

# Analisis Sentimen Menggunakan *Long Short-Term Memory* Terkait Vaksinasi Covid-19 Di Indonesia

Kharisma Kusuma Dewi<sup>a,1,\*</sup>, Sri Winiarti<sup>b,2</sup>

<sup>a</sup> Program Studi Informatika Universitas Ahmad Dahlan, Jl. Ringroad Selatan, Kragilan, Tamanan, Bantul, Yogyakarta, dan 55191, Indonesia

<sup>1</sup> [kharisma1900018299@webmail.uad.ac.id](mailto:kharisma1900018299@webmail.uad.ac.id), <sup>2</sup> [sri.winiarti@tif.uad.ac.id](mailto:sri.winiarti@tif.uad.ac.id);

\* Penulis Korespondensi

## ABSTRAK

Pro dan kontra masyarakat terkait program vaksinasi Covid-19 di Indonesia belum dikelola dengan baik oleh pemerintah. Pengelolaan opini dapat dilakukan dengan analisis sentimen untuk mendapatkan rekomendasi yang terbaik. Penelitian dengan topik yang sama banyak yang menggunakan *machine learning*, dan sedikit yang menggunakan *deep learning*. Pada penelitian ini memilih *deep learning* dengan algoritma *Long Short-Term Memory* (LSTM) untuk analisis sentimen. Penelitian ini bertujuan untuk mengetahui bagaimana melakukan analisis sentimen menggunakan LSTM terhadap vaksinasi Covid-19 di Indonesia. Serta dapat mengetahui performa LSTM untuk analisis sentiment terkait vaksinasi covid19 di Indonesia. Tahapan pertama dilakukan dengan pengumpulan data yang diambil dari Kaggle dengan topik yang sama. Kemudian dilakukan *preprocessing*. Setelah itu dilakukan klasifikasi dengan menggunakan LSTM. Proses akhir dari analisis sentimen yaitu pengujian metode klasifikasi untuk mengetahui performa model menggunakan *confusion matrix* dan *classification report*. Penelitian ini menggunakan 3000 data, dan dari banyak percobaan modifikasi model LSTM, dipilih model *Bidirectional LSTM* dan GloVe untuk *word embedding*, dengan menambahkan regularisasi berupa *dropout* dan *pooling layer* berupa GlobalMaxPool1D. Performa yang dihasilkan yaitu akurasi 71%, dengan rincian untuk sentimen negatif (presisi: 89%, recall: 20%, dan f1-score: 33%), sentimen netral (presisi: 72%, recall: 86%, dan f1-score: 78%), sentimen positif (presisi: 67%, recall: 72%, dan f1-score: 70%).



## Kata Kunci

Analisis sentiment  
Vaksinasi covid-19  
Long short-term memory



This is an open-access article under the [CC-BY-SA](https://creativecommons.org/licenses/by-sa/4.0/) license

## 1. Pendahuluan

*Coronavirus* merupakan virus keluarga besar virus yang menyebabkan flu ringan hingga penyakit pernapasan yang parah, seperti *Middle East Respiratory Syndrome (MERS)* dan *Severe Acute Respiratory Syndrome (SARS)*. Virus ini mulai menjadi wabah di kota Wuhan, China pada bulan November–Desember 2019. Pada 11 Maret 2020, *World Health Organization* [1] menyatakan virus ini sebagai pandemi global, karena telah menginfeksi banyak orang di berbagai negara. Hingga 25 April 2022, Satgas Covid19 Indonesia [2] mengonfirmasi ada 6.044.467 kasus covid-19 dengan jumlah kematian 156.133. Upaya pemerintah dalam menangani kasus Covid-19 ini yaitu dengan vaksinasi.

Program vaksinasi Covid-19 yang diselenggarakan pemerintah mendapat berbagai macam respon dari masyarakat. Dari banyak media sosial, twitter menjadi salah satu yang paling sering digunakan oleh masyarakat untuk menyampaikan responnya. Jemadu dalam artikel Suara.com [3] menyebutkan bahwa berdasarkan data jangkauan iklan, jumlah pengguna twitter di Indonesia mencapai 18,45 juta orang pada tahun 2022. Terdapat tweet positif, netral, dan negatif terkait vaksinasi Covid-19 ini. Karena adanya respon pro dan kontra tersebut sehingga peneliti melakukan analisis sentimen media sosial twitter terhadap program vaksin Covid-19 di Indonesia.

