BAB 2
LANDASAN TEORI

2.1 Sarkasme

Sarkasme merupakan bentuk sentimen negatif yang disampaikan melalui ungkapan yang secara permukaan terlihar positif. Berbeda dengan kebohongan, sarkasme tidak bertujuan untuk menipu, melainkan digunakan sebagai sarana untuk menyampaikan kritik atau komentar secara humoris. Menurut beberapa penelitian, sarkasme dapat dipahami sebagai bentuk ujaran yang memiliki makna semantik yang berlawanan dengan arti literalnya. Selain itu, sarkasme juga dapat diartikan sebagai ungkapan yang bersifat tajam, menyindir, atau bernada sindiran yang digunakan untuk mengkritik suatu hal secara tidak langsung [7] .

Lebih lanjut, sarkasme merupakan konsep yang kompleks karena sangat bergantung pada bahasa, konteks, serta pengetahuan umum yang dimiliki oleh penutur dan pendengar. Dalam hal ini, pemahaman terhadap sarkasme tidak hanya ditentukan oleh teks yang diucapkan, tetapi juga oleh situasi dan latar belakang komunikasi.

Dalam NLP, deteksi sarkasme umumnya dipandang sebagai suatu tugas klasifikasi. Tugas ini bertujuan untuk mengelompokkan suatu teks ke dalam kategori sarkasme atau non-sarkasme berdasarkan isi dan konteksnya. Meskipun termasuk bidang penelitian yang relatif baru, deteksi sarkasme memiliki potensi yang besar dan menjadi bagian penting dalam meningkatkan akurasi pemahaman teks, khususnya dalam analisis sentimen [8].

2.2 BERT

BERT (*Bidirectional Encoder Representations from Transformers*) merupakan model representasi bahasa berbasis *deep learning* yang diperkenalkan oleh Jacob Devlin pada tahun 2018. Model ini dirancang untuk memahami konteks kata dalam suatu kalimat secara dua arah , yaitu dengan mempertimbangkan kata sebelum dan sesudahnya secara simultan. Berbeda dengan model sebelumnya seperti *Word2Vec* atau *GloVe* yang bersifat statis, BERT menghasilkan representasi kata yang dinamis tergantung pada konteks kalimat. Pendekatan ini terbukti meningkatkan performa secara signifikan pada berbagai tugas NLP.

BERT tidak hanya mengandalkan kata-kata mentah, tetapi menggunakan

skema embedding yang terstruktur. Setiap urutan input terdiri dari gabungan beberapa jenis embedding yang dijumlahkan untuk membentuk representasi akhir yang dimasukkan ke dalam model [9].

- ***Segment Embeddings*** untuk membedakan antara kalimat pertama dan kalimat kedua dalam satu urutan [9].
- ***Position Embeddings*** berguna memberikan informasi mengenai urutan atau posisi dari setip kata [9].

2.3 Representasi input BERT

Representasi input dalam model BERT dirancang untuk dapat menangani berbagai tugas, dari satu kalimat tunggal maupun berbagai kalimat, melalui struktur yang sangat terorganisasi. Vektor input akhir yang dimasukkan ke dalam blok *Transformer encoder* merupakan hasil penjumlahan elemen demi elemen dari beberapa jenis *embedding* utama: *Segment Embeddings*, dan *Position Embeddings*. Keunggulan algoritma ini terletak pada kemampuannya menangani masalah kata di luar kamus dengan memecah kata menjadi unit sub-kata, sehingga model tetap dapat mengekstraksi fitur semantik dari akar kata tersebut. Selain itu, BERT memperkenalkan token spesial, yaitu token di awal setiap urutan sebagai representasi untuk tugas klasifikasi, dan token yang berfungsi sebagai pemisah antar kalimat atau penanda akhir urutan [10].

Segment Embeddings, berfungsi untuk memberikan informasi kepada model mengenai perbedaan antara dua kalimat dalam satu urutan input, seperti pada tugas tanya-jawab. Terakhir, Position Embeddings ditambahkan karena arsitektur *Transformer* memproses seluruh token secara paralel dan tidak memiliki urutan bawaan seperti pada RNN atau LSTM. Berbeda dengan model *Transformer* asli yang menggunakan fungsi matematika statis, BERT menerapkan *Learned Position Embeddings*. Penggunaan representasi posisi yang dipelajari ini memberikan fleksibilitas lebih bagi model untuk menemukan pola posisi yang paling sesuai dengan data pelatihan selama fase pralatih, yang terbukti meningkatkan performa model dalam memahami struktur kalimat yang kompleks [11] .

2.4 Confusion Matrix

Confusion Matrix adalah tabel yang membandingkan hasil prediksi model dengan nilai sebenarnya dalam dataset pengujian. Tabel 2.1 menjelaskan bagian

merinci jumlah data yang diklasifikasikan dengan benar dan salah untuk setiap kelas.

Tabel 2.1. Confusion Matrix

Kelas	**Klasifikasi Positif**	**Klasifikasi Negatif**
Positif	TP	FP
Negatif	FN	TN

Keterangan :

- *True Positive (TP)*: Jumlah data yang positif dan diprediksi dengan benar sebagai positif oleh model.
- *True Negative (TN)*: Jumlah data yang negatif dan diprediksi dengan benar sebagai negatif oleh model.
- *False Positive (FP)*: Jumlah data yang negatif dan salah diprediksi sebagai positif .
- *False Negative (FN)*: Jumlah data yang positif dan salah diprediksi sebagai negatif .

Berdasarkan komponen di atas, beberapa metriks evaluasi dapat dihitung untuk memberikan gambaran performa model secara menyeluruh:

1. *Accuracy*

   Accuracy (Akurasi) merukapan metrik yang menggambarkan proporsi total prediksi yang benar (positif dan negatif) dibandingkan dengan keseluruhan jumlah data.

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \tag{2.1}$$

2. *Precision*

   Precision (presisi) merupakan metrik yang mengukur tingkat ketepatan antara data yang diminta dengan hasil prediksi yang diberikan oleh model. Metrik ini sangat krusial dalam skenario di mana kesalahan False Positive sangat tinggi.

$$Precision = \frac{TP}{TP + FP} \tag{2.2}$$

3. *Recall*

   Recall (sensifitas) merupakan metrik yang mengukur keberhasilan model dalam menemukan kembali semua informasi relevan dari kelas positif. Metrik ini diprioritaskan ketika kegagalan dalam mendeteksi kelas positif (*False Negative*)

$$Recall = \frac{TP}{TP + FN} \tag{2.3}$$

4. *F1-Score*

   *F1-Score*Merupakan metrik yang menggabungkan rata-rata harmonik dari presisi dan *recall*. Metrik ini digunakan untuk menyelesaikan masalah hubungan terbalik yang sering terjadi antara presisi dan *recall*.

$$F1Score = 2 * \frac{Precision * Recall}{Precision + Recall} \tag{2.4}$$