

Analisis Sentimen *Review* Film Berbahasa Inggris Dengan Pendekatan *Bidirectional Encoder Representations from Transformers*

Cindy Alifia Putri*¹, Adiwijaya², Said Al Faraby³

^{1,2,3}Universitas Telkom; Jl. Telekomunikasi Bandung, Telp(022)7565930

School of Computing Telkom University, Bandung 40257

e-mail: ¹cindyalifiaputri@student.telkomuniversity.ac.id, ²adiwijaya@telkomuniversity.ac.id,

³saidalfaraby@telkomuniversity.ac.id

Abstract

Sentiment Analysis is the process of analyzing a person's opinion or attitude. Sentiment analysis is used to get the results of an analysis of various opinions or judgments of someone in providing comments or opinions. In this study, the authors conducted a sentiment analysis of film reviews using the cornelledu dataset from pabo for film reviews with a classification process using the Bidirectional Encoder Representations from Transformers (BERT) algorithm that performs fine-tuning with some layer for classification. From this study, obtained the accuracy of the results calculated by the confusion matrix of 73%.

Keywords—Accuracy, Bidirectional Encoder Representations from Transformers, Confusion Matrix, Dataset, Sentiment Analysis

Abstrak

Analisis Sentimen adalah proses analisis terhadap suatu pendapat atau sikap seseorang. Analisis sentimen digunakan untuk mendapatkan suatu hasil analisa terhadap berbagai macam pendapat atau sikap seseorang dalam memberikan komentar atau opininya. Pada penelitian ini, penulis melakukan klasifikasi sentiment analysis terhadap review film dengan menggunakan Dataset cornelledu dari pabo untuk movie review dengan proses klasifikasi menggunakan algoritma Bidirectional Encoder Representations from Transformers (BERT) yang dilakukan fine-tuning dengan beberapa layer untuk klasifikasi. Dari penelitian ini, didapatkan hasil akurasi yang dihitung dengan menggunakan confusion matrix sebesar 73%.

Kata kunci—Akurasi, Bidirectional Encoder Representations from Transformers, Confusion Matrix, Dataset, Sentiment Analysis

1. PENDAHULUAN

Pesatnya perkembangan teknologi saat ini menyebabkan tidak ada satupun informasi yang tidak dapat diakses dan dicari tahu. Salah satu informasi yang sering diakses adalah *review* dari sebuah film. *Review* film adalah suatu hal yang pasti pernah dibaca dan diungkapkan oleh semua orang baik itu dalam ucapan atau tulisan. Banyak sekali situs yang menyediakan tentang suatu produk yang dapat mencerminkan tentang pendapat pengguna [1] salah satunya adalah tentang *review* film. Pengguna situs-situs *website* untuk *review* film semakin lama semakin bertambah hal tersebut juga diiringi dengan banyaknya komentar-komentar yang ditinggalkan oleh pengguna *website* tersebut. Banyaknya *review* film semakin hari semakin membuat

pengguna tidak dapat mengetahui kualitas dari film tersebut, maka dalam studi ini akan dibahas bagaimana melakukan analisa terhadap *review* film agar dapat dilakukan teknik pengklasifikasian *review* film sehingga diketahui kualitas baik buruknya suatu film dengan lebih mudah.

Proses untuk menganalisa *review* film tersebut disebut dengan sentiment analisis. Jadi, *sentiment analysis* adalah proses yang bertujuan untuk menentukan isi dari dataset yang berbentuk teks (dokumen, kalimat, paragraf, dll) yang memiliki sifat positif, negatif, atau netral [2]. Pendekatan yang digunakan untuk melakukan *sentiment analysis* terhadap data dari *review* film adalah dengan menggunakan pendekatan klasifikasi menggunakan algoritma Bidirectional Encoder Representations from Transformers (BERT). Data *review* film yang digunakan merupakan dataset *review* film yang diunduh dari situs *website cornelledu* (Data Movie review polarity dataset v2.0) [3]. Dataset ini merupakan dataset dimana satu

baris datanya adalah berupa dokumen, tidak seperti dataset dari IMDb yang datasetnya berupa kalimat untuk satu barisnya.

Dalam penelitian-penelitian terdahulu, telah banyak sekali contoh penelitian yang membahas tentang *sentiment analysis*. Terdapat bermacam-macam data yang digunakan dalam implementasi dalam penyelesaian *task* terhadap analisis sentiment tersebut. Macam-macam data yang digunakan adalah *movie review* [4], teks hadits [5, 6], dan bahkan komentar di sosial media [7, 8].

Sejauh ini, baru terdapat sedikit penelitian terkait dengan analisis sentiment dengan menggunakan algoritma BERT yang dilakukan *fine-tuning* dengan beberapa layer. Penelitian sebelumnya adalah yang dilakukan oleh Alexander Rietzler [9] yang berhasil memperoleh akurasi mendekati *state-of-the-art* 79.19% dengan menggunakan dataset dari Pontiki [10] yaitu laptops dataset. Pada penelitian tersebut, Alexander Rietzler mendapatkan hasil yang sangat bagus akan tetapi dataset yang digunakan adalah dataset dimana untuk satu barisnya adalah berupa kalimat, sedangkan dalam penelitian ini dataset untuk setiap barisnya adalah satu dokumen.

Selanjutnya terdapat penelitian dari Victor SanH [11] yang mendapatkan akurasi sangat tinggi dengan menggunakan dataset dari IMDb yaitu sebesar 93.46%. Hasil dari penelitian yang dilakukan oleh Victor SanH memang mendapatkan hasil yang sangat bagus, akan tetapi masih belum dapat mengalahkan *state-of-the-art* untuk klasifikasi analisis sentiment terhadap dataset dari IMDb yang dilakukan oleh Tan Thongtan dan Tanasanee Phienthrakul [12] yang mendapatkan akurasi sebesar 97.42%.

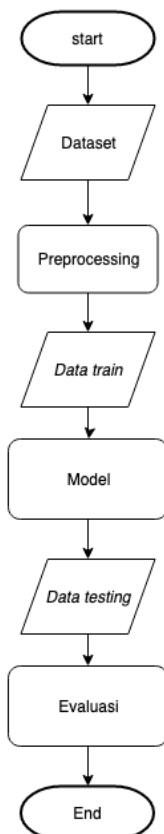
Penelitian lainnya adalah dari Asriyanti Indah Pratiwi [4] yang memakai dataset sama dengan yang digunakan dalam penelitian ini akan tetapi algoritma yang digunakan bukanlah menggunakan BERT. Penelitian tersebut menggunakan *Information Gain* untuk proses klasifikasi dan didapatkan hasil akurasi yang besar yaitu sebesar 88.5%.

Sebelum menggunakan metode BERT, penulis membuat model klasifikasi menggunakan algoritma *naïve bayes* sebagai *baseline* dengan *data train*, *data test*, dan *preprocessing* yang digunakan sama dengan yang digunakan dalam metode BERT dan memperoleh hasil akurasi sebesar 48%. Dengan akurasi sebesar 48% tentu perlu dilakukan improvisasi untuk meningkatkan hasil akurasi tersebut. Disisi lain, penggunaan metode BERT masih sangat jarang digunakan untuk proses klasifikasi dengan dataset berbentuk dokumen sebelumnya, hal tersebut dikarenakan metode ini memiliki kekurangan dalam hal waktu training dan biaya untuk running program yang sangat besar. Untuk itu, penulis menggunakan BERT-base yang memiliki keterbatasannya itu hanya dapat melakukan *learning* terhadap dataset dengan jumlah hanya 128 karakter dari setiap datanya. Hal tersebut, yang memberikan tanda tanya apakah BERT-base dapat digunakan klasifikasian *analysis sentiment* untuk dataset berupa dokumen? Oleh karena itu, penelitian ini memiliki arti yang penting untuk mengetahui hasil

apakah BERT-base dapat digunakan untuk proses klasifikasi *analysis sentiment* terhadap dataset berupa dokumen.

2. METODE PENELITIAN

Flowchart pada penelitian ini, digambarkan oleh gambar 1 berikut ini:



Gambar 1. *Flowchart System*

Langkah penting dalam penelitian ini adalah dataset yang digunakan, *preprocessing*, pembuatan model, dan juga evaluasi.

2.1 Dataset

Pengumpulan dataset dilakukan dengan cara mengunduh dari situs *website cornelledu* [3] dimana data yang digunakan adalah data *movie review polarity dataset v2.0*. Ketika data selesai diunduh, maka akan terdapat sebanyak 2000 *review* film dimana 1000 diantaranya adalah *review* positif dan 1000 data sisanya adalah *review* negatif.

Tabel 1. Dataset Awal

No	Dataset
1	how do films like mouse hunt get into theatres ? isn't there a law or something ? this diabolical load of claptrap . . .
2	it was with a huge lack of something to do that i decided to watch this on good old upn on sundayafternoon , when the only good things on tv are the second-rate movies they

	show (some are good . . .
3	note : ordinarily , moviereviews . org will not give away any critical plot points of a film that could be interpreted as " spoilers
4	i have to admit that i disliked this film initially . it certianly isn't for every taste , and it's sheer torture to sit through
5	when i first heard that kevincostner was making a movie called " the postman , " i thought , " an american version of 'ilpostino ?

Pada Tabel 1 terdapat beberapa contoh dataset yang diambil dari *file negative* dan *file positive*. Contoh nomor 1 dan 2 adalah contoh dataset yang diambil dari *file negative*, sedangkan nomor 3, 4, dan 5 diambil dari *file positive*.

2.2 Preprocessing

Preprocessing teks yang dilakukan pada tahapan ini berguna untuk menyiapkan dataset atau membersihkan data yang akan digunakan untuk proses *training*. Dalam proses training, data yang diberikan dan siap untuk dilakukan pelatihan dalam pembuatan model adalah data yang bersih dan bebas dari *noise*. *Noise* dalam sebuah teks dapat berupa *stopwords*, ataupun kata-kata yang muncul hanya 1 atau 2 kali dalam dataset, sehingga perlu dilakukan penghapusan kata-kata yang tidak signifikan memberikan berpengaruh dalam proses pembuatan model.

2.2.1 Penghapusan Stopwords

Istilah *stopword* mengacu kesejumlah kata dalam bahasa Inggris yang dianggap tidak penting [13] seperti *after*, *also*, *an*, *and*, *as* dan masih banyak lagi. Dalam tahap ini, akan dilakukan penghapusan stopwords tersebut [14]. Penghapusan tanda baca dilakukan karena tanda baca tidak mempengaruhi *training* program sehingga dengan dihapusnya *stopwords* kata-kata yang masuk kedalam program adalah memang kata yang memiliki makna penting didalam proses sentiment ini.

Tabel 2. Dataset Setelah Penghapusan *Stopwords*

No	Dataset
1	how do films like mouse hunt get into theatres ? isn't law something ? this diabolical load claptrap . . .
2	it was with huge lack something to do i decided watch this good old upnsundayafternoon , only good things tv are the second-rate movies they show (some are good . . .
3	note : ordinarily , moviereviews . org will not give away any critical plot points film that could be interpreted as " spoilers
4	i have to admit that i disliked this film initially .certianly isn't every taste , it's sheer torture to sit through
5	when i first heard that kevincostner was making movie called " postman , " i thought , " american version 'ilpostino ?

Pada Tabel 2 terlihat bahwa terdapat penghapusan beberapa kata setelah proses penghapusan *stopwords*. Beberapa kata yang telah dihapus contohnya adalah at, the, dan for.

2.2.2 Lowercase

Tahapan ini, berfungsi untuk mengubah semua teks kedalam *lowercase*. Proses ini, berguna apabila terdapat dua kata yang sama akan tetapi terdapat perbedaan hanya pada letak besar dan kecil kedua katanya saja, maka system tidak akan memproses kedua kata tersebut menjadi dua kata yang berbeda.

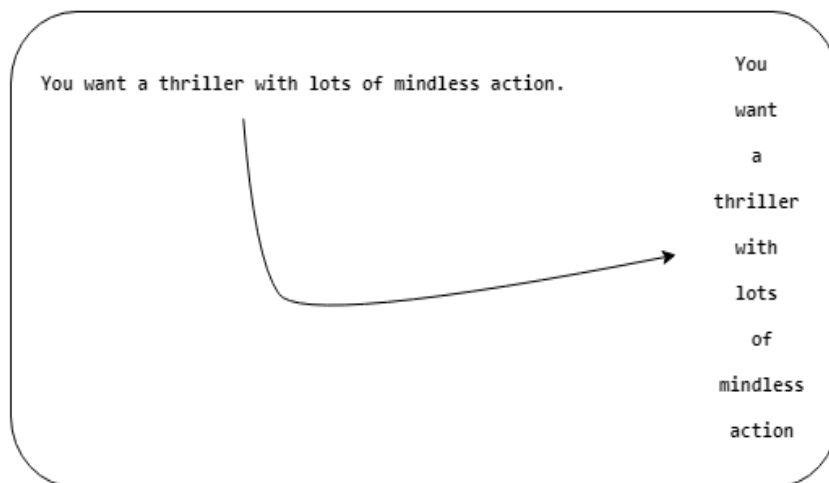
Tabel 3. *Lowercase Dataset*

No	Dataset
1	how do films like mouse hunt get into theatres ? isn't law something ? this diabolical load claptrap . . .
2	it was with huge lack something to do i decided watch this good old upnsunday afternoon , only good things tv are the second-rate movies they show (some are good . . .
3	note: ordinarily , moviereviews . org will not give away any critical plot points film that could be interpreted as " spoilers
4	i have to admit that i disliked this film initially .certainly isn't every taste , it's sheer torture to sit through
5	when i first heard that kevincostner was making movie called " postman , " i thought , " american version 'ilpostino ?

Pada Tabel 3 memang tidak terlihat perbedaan setelah dilakukan proses *lowercase* untuk dataset. Hal tersebut, dikarenakan memang tidak terdapat contoh kata yang memiliki huruf besar pada Tabel 1.

2.2.3 Tokenisasi

Setelah semua data dibersihkan, selanjutnya dilakukan proses tokenisasi. Proses tokenisasi dilakukan agar setiap *record* di dalam data tersebut, diubah menjadi setiap kata yang berdiri sendiri atau dengan kata lain untuk memisah-misahkan kata. Setiap potongan kata tersebut dinamakan dengan token [15].



Gambar 3. Tokenisasi

Pada Gambar 2 terlihat jelas bahwa yang dihasilkan dari proses tokenisasi ini adalah pengubahan format dari sebuah kata yang awalnya adalah kalimat menjadi sebuah kata yang berdiri sendiri. Proses tokenisasi digunakan untuk memudahkan dalam hal *training* algoritma.

2.2.4 Pengambilan 128 Karakter

Setelah semua data bersih, sudah dalam bentuk lower case, dan juga sudah di tokenisasi, maka selanjutnya perlu dilakukan pemotongan data. Pemotongan data dilakukan karena apabila menggunakan algoritma BERT-base, jumlah character yang digunakan sangat mempengaruhi waktu *training* dan *cost* yang dibutuhkan. *Resource* dan *cost* yang digunakan untuk pemrosesan 512 karakter sangatlah besar, oleh karena itu penulis menggunakan 128 karakter saja. Dalam implementasinya, penulis melakukan 3x percobaan dalam melakukan sebanyak 128 karakter terhadap data yang akan digunakan untuk proses *training* algoritma BERT. Pertama, data diambil sebanyak 128 karakter pertama untuk setiap dokumennya. Kedua, data diambil 64 karakter di awal dokumen dan 64 karakter di akhir dokumen. Ketiga, data diambil 128 karakter di akhir dokumen. Tujuannya adalah karena tidak semua sentimen yang dihasilkan setiap data ada pada bagian tertentu.

2.2.5 Pelabelan Data

Selanjutnya yang perlu dilakukan dalam tahapan *preprocessing* data ini adalah pemberian label untuk setiap *record* data. Hal tersebut dikarenakan ketika dataset selesai diunduh, dataset telah dikelompokkan kedalam folder negatif dan positif tetapi untuk setiap *review* filmnya belum memiliki label negative atau positif. Sehingga perlu dilakukan pelabelan untuk dataset yang akan dipakai. Pelabelan dataset dilakukan dengan tujuan pemberian *class* pada setiap *review* data sehingga hal tersebut akan memudahkan proses *training* data agar didapatkan hasil akurasi yang baik.

Tabel 4. Pelabelan Dataset

No	Dataset	Class
1	how do films like mouse hunt get into theatres ? isn't law something ? this diabolical load claptrap . . .	negative
2	it was with huge lack something to do i decided watch this good old upnsundayafternoon , only good things tv are the second-rate movies they show (some are good . . .	negative
3	note : ordinarily , moviereviews . org will not give away any critical plot points film that could be interpreted as " spoilers	positive
4	i have to admit that i disliked this film initially .certainly isn't every taste , it's sheer torture to sit through	positive
5	when i first heard that kevincostner was making movie called " postman , " i thought , " american version 'ilpostino ?	positive

Pada Tabel 4, data yang digunakan untuk proses *training* telah memiliki label. Pelabelan pada setiap data dilakukan secara *brute force*. File negative akan diberikan label 'negative', sedangkan untuk file positive akan diberikan label 'positive'.

2.2.6 Split Data

Total data yang dipakai adalah sebanyak 2000 data. Dimana untuk satu data adalah dokumen yang hanya memiliki 128 karakter. Pembagian data kedalam *data training* dan *data testing* adalah sebanyak 70% *data training* dan 30% *data testing*. Sehingga didapatkan *data training* sebanyak 1400 data, sedangkan terdapat 600 data untuk *data testing*.

2.3 Bidirectional Encoder Representations from Transformers

Metode yang dipakai dalam penelitian ini adalah menggunakan metode BERT (*Bidirectional Encoder Transformers for Language Understanding*). Metode ini dipopulerkan oleh Google pada 11 Oktober 2018 tahun lalu dan revisi terakhir dilakukan pada 24 Mei 2019. BERT adalah model representasi bahasa baru yang menghasilkan model *pre-train* representasi *bidirectional* dari teks yang tidak berlabel dengan bersama-sama mengkondisikan dari kedua konteks di semua *layer*. Sehingga, model BERT yang telah *ditrain*, dapat disesuaikan dengan hanya menambahkan satu *layer output* saja.

Terdapat 2 *steps* penting yang digunakan dalam metode ini, yaitu *pre-train* dan *fine-tuning*. Dalam step *pre-train*, dilakukan *training* model terhadap data yang tidak berlabel, sedangkan dalam step *fine-tuning* model BERT diinisialisasi dengan parameter *pre-train*, dan semua parameter tersebut disesuaikan dengan menggunakan data yang berlabel.

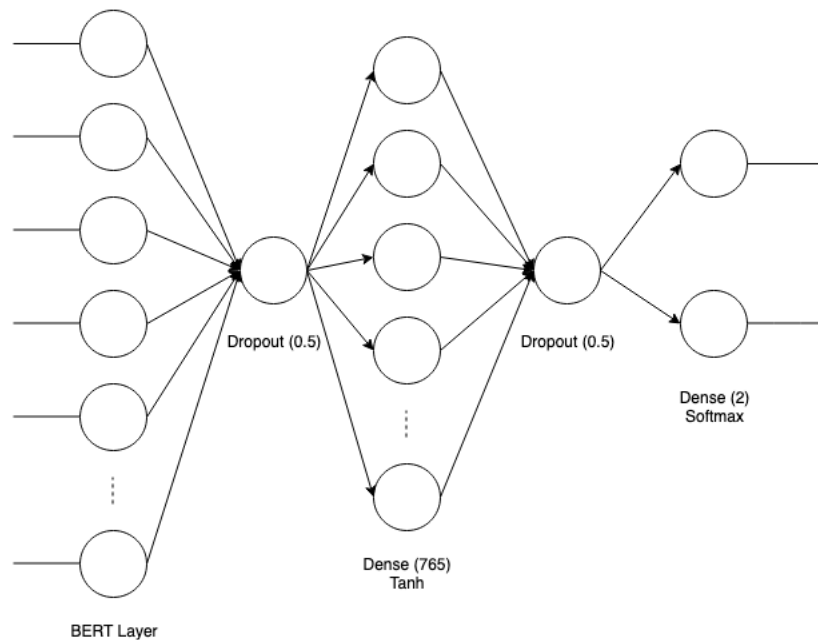
Dalam penelitian ini, model BERT yang digunakan adalah model BERT-base Multilingual Cased (multi_cased_L-12_H-768_A-12) yang memiliki 128 karakter untuk diproses. Layer BERT-base yang digunakan dalam penelitian ini memiliki 12-layer, 768-hidden, 12-heads, 110M parameters [16]. Dalam implementasinya, model pre trained untuk BERT ini tidak akan diikuti dalam proses learning. Sehingga dalam pengaplikasiannya, layer BERT ini akan di *freeze* sehingga bobot-bobot yang ada tidak akan ikut *ter-update*.

2.4 Fine-tuning Bidirectional Encoder Representations from Transformers

Bidirectional Encoder Representations from Transformers adalah arsitektur jaringan yang sudah di *training* dengan menggunakan banyak sekali dataset daribanyak sekali artikel dengan banyaksekali bahasa [16]. Oleh karena itu, daripada melakukan *training* terhadap layer BERT yang sudah memiliki bobot-bobot yang sangat baik untuk segala kasus natural *language processing*, penelitiannya perlu menambahkan layer yang akan ditujukan untuk proses klasifikasi.

Tambahan layer yang digunakan oleh penulis adalah 1 Dense layer dengan *activation function* tanh, 2 Dropout layer 0.5, dan satu layer *output* dengan menggunakan *activation function softmax* dan *loss* yang digunakan adalah *crossentropy loss*. Penambahan 2 dropout layer bertujuan agar program tidak mengalami *overfitting*. *Overfit* adalah [17] model yang dihasilkan dari proses *training* terlalu bagus, akan tetapi kekurangannya adalah karena model terlalu focus belajar pada data *training*, akibatnya saat diberikan data yang baru untuk dilakukan klasifikasi hasilnya salah.

Berikut adalah arsitektur sistem yang digunakan dalam penelitian ini:



Gambar 2. Arsitektur Sistem

Dapat terlihat bahwa layer *fine-tuning* yang ditambahkan pada *pre-train* BERT adalah dengan menambahkan 2 Dropout (0.5), 1 Dense layer, dan juga 1 layer *output*.

2.5 Akurasi

Apabila telah dilakukan *training* untuk pembuatan model, maka hal selanjutnya adalah evaluasi. Evaluasi sangat perlu dilakukan, dikarenakan untuk mengetahui bagus atau tidaknya suatu program dapat dilihat pada akurasi yang dihasilkan. Perhitungan akurasi pada penelitian ini menggunakan *confusion matrix*, karena *class* dari dataset hanya berupa *class* positif dan negative saja. Gambaran dari *confusion matrix* dapat dilihat pada Gambar 4 berikut ini. Keterangan (p,n) adalah positif dan negatif *class* label. Sedangkan (y,n) adalah *class* prediksi yang dihasilkan [18]:

		True class	
		p	n
Hypothesized class	Y	True Positives	False Positives
	N	False Negatives	True Negatives

Gambar 4. Confusion Matrix

Dapat dilihat pada tabel 4 yang merupakan contoh dataset asli yang sudah memiliki label. Sebagai contoh, misalkan sudah terdapat model dan terdapat data baru yang hasil prediksi

classnya akan di bandingkan dengan hasil class yang asli pada tabel 4. Contoh hasil yang didapatkan:

Tabel 5. Contoh Hasil Prediksi

No	Dataset	Class
1	how do films like mouse hunt get into theatres ? isn't law something ? this diabolical load claptrap . . .	positive
2	it was with huge lack something to do i decided watch this good old upnsundayafternoon , only good things tv are the second-rate movies they show (some are good . . .	negative
3	note : ordinarily , moviereviews . org will not give away any critical plot points film that could be interpreted as " spoilers	negative
4	i have to admit that i disliked this film initially .certainly isn't every taste , it's sheer torture to sit through	positive
5	when i first heard that kevincostner was making movie called " postman , " i thought , " american version 'ilpostino ?	positive

Sehingga didapatkan *confusion matrix* seperti berikut:

Tabel 6. Contoh Confusion Matrix

Keterangan		True Class	
		P	N
Hypothesized Class	Y	2	1
	N	1	1

Rumus untuk melakukan pengukuran akurasi dengan menggunakan *confusion matrix* adalah sebagai berikut:

$$Acc = \frac{TP + TN}{TP + TN + FP + FN} * 100\%$$

Dimana:

1. TP / True positive, yaitu jumlah data positif yang terklasifikasi dengan benar.
2. TN / True negative, yaitu jumlah data negative yang terklasifikasi dengan benar.
3. FP / False positive, yaitu jumlah data positive yang terklasifikasi dengan salah.
4. FN / False negative, yaitu jumlah data negative yang terklasifikasi dengan salah.

Sehingga didapatkan:

$$Acc = \frac{2+1}{2+1+1+1} * 100\%$$

$$Acc = 0.6 * 100\%$$

$$Acc = 60\%$$

Dari contoh tersebut akurasi program adalah sebesar 60% dimana dari 100 data yang diberikan, program akan salah prediksi sebesar 40 data.

3. HASIL DAN PEMBAHASAN

Hasil komparasi dari pengambilan 128 karakter, dengan menggunakan *loss function Sparse categorical cross entropy* mendapatkan hasil seperti berikut ini:

Tabel 5. Hasil Akurasi

Keterangan	Hasil Akurasi	
	<i>Training</i>	<i>Testing</i>
128 Karakter Pertama	0,87214285	0,55166670
64 Karakter Pertama + 64 Karakter Terakhir	0,82357144	0.65
128 Karakter Terakhir	0,95285714	0,73333335

Dapat terlihat bahwa hasil akurasi yang paling tinggi terdapat pada pemilihan 128 kata terakhir pada dataset Movie review polarity dataset v2.0 [3]. Hal tersebut dikarenakan pada kalimat-kalimat terakhir dataset Movie review polarity dataset v2.0 banyak sekali terdapat kata yang mengandung opini seperti *like*, *hate*, *love*, dan *adorable* sehingga program akan lebih banyak belajar dan memprediksi dengan benar.

Tabel 6. Hasil Prediksi

Contoh kalimat	128 Karakter Pertama	64 Karakter Pertama + 64 Karakter Terakhir	128 Karakter Terakhir
I love this moview	positive	positive	positive
Omg this movie is sukcs	positive	negative	negative
I dont like this, but I'm not loving this either	negative	negative	negative
This movide was very badass	negative	negative	negative
I can't explain how terrible you are	negative	negative	negative
dissappointed	negative	negative	negative
y'llhve to watch this	negative	positive	negative
recommend	positive	negative	positive

Tabel 6 adalah contoh output yang dihasilkan dari setiap *data test* yang telah disediakan. Terlihat bahwa pada 128 karakter pertama memang hasil klasifikasi yang dihasilkan masih terdapat banyak kesalahan. Hal tersebut dikarenakan pada 128 kalimat pertama pada dataset [3] hanya membicarakan alur cerita dari film yang dibicarakan. Penulis tidak banyak memberikan opini pada awal kalimat, hal itulah yang kemudian membuat akurasi dari pengambilan 128 karakter pertama rendah sekali.

Contoh hasil klasifikasi yang dihasilkan dari *input* yang diberikan oleh penulis secara *random* adalah sebagai berikut:

```
pred_token_ids (8, 128)
text: I love this movie
res: positive
text: Omg this movie is sukcs
res: negative
text: I dont like this, but I'm not loving this either
res: negative
text: This movide was very badass
res: negative
text: I can't explain how terrible you are
res: negative
text: dissappointed
res: negative
text: y'll hve to watch this
res: negative
text: recommend
res: positive
```

Gambar 5. Hasil Pengujian

Terlihat pada Gambar 5 bahwa terdapat data yang salah untuk diklasifikasikan. Pada *data test* ke-7 yang kalimatnya “y’allhve to watch this”. Dalam review tersebut mencerminkan bahwa pemberi *review* memberikan rekomendasi kepada temannya untuk menonton film tersebut, dan tentu saja rekomendasi ada karena film tersebut bagus. Akan tetapi, karena pemberi *review* tidak memberikan kata yang memberikan makna positif yang kuat seperti ‘great’, ‘good’, atau ‘wonderful’, sehingga kalimat tersebut kurang begitu jelas maknanya bagi program untuk diklasifikasikan kedalam *class* positive.

4. KESIMPULAN

Berdasarkan dataset yang digunakan yaitu berbentuk dokumen, model pre-trained dari BERT-base terbukti dapat digunakan. BERT-base terbukti dapat digunakan tidak hanya ditujukan kepada kalimat-kalimat pendek untuk diklasifikasikan. Walaupun hanya dilakukan pengambilan 128 karakterter akhir dari total karakter yang ada pada dokumen, BERT-base masih dapat melakukan *training* dengan baik sehingga didapatkan hasil akurasi yang cukup tinggi yaitu sebesar sebesar 73.7%. Akurasi ini sudah terbukti cukup bagus dan cukup jauh dibandingkan dengan penggunaan algoritma *naïve bayes* untuk proses klasifikasi.

5. SARAN

Sangat penting mengenali dataset sebelum membuat model algoritma untuk klasifikasi ataupun kasus lainnya. Sehingga diharapkan untuk selanjutnya, peneliti dapat mengenali opini dan bagian-bagian dari dataset yang penting dan memberikan pengaruh yang besar terhadap proses *training*.

UCAPAN TERIMA KASIH

Dengan penuh rasa syukur kehadiran Allah SWT, saya selaku penulis dan penyusun mengucapkan terimakasih yang sebesar-besarnya kepada pihak-pihak yang telah membantu penulis untuk menyelesaikan penelitian ini, ucapan terima kasih penulis ucapkan kepada Allah SWT yang telah memberikan segala kemudahan, kelancaran dan kesehatan sampai sekarang.

DAFTAR PUSTAKA

- [1] W. H. H. C. H. L. G. C. Lu C L Liu and E. Jou 2012, "Movie Rating and Review Summarization in Mobile Environment," *IEEE Trans. Syst. Man, Cybern. Pert C (Applications Rev.)*, Vol. 42, No. 3, pp. 397–407.
- [2] C. B. T. Dergiades E Kantopaulus and N. Bassiliades 2013, "Ontology-Based Sentiment Analysis of Twitter Posts," *Expert Syst. with Appl.*, Vol. 40, No. 10, pp. 4065–4074
- [3] B. Pang and L. Lee 2004, "A Sentimental Education: Sentiment Analysis Using Subjectivity Summarization Based on Minimum Cuts," in *Proceedings of The ACL*.
- [4] A. I. Pratiwi and Others 2018, "On The Feature Selection and Classification Based on Information Gain for Document Sentiment Analysis," *Appl. Comput. Intell. Soft Comput.*, Vol.
- [5] S. Al Faraby, E. R. R. Jasin, A. Kusumaningrum, and Adiwijaya 2018, "Classification of Hadith Into Positive Suggestion, Negative Suggestion, and Information," *J. Phys. Conf. Ser.*, Vol. 971, p. 12046, Mar, doi: 10.1088/1742-6596/971/1/012046.
- [6] H. Prasetyo, A. Adiwijaya, and W. Astuti 2019, "Klasifikasi Multi-label pada Hadis Bukhari Dalam Terjemahan Bahasa Indonesia Menggunakan Mutual Information dan Backpropagation Neural Network," *eProceedings Eng.*, Vol. 6, No. 2.
- [7] B. Manuel and D. Tricahyono 2018, "Classifying Electronic Word of Mouth and Competitive Position in Online Game Industry," *J. Data Sci. Its Appl.*, Vol. 1, No. 1, pp. 20–27.
- [8] M. Z. Naf'an, A. A. Bimantara, A. Larasati, E. M. Risondang, and N. A. S. Nugraha, 2019, "Sentiment Analysis of Cyberbullying on Instagram User Comments," *J. Data Sci. Its Appl.*, Vol. 2, No. 1, pp. 88–98.
- [9] A. Rietzler, S. Stabinger, P. Opitz, and S. Engl 2019, "Adapt or Get Left Behind: Domain Adaptation Through Bert Language Model Finetuning for Aspect-Target Sentiment Classification," *arXiv Prepr. arXiv1908.11860*.
- [10] M. Pontiki, D. Galanis, J. Pavlopoulos, H. Papageorgiou, I. Androutsopoulos, and S. Manandhar 2014, "S_{em}E_{val}-2014 Task 4: Aspect Based Sentiment Analysis," in *Proceedings of The 8th International Workshop on Semantic Evaluation (S_{em}E_{val} 2014)*, pp. 27–35, doi: 10.3115/v1/S14-2004.
- [11] V. Sanh, L. Debut, J. Chaumond, and T. Wolf 2019, "DistilBERT, A Distilled Version of BERT: Smaller, Faster, Cheaper and Lighter," *arXiv Prepr. arXiv1910.01108*.
- [12] T. Thongtan and T. Phienthrakul 2019, "Sentiment Classification Using Document Embeddings trained with Cosine Similarity," in *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics: Student Research Workshop*, pp. 407–414.

- [13] Y. Sopyan 2007, *Mengenal dan Mengoptimalkan Google*. Mediakita
- [14] H. Pouransari and S. Ghili 2014, “*Deep Learning for Sentiment Analysis of Movie Reviews*.”
- [15] P. Raghavan C D Manning and H. Schutze 2008, *Introduction to Information Retrieval*. Press Syndicate of the University of Cambridge.
- [16] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova 2018, “*Bert: Pre-training of Deep Bidirectional Transformers for Language Understanding*,” *arXiv Prepr. arXiv1810.04805*.
- [17] A. B. Santoso 2018, *Tutorial & Solusi Pengolahan Data Regresi*. Catatan Budi.
- [18] T. Fawcett 2006, “*An Introduction to ROC Analysis*,” *Pattern Recognit. Lett.*, Vol. 27, pp. 861–874.