Untuk menganalisis respon atau opini masyarakat pada twitter, dapat menggunakan metode analisis sentimen. Analisis sentimen atau opinion mining merupakan salah satu bidang dari *Natural Language Processing*, yang membangun sistem untuk mengumpulkan, mengekstrak, mengolah data atau informasi dari internet terkait suatu isu sehingga dapat diketahui polaritas yang ada dalam kalimat opini dari data tersebut. Dengan menggunakan analisis sentimen, dapat diketahui bagaimana opini masyarakat terkait vaksinasi Covid-19, apakah opini tersebut bersifat positif atau negatif. Berdasarkan penelitian yang dilakukan oleh Zucco dkk. [4] tahapan dalam analisis sentimen yaitu: *data collection, data extraction, data preprocessing, classification*.

Menurut penelitian yang telah dilakukan oleh Yulita [5], analisis sentimen pada data twitter terkait vaksinasi Covid-19 dengan *Naïve Bayes*, mendapatkan akurasi mencapai 0.93 (93%). Menurut Lestandy dkk. [6], analisis sentimen terkait vaksinasi Covid-19 dengan *Recurrent Neural Networks* dan *Naïve Bayes* mencapai akurasi 97,77% untuk RNN dan akurasi 80% untuk *Naïve Bayes*. Selanjutnya dalam penelitian yang dilakukan oleh Aryal dan Bhattarai [7] mengenai analisis sentimen tweet vaksinasi COVID-19 di Nepal, India, dan Singapura, menggunakan *Naïve Bayes* dan *Long Short-Term Memory (LSTM)*, diperoleh hasil bahwa akurasi model LSTM 7% lebih baik dibandingkan dengan *Naïve Bayes*.

Penelitian lain yang di Laurensz dan Sedyono [8], dengan 845 data twitter, tweet "vaksinmerahputih" dan "vaksininovac" menggunakan *Naïve Bayes* dan *Support Vector Machine*. Pada penelitian ini diperoleh rata-rata akurasi dengan menggunakan metode uji *K-Fold Cross Validation* sebesar 85,59% untuk *Naïve Bayes*, sedangkan SVM sebesar 84,41%. Kajian lain yang menggunakan pendekatan *Naïve Bayes* untuk analisis sentimen adalah penelitian oleh Harun dan Ananda [9]. Selain *Naive Bayes*, penelitian ini juga menggunakan pendekatan pohon keputusan. Hasil dari penelitian ini dengan metode *Confusion Matrix Test*, *Naïve Bayes* menghasilkan akurasi 100% lebih tinggi dibandingkan dengan *Decision Tree* yang hanya mendapatkan akurasi 50.39%. Selanjutnya penelitian yang dilakukan oleh Salsabila [10] terkait tweet vaksinasi Covid19 menggunakan *Naïve Bayes*, hasil yang diperoleh yaitu akurasi sebesar 71%.

Penelitian lain yang menggunakan metode deep learning seperti penelitian yang dilakukan oleh Ihsan dkk. [11] terkait tweet vaksinasi Covid19 menggunakan *Long Short-Term Memory* dan *word embeddings word2vec* diperoleh akurasi sebesar 66%. Kemudian penelitian yang dilakukan oleh Hidayatullah dkk. [12] terkait Pemilu 2019 pada twitter, menggunakan *Naïve Bayes*, *SVM*, *LSTM*, dan *CNN*. Hasil yang diperoleh yaitu akurasi *Naïve Bayes*: 82.08%, akurasi *SVM*: 83.23%, akurasi *LSTM*: 84.20%, dan akurasi *CNN*: 84.05%. Selanjutnya penelitian yang dilakukan oleh Pasaribu dkk. [13] menggunakan *dataset* Kaggle *Amazon Fine Food* dengan *Bidirectional Lstm* dan *Bert Embedding* menghasilkan akurasi sebesar 93%.

Berdasarkan penelitian-penelitian sebelumnya, penelitian analisis sentimen dengan topik vaksinasi covid19 menggunakan *machine learning* dengan algoritma *Naïve Bayes* sudah banyak dilakukan karena memiliki tingkat akurasi yang lebih tinggi daripada algoritma lainnya. Dan tidak banyak penggunaan metode *deep learning*. Oleh karena itu, tujuan penelitian ini untuk melakukan analisis sentimen *Long Short-Term Memory (LSTM)* terkait program vaksinasi covid19 di Indonesia. Serta mengetahui performa yang dihasilkan. Metode LSTM dipilih karena kemampuannya dalam mengatasi masalah *vanishing gradient* pada jaringan RNN dan kemampuannya dalam mempertahankan informasi jangka panjang pada teks yang panjang. Dalam analisis sentimen Twitter, metode LSTM digunakan untuk mengklasifikasikan teks ke dalam kategori sentimen positif, negatif, atau netral.

## 2. Metode

### 2.1. Pengumpulan Data

Metode pengumpulan data dalam penelitian ini yaitu menggunakan data dari situs Kaggle dengan studi kasus yang sama yaitu tweet vaksinasi covid19 di Indonesia. Dalam penelitian ini, subjeknya adalah masyarakat Indonesia yang menggunakan twitter untuk menyampaikan pendapatnya tentang vaksin Covid-19. Objek dalam penelitian ini yaitu menggunakan tweet berbahasa Indonesia untuk memahami emosi dan pendapat masyarakat tentang vaksin covid-19.

### 2.2. Penerapan Metode

Dalam tahap ini juga dilakukan preprocessing teks berupa *cleaning* dan *case folding, stop word*, dan lematisasi. Kemudian dilanjutkan dengan analisis sentimen menggunakan LSTM. Dalam review yang dilakukan oleh Saxena [14], *Long Short-Term Memory* merupakan jenis *recurrent neural network*. LSTM

digunakan untuk mengatasi masalah ketergantungan jangka panjang RNN, di mana RNN tidak dapat memprediksi kata yang disimpan dalam memori jangka panjang tetapi dapat membuat prediksi yang lebih akurat berdasarkan input saat ini. Secara default, LSTM dapat menyimpan informasi untuk jangka waktu yang lama yang digunakan untuk menganalisis, memprediksi, dan mengklasifikasikan berdasarkan data deret waktu. LSTM adalah jenis pendekatan pembelajaran mendalam yang dapat membantu mengingat nilai baca sebelumnya dari waktu ke waktu. Analisis teks dapat dilakukan menggunakan LSTM karena membantu mengingat nilai baca sebelumnya secara efektif. LSTM akan lebih mampu menghadapi perubahan sentimen dalam sebuah tweet.

### 2.3. Evaluasi Model

Untuk mengetahui performa sebagai evaluasi model LSTM, digunakan *confusion matrix* dan *classification\_report* untuk menampilkan presisi, *recall*, *f1-score*, dan akurasi. Pengertian dari *confusion matrix* adalah matriks yang menggunakan data dalam matriks untuk memvisualisasikan kinerja algoritma klasifikasi. *Confusion matrix* bekerja dengan membandingkan nilai aktual dengan nilai prediksi. Yang mana dapat ditampilkan dalam bentuk tabel dengan empat kombinasi berbeda antara nilai aktual dan prediksinya [15].

## 3. Hasil dan Pembahasan

### 3.1. Pengumpulan Data

Pada penelitian ini diperoleh data dari situs Kaggle yang mana situs tersebut merupakan situs penyedia data terpercaya. Data ini merupakan data bersentimen hasil *crawling* dari twitter terkait vaksinasi covid19 di Indonesia. Hasil yang diperoleh yaitu terdapat 3000 data tweet dengan kolom: *datetime*, *tweet id*, *text*, *translate text*, *username*, *retweetedTweet*, *inReplyToUser*, *clean\_tweet*, *subjectivity*, *polarity*, *analysis*.

### 3.2. Preprocessing

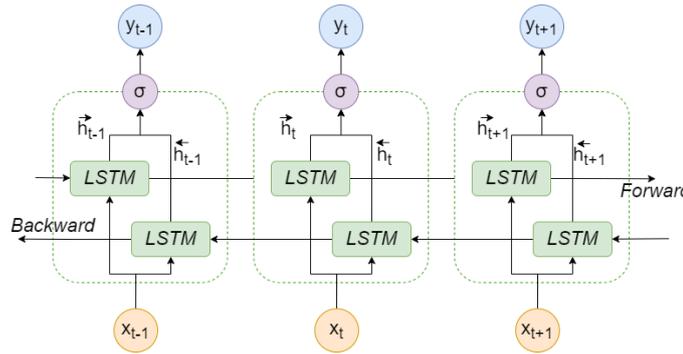
Tahap *preprocessing* data dilakukan sebelum data digunakan untuk klasifikasi. Pada tahapan ini dilakukan *cleaning*, *stop word*, dan lematisasi. Proses *cleaning* yang dilakukan yaitu menghapus duplikasi data dan *missing value*, juga merubah kata menjadi *lowercase*. Kemudian dilanjutkan proses *stop word* menggunakan *library* nltk dan lematisasi. Kolom yang akan di proses pada tahap ini yaitu *'clean\_tweet'*. Hasil dari *preprocessing* data dapat dilihat pada Gambar 1.

	analysis	clean_tweet	punctuation	stop_word	lemmatize
0	Positif	Norway Unplug Most Restrict Covid-19	norway unplug most restrict covid19	norway unplug restrict covid19	norway unplug restrict covid19
1	Netral	The East Java Provincial Government also prepa...	the east java provincial government also prepa...	east java provincial government also prepares ...	east java provincial government also prepare c...
2	Netral	The Indonesian government is tireless, continu...	the indonesian government tireless continues m...	indonesian government tireless continues make ...	indonesian government tireless continue make m...
3	Positif	Good morning bro n sis. Better Stayathome isol...	good morning better stayathome isolateyourself...	good morning better stayathome isolateyourself...	good morning better stayathome isolateyourself...
4	Netral	Hospital drill in Banten is still below 50 per...	hospital drill banten still below percent this...	hospital drill banten still percent criteria c...	hospital drill banten still percent criteria c...

Gambar 1. Hasil Preprocessing

### 3.3. Penerapan Metode LSTM

Arsitektur LSTM yang digunakan yaitu dengan *Bidirectional LSTM*, dengan menambahkan *pooling layer* dan GloVe (*Global Vectors for Word Representation*) untuk *word embedding*. *Pooling layer* sering digunakan dalam tugas klasifikasi teks, di mana data input adalah urutan kata. Penambahan lapisan *pooling* digunakan untuk mengurangi dimensi spasial peta fitur sambil mempertahankan informasi penting. Tujuan utama dari *pooling layer* adalah untuk mengurangi ukuran peta fitur, yang membantu dalam membuat perhitungan lebih cepat dengan mengurangi jumlah parameter yang dapat dilatih. Lapisan *pooling* merangkum fitur di wilayah peta fitur, memungkinkan representasi yang lebih ringkas. GloVe merupakan salah satu metode *word embedding* yang digunakan untuk mengubah kata-kata menjadi vektor numerik. Dengan menggunakan GloVe, LSTM dapat memahami makna kata-kata dalam konteks yang lebih baik dan menghasilkan representasi vektor yang lebih baik pula. Hal ini dapat meningkatkan kinerja dan akurasi *Bidirectional LSTM* pada deteksi kalimat abusive pada analisis sentimen teks. Arsitektur LSTM yang digunakan dapat dilihat pada Gambar 2.



**Gambar 2.** Arsitektur BiLSTM yang Digunakan

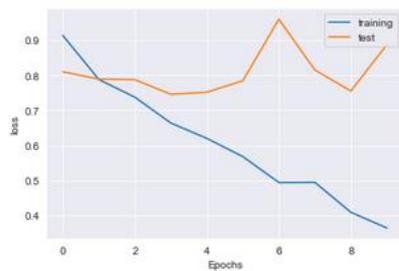
Membuat model LSTM pertama dengan menggunakan *Bidirectional LSTM*, dengan tambahan *pooling layer* yaitu *GlobalMaxPool1D*. Lapisan ini mengambil nilai maksimum dari semua fitur dalam urutan input dan mengembalikan nilai tunggal. Teknik ini digunakan untuk meringkas fitur yang ada di wilayah peta fitur yang dihasilkan oleh lapisan konvolusi. *Global pooling* mengurangi setiap saluran di peta fitur menjadi satu nilai, yang setara dengan menggunakan filter dimensi  $n_h \times n_w$ , yaitu dimensi peta fitur. Summary dari permodelan LSTM pertama yang sudah dibentuk dapat dilihat pada Gambar 3. Dan hasil performanya dapat dilihat pada Gambar 4 – 5.

```
Model: "sequential"
-----
```

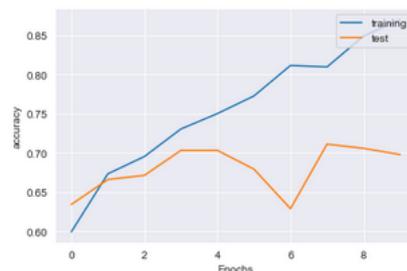
Layer (type)	Output Shape	Param #
embedding (Embedding)	(None, 40, 100)	650000
bidirectional (Bidirectional)	(None, 40, 256)	234496
global_max_pooling1d (GlobalMaxPooling1D)	(None, 256)	0
dense (Dense)	(None, 128)	32896
dense_1 (Dense)	(None, 3)	387

```
-----
Total params: 917,779
Trainable params: 267,779
Non-trainable params: 650,000
-----
```

**Gambar 3.** Summary LSTM Model Pertama



**Gambar 4.** Grafik Loss LSTM Pertama



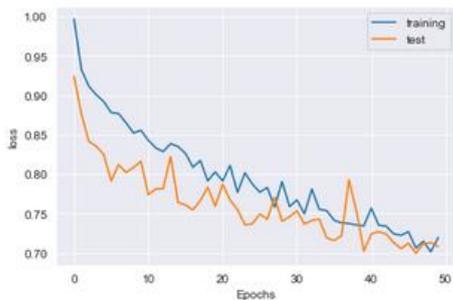
**Gambar 5.** Grafik Akurasi LSTM Pertama

Dari Gambar 4. dan 5, dapat dilihat bahwa model LSTM pertama mengalami *overfitting*. *Overfitting* dalam model LSTM yang dibuat mengacu pada situasi di mana model menjadi terlalu khusus dalam mempelajari data pelatihan dan gagal menggeneralisasi dengan baik ke data yang tidak terlihat. Sehingga menghasilkan performa yang buruk pada data pengujian, meskipun performa pada data pelatihan sangat baik. Untuk mengatasi masalah *overfitting* dilakukan percobaan pada model pertama dengan menambahkan *layer lstm* baru dan menambahkan regularisasi dengan *dropout*. Lapisan *dropout* dalam model LSTM dapat secara acak menonaktifkan persentase tertentu dari unit LSTM selama pelatihan, yang membantu mengurangi *overfitting*. Summary dari permodelan LSTM kedua dapat dilihat pada Gambar 6. Dan hasil performanya dapat dilihat pada Gambar 7 – 9.

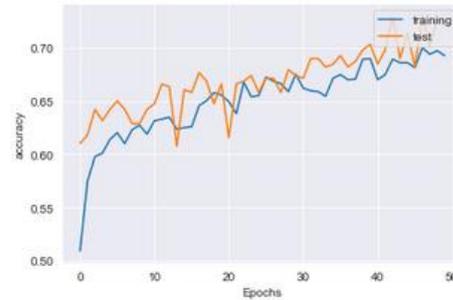
```

Model: "sequential"
-----
Layer (type)                Output Shape              Param #
-----
embedding (Embedding)       (None, 40, 100)          650000
dropout (Dropout)           (None, 40, 100)          0
lstm (LSTM)                  (None, 40, 128)          117248
dropout_1 (Dropout)         (None, 40, 128)          0
bidirectional (Bidirectiona (None, 40, 256)          263168
1)
global_max_pooling1d (Globa (None, 256)              0
lMaxPooling1D)
dropout_2 (Dropout)         (None, 256)              0
dense (Dense)                (None, 128)              32896
dense_1 (Dense)              (None, 3)                387
-----
Total params: 1,063,699
Trainable params: 413,699
Non-trainable params: 650,000
    
```

**Gambar 6.** Summary Model LSTM Kedua



**Gambar 7.** Grafik Loss LSTM Kedua



**Gambar 8.** Grafik Akurasi LSTM Kedua

	precision	recall	f1-score	support
Negatif	0.89	0.20	0.33	40
Netral	0.72	0.86	0.78	125
Positif	0.67	0.72	0.70	115
accuracy			0.71	280
macro avg	0.76	0.59	0.60	280
weighted avg	0.73	0.71	0.68	280

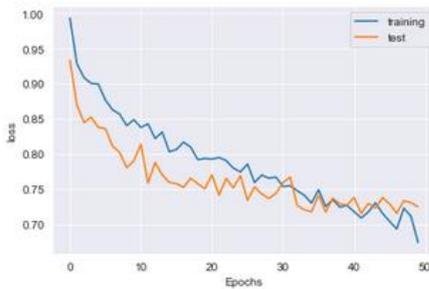
**Gambar 9.** Performa LSTM Kedua

Dari Gambar 7 dan 8, dapat dilihat bahwa model LSTM kedua berhasil mengatasi permasalahan *overfitting*. Namun dapat dilihat pada Gambar 9, performa yang dihasilkan hanya rentan 70%. Untuk menaikkan performa dilakukan percobaan dengan modifikasi model lstm kedua yaitu dengan mengganti jenis *pooling layer* menjadi *GlobalAveragePooling1D*. Lapisan ini mirip dengan lapisan *GlobalMaxPool1D* karena mengurangi dimensi peta fitur output dalam jaringan saraf. Namun, alih-alih mengambil nilai maksimum dari semua fitur dalam urutan input, *GlobalAveragePooling1D* mengambil nilai rata-rata dari semua fitur dalam urutan input dan mengembalikan satu nilai. *Summary* dari permodelan LSTM ketiga dapat dilihat pada Gambar 10. Dan hasil performanya dapat dilihat pada Gambar 11 – 13.

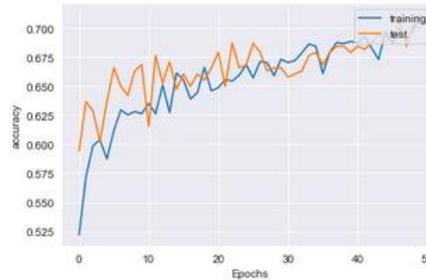
```

Model: "sequential"
-----
Layer (type)                Output Shape                Param #
-----
embedding (Embedding)       (None, 40, 100)           650000
dropout (Dropout)           (None, 40, 100)           0
lstm (LSTM)                  (None, 40, 128)           117248
dropout_1 (Dropout)         (None, 40, 128)           0
bidirectional (Bidirectiona (None, 40, 256)           263168
1)
global_average_pooling1d (G (None, 256)                 0
lobalAveragePooling1D)
dropout_2 (Dropout)         (None, 256)                 0
dense (Dense)                (None, 128)                 32896
dense_1 (Dense)              (None, 3)                   387
-----
Total params: 1,063,699
Trainable params: 413,699
Non-trainable params: 650,000
    
```

Gambar 10. Summary Model LSTM Ketiga



Gambar 11. Grafik Loss LSTM Ketiga



Gambar 12. Grafik Akurasi LSTM Ketiga

	precision	recall	f1-score	support
Negatif	1.00	0.20	0.33	40
Netral	0.65	0.93	0.76	125
Positif	0.75	0.61	0.67	115
accuracy			0.69	280
macro avg	0.80	0.58	0.59	280
weighted avg	0.74	0.69	0.66	280

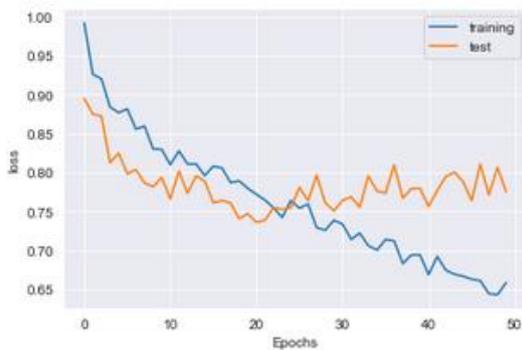
Gambar 13. Performa LSTM Ketiga

Dari Gambar 11 dan 12, dapat dilihat bahwa model LSTM ketiga berhasil mengatasi permasalahan *overfitting* juga. Namun dapat dilihat pada Gambar 13, performa yang dihasilkan kurang dari model LSTM kedua. Untuk menaikkan performa dilakukan percobaan dengan modifikasi model lstm ketiga yaitu dengan mengganti *GlobalAveragePooling1D* menjadi *Flatten. Layer Flatten* pada model LSTM digunakan untuk mengubah output dari lapisan LSTM, yang merupakan urutan vektor, ke dalam format yang dapat dimasukkan ke dalam lapisan yang terhubung sepenuhnya untuk klasifikasi. Summary dari permodelan LSTM keempat dapat dilihat pada Gambar 14. Dan hasil performanya dapat dilihat pada Gambar 15 – 17.

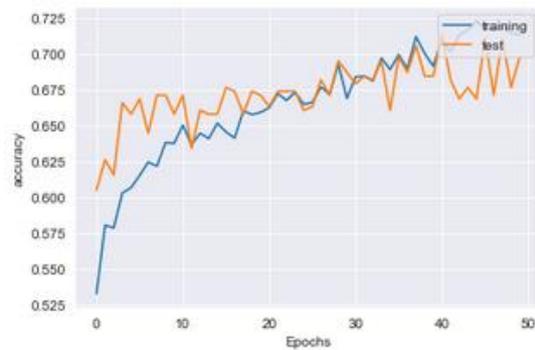
```

Model: "sequential"
-----
Layer (type)                Output Shape                Param #
-----
embedding (Embedding)       (None, 40, 100)            650000
dropout (Dropout)           (None, 40, 100)            0
lstm (LSTM)                  (None, 40, 128)            117248
dropout_1 (Dropout)         (None, 40, 128)            0
bidirectional (Bidirectiona (None, 40, 256)            263168
1)
flatten (Flatten)           (None, 10240)               0
dropout_2 (Dropout)         (None, 10240)               0
dense (Dense)                (None, 128)                 1310848
dense_1 (Dense)             (None, 3)                   387
-----
Total params: 2,341,651
Trainable params: 1,691,651
Non-trainable params: 650,000
    
```

Gambar 14. Summary Model LSTM Keempat



Gambar 15. Grafik Loss LSTM Keempat



Gambar 16. Grafik Akurasi LSTM Keempat

	precision	recall	f1-score	support
Negatif	0.71	0.12	0.21	40
Netral	0.67	0.85	0.75	125
Positif	0.65	0.64	0.65	115
accuracy			0.66	280
macro avg	0.68	0.54	0.54	280
weighted avg	0.67	0.66	0.63	280

Gambar 17. Performa LSTM Keempat

### 3.4. Evaluasi

Proses pemilihan permodelan LSTM untuk mendapatkan performa yang tinggi dilakukan dengan percobaan menggunakan model *Bidirectional LSTM* dengan tambahan GloVe tanpa *dropout*, dengan *dropout*, modifikasi pada *pooling layer*, dan *layer Flatten*. Dilakukan dengan ukuran *epoch* 20 untuk model pertama 50 untuk model lainnya. Berikut disajikan Tabel 1 perbandingan akurasi arsitektur LSTM yang sudah dilakukan.

Tabel 1. Perbandingan Akurasi LSTM

Model	Epoch	Dropout	Jenis modifikasi layer	train	val	test
1.	20	Tidak	GlobalMaxPool1D	85%	70%	68%
2.	50	Ya	Penambahan Layer LSTM + GlobalMaxPool1D	79%	72%	71%
3.	50	Ya	Penambahan Layer LSTM + GlobalAveragePooling1D	78%	70%	69%
4.	50	Ya	Penambahan Layer LSTM + Flatten	83%	70%	66%

Dari Tabel diatas, dapat dilihat bahwa model LSTM 1, performa data pelatihan lebih tinggi dari pada performa data validasi dan pengujian. Hal tersebut menunjukkan bahwa model tersebut mengalami *overfitting*. Untuk mengatasi permasalahan tersebut dilakukan dengan penambahan *layer* LSTM dan *dropout* seperti pada model LSTM 2 – 4. Namun meskipun sudah dilakukan penambahan *dropout* untuk regularisasi, model LSTM 4 juga menunjukkan adanya *overfitting*. Pada model LSTM 2 dan 3 dilakukan modifikasi *pooling layer*, hasil yang diperoleh yaitu dengan menggunakan *GlobalMaxPool1D* dan *GlobalAveragePooling1D*, performa yang dihasilkan yaitu pada rentan 70%. Dari hasil perbandingan percobaan model LSTM 1 – 4 dapat disimpulkan bahwa pada penelitian ini model yang lebih baik digunakan yaitu model LSTM 2. Model tersebut dipilih karena menghasilkan performa tinggi dan tidak *overfitting* pada penelitian ini.

#### 4. Kesimpulan

Analisis sentimen menggunakan LSTM terkait program vaksinasi covid19 di Indonesia dapat dilakukan dengan data twitter dari Kaggle. Kemudian dilakukan *preprocessing*, penerapan metode LSTM, dan evaluasi. Pada penelitian ini dari banyak percobaan modifikasi model LSTM, dipilih model *Bidirectional LSTM* dan *GloVe* untuk *word embedding*, dengan menambahkan regularisasi berupa *dropout* dan *pooling layer* berupa *GlobalMaxPool1D*. Performa dari model LSTM yang dipilih menghasilkan akurasi 71%, dengan rincian untuk sentimen negatif (presisi: 89%, *recall*: 20%, dan *f1-score*: 33%), sentimen netral (presisi: 72%, *recall*: 86%, dan *f1-score*: 78%), sentimen positif (presisi: 67%, *recall*: 72%, dan *f1-score*: 70%).

#### Daftar Pustaka

- [1] "Coronavirus disease 2019 ( COVID-19 )," in *World Health Organization*, 2020.
- [2] K. RI and S. COVID19, 25 April 2022. [Online]. Available: <https://covid19.go.id/index.php/artikel/2022/04/25/situasi-covid-19-di-indonesia-update-25-april-2022>.
- [3] L. Jemadu and D. Prastya, 23 Februari 2022. [Online]. Available: <https://www.suara.com/tekno/2022/02/23/191809/jumlah-pengguna-media-sosial-indonesia-capai-1914-juta-per-2022?page=2>.
- [4] C. Zucco, B. Calabrese, G. Agapito, P. H. Guzzi and M. Cannataro, "Sentiment Analysis For Mining Texts and Social Networks Data: Methods and tools," *Wiley Interdisciplinary Reviews: Data Mining and Knowledge Discovery*, pp. 1-32, 2019.
- [5] W. Yulita, "ANALISIS SENTIMEN TERHADAP OPINI MASYARAKAT TENTANG VAKSIN COVID-19 MENGGUNAKAN ALGORITMA NAÏVE BAYES CLASSIFIER," *JDMSI*, Vol. 2, No. 2, 2021, pp. 1-9, 2021.
- [6] M. Lestandy, A. Abdurrahim and L. Syafa'ah, "Analisis Sentimen Tweet Vaksin COVID-19 Menggunakan Recurrent," *Rekayasa Sistem dan Teknologi Informasi*, vol. V, pp. 802-808, 2021.
- [7] R. R. Aryal and A. Bhattarai, "Sentiment Analysis on Covid-19 Vaccination Tweets using Naïve," *Advances in Engineering and Technology: An International Journal*, vol. I, pp. 57-70, 2021.

- [8] B. Laurensz and E. Sedyono, "Analisis Sentimen Masyarakat terhadap Tindakan Vaksinasi dalam Upaya Mengatasi Pandemi Covid-19," *Jurnal Nasional Teknik Elektro dan Teknologi Informasi* Vol. 10, No. 2, Mei 2021, pp. 118-123, 2021.
- [9] A. Harun and D. P. Ananda, "Analisa Sentimen Opini Publik Tentang Vaksinasi Covid-19 di Indonesia Menggunakan Naïve Bayes dan Decision Tree," *MALCOM: Indonesian Journal of Machine Learning and Computer Science* Vol. 1 Iss. 1 April 2021, pp. 58-63, 2021.
- [10] N. SALSABILA, "ANALISIS SENTIMEN MEDIA SOSIAL TWITTER TERHADAP PROGRAM VAKSINASI COVID-19 DI INDONESIA DENGAN ALGORITMA NAIVE BAYES," Universitas Telkom, S1 Sistem Informasi, Bandung, 2021.
- [11] M. Ihsan, B. S. Negara and S. Agustian, "LSTM (Long Short Term Memory) for Sentiment COVID-19 Vaccine Classification on Twitter," *Jurnal Teknologi Informasi dan Komunikasi*, vol. XIII, pp. 79-89, 2022.
- [12] A. F. Hidayatullah, S. Cahyaningtyas and A. M. Hakim, "Sentiment Analysis on Twitter using Neural Network:," in *IOP Conference Series: Materials Science and Engineering*, 2021.
- [13] D. J. M. Pasaribu, Kusri and Sudarmawan, "PENINGKATAN AKURASI KLASIFIKASI SENTIMEN ULASAN MAKANAN AMAZON DENGAN BIDIRECTIONAL LSTM DAN BERT EMBEDDING," *Inspiration: Jurnal Teknologi Informasi dan Komunikasi*, vol. Volume 10, pp. 9-20, 2020.
- [14] S. Saxena, "What is LSTM? Introduction to Long Short-Term Memory," Analytics Vidhya, 16 Maret 2021. [Online]. Available: <https://www.analyticsvidhya.com/blog/2021/03/introduction-to-long-short-term-memory-lstm/>.
- [15] S. Narkhede, "Understanding Confusion Matrix," Towards Data Science, 9 May 2018. [Online]. Available: <https://towardsdatascience.com/understanding-confusion-matrix-a9ad42dcfd62>.