

**KLASIFIKASI KALIMAT PERBINCANGAN MASYARAKAT TERHADAP  
PANDEMI COVID-19 PADA TWITTER DENGAN METODE *LONG SHORT-  
TERM MEMORY***

**SKRIPSI**

Oleh:  
**YULIANA ROMADHONI**  
**NIM. 18650012**



**JURUSAN TEKNIK INFORMATIKA  
FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM  
MALANG  
2022**

**KLASIFIKASI KALIMAT PERBINCANGAN MASYARAKAT  
TERHADAP PANDEMI COVID-19 PADA TWITTER DENGAN METODE  
*LONG SHORT-TERM MEMORY***

**SKRIPSI**

Oleh:  
**YULIANA ROMADHONI**  
NIM. 18650012

**Diajukan Kepada:  
Fakultas Sains dan Teknologi Universitas Islam Negeri (UIN) Malang  
untuk Memenuhi Salah Satu Persyaratan Dalam  
Memperoleh Gelar Sarjana Komputer (S.Kom)**

**URUSAN TEKNIK INFORMATIKA  
FAKULTAS SAINS DAN TEKNOLOGI  
UNIVERSITAS ISLAM NEGERI MAULANA MALIK IBRAHIM  
MALANG  
2022**

**HALAMAN PERSETUJUAN**

**KLASIFIKASI KALIMAT PERBINCANGAN MASYARAKAT  
TERHADAP PANDEMI COVID-19 PADA TWITTER DENGAN METODE  
LONG SHORT-TERM MEMORY**

**SKRIPSI**

Oleh :  
**YULIANA ROMADHONI**  
**NIM. 18650012**

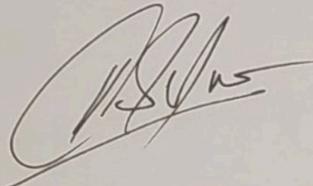
Telah Diperiksa dan Disetujui untuk Diuji  
Tanggal : 3 Juni 2022

Dosen Pembimbing I



Dr. Fachrul Kurniawan, M.MT IPM  
NIP. 19771020 200912 1 001

Dosen Pembimbing II



Okta Qomaruddin Aziz, M.Kom  
NIP. 19911019 201903 1 013

Mengetahui,  
Ketua Jurusan Teknik Informatika  
Fakultas Sains dan Teknologi  
Universitas Islam Negeri Maulana Malik Ibrahim Malang



Dr. Fachrul Kurniawan, M.MT  
NIP. 19771020 200912 1 001

HALAMAN PENGESAHAN

KLASIFIKASI KALIMAT PERBINCANGAN MASYARAKAT  
TERHADAP PANDEMI COVID-19 PADA TWITTER DENGAN METODE  
LONG SHORT-TERM MEMORY

SKRIPSI

Oleh:  
**YULIANA ROMADHONI**  
NIM. 18650012

Telah Dipertahankan di Depan Dewan Penguji dan Dinyatakan Diterima Sebagai  
Salah Satu Persyaratan untuk Memperoleh Gelar Sarjana Komputer (S.Kom)  
Pada Tanggal: 3 Juni 2022

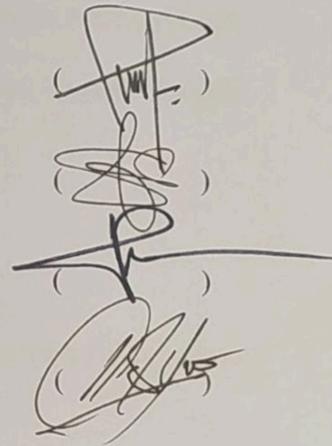
**Susunan Dewan Penguji**

Penguji Utama : Dr. Yunifa Miftachul Arif, M.T  
NIP. 19830616 201101 1 004

Ketua Penguji : Dr. M. Amin Hariyadi M.T  
NIP. 19670018 200501 1 001

Sekretaris Penguji : Dr. Fachrul Kurniawan, M.MT IPM  
NIP. 19771020 200912 1 001

Anggota Penguji : Okta Qomaruddin Aziz, M.Kom  
NIP. 19911019 201903 1 013



Mengetahui,  
Ketua Jurusan Teknik Informatika  
Fakultas Sains dan Teknologi  
Universitas Islam Negeri Maulana Malik Ibrahim Malang



Dr. Fachrul Kurniawan, M.MT IPM  
NIP. 19771020 200912 1 001

## PERNYATAAN KEASLIAN TULISAN

Saya yang bertanda tangan di bawah ini :

Nama : Yuliana Romadhoni  
NIM : 18650012  
Fakultas : Sains dan Teknologi  
Jurusan : Teknik Informatika  
Judul Skripsi : Klasifikasi Kalimat Perbincangan Masyarakat Terhadap Pandemi Covid-19 Pada Twitter Dengan Metode *Long short-term memory*

Menyatakan dengan sebenarnya bahwa Skripsi yang saya tulis ini benar-benar hasil karya saya sendiri, bukan merupakan pengambilalihan data, tulisan atau pikiran orang lain yang saya akui hasil tulisan atau pikiran saya sendiri, kecuali dengan mencantumkan sumber cuplikan pada daftar pustaka.

Apabila di kemudian hari terbukti atau dapat dibuktikan Skripsi ini hasil jiplakan, maka saya bersedia menerima sanksi atas perbuatan tersebut.

Malang, 3 Juni 2022  
Yang Membuat pernyataan,



Yuliana Romadhoni  
18650012

## HALAMAN MOTTO

لَئِنْ شَكَرْتُمْ لَأَزِيدَنَّكُمْ

*“Jika kalian mau bersyukur, maka Aku sungguh akan menambah nikmat bagi kalian.” (QS. Ibrahim: 7)*

## HALAMAN PERSEMBAHAN

بِسْمِ اللَّهِ الرَّحْمَنِ الرَّحِيمِ الْحَمْدُ لِلَّهِ رَبِّ الْعَالَمِينَ

Skripsi ini saya persembahkan untuk Kedua orang tua, Saudara, Keluarga,  
Seluruh guru, Dosen, Sahabat, Teman-teman seperjuangan, Orang-orang yang  
selalu memberikan dukungan dan selalu optimis pada penulis dan Diri saya  
sendiri.

**Terima kasih.**

## **KATA PENGANTAR**

Puji dan syukur atas kehadiran Tuhan semesta alam Allah SWT yang telah memberikan Rahmat dan hidayah-Nya, sehingga peneliti diberikan kemudahan dan keberkahan dalam setiap menyelesaikan skripsi ini. Penyusunan skripsi ini bertujuan untuk memenuhi syarat kelulusan bagi mahasiswa Teknik Informatika Universitas Islam Negeri Maulana Malik Ibrahim Malang. Keberhasilan penulisan skripsi ini tidak lepas dari dorongan dan bimbingan dari berbagai pihak. Untuk itu dalam kesempatan ini peneliti mengucapkan terima kasih yang sebesar-besarnya kepada :

1. Prof. Dr. H.M. Zainuddin, MA selaku rektor Universitas Islam Negeri Maulana Malik Ibrahim.
2. Dr. Sri Hariani, M.Si selalu dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Maulana Malik Ibrahim.
3. Bapak Dr. Fachrul Kurniawan, M.MT., IPM selaku Ketua Jurusan Teknik Informatika Universitas Islam Negeri Ibrahim Malang sekaligus sebagai dosen pembimbing I yang telah bersedia meluangkan waktunya dalam membimbing dan memberikan dorongan dan arahan kepada peneliti.
4. Bapak Okta Qomaruddin Aziz, M.Kom selaku dosen pembimbing II yang telah bersedia memluangkan waktunya dalam memberikan arahan dan membimbing kepada peneliti sehingga dapat menyelesaikan skripsi ini.
5. Bapak HARIYONO dan ibu MURJIATI selaku orang tua penulis yang telah memberikan dukungan yang berlimpah, baik dari segi ekonomi, moral dan spiritual serta Saudara kandung penulis Lailatul Maulidya yang selalu

memberi semangat, dukungan dengan menyiapkan konsumsi sehingga penulis bisa mengerjakan skripsi dengan lancar.

6. Seluruh dosen dan staff Jurusan Teknik Informatika yang telah memberikan ilmu dan pengalaman yang berharga.
7. Sahabat-sahabat saya terutama pada anggota grup “SUKSES DIWAKTU DEKAT”, “ROAD TO SKRIPSI” dan “ESKULIN” yang selalu membantu, memotivasi dan memberikan semangat terkait pengerjaan skripsi ini.
8. Saudara Teknik Informatika khususnya anggota UFO yang sama-sama mengejar gelar S.Kom di Universitas yang sama.
9. Sahabat dan teman-teman yang memberikan pencerahan dan semangat yang berlimpah kepada peneliti sehingga penelitian ini mampu saya kerjakan dengan lancar.
10. Diri saya sendiri yang mampu terus maju dan semangat walaupun terdapat beberapa kendala dan halangan ditengah-tengah perjalanan pengerjaan skripsi ini.

Dalam penulisan skripsi ini penulis menyadari kekurangan dalam penulisan dan penelitian ini, maka dari itu penulis berterimakasih kepada pihak yang membantu, penulis juga mengharapkan saran dan kritik yang membangun. Serta memohon maaf apabila terdapat kesalahan dalam penulisan.

Malang, 03 Juni 2022

Penulis

## DAFTAR ISI

<b>HALAMAN JUDUL .....</b>	<b>ii</b>
<b>HALAMAN PERSETUJUAN .....</b>	<b>iii</b>
<b>HALAMAN PENGESAHAN .....</b>	<b>iv</b>
<b>PERNYATAAN KEASLIAN TULISAN .....</b>	<b>v</b>
<b>HALAMAN MOTTO .....</b>	<b>vi</b>
<b>HALAMAN PERSEMBAHAN .....</b>	<b>vii</b>
<b>KATA PENGANTAR.....</b>	<b>viii</b>
<b>DAFTAR ISI.....</b>	<b>x</b>
<b>DAFTAR GAMBAR.....</b>	<b>xii</b>
<b>DAFTAR TABEL .....</b>	<b>xiii</b>
<b>ABSTRAK .....</b>	<b>xiv</b>
<b>ABSTRACT .....</b>	<b>xv</b>
<b>مستخلص البحث.....</b>	<b>xvi</b>
<b>BAB I PENDAHULUAN.....</b>	<b>1</b>
1.1 Latar Belakang .....	1
1.2 Pernyataan Masalah.....	6
1.3 Tujuan Penelitian.....	6
1.4 Batasan Masalah.....	7
1.5 Manfaat Penelitian.....	7
<b>BAB II STUDI PUSTAKA .....</b>	<b>8</b>
2.1 Penelitian Terkait .....	8
2.2 <i>Deep learning</i> .....	11
2.3 Natural Language Processing .....	13
2.4 <i>Long short-term memory (LSTM)</i> .....	14
<b>BAB III METODE PENELITIAN .....</b>	<b>20</b>
3.1 Prosedur Penelitian.....	20
3.1.1 Identifikasi Masalah.....	21
3.1.2 Studi Literatur .....	21
3.1.3 Pengumpulan Data.....	21
3.1.4 Desain Sistem .....	22
3.1.5 Implementasi dan Pengujian Sistem .....	22
3.1.6 Analisa Hasil Pengujian dan Kesimpulan.....	22

3.2 Desain Sistem .....	22
3.3 <i>Crawling</i> Data .....	23
3.4 Pelabelan Data .....	24
3.5 <i>Preprocessing</i> .....	24
3.5.1 <i>Cleansing</i> .....	25
3.5.2 <i>Case folding</i> .....	26
3.5.3 <i>Tokenizing</i> .....	26
3.5.4 <i>Stemming</i> .....	26
3.6 <i>Term Frequency–Inverse Document Frequency</i> (TF-IDF) .....	27
3.7 Alur Perhitungan <i>Long Short Term Memory</i> .....	29
3.8 Skenario Pengujian .....	33
<b>BAB IV IMPLEMENTASI DAN PEMBAHASAN .....</b>	<b>36</b>
4.1 Implementasi .....	36
4.1.1 <i>Input Data</i> .....	36
4.1.2 <i>Preprocessing</i> .....	38
4.1.3 Pembagian Data .....	40
4.1.4 Arsitektur Jaringan .....	41
4.1.5 Implementasi LSTM .....	43
4.2 Hasil Implementasi .....	55
4.3 Pembahasan .....	58
<b>BAB V SIMPULAN DAN SARAN .....</b>	<b>64</b>
5.1 Simpulan .....	64
5.2 Saran .....	64
<b>DAFTAR PUSTAKA</b>	
<b>LAMPIRAN</b>	

## DAFTAR GAMBAR

Gambar 2.1 Struktur jaringan saraf LSTM .....	16
Gambar 2.2 Flowchart LSTM.....	17
Gambar 3.1 Diagram Blok Penelitian .....	20
Gambar 3.2 Desain Sistem.....	23
Gambar 4. 1 Grafik Perbandingan Kelas Sentimen .....	38
Gambar 4. 2 Arsitektur LSTM.....	43
Gambar 4. 3 Grafik MSE pada <i>Learning_rate</i> 0.01 .....	45
Gambar 4.4 Grafik MSE pada <i>Learning_rate</i> 0.02 .....	45
Gambar 4.5 Grafik MSE pada <i>Learning_rate</i> 0.03 .....	46
Gambar 4.6 Grafik MSE pada <i>Learning_rate</i> 0.04 .....	46
Gambar 4.7 Grafik MSE pada <i>Learning_rate</i> 0.05 .....	47
Gambar 4.8 Grafik MSE pada <i>Learning_rate</i> 0.06 .....	47
Gambar 4. 9 Grafik MSE pada <i>Learning_rate</i> 0.07 .....	48
Gambar 4. 10 Grafik MSE pada <i>Learning_rate</i> 0.08 .....	48
Gambar 4. 11 Grafik MSE pada <i>Learning_rate</i> 0.09 .....	49
Gambar 4. 12 Grafik Akurasi dan <i>Loss</i> data <i>Training</i> pada <i>Epoch</i> 50 .....	50
Gambar 4. 13 Grafik Akurasi dan <i>Loss</i> data <i>Training</i> pada <i>Epoch</i> 100 .....	51
Gambar 4. 14 Grafik Akurasi dan <i>Loss</i> data <i>Training</i> pada <i>Epoch</i> 150 .....	51
Gambar 4. 15 Grafik Akurasi dan <i>Loss</i> data <i>Training</i> pada <i>Epoch</i> 200 .....	52
Gambar 4. 16 Grafik Akurasi dan <i>Loss</i> data <i>Training</i> pada <i>Epoch</i> 250 .....	52
Gambar 4. 17 Grafik Akurasi dan <i>Loss</i> data <i>Training</i> pada <i>Epoch</i> 300 .....	53
Gambar 4. 18 Confusion matrix data <i>training</i> .....	54
Gambar 4. 19 Golongan Klasifikasi.....	60

## DAFTAR TABEL

Tabel 2. 1 Perbedaan dengan penelitian terkait .....	10
Tabel 3.1 Data Labeling .....	24
Tabel 3.2 <i>Cleansing data</i> .....	25
Tabel 3.3 Proses <i>Case folding</i> .....	26
Tabel 3.4 Proses <i>Tokenizing</i> .....	26
Tabel 3.5 Proses <i>Stemming</i> .....	27
Tabel 3.6 Perhitungan TF .....	28
Tabel 3.7 Perhitungan IDF .....	28
Tabel 3.8 Perhitungan TF-IDF .....	28
Tabel 3.9 Hasil Perhitungan TF-IDF .....	29
Tabel 3.10 Hasil perhitungan LSTM .....	33
Tabel 3.11 <i>Confusion matrix</i> .....	34
Tabel 4.1 Sampel Data <i>Tweets</i> .....	37
Tabel 4. 2 Rasio Pembagian Data .....	41
Tabel 4. 3 Perbandingan <i>Layer</i> .....	42
Tabel 4.4 Perbandingan MSE berdasarkan <i>learning rate</i> .....	49
Tabel 4. 5 Menentukan Akurasi Optimum dengan <i>Epoch</i> .....	53
Tabel 4. 6 Penjabaran <i>Confusion matrix Training</i> .....	54
Tabel 4. 7 Confusion Matrix data testing .....	56
Tabel 4.8 Hasil Klasifikasi .....	57

## ABSTRAK

Romadhoni, Yuliana. 2022. **Klasifikasi Kalimat Perbincangan Masyarakat Terhadap Pandemi Covid-19 Pada Twitter Dengan Metode *Long short-term memory***. Skripsi. Jurusan Teknik Informatika, Fakultas Sains Dan Teknologi, Universitas Islam Negeri Maulana Malik Ibrahim Malang. Pembimbing: (I) Dr. Fachrul Kurniawan, M.MT. (II) Okta Qomaruddin Aziz, M.Kom.

---

Kata Kunci: *Klasifikasi Teks, Sosial Media, Pandemi Covid-19, Deep learning, Long short-term memory*.

Ditemukan virus baru yang bernama Covid-19 pada tanggal 31 Desember 2019 di daerah Wuhan, China. virus tersebut ramai diperbincangkan masyarakat pada media sosial. Perbincangan masyarakat ini dapat digunakan pemerintahan untuk memperbaiki kebijakan pada saat pandemi dengan cara klasifikasi teks. Klasifikasi teks ini membutuhkan metode yang terbaik untuk mengukur nilai akurasi, presisi, *Recall* dan *F1-Score*. *Long short-term memory* merupakan metode *deep learning* yang biasa digunakan proses klasifikasi dengan nilai akurasi yang tinggi. oleh karena itu, penelitian ini menggunakan metode tersebut untuk mengukur nilai akurasi dengan menggunakan Bahasa pemrograman python. Sebelum dilakukan perhitungan menggunakan metode LSTM, data yang akan dilakukan percobaan diambil dengan cara *crawling* data pada Twitter menggunakