Analisis Sentimen Komentar Berdasarkan Geo Tagged Menggunakan Algoritma Bilstm

1st Raka Zia Insani
Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia
rakaziainsani@student.telkomuniversity.ac.id

2nd Casi Setianingsih
Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia
setiacasie@telkomuniversity.ac.id

3rd Tito Waluyo Purboyo
Fakultas Teknik Elektro
Universitas Telkom
Bandung, Indonesia
titowaluyo@telkomuniversity.ac.id

Abstrak–Instagram merupakan salah satu media sosial terbesar yang saat ini populer digunakan. Di sana pengguna dapat mengirim gambar, video, berbagi pesan dan bahkan menandai lokasi juga terdapat di dalamnya. Dalam sebuah postingan, terdapat kolom komentar. Jenis komentarnya sangat beragam, ada yang positif dan negatif. Komentar digunakan untuk menganalisis sentimen positif dan negatif. Dalam penelitian ini digunakan algoritma bidirectional lstm untuk proses klasifikasi. Algoritma bilstm adalah algoritma turunan dari lstm, tetapi memiliki struktur yang berbeda. BiLSTM memiliki dua lapisan lstm untuk digunakan sebagai lapisan maju dan lapisan mundur. Dataset untuk melatih model diperoleh dari data komentar yang diambil dari postingan Instagram yang berisi tag lokasi wisata. Dalam penelitian ini, Model terbaik pada sistem yang dibuat menggunakan rasio data uji dan latih dengan perbandingan 65% dan 35%, parameter learning rate sebesar 0.0001, batch size sebesar 400 dan epoch dengan jumlah 10. Pada pengujian sistem dengan model tersebut menghasilkan nilai presisi sebesar 52%, recall 75%, f1-score sebesar 52% dan akurasi sebesar 88%. Sehingga, hasil dari klasifikasi tersebut diharapkan dapat menjadi acuan masyarakat untuk mengunjungi sebuah tempat wisata.

Kata kunci— instagram, komentar sentimen, BiLSTM, website.

Abstract – Instagram is one of the largest social media that is currently popularly used. There users can send a picture, video, share messages and even tagging location are also contained in it. In a post, there is a comment section. The types of comments are very diverse, there are positive and negative. The comments are used to analyze positive and negative sentiments. In this study, a bidirectional lstm algorithm was used for the classification process. The bilstm algorithm is a derivative algorithm of lstm, but it has a different structure. BiLSTM has two lstm layers to use as forward layers and backward layers. The dataset for the training model is taken from comment data taken from Instagram posts that contain tourist location tags. In this study, the best model was obtained when the amount of training data was 65% and testing data was 35%. Other parameters, learning rate of 0.0001, batch size of 400 and epoch of 10. Testing of the system with the model resulted in a precision value of 52%, recall of 75%, an f1-score of 52% and an accuracy of 88%. Thus, the results of this classification are expected to be a reference for people to visit a tourist attraction.

Keywords— instagram, sentiment comment, BiLSTM, website.

I. PENDAHULUAN

Media Sosial merupakan salah satu media yang sangat besar dalam berbagi cerita dari setiap individu, baik berupa gambar maupun tulisan. Dalam perkembangannya sudah muncul lebih dari puluhan media sosial yang tersebar di seluruh penjuru negeri ini. Indonesia sendiri merupakan salah satu negara dengan tingkat pengguna media sosial tertinggi, lebih dari setengan dari total jumlah keseluruhan penduduk di Indonesia[1] . Dengan kondisi saat ini media sosial menjadi sangat penting dalam penyebaran informasi.

Instagram merupakan salah satu dari sekian banyak media sosial yang banyak diminati oleh masyarakat, terutama untuk anak-anak muda. Dengan berbagai fitur dan tampilan yang menarik menjadi salah satu poin yang menjadikan instagram merupakan media sosial favorit bagi mereka. Didalam sini pengguna tidak hanya dapat membagikan gambar saja,

melainkan mereka juga dapat membuat deskripsi pada objek foto tersebut. Dalam foto yang pengguna bagikan juga, terdapat fitur *tagging location* agar orang-orang melihat foto tersebut dapat mengetahui dimana lokasi dari foto tersebut. Selain itu pengguna yang melihat foto tersebut dapat memberikan komentar mereka sebagai kesan terhadap objek yang dilihat, baik itu positif maupun negatif.

