

Multi-Aspect Sentiment Analysis Pada Review Film Menggunakan Metode Bidirectional Encoder Representations From Transformers (BERT)

Nur Karimah^{1*}, Anna Baita²

^{1,2)} Program Studi Informatika, Fakultas Ilmu Komputer, Universitas Amikom Yogyakarta
Jl. Ring Road Utara, Yogyakarta, Indonesia 55281

*email: nurkarimah@students.amikom.ac.id

(Naskah masuk: 29 September 2023; diterima untuk diterbitkan: 30 Januari 2024)

ABSTRAK – Film telah menjadi kebutuhan utama bagi setiap orang. Film yang ditonton melalui bioskop maupun streaming online membutuhkan biaya sehingga potensial calon penonton akan mempertimbangkan melihat ulasan dari penonton yang telah menonton film tersebut. Untuk dapat menganalisis suatu komentar bernilai positif, negatif, atau netral dapat dilakukan dengan teknik analisis sentimen. Namun, dalam memprediksi polaritas secara keseluruhan pada setiap ulasan tidaklah cukup karena ulasan tersebut dapat memberikan informasi tentang berbagai aspek. Oleh karena itu peneliti mengusulkan pengaplikasian metode Bidirectional Encoder Representation from Transformer (BERT) pada multi-aspect sentiment analysis untuk mengetahui polaritas terhadap aspek-aspek komentar film. Peneliti melakukan percobaan menggunakan model BERT_{BASE-UNCASE} dengan hyperparameters max_epoch 10, batch size 16, dan learning rate 1e-4, 5e-5, 3e-5, dan 2e-5. Hasil Penelitian menunjukkan bahwa, dari semua percobaan didapatkan bahwa nilai accuracy terbaik berada pada percobaan ketiga dengan menggunakan learning rate 3e-5 sebesar 82,32%. Sedangkan nilai precision, recall, dan f1-score terbaik terletak pada aspek animation sebesar 86%, 85%, dan 85%.

Kata Kunci – Analisis sentimen; BERT Base-Uncase; Bidirectional Encoder Representations from Transformer (BERT); Multi-aspect; Review film.

Multi-Aspect Sentiment Analysis of Film Review Using Bidirectional Encoder Representations from Transformers (BERT)

ABSTRACT – Movies have become an essential part of our lives, and people often rely on reviews before watching them, either in theaters or through online streaming platforms. To analyze a comment's sentiment as positive, negative, or neutral, sentiment analysis techniques can be used. However, predicting the overall polarity of reviews is insufficient, as reviews provide information on various aspects of the film. Therefore, the researcher suggests using the Bidirectional Encoder Representation from Transformer (BERT) method in multi-aspect sentiment analysis to determine polarity toward different aspects of film comments. The researcher conducted experiments using the BERT_{BASE-UNCASE} model with hyperparameters, including max_epoch 10, batch size 16, and learning rates of 1e-4, 5e-5, 3e-5, and 2e-5. The results of the study indicate that the third experiment, with a learning rate of 3e-5, achieved the best accuracy, amounting to 82.32%. Meanwhile, the best precision, recall, and f1-score values were obtained for the animation aspect, at 86%, 85%, and 85%, respectively.

Keywords – BERT Base-Uncase; Bidirectional Encoder Representations from Transformer (BERT); Multi-aspect; Review film; Sentiment analysis.

1. PENDAHULUAN

Sentiment analysis adalah proses pengolahan informasi teks untuk menganalisis emosi yang

terkandung di dalam teks [1]. Tujuan dari analisis sentimen adalah untuk mengungkapkan pendapat dari suatu teks khusus [2]. Dalam melakukan sentiment analysis diperlukan pengetahuan tentang

pemrosesan *Neural Network Language (NLP)*, analisis teks, dan komputasi linguistik untuk mendapatkan pengukuran yang bersifat semantik dari informasi yang sedang dilatih [3].

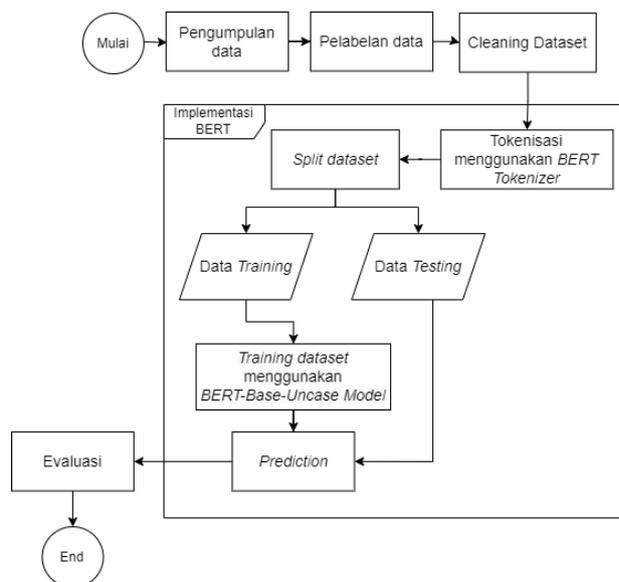
Saat ini terdapat banyak penelitian terkait dengan *sentiment analysis*, khususnya pada *review film* yang dilakukan dengan menggunakan metode *machine learning* seperti *naïve bayes* [4], *SVM* [5], dan *random forest* [6]. Metode *SVM* menjadi metode *machine learning* yang memiliki akurasi yang lebih baik pada beberapa penelitian analisis sentimen [7]. Selain menggunakan metode *machine learning*, penelitian terhadap sentimen analisis juga mulai banyak dilakukan dengan menggunakan metode *deep learning* seperti *LSTM & CNN* [2]; [8]; [9] dan *BERT* [10]. Disisi lain, terdapat penelitian yang dilakukan untuk membandingkan hasil kinerja model dengan menggunakan algoritma *Naïve Bayes*, *Random Forest*, dan *BERT*. Hasil dari penelitian tersebut menyatakan bahwa algoritma *BERT* memiliki performa yang lebih baik dibandingkan dengan menggunakan algoritma *Naïve Bayes* dan *Random Forest* [11].

Namun, dalam memprediksi polaritas secara keseluruhan pada setiap ulasan tidaklah cukup karena ulasan tersebut dapat memberikan komentar tentang berbagai aspek [12] dari film yang bersangkutan. Misalnya, suatu ulasan *review film* dapat menyebutkan *plot*, *acting*, *visual*, *casting*, dan lain-lain. Menganalisis aspek-aspek, daripada ulasan secara keseluruhan, memberikan pemahaman yang lebih baik tentang kelebihan dan kekurangan utama dari suatu film. Oleh karena itu, penelitian ini fokus pada klasifikasi sentimen pada tingkat aspek yang memprediksi setiap aspek.

Dari permasalahan diatas, penelitian ini mengusulkan model klasifikasi sentimen berorientasi *multi-aspect* dengan menggunakan metode *Bidirectional Encoder Representations from Transformers (BERT)*. Dari hasil penelitian ini diharapkan dapat mengetahui kecenderungan polaritas berdasarkan aspek-aspek yang terdapat pada *review film* serta mengetahui kinerja model *BERT* dalam sentimen analisis *review film*.

2. METODE DAN BAHAN

Gambar 1. merupakan alur diagram sistem yang diterapkan pada penelitian ini.



Gambar 1. Alur diagram sistem

Pengumpulan data

Data yang digunakan dalam penelitian ini berasal dari *review film* “*The Little Mermaid*” berbahasa Inggris yang diambil dari situs *IMDB*, *Rotten Tomatoes*, dan *Metacritics*. Data diambil dari tanggal 24 Mei - 05 Juli 2023.

Untuk mendapatkan data dilakukan dengan *web scraping* [21] data dari masing-masing website. Komentar yang digunakan adalah komentar dari *user review* pada masing-masing penyedia platform. Adapun *class* yang digunakan dalam mengambil data pada *IMDB* menggunakan “*text show-more_control*”, pada *Rotten Tomatoes* menggunakan *class* “*audience-reviews_review js-review-text*”, dan pada *Metacritics* menggunakan *class* “*blurb blurb_collapsed*”.

Pelabelan data

Pelabelan dilakukan secara manual yang dilakukan oleh peneliti sendiri. Pada setiap ulasan akan diberikan dua label yaitu label aspek dan label sentimen. Untuk label sentimen terbagi menjadi tiga yaitu positif dengan simbol angka (1), negatif dengan simbol angka (-1), dan netral dengan simbol angka (0). Sedangkan untuk label aspek terbagi menjadi lima yaitu *acting*, *plot*, *cast*, *animation*, dan *music*. Pemilihan aspek dilakukan dengan melihat topik yang sering dibahas oleh para *reviewers*.

Cleaning data

Tahap *cleaning data* akan membuang karakter yang tidak memiliki makna dan membuat data menjadi lebih konsisten. Pada penelitian ini *cleaning data* dilakukan dengan menghapus *tag HTML*, *case folding*, menghapus angka, tanda baca, dan karakter.

Multi-Aspect Sentiment Analysis

Pada tugas klasifikasi ini menggunakan klasifikasi tingkat aspek atau *Aspect Level Sentiment Classification (ALCS)* yang bertujuan untuk mendeteksi sentimen terhadap aspek tertentu dalam sebuah kalimat. *ALCS* sangat penting untuk kalimat multi-aspect yang mengandung banyak aspek [13].

Multi-aspect sentiment analysis memformalkan ulang dari klasifikasi tingkat aspek. Diberikan sebuah kalimat $s = \{w_1, w_2, \dots, w_n\}$ dan aspect $m = \{a_1, a_2, \dots, a_m\}$ terdapat di s , tujuan dari multi-aspect sentiment analysis adalah untuk mendeteksi sentimen polaritas $\{y_1, y_2, \dots, y_m\}$ dari semua aspek $\{a_1, a_2, \dots, a_m\}$, dimana y_i berhubungan dengan label sentimen pada aspect a_i [14]. Contohnya pada kalimat review film yang berbunyi "The special effect was fantastic but the acting was bad". Kalimat tersebut menunjukkan terdapat dua aspek yang terkandung didalamnya, yaitu *effect* yang memiliki kelas positif dan *acting* memiliki kelas negatif.

Implementasi BERT

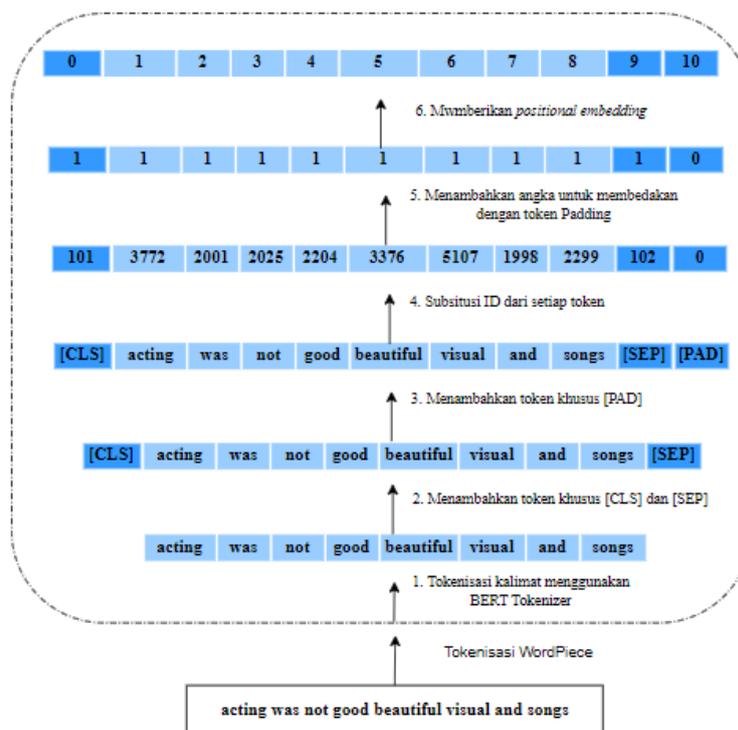
Sebelum melakukan *training*, data yang dikumpulkan harus diubah menjadi bentuk representasi *input* yang diterima oleh *BERT*, yakni dengan menggunakan *tokenizer*. *Tokenizer* dilakukan untuk mengubah suatu kata ke dalam sebuah token yang dapat dipahami oleh model. Hal ini dilakukan dengan menggunakan *vocabulary* tertentu tergantung pada model yang digunakan. *Vocabulary* yang dibuat

menggunakan model *WordPiece* [15].

BERT akan menerima urutan kalimat dengan panjang kalimat yang sama pada setiap *input*. Panjang *input* maksimum yang dapat diterima oleh *BERT* adalah 512 kata [15]. Hal ini disebabkan karena *encoder* pada *transformer* hanya dapat menghasilkan *output* dengan dimensi 512 kata. Apabila panjang kalimat yang diterima kurang dari 512 kata, maka kalimat harus ditambahkan. Sebaliknya, jika panjang kalimat yang diterima melebihi dari 512 kata, maka kalimat akan dipotong.

Terdapat beberapa token khusus yang diinputkan ke dalam *BERT* untuk menginisiasi hubungan antara satu kata dengan kata berikutnya ataupun satu kalimat dengan kalimat berikutnya. Kata-kata yang tidak ada dalam *vocabulary* akan diberikan label *unknown (UNK)*. Setiap awal kalimat diberikan tanda khusus berupa token *classification [CLS]* yang digunakan sebagai indikator sebuah awal kalimat. Pada bagian akhir kalimat diberikan token khusus *separator [SEP]* yang berfungsi sebagai pemisah kalimat satu dengan kalimat berikutnya. Untuk kalimat yang memiliki panjang kalimat yang lebih sedikit dari panjang kalimat yang telah ditentukan, perlu token *padding [PAD]*.

Proses tahapan tokenisasi akan menghasilkan *input* pada *BERT*. Gambar 2. menggambarkan proses keseluruhan dari proses tokenisasi menggunakan *BERT Tokenizer*. Sedangkan Gambar 3. merupakan gambaran *representasi Input* pada *BERT*.



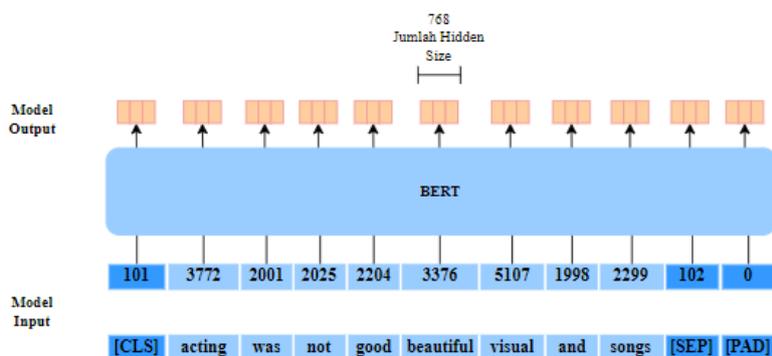
Gambar 2. Proses tokenisasi menggunakan *BERT tokenizer*

<i>Input</i>	[CLS]	acting	was	not	good	beautiful	visual	and	songs	[SEP]	[PAD]
<i>Token</i>	E[CLS]	Eacting	Ewas	Enot	Egood	Ebeautiful	Evisual	Eand	Esongs	E[SEP]	E[PAD]
<i>Embedding</i>	+	+	+	+	+	+	+	+	+	+	+
<i>Segmen</i>	E1	E1	E1	E1	E1	E1	E1	E1	E1	E1	E0
<i>Embedding</i>	+	+	+	+	+	+	+	+	+	+	+
<i>Position</i>	E0	E1	E2	E3	E4	E5	E6	E7	E8	E9	E10
<i>Embedding</i>											

Gambar 3. Representasi *Input* pada *BERT* [15]

Penelitian ini menggunakan model *BERT*_{BASE-UNCASE} [16]. Model ini dibangun menggunakan 12 *layer* dan memberikan *output* berupa *vektor* dengan ukuran *hidden size* sebesar 768

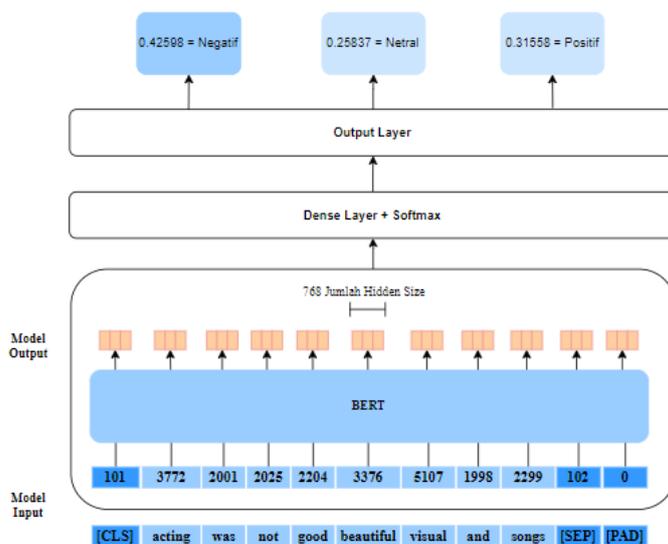
yang ditunjukkan pada Gambar 4. dalam proses *sentiment analysis output* dari token [CLS] akan digunakan sebagai *input* untuk klasifikasi.



Gambar 4. *Input* dan *output* dalam *BERT*

Dalam melakukan klasifikasi, penelitian ini menggunakan *layer Dense/Fully Connected Layer* seperti pada Gambar 5. yang terhubung pada *output BERT*. Lapisan *Dense* ini dapat digunakan untuk mengubah representasi teks menjadi menjadi prediksi kelas yang spesifik. Lapisan *Dense* akan menerima *input* dari token [CLS] sebagai representasi

kalimat. Vektor ini mewakili pemahaman *BERT* terhadap kalimat tersebut. Penelitian ini menggunakan tiga kelas, yakni positif, negatif, dan netral. Maka lapisan *Dense* terakhir akan memiliki tiga *neuron*. Gambar 5. menunjukkan proses klasifikasi menggunakan klasifikasi *BERT*.



Gambar 5. Ilustrasi Proses Klasifikasi menggunakan *BERT*

Setelah BERT melakukan *pre-train* dengan *dataset* yang ada, model akan melakukan *training*. Sebelum melakukan *training*, *dataset* akan dibagi menjadi dua, yakni data *training* dan data *testing*. Penelitian ini menggunakan data *loader* untuk membantu menghemat memori dan meningkatkan kecepatan selama *training*. *DataLoader* merupakan fungsi pada *PyTorch* yang berperan sebagai *iterator*. *AdamW* digunakan sebagai *optimizer* untuk mengoreksi *weight* dari kalimat [15].

Pada penelitian yang dilakukan oleh Devlin dkk [15] merekomendasikan *hyperparameters* untuk *fine-tuning* yang menunjukkan performa yang optimal untuk tugas spesifik, yaitu:

1. *Batch size*: 16 dan 32
2. *Epoch*: 2, 3, dan 4
3. *Learning rate (AdamW)*: $5e^{-5}$, $3e^{-5}$, dan $2e^{-5}$

Misalkan panjang maksimal urutan kalimat yang akan dianalisa telah ditentukan yaitu 15. Maka hasil dari tahap preprocessing yang dapat diterima oleh BERT dapat dilihat pada Tabel 1. berikut:

Tabel 1. Input pada BERT

Kalimat	acting and characteristically was not good but beautiful visual and song
Kalimat ditokenisasi menggunakan <i>WordPiece</i>	'acting', 'and', 'characteristic', '##ally', 'was', 'not', 'good', 'but', 'beautiful', 'visual', 'and', 'song'
Kalimat diberikan token khusus [CLS], [SEP], dan [PAD]	[[CLS]', 'acting', 'and', 'characteristic', '##ally', 'was', 'not', 'good', 'but', 'beautiful', 'visual', 'and', 'song', '[SEP]', '[PAD]']
ID dari tiap token	[101, 3772, 1998, 8281, 3973, 2001, 2025, 2204, 2021, 3376, 5107, 1998, 2299, 102, 0]
<i>Attention Mask</i>	[1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 1, 0]

Pada Tabel 1. kalimat “*acting and characteristically was not good but beautiful visual and song*” ditokenisasi menggunakan *WordPiece* dan terdapat kata yang tidak masuk ke dalam *vocabulary* atau disebut juga dengan *Out-Of-Vocabulary (OOV)*. Kata yang tidak terdapat pada *vocabulary* akan dipecah menjadi dua *sub* kata yang ditandai menggunakan tanda *##* sebagai kata yang mengikuti kata sebelumnya. Setelah memecah kalimat menjadi token-token, indeks dari *vocabulary* disesuaikan dengan tiap token yang ada pada kalimat yang disebut *token ids*. *Attention Mask* berfungsi untuk memberitahu model

mana yang harus diperhatikan dan yang tidak diperhatikan.

Evaluasi

Tahapan evaluasi yang digunakan pada penelitian ini adalah dengan menggunakan *confusion matrix 3x3*. *Confusion matrix* merupakan sebuah teknik evaluasi yang berisi informasi hasil kelas dari prediksi yang sebenarnya. *Confusion matrix* memiliki dua kelas, yakni berisi kelas yang sebenarnya dan kelas hasil prediksi model [17].

Nilai akurasi tertinggi yang didapat dari proses *training* sebelumnya akan menjadi nilai akurasi model. Parameter yang digunakan untuk data uji adalah *True Positive (TP)*, *False Positive (FP)*, *True Neutral (TNt)*, *False Neutral (FNt)*, *True Negative (TN)*, dan *False Negative (FN)*. Untuk memperoleh prediksi dari model, *confusion matrix* yang digunakan dapat dilihat pada Tabel 2.

Tabel 2. Confusion Matrix

Confusion Matrix	True Class		
	Positive	Neutral	Negative
Positive	True Positive (TP)	False Negative (FNt)	False Positive (FP)
Neutral	False Neutral (FNt)	True Neutral (TNt)	False Neutral (FNt)
Negative	False Negative (FN)	False Neutral (FNt)	True Negative (TN)

Beberapa *matrix* yang digunakan pada penelitian ini adalah *accuracy*, *f1-score*, *precision*, dan *recall*.

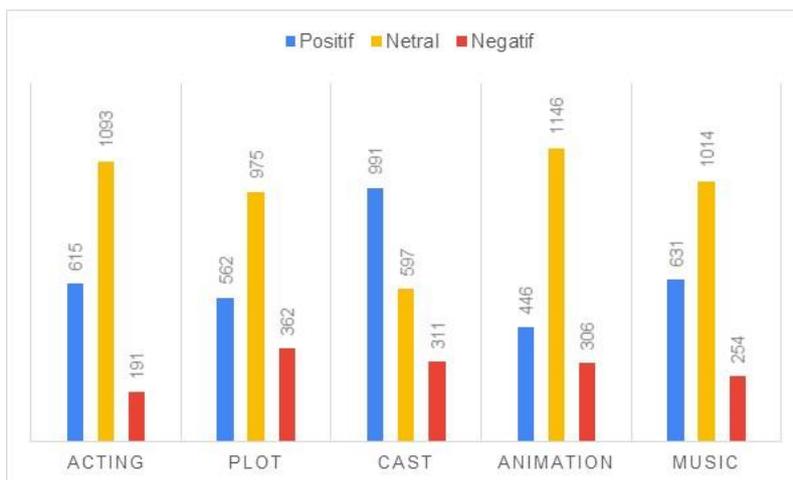
3. HASIL DAN PEMBAHASAN

Pengumpulan Data

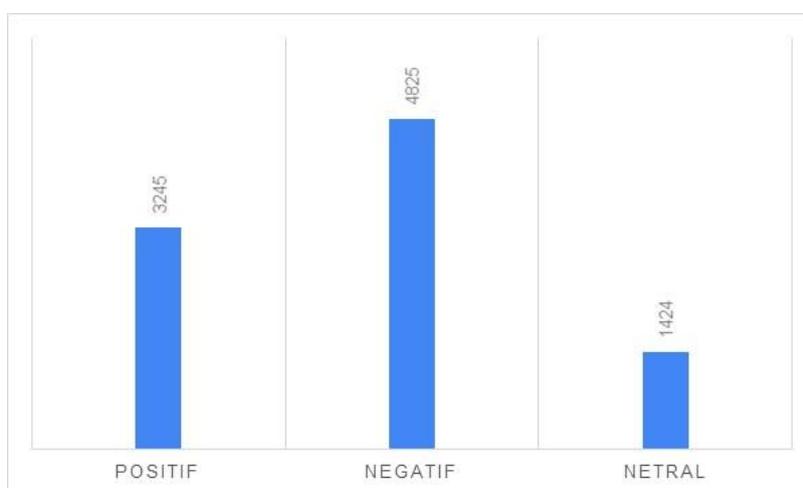
Data dikumpulkan dengan menggunakan teknik *scraping* pada beberapa situs penyedia layanan rating film. *Scraping* akan menghasilkan data dalam bentuk *JSON*. Oleh karena itu, peneliti perlu mengekstrak data tersebut ke dalam bentuk *CSV*, kemudian data dikumpulkan menjadi satu dokumen. Adapun detail data yang berhasil dikumpulkan oleh peneliti dapat dilihat pada Tabel 3.

Tabel 3 Pengumpulan data

No	Situs	Total
1	IMDB	810
2	Rotten Tomatoes	768
3	Metacritic	320



Gambar 7. Visualisasi *dataset* berdasarkan jumlah aspek dan sentimen



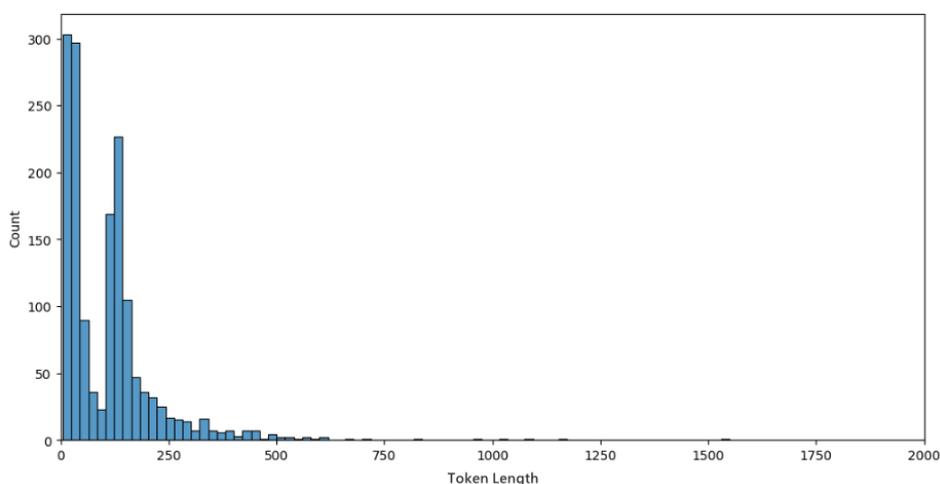
Gambar 8. Visualisasi *dataset* berdasarkan jumlah sentimen

Komentar yang memiliki sentimen positif sebanyak 3245, sentimen netral sebanyak 4825, dan sentimen negatif sebanyak 1424.

Implementasi BERT

Implementasi *BERT* akan dilakukan beberapa tahapan, yakni *split data*, *tokenisasi*, pemodelan,

training, dan *testing*. Pembagian data *training* dan data *testing* dilakukan dengan menggunakan perbandingan 80:20. Setelah *dataset* dibagi, kemudian model akan melakukan tokenisasi pada *dataset*. Proses tokenisasi menggunakan *BERT Tokenizer*. Gambar 9. adalah panjang token yang dimiliki oleh *dataset* dalam penelitian ini.



Gambar 9. Grafik Panjang Token

Dari grafik pada Gambar 9. menunjukkan bahwa *dataset* memiliki panjang token lebih dari 512 kata. Oleh sebab itu, penelitian ini akan menggunakan panjang maksimal *BERT* yakni 512 kata dalam *train dataset*. Setelah itu, akan dilakukan pembungkusan proses tokenisasi dalam *Dataset Pytorch*, serta mengonversi label menjadi *tensor*.

Model yang digunakan pada penelitian ini adalah *BERT Model*, yakni *BERT-Base-Uncase* yang telah dilatih sebelumnya.

Hasil Klasifikasi Sentimen

Penelitian ini menggunakan *fine tuning* dengan menggunakan *hyperparameters*. *Hyperparameters* ditentukan melalui beberapa percobaan untuk menentukan perbandingan serta mencari hasil terbaik. Pada percobaan kali ini, peneliti akan menggunakan *hyperparameters* sebagai berikut:

- *Batch Size*: 16
- *Learning rate*: $1e^{-4}$, $5e^{-5}$, $3e^{-5}$, $2e^{-5}$
- *Max_Epoch*: 10

Learning rate dipilih berdasarkan rekomendasi *BERT* [15]. *Batch Size* dipilih karena jika menggunakan *batch size* yang lebih besar, semakin memakan waktu yang lebih lama.

Train loss, *validation loss/val loss*, dan *Test loss* merupakan *matrix* yang digunakan dalam proses *training* dan evaluasi model. *Train loss* digunakan untuk mengukur seberapa besar selisih antara prediksi model dan label yang sebenarnya pada data pelatihan. *Val loss* digunakan untuk membantu mengevaluasi kinerja model pada data yang tidak pernah dilihat selama pelatihan. Jika nilai *val loss* meningkat atau tetap stabil menandakan terdapat indikasi *overfitting*. *Test loss* memberikan informasi tentang seberapa baik model melakukan generalisasi pada data baru.

Semakin kecil nilai *loss*, maka semakin baik performa model tersebut. Hal ini menunjukkan bahwa prediksi model semakin mendekati nilai label yang sebenarnya pada data yang digunakan. Tabel 8. menunjukkan nilai *train loss*, *val loss*, dan *test loss* pada masing-masing *learning rate* yang digunakan pada penelitian ini.

Tabel 8. Nilai *train loss*, *val loss*, dan *test loss*

Learning Rate	$1e^{-4}$	$5e^{-5}$	$3e^{-5}$	$2e^{-5}$
<i>Train Loss</i>	0.005	0.1	0.073	0.096
<i>Val Los</i>	0.579	0.308	0.306	0.31
<i>Test Loss</i>	0.603	0.318	0.31	0.318

Dari Tabel 8. disimpulkan bahwa *learning rate* yang memiliki *test loss* dan *val loss* terbaik berada pada *learning rate* $3e^{-5}$ dengan nilai *test loss* sebesar 0.31. Sedangkan *train loss* terbaik berada pada

learning rate $1e^{-4}$ senilai 0.005.

Selain mengetahui nilai *train loss*, *val loss*, dan *test loss* terbaik pada masing-masing percobaan, diketahui juga hasil dari *ouput tensor BERT* dalam melakukan prediksi probabilitas sentimen. Tabel 9. menunjukkan ringkasan dari seluruh nilai *output probabilitas* yang didapat pada tugas *multi-aspect sentiment analysis* menggunakan *BERT*.

Tabel 9. Nilai *Output Probabilitas* pada Setiap Percobaan

Aspek	Learning Rate			
	$1e^{-4}$	$5e^{-5}$	$3e^{-5}$	$2e^{-5}$
<i>Acting</i>	0.5756	0.5329	0.4826	0.4170
<i>Plot</i>	0.5673	0.5336	0.5265	0.5317
<i>Cast</i>	0.5754	0.4140	0.5308	0.4935
<i>Animation</i>	0.5749	0.3837	0.4213	0.4596
<i>Music</i>	0.5701	0.5277	0.5366	0.5271

Dari Tabel 9. menunjukkan bahwa probabilitas terbaik pada aspek *acting*, *plot*, *cast*, *animation*, dan *music* berada pada percobaan menggunakan *learning rate* $1e^{-4}$.

Evaluasi Model

Dalam melakukan analisis sentimen menggunakan *BERT*, Model yang memiliki nilai akurasi terbaik kemudian diuji dengan menggunakan *dataset testing*. Setelah mendapatkan nilai akurasi, model kemudian akan mencoba untuk memprediksi menggunakan data *testing*.

Tabel 10. menunjukkan perbandingan akurasi yang dilakukan selama pelatihan *dataset* dengan menggunakan *hyperparameters learning rate* $1e^{-4}$, $5e^{-5}$, $3e^{-5}$, dan $2e^{-5}$.

Tabel 10. Nilai *Accuracy* pada setiap percobaan

Learning Rate	Accuracy
$1e^{-4}$	0.8063
$5e^{-5}$	0.8126
$3e^{-5}$	0.8232
$2e^{-5}$	0.8100

Setelah melakukan *testing* terhadap *dataset*, diperoleh hasil akurasi secara keseluruhan menggunakan *BERT*. Dari semua percobaan didapatkan bahwa akurasi terbaik berada pada percobaan keempat dengan menggunakan *learning rate* $3e^{-5}$ dengan hasil akurasi sebesar 0.8232.

Peneliti juga menghitung *confusion matrix* pada parameter yang memiliki nilai akurasi terbaik dari masing-masing percobaan, yakni pada *learning rate* $3e^{-5}$. Pada Tabel 11. menunjukkan nilai *confusion matrix* pada *learning rate* $3e^{-5}$.

Tabel 11. Nilai *confusion matrix*

Learning Rate: 3e-5			
Aspek	Precision	Recall	F1-score
Acting	0.79	0.79	0.79
Plot	0.73	0.71	0.72
Cast	0.81	0.79	0.80
Animation	0.86	0.85	0.85
Music	0.79	0.82	0.81
Average	0.796	0.792	0.794

Tabel 11. menunjukkan bahwa aspek yang memiliki kinerja paling baik dalam melakukan prediksi terletak pada aspek *animation*. Sedangkan kinerja model yang kurang baik terletak pada aspek *plot*. Hal ini disebabkan karena aspek *plot* memiliki data yang bersifat ambiguitas ketika menentukan sentimen.

4. KESIMPULAN

Penerapan sentimen analisis berfokus pada *multi-aspect* untuk melihat aspek yang memengaruhi penilaian terhadap ulasan film. Aspek yang digunakan antara lain aspek *acting*, *plot*, *cast*, *animation*, dan *music*. Analisis sentimen dilakukan dengan menggunakan Metode *Bidirectional Encoder Representations from Transformers (BERT)*. Penelitian ini melakukan beberapa percobaan dengan menggunakan *hyperparameters learning rate* $1e^{-4}$, $5e^{-5}$, $3e^{-5}$, dan $2e^{-5}$. Hasil terbaik pada percobaan *multi-aspect sentiment analysis* terdapat pada percobaan menggunakan *learning rate* $3e^{-5}$ dengan nilai *accuracy* sebesar 82,32%. Sedangkan kinerja model untuk masing-masing aspek yang memiliki nilai terbaik terletak pada aspek *animation* dan aspek yang memiliki kinerja model kurang baik terletak pada aspek *plot*. *Dataset* yang digunakan memiliki data yang tidak seimbang pada setiap aspek dan kelas sehingga memengaruhi hasil kinerja model yang dibangun. Dengan sebaran *dataset* yang tidak seimbang, kinerja model yang diperoleh sudah cukup baik dalam memprediksi polaritas data. Untuk meningkatkan kinerja model, saran yang dapat penulis berikan yaitu peneliti dapat menggunakan metode dalam menangani ketidakseimbangan data dalam distribusi kelas maupun aspek. Selain itu, peneliti dapat menggunakan model *pre-train BERT* yang lainnya, seperti *BERT-Large-Uncase*, *Distil BERT*, dan yang lainnya untuk mendapatkan kinerja model yang lebih baik.

DAFTAR PUSTAKA

- [1] J. D. Bodapati, N. Veeranjanyulu, and S. Shaik, "Sentiment analysis from movie reviews using LSTMs," *Ing. des Syst. d'Information*, vol. 24, no. 1, pp. 125–129, 2019, doi: 10.18280/isi.240119.
- [2] N. Mohamed Ali, M. M. A. El Hamid, and A. Youssif, "Sentiment Analysis for Movies Reviews Dataset Using Deep Learning Models," *Int. J. Data Min. Knowl. Manag. Process*, vol. 09, no. 03, pp. 19–27, 2019, doi: 10.5121/ijdkp.2019.9302.
- [3] H. R. Alhakiem and E. B. Setiawan, "Aspect-Based Sentiment Analysis on Twitter Using Logistic Regression with FastText Feature Expansion," *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 6, no. 5, pp. 840–846, 2022, doi: 10.29207/resti.v6i5.4429.
- [4] M. T. Razaq, D. Nurjanah, and H. Nurrahmi, "Analisis Sentimen Review Film menggunakan Naive Bayes Classifier dengan fitur Tf-Idf," vol. 9, no. 5, pp. 6053–6071, 2022.
- [5] N. G. Ramadhan and T. I. Ramadhan, "Analysis Sentiment Based on IMDB Aspects from Movie Reviews using SVM," *Sinkron*, vol. 7, no. 1, pp. 39–45, 2022, doi: 10.33395/sinkron.v7i1.11204.
- [6] M. A. A. Jihad, Adiwijaya, and W. Astuti, "Analisis Sentimen Terhadap Ulasan Film Menggunakan Algoritma Random Forest Muhammad," *e-Proceeding Eng.*, vol. 8, p. 10153, 2021.
- [7] G. Cahyani, W. Widayani, S. D. Anggita, Y. Pristyanto, I. Ikmah, and A. Sidauruk, "Klasifikasi Data Review IMDb Berdasarkan Analisis Sentimen Menggunakan Algoritma Support Vector Machine," *J. Media Inform. Budidarma*, vol. 6, no. 3, p. 1418, 2022, doi: 10.30865/mib.v6i3.4023.
- [8] K. Dashtipour, M. Gogate, A. Adeel, H. Larijani, and A. Hussain, "Sentiment analysis of persian movie reviews using deep learning," *Entropy*, vol. 23, no. 5, pp. 1–16, 2021, doi: 10.3390/e23050596.
- [9] W. Widayat, "Analisis Sentimen Movie Review menggunakan Word2Vec dan metode LSTM Deep Learning," *J. Media Inform. Budidarma*, vol. 5, no. 3, p. 1018, 2021, doi: 10.30865/mib.v5i3.3111.
- [10] H. Jayadianti, W. Kaswidjanti, A. T. Utomo, and S. Saifullah, "Sentiment analysis of Indonesian reviews using fine-tuning IndoBERT and R-CNN," *Ilk. J. Ilm.*, vol. 14, no. 3, pp. 348–354, 2022, [Online]. Available: <http://dx.doi.org/10.33096/ilkom.v14i3.1505.348-354>
- [11] N. Husin, "Komparasi Algoritma Random Forest, Naïve Bayes, dan Bert Untuk Multi-Class Classification Pada Artikel Cable News Network (CNN)," *J. Esensi Infokom*, vol. 7, no. 1, pp. 75–84, 2023.

- [12] N. Alghamdi, S. Khatoon, and M. Alshamari, "Multi-Aspect Oriented Sentiment Classification: Prior Knowledge Topic Modelling and Ensemble Learning Classifier Approach," *Appl. Sci.*, vol. 12, no. 8, 2022, doi: 10.3390/app12084066.
- [13] M. Hu *et al.*, "Can: Constrained attention networks for multi-aspect sentiment analysis," *EMNLP-IJCNLP 2019 - 2019 Conf. Empir. Methods Nat. Lang. Process. 9th Int. Jt. Conf. Nat. Lang. Process. Proc. Conf.*, pp. 4601-4610, 2019, doi: 10.18653/v1/d19-1467.
- [14] Z. Wu, C. Ying, X. Dai, S. Huang, and J. Chen, "Transformer-Based Multi-aspect Modeling for Multi-aspect Multi-sentiment Analysis," *Nat. Lang. Process. Chinese Comput.*, vol. 12431, 2020.
- [15] J. Devlin, M. W. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of deep bidirectional transformers for language understanding," *Proc. naacL-HLT*, vol. 1, no. Mlm, pp. 4171-4186, 2019.
- [16] M. P. Geetha and D. Karthika Renuka, "Improving the performance of aspect based sentiment analysis using fine-tuned Bert Base Uncased model," *Int. J. Intell. Networks*, vol. 2, no. March, pp. 64-69, 2021, doi: 10.1016/j.ijin.2021.06.005.
- [17] A. N. Azhar and M. L. Khodra, "Fine-tuning Pretrained Multilingual BERT Model for Indonesian Aspect-based Sentiment Analysis," *2020 7th Int. Conf. Adv. Informatics Concepts, Theory Appl. ICAICTA 2020*, 2020, doi: 10.1109/ICAICTA49861.2020.9428882.