + GloVe, BERT, dan feature fusion (sentimen, sintaksis, konteks)	Akurasi dan presisi sangat tinggi (hingga 98% pada dataset Twitter), menunjukkan pentingnya konteks dan BERT dalam deteksi sarkasme	Mampu menangkap makna implisit dan konteks kalimat	Dataset terbatas pada benchmark berbahasa Inggris; belum diuji pada konteks kebijakan publik Indonesia

No	Judul	Peneliti	Riset	Metode	Hasil Penelitian	Kelebihan	Keterbatasan
10	Performance Comparison of 10 Machine Learning Algorithms in Sentiment Classification on Platform X Regarding the Government's Priority Program: MBG	(Utama dkk., 2025a)	Analisis sentimen masyarakat terhadap program MBG di Platform X	10 algoritma ML (NB, RF, GB, AdaBoost, LR, SVM, SGD, Ridge, EXtra Trees, Bagging) + TF-IDF	Model Bagging dan Gradient Boosting menghasilkan akurasi tertinggi sekitar 81%, serta mampu mengidentifikasi kata kunci sentimen positif dan negatif	Perbandingan model sangat komprehensif	Belum mempertimbangkan konteks sarkasme secara eksplisit; berbasis fitur leksikal sehingga makna implisit sulit ditangkap

No	Judul	Peneliti	Riset	Metode	Hasil Penelitian	Kelebihan	Keterbatasan
11	Sentiment Perspective of Government's Free Nutritious Meal Policy on Social Media X using Indo-BERT and Bi-LSTM	(Subarkah dkk., 2025)	Analisis sentimen kebijakan MBG di media sosial X	Indo-BERT dan Bi-LSTM	Indo-BERT mencapai akurasi 80%, lebih baik dibanding Bi-LSTM (78%), dan efektif untuk bahasa Indonesia	Metode sederhana dan mudah diimplementasikan	Kelas netral sangat kecil sehingga sulit terklasifikasi; belum menangani sarkasme secara khusus
12	The Influence of User Sentiment Analysis on the Free Nutritious Meal Program Using K-Nearest Neighbors and Logistic	(Mardiana & Abidin, 2025)	Analisis sentimen publik terhadap program Makan Bergizi Gratis (MBG) di media sosial X	TF-IDF, KNN, Logistic Regression, SMOTE	Akurasi mencapai 78–81%. Model sangat baik mendeteksi sentimen positif (*recall* > 0.96), namun lemah pada sentimen negatif	Metode sederhana dan mudah diimplementasikan, TF-IDF efektif menonjolkan kata penting	Model bias terhadap kelas mayoritas (positif), deteksi sentimen negatif masih lemah meskipun sudah menggunakan SMOTE, belum menangani konteks

No	Judul	Peneliti	Riset	Metode	Hasil Penelitian	Kelebihan	Keterbatasan
	Regression Methods						mendalam seperti sarkasme atau ironi
13	Federated Learning for Sarcasm Detection: A Study of Attention-Enhanced BiLSTM, GRU, and LSTM Architectures	(Sami dkk., 2024)	Deteksi sarkasme pada teks (NLP) dengan fokus privasi data	Federated Learning (FedAvg) + Attention-based BiLSTM, GRU, dan LSTM	Model BiLSTM dengan attention menghasilkan performa terbaik dengan akurasi, *precision*, *recall*, dan *F1-Score* mendekati 99% pada dataset News Headlines	Menjaga privasi data karena pelatihan terdesentralisasi; attention mechanism mampu menangkap konteks dan kontradiksi emosional; BiLSTM unggul dalam memahami konteks dua arah	BiLSTM memiliki kompleksitas komputasi dan waktu pelatihan tinggi
14	Analisis Sentimen Cuitan Di Media Sosial X Tentang Program Makan	(Munir, 2025)	Analisis opini publik terhadap kebijakan Program Makan	NLP, TF-IDF, K-Means, SVM, BERT	SVM mencapai akurasi **92%**, BERT kurang optimal akibat	SVM efektif pada teks pendek dan data tidak seimbang	BERT membutuhkan data besar dan seimbang

No	Judul	Peneliti	Riset	Metode	Hasil Penelitian	Kelebihan	Keterbatasan
	Bergizi Gratis Dengan Metode Nlp		Bergizi Gratis di media sosial X		data tidak seimbang		
15	Deep Learning-based Sentiment Analysis of Public Comments on Military Education Using RoBERTa Algorithm and Rule-Based *Hybrid* Parameters	(Hidayat dkk., 2025)	Analisis sentimen komentar publik Instagram terkait kebijakan pendidikan militer	RoBERTa + Rule-Based *Hybrid*	Model mencapai akurasi tinggi (±96–97%) dalam klasifikasi sentimen positif, netral, dan negatif	Mampu memahami konteks bahasa Indonesia dan menangani noise bahasa media sosial	Belum secara khusus mendeteksi sarkasme, sehingga potensi salah klasifikasi pada komentar sarkastik masih ada

2.3 Matriks Penelitian

Tabel 2.3 Matriks Penelitian

NO	Penelitian	Tujuan Penelitian		Model												
		Analisis Sentimen	Deteksi Sarkasme	TF-IDF	Glove	Indo BERT	BERT	RoBERTa	LSTM	BiLSTM	SVM	KNN	Naïve Bayes	RF	LG	SVM
1	(Tan dkk., 2022)	✓	-	-	-	-	-	✓	✓	✓	-	-	-	-	-	-
2	(Long Tan dkk., 2022)	✓	-	-	-	-	-	✓	-	✓	-	-	-	-	-	-
3	(Kumar dkk., 2020)	-	✓	-	-	-	-	-	-	✓	-	-	-	-	-	-
4	(Sharma dkk., 2022)	-	✓	-	-	-	-	✓	-	✓	-	-	-	-	-	-
5	(Hassan dkk., 2024)	-	✓	-	-	-	-	✓	-	✓	-	-	-	-	-	-
6	(Handoyo & Suhartono, 2021)	✓	-	-	-	-	-	✓	✓	-	-	-	-	-	-	-
7	(Onan & Tocoglu, 2021)	-	✓	✓	-	-	-	-	-	✓	-	-	-	-	-	-
8	(Modianus Laia dkk., 2025)	✓	-	-	-	-	-	-	-	-	✓	-	-	-	-	-
9	(Eke dkk., 2021)	-	✓	-	-	-	-	-	-	-	-	-	✓	✓	✓	✓
10	(Utama dkk., 2025b)	✓	-	-	-	-	✓	-	-	✓	-	-	-	-	-	-

NO	Penelitian	Tujuan Penelitian		Model												
		Analisis Sentimen	Deteksi Sarkasme	TF-IDF	Glove	Indo BERT	BERT	RoBERTa	LSTM	BiLSTM	SVM	KNN	Naïve Bayes	RF	LG	SVM
11	(Subarkah dkk., 2025)	✓	-	-	-	✓	-	-	-	✓	-	-	-	-	-	-
12	(Mardiana & Abidin, 2025)	✓	-	✓	-	-	-	-	-	-	-	✓	-	-	✓	-
13	(Sami dkk., 2024)	-	✓	-	-	-	-	-	✓	✓	-	-	-	-	-	-
14	(Munir, 2025)	-	-	-	-	-	-	-	-	-	-	-	✓	-	-	-
15	(Hidayat dkk., 2025)	✓	-	-	-	-	-	✓	-	-	-	-	-	-	-	-
16	Hasil Penelitian	✓	✓	✓	✓	-	-	✓	-	✓	-	-	-	-	-	-

Sejalan dengan tujuan penelitian ini, yaitu memperbaiki kesalahan klasifikasi sentimen pada komentar media sosial X (twitter) yang mengandung sarkasme melalui pendekatan RoBERTa-BiLSTM, penelusuran terhadap penelitian-penelitian sebelumnya menunjukkan bahwa penggabungan pemahaman konteks makna teks dan informasi urutan kata masih belum dimanfaatkan secara maksimal. Meskipun penelitian terdahulu telah memberikan kontribusi penting, pendekatan yang digunakan umumnya masih belum mengintegrasikan analisis sentimen dan deteksi sarkasme dalam satu proses klasifikasi, sehingga berpotensi menimbulkan kesalahan klasifikasi pada komentar bernada sindiran.

Berdasarkan kondisi tersebut, penelitian ini memfokuskan pada deteksi sarkasme pada multi-kelas yang dikaitkan langsung dengan analisis sentimen melalui penerapan model *hybrid* RoBERTa-BiLSTM, di mana RoBERTa digunakan untuk analisis sentimen dan BiLSTM digunakan untuk deteksi sarkasme. Pendekatan ini diharapkan mampu meningkatkan ketepatan klasifikasi sentimen positif dan negatif, khususnya pada komentar sarkastik, sehingga hasil analisis sentimen yang diperoleh menjadi lebih akurat dan mencerminkan opini pengguna media sosial secara lebih nyata.