Terdapat banyak penelitian terkait penelitian mengenai *tagging location*, seperti misalnya mengklasifikasikan daya tarik turis terhadap suatu kota[2]. Penelitian tersebut menggunakan algoritma X-means dalam penerapannya untuk mengklasifikan gambar dari media sosial flickr. Oleh karena itu pada Tugas Akhir ini akan berfokus pada menganalisa sentiment berdasarkan komentar dari pengguna di Instagram dengan menggunaka algoritma Bidirectional LSTM. Algoritma ini merupakan algoritma yang cocok dalam melakukan analisa tulisan, terutama tulisan yang panjang[3]. Analisa yang dilakukan untuk mengetahui komentar persentasi komentar positif dan negatif dari objek wista yang telah ditentukan. Hasilnya nanti akan dijadikan sebuah acuan terhadap daya tarik turis terhadap suatu daerah.

II. KAJIAN TEORI

A. *Geo Tagged*

Geo Tagged merupakan sebuah tanda letak geografi pada sebuah objek, baik itu berupa gambar, teks ataupun video. Fitur *geo tagging* muncul karena berkembangnya teknologi untuk mengetahui atau mencari lokasi yaitu *Global Positioning System* (GPS). Sensor ini memungkinkan penggunanya untuk menandai lokasi atau melakukan “geo tagging” pada social netrowking service(SNSs) [2]. *Geo tagged* juga banyak digunakan pada media sosial agar para penggunnnanya dapat memberitahu lokasi tempat mereka meng-*upload* gambar, video ataupun tulisan dan membagikannya pada pengguna lain.

Pada penelitian [4] *geo tagged* digunakan untuk menganalisa sentiment pada twitter. *Geo tagged tweet* merupakan sebuah *tweet* yang didalamnya terdapat koordinat geografi yang mengindikasikan lokasi dimana *tweet* tersebut di unggah. Pada penelitian lain, *geo tagged* digunakan untuk mengukur keakuratan estimasilintasan perjalanan waktu seseorang [5].

B. *Analisa Sentimen Tekstual*

Analisa sentiment tekstual merupakan sebuah proses pengolahan data dari bentuk tekstual untuk mendapatkan sebuah informasi sentimen dalam sebuah kalimat. Dalam perkembangan teknologi analisa sentiment tekstual dapat dilakukan secara otomatis. Pada penelitian [6] disebutkan sentiment tekstual dibagi dalam 3 level yaitu, kata, kalimat dan bab. Menurut penelitian terdapat beberapa metode yang dapat digunakan ,yaitu Kamus sentiment, *Machine Learning* dan *Deep Learning*. Selain 3 metode tersebut terdapat satu metode lainnya yaitu *Lexicon Learning* [7].

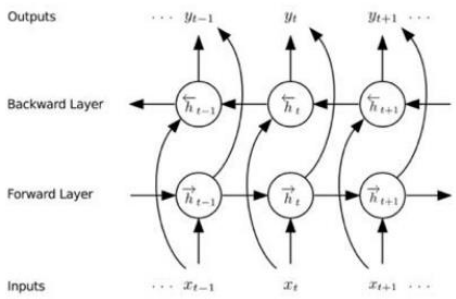
Deep Learning merupakan salah satu metode yang saat ini banyak digunakan dalam proses analisa sentimental tekstual. Terdapat banyak algoritma yang dapat digunakan untuk melakukannya. Salah satu algoritma yang popular digunakan yaitu algoritma *Naïve Bayes* seperti yang digunakan dalam penelitian [8]. Selain *Naïve Bayes*, terdapat beberapa algoritma lain seperti *Long-Short Term Memory* yang dapat digunakan dalam analisa sentiment tekstual.

C. *LSTM*

Algoritma Long-Short Term Memory (LSTM) merupakan sebuah varian dari algoritma Reccurant Neuran Network (RNN). Algoritma ini diciptakan karena RNN tidak dapat memenuhi kebutuhan untuk menyimpan data dalam jangka yang panjang [9]. Pada struktur arsitekturnya, algoritma ini tidak jauh berbeda dari RNN karena memiliki dasar yang sama. Dalam perkembangannya terdapat dua variasi popular LSTM yaitu Bidirectional LSTM dan Multilayer LSTM [10].

1. Bidirectional LSTM

Bidirectional LSTM merupakan salah satu variasi dari LSTM. Pada varian LSTM ini, menggunakan dua unit LSTM pada prosesnya, satu unit untuk proses dari kiri-kanan dan satu unit lagi untuk proses dari kanan-kiri ketika proses inputnya berlangsung [10]. LSTM pertama berjalan pada urutan input pertama dan LSTM kedua berjalan dengan urutan input terbalik. Algoritma ini juga dinilai memiliki tingkat akurasi yang baik pada penerapan dalam mengklasifikasian kalimat [11].



GAMBAR 1 Arsitektur BiLSTM [12]

Dari ilustrasi arsitektur BiLSTM diatas dapat dilihat output yang dihasilkan merupakan keluaran dari forward dan backward layer, sehingga data yang dihasilkan menjadi lebih akurat. Hal ini dikarenakan model dapat memahami data yang diterima dari depan dan belakang. Dalam fungsi matematis keluaran dari algoritma BiLSTM dapat dituliskan sebagai berikut [13]:

$$y_t = W_{hy}^{\rightarrow} \overrightarrow{h_t} + W_{hy}^{\leftarrow} \overleftarrow{h_t} \quad \textbf{(1)}$$

Dimana :

y_t = *output* dari BiLSTM

$W_{hy}^{\rightarrow}$ = nilai bobot untuk *output* LSTM Maju

$W_{hy}^{\leftarrow}$ = nilai bobot untuk *output* LSTM Mundur

$\overrightarrow{h_t}$ = nilai keluaran dari LSTM maju

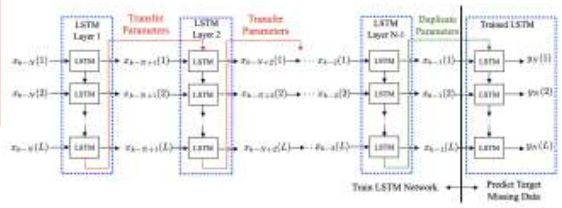
$\overleftarrow{h_t}$ = Nilai keluaran dari LSTM mundur

BiLSTM teridiri dari dua LSTM independent. Dua LSTM tersebut dapat meringkas informasi dari arah depan (*forward*) dan belakang (*Backward*) kalimat dan kemudian menggabungkan kedua informasi tersebut ke dalam satu informasi yang utuh. *Forward* LSTM menghitung data yang ada pada *hidden layer* (*fh)* berdasarkan *hidden layer*(*fh)* sebelumnya. *Backward* LSTM menghitung data yang ada pada *hidden layer* (*bh*) berdasarkan *hidden layer* (*bh*) sebelumnya. Setelah kedua proses itu terjadi maka hasil dari masing-masing layer akan digabungkan. Pada model BiLSTM kedua parameter berdiri sendiri-sendiri (independen), tetapi untuk kata yang diolah sama dari sebuah kalimat [14] .

2. Multilayer LSTM

Multilayer LSTM merupakan salah satu variasi dari LSTM yang juga popular digunakan pada pembelajaran deep learning. LSTM ini memiliki lapisan tersembunyi dibalik lapisan utamanya. Sehingga output dari lapisan pertama tidak langsung masuk ke lapisan kedua melainkan ke lapisan tersembunyi yang ada pada lapisan pertama. Pada arsitektur ini hidden state dari satu unit LSTM akan menjadi input untuk satuan LSTM pada lapisan I+1 [10].

Pada lapisan I+1 digunakan sebagai parameter awal untuk *training*. Proses ini berulang hingga (N – 1) LSTM *layer*, Gambar 2.3 menunjukkan arsitektur *Multilayer* LSTM dengan transfer parameter yang dapat mengurangi secara signifikan kompleksitas komputasi sebagai parameter lapisan LSTM [15].



GAMBAR 2 Multilayer LSTM [15]

D. SMOTE

SMOTE atau Synthetic minority oversampling technique merupakan sebuah teknik untuk mengatasi permasalahan data yang tidak seimbang. SMOTE akan melakukan mengambil sampel dari class minoritas kemudian data tersebut diduplikasi hingga dapat menghasilkan data yang seimbang diantara class [16].

Menurut [17] proses SMOTE secara rinci dapat diurutkan sebagai berikut:

a. Setiap sample minoritas $x_i \in S_{min}$, hitung k terdekat dengan sample kelas minoritas menurut jarak Euclidean

b. Pilih nilai x_j acak dari k terdekat dari x_i.

c. Sampel baru dihasilkan dari x_i dan x_j berdasarkan rumus:

$$x_{new} = x_i + |x_i + x_j| \times \delta \quad \textbf{(2)}$$

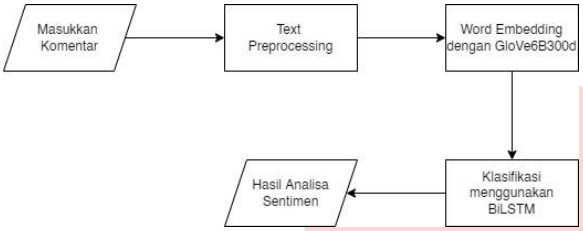
III. METODE

A. Gambaran Umum Sistem

Gambar 3 merupakan alur kerja sistem analisis sentiment komentar. Ketika sistem dimulai, inputan yang masuk berupa dataset yang di dalamnya berisi mengenai komentar dari postingan. Kemudian memasuki tahap preprocessing dilakukan proses penghapusan kata-kata yang tidak relevan serta dapat

menganggu proses klasifikasi. Hasil output dari tahap preprocessing berupa kalimat komentar saja.

Tahapan selanjutnya yaitu proses pembobotan kata. Proses pembobotan kata dilakukan untuk mengubah kata menjadi sebuah vektor angka. Setelah pembobotan kata maka akan dilakukan klasifikasi menggunakan algoritma BiLSTM. Hasil klasifikasi akan berupa nilai positif dan negatif dari komentar yang sudah di *input* kan.



GAMBAR 3 Diagram Alur Sistem

B. Kebutuhan Data

Dataset diambil komentar pada sebuah postingan di Instagram yang memiliki *tagging location* yang sudah ditentukan. Data yang terkumpul berjumlah 3919 komentar yang terdiri dari 3880 komentar berlabel positif dan 39 komentar berlabel negatif dan diambil dengan rentang waktu selama tiga minggu dari tanggal 25 Februari 2022 hingga 14 Maret 2022 . Label untuk setiap komentar didapatkan dari hasil validasi yang dilakukan oleh ahli bahasa di Balai Bahasa Universitas Pendidikan Indonesia. Data yang sudah divalidasi ini kemudian akan dilanjutkan ke proses selanjutnya sebelum masuk kedalam proses klasifikasi.

TABLE I Jumlah Dataset

Lokasi	Positif	Negatif
Bunker Batujajar	485	11
Kawah Putih	490	5
Padaawas	480	4
Tebing Keraton	485	4
Taman Langit	483	3
Ranca Upas	487	2
Wayang Windu	491	
Tangkuban Perahu	479	10

TABLE II Contah Dataset

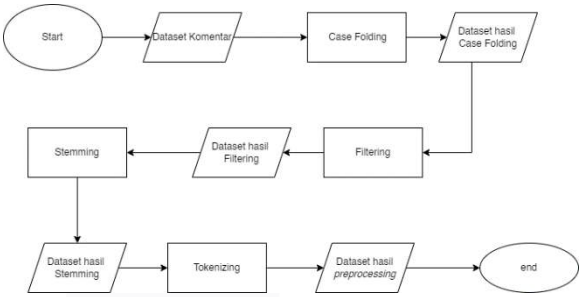
Tanggal	Komentar
Versi Indonesia	
26/2/2022	viewnya keren banget ini
28/2/2022	Juara laah 😍

TABLE III Jumlah Data Uji Sebelum dan Sesudah SMOTE

Sebelum		Sesudah	
Positif	Negatif	Positif	Negatif
2259	28	2259	2259

C. Text Processing

Tahap pre-processing ini dilakukan untuk menghilangkan hal-hal yang dapat mengganggu proses klasifikasi dari sistem yang dibuat. *Flowchart* dari proses pre-processing teks dapat dilihat pada Gambar 4 :



GAMBAR 4 Diagram Alur Text Preprocessing

1. *Case Folding*

Case Folding merupakan sebuah proses mengubah semua data masukan yang berbentuk huruf menjadi karakter kecil. Proses ini juga dilakukan agar meminimalisir fitur yang berbeda pada sebuah dokumen, karena pada sebuah sistem, kaliamat atau kata dengan karakter yang berbeda adalah sebuah fitur yang berbeda juga.

2. *Filtering*

Filtering secara bahasa berarti menyaring. Hal yang sama juga berlaku pada proses ini. Pada tahap ini kata-kata, tanda baca serta angka yang tidak ada kaitannya dengan proses selanjutnya akan dibuang. Kata-kata tersebut berupa tanda baca, tanda hubung dan sebaginya. Dengan adanya prose ini, dapat mengurangi gangguan dalam proses klasifikasi yang dilakukan. Akurasi yang didapatkan pun akan semakin

baik.

3. *Stemming*

Stemming merupakan proses untuk mengembalikan kata-kata menjadi kedalam bentuk awal. Proses ini dilakukan dengan menggunakan sebuah *library* dari *Sastrawi Module* yang ada pada python. Tujuan dilakukannya proses ini adalah untuk menghindari makna ganda dari sebuah kata.

4. *Tokenizing*

Tokenizing merupakan sebuah proses memecah kalimat menjadi sebuah kata-kata. Proses ini dilakukan agar dapat diproses pada tahap selanjutnya.

TABLE IV Contoh Text Preprocessing

Input	Case Folding	Filtering	Stemming	Tokenizing
Awas ada Buaya 🐊	awas ada buaya 🐊	awas ada buaya	awas buaya	"awas", "buaya"

D. Word embedding

Pada tahap ini kata-kata dari proses pre-processing akan dijadikan sebuah vektor. Pada penelitian ini digunaka Glove 6B *pre-trained model word embedding* untuk memberi nilai input kata. Terdapat banyak vektor kata dengan berbagai macam dimensi pada GloVe 6B *pre-trained model word embedding*, yaitu GloVe 6B 50d, GloVe 6B 100d, GloVe 6B 200d, GloVe 6B 300d .Vektor kata yang digunakan adalah GloVe 6B 300d, yang memiliki 300 Dimensi dan berjumlah sebanyak 400000 kata. Vektor kata ini dipilih karena menghasilkan akurasi yang paling baik diantar vektor kata dengan dimensi yang lain. **Gambar 5** dan **Gambar 6** adalah contoh hasil keluaran dari *word embedding*.

```
(embedding_matrix[word_index["awas"]])
array([ 3.00040007e-01, -1.86059996e-01, -5.59199989e-01, -2.26520002e-01,
```

GAMBAR 5 Word Embedding kata "awas'

```
(embedding_matrix[word_index["buaya"]])
array([ 0.56290001, -0.043101  ,  0.075601  ,  0.12939   , -0.13519   ,
```

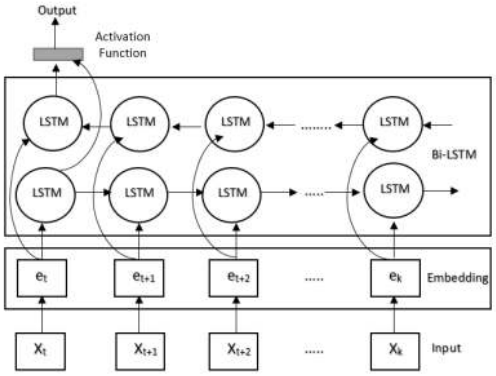
GAMBAR 6 Word Embedding kata "buaya"

Word Embedding dari kata “awas” = [0.3000 -0.1860 -0.5591 -0.2265 0]

Word Embedding dari kata “buaya” = [0.5629 -0.0431 0.0756 0.1293 -0.1351]

E. Aristektur Model

Pada penelitian ini akan menggunakan pemodelan BiLSTM untuk proses klasifikasinya. Model yang dibuat untuk penelitian ini merupakan model *sequential* yang berarti terdapat beberapa *layer* dalam sebuah model. Terapat empat layer pada model yang dibuat, yaitu *Embedding layer, LSTM layer, BiLSTM layer* dan *Dense layer*. Embedding layer berfungsi untuk mengubah inputan kata atau kaliamat yang masuk menjadi sebuah vektor numerik atau bisa disebut *word embedding*. *LSTM layer* berfungsi untuk memanggil fungsi LSTM. Pada *BiLSTM layer* terdapat dua LSTM, yaitu *forward* dan *backward* LSTM yang digunakan untuk menangkap informasi dari dua arah. Dan terakhir terdapat *Dense layer* yang berfungsi sebagai *output layer*. Bentuk dari model BiLSTM ditunjukkan pada Gambar 7.



GAMBAR 7 Arsitektur Model BiLSTM[18]

IV. HASIL DAN PEMBAHASAN

1. Pengujian Ratio

Pengujian ini dilakukan dengan membagi jumlah data untuk train dan test pada model yang telah dibuat dan di seimbangkan dengan SMOTE. Penguian ini bertujuan untuk mengetahui tingkat akurasi model menggunakan confusion matriks, jika rasio jumlah train dan test diubah. Tabel dibawah menunjukkan hasil dari pengujian ini. Hasil terbaik didapatkan ketika jumlah data uji dan data tes sebanyak 65% dan 35%. Nilai *precission* = 52%, *recall* = 70%, *f1-score* = 50% dan *accuracy* =85%.

TABLE V Tabel Pengujian Rasio

Ratio	Precision	Recall	F1-Score	Accuracy
50 : 50	51%	57%	47%	84%

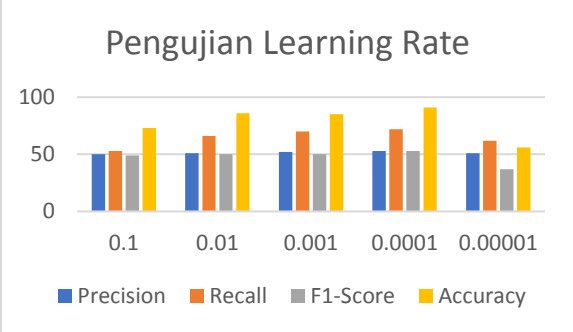
Ratio	Precision	Recall	F1-Score	Accuracy
55 : 45	51%	61%	48%	85%
60 : 40	51%	60%	48%	84%
65 : 35	52%	70%	50%	85%
70 : 30	51%	61%	48%	85%
75 : 25	51%	58%	47%	83%
80 : 20	51%	64%	48%	83%
85 : 15	51%	60%	48%	82%
90 : 10	52%	71%	50%	83%
95 : 5	53%	67%	52%	84%

2. Pengujian Learning Rate

Pengujian ini menggunakan masukan dari hasil pengujian sebelumnya dengan menggunakan rasio data terbaik yaitu 65% data uji dan 35% data latih. Pengujian dilakukan sebanyak lima kali dimulai dari *learning rate* sebesar 0.1 sampai 0.00001. Hasil dari pengujian *learning rate* ini mendapatkan nilai terbaik pada saat *learning rate* sebesar 0.0001. Nilai yang didapatkan pada *Precission*= 53%, *Recall* = 72%, *F1-Score* = 53% dan *Accuracy* = 91%. Hasil pegujian lengkap terdapat pada Tabel VI.

TABLE VI Tabel Pengujian Learning Rate

LR	Precision	Recall	F1-Score	Accuracy
0.1	50%	53%	49%	89%
0.01	51%	66%	50%	86%
0.001	52%	70%	50%	85%
0.0001	53%	72%	53%	91%
0.00001	51%	62%	37%	56%



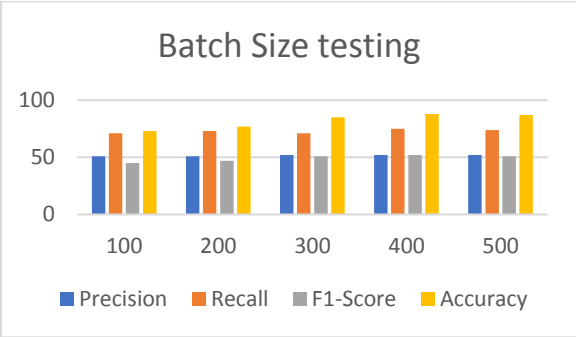
GAMBAR 8 Grafik Pengujian Learning Rate

3. Penguian Batch Size

Pengujian *Batch Size* dilakukan dengan menggunkan masukan dari hasil terbaik setalah pengujian *learning rate* yaitu 0.0001. Pengujian *batch size* ini dilakukan sebanyak lima kali dimulai dari 100 hingga 500. Hasil terbaik dari pengujian *Batch Size* didapatkan ketika jumlah *batch size* sebesar 400. Nilai yang didapatkan pada *precision*= 52%, *recall* = 75%, *f1-score* = 52% dan *accuracy* = 88%. Hasil pegujian lengkap terdapat pada Tabel VII.

TABLE VII Tabel Pengujian Batch Size

BATCH SIZE	Precision	Recall	F1-Score	Accuracy
100	51%	71%	45%	73%
200	51%	73%	47%	77%
300	52%	71%	51%	85%
400	52%	75%	52%	88%
500	52%	74%	51%	87%



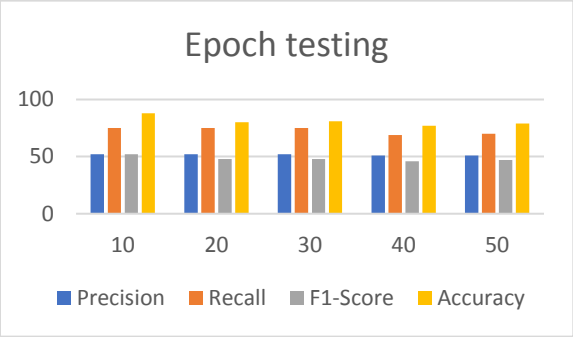
GAMBAR 9 Grafik Pengujian Batch Size

4. Pengujian Epoch

Pengujian *Epoch* ini dengan masukan dari pengujian sebelumnya. Pengujian ini bertujuan untuk mendapatkan nilai akurasi terbaik dari sebuah model ketika menjalani proses training. Pada pengujian kali ini akan dilakukan sebanyak lima kali dengan hasil nilai *epoch* dimulai dari 10 hingga 50. Hasil terbaik didapatkan ketika jumlah *Epoch* sebesar 10. Ketika *epoch* dengan 10 nilai *precision*= 52%, *recall* = 75%, *f1-score* = 52% dan *accuracy* = 88%. Hasil pegujian lengkap terdapat pada Tabel VIII.

TABLE VIII Table Pengujian Epoch

EPOCH	Precision	Recall	F1-Score	Accuracy
10	52%	75%	52%	88%
20	52%	75%	48%	80%
30	52%	75%	48%	81%
40	51%	69%	46%	77%
50	51%	70%	47%	79%



GAMBAR 10 Grafik Pengujian Epoch

V. KESIMPULAN

Dari rangkaian pengujian yang telah dilakukan dengan metode confusion matrix Performansi terbaik dari sistem klasifikasi sentimen objek wisata di kota Bandung dengan algoritma BiLSTM ketika rasio data untuk uji dan latih sebesar 65% dan 35% dengan parameter *learning rate* = 0.0001, *batch size* = 400 dan *epoch* = 10. Hasil yang didapatkan adalah *Precission* = 52%, *Recall* = 75%, *F1-Score* = 52%, dan *Accuracy* = 88%.

REFERENSI

[1] Supratman, L. P. "Penggunaan Media Sosial oleh Digital Native". Jurnal Ilmu Komunikasi, vol. 15. 2018

[2] J.-Y. Lin, S.-M. Wen, M. Hirota, T. Araki, and H. Ishikawa, "Less-Known Tourist Attraction Discovery Based on Geo-Tagged Photographs," *Machine Learning and Knowledge Extraction*, vol. 2, no. 4, pp. 414–435, Oct. 2020, doi: 10.3390/make2040023.

[3] A. U. Rehman, A. K. Malik, B. Raza, and W. Ali, "A Hybrid CNN-LSTM Model for Improving Accuracy of Movie Reviews Sentiment Analysis," Multimedia Tools and Applications, vol. 78, no. 18, pp. 26597–26613, Sep. 2019.

[4] W. L. Lim, C. C. Ho, and C. Y. Ting, "Sentiment analysis by fusing text and location features of geo-tagged tweets," IEEE Access, vol. 8, pp. 181014–181027, 2020.

[5] M. Parsafard, G. Chi, X. Qu, X. Li, and H. Wang, "Error Measures for Trajectory Estimations with Geo-Tagged Mobility Sample Data," IEEE Transactions on Intelligent Transportation Systems, vol. 20, no. 7, pp. 2566–2583, Jul. 2019.

[6] G. Xu, Y. Meng, X. Qiu, Z. Yu, and X. Wu, "Sentiment analysis of comment texts based on BiLSTM," IEEE Access, vol. 7, pp. 51522–51532, 2019.

[7] K. Z. Aung and N. N. Myo, "Sentiment analysis of students' comment using lexicon based approach," 2017 IEEE/ACIS 16th International Conference on Computer and Information Science (ICIS), 2017, pp. 149-154.

[8] Z. Li, R. Li, and G. Jin, "Sentiment analysis of danmaku videos based on naïve bayes and sentiment dictionary," IEEE Access, vol. 8, pp. 75073–75084, 2020.

[9] G. Jain, M. Sharma, and B. Agarwal, "Optimizing semantic LSTM for spam detection," International Journal of Information Technology (Singapore), vol. 11, no. 2, pp. 239–250, Jun. 2019.

[10] M. K. Balwant, "Bidirectional LSTM Based on POS tags and CNN Architecture for Fake News Detection," *2019 10th International Conference on Computing, Communication and Networking Technologies (ICCCNT)*, 2019, pp. 1-6.

[11] R. Dwi, W. Santosa, M. Arif Bijaksana, and A. Romadhony, "Implementasi Algoritma Long Short-Term Memory (LSTM) untuk Mendeteksi Penggunaan Kalimat Abusive Pada Teks Bahasa Indonesia."

[12] Pasaribu, David Junggu Manggala; Kusrini, Kusrini; Sudarmawan, Sudarmawan. Peningkatan Akurasi Klasifikasi Sentimen Ulasan Makanan Amazon Dengan Bidirectional LSTM Dan Bert Embedding. Inspiration: Jurnal Teknologi Informasi Dan Komunikasi, vol. 10. 2020.

[13] Rizky, Muhammad Gerald (2021) TA : Analisis Perbandingan Metode LSTM dan BiLSTM untuk Klasifikasi Sinyal Jantung Phonocardiogram. Undergraduate thesis, Universitas Dinamika.

[14] J. Xie, B. Chen, X. Gu, F. Liang and X. Xu, "Self-Attention-Based BiLSTM Model for Short Text Fine-Grained Sentiment Classification," in *IEEE Access*, vol. 7, pp. 180558-180570, 2019.

[15] J. Kwon, C. Cha and H. Park, "Multilayered LSTM with Parameter Transfer for Vehicle Speed Data Imputation," 2021 IEEE International Symposium on Circuits and Systems (ISCAS), 2021, pp. 1-5.

[16] C. Meng, L. Zhou, and B. Liu, “A case study in credit fraud detection with SMOTE and XGboost,” in Journal of Physics: Conference Series, Aug. 2020, vol. 1601, no. 5.

[17] Y. Yan, R. Liu, Z. Ding, X. Du, J. Chen and Y. Zhang, "A Parameter-Free Cleaning Method for SMOTE in Imbalanced Classification," in *IEEE Access*, vol. 7, pp. 23537-23548, 2019.

[18] A. R. Isnain, A. Sihabuddin, and Y. Suyanto, “Bidirectional Long Short Term Memory Method and Word2vec Extraction Approach for Hate Speech Detection,” IJCCS (Indonesian Journal of Computing and Cybernetics Systems), vol. 14, no. 2, p. 169, Apr. 2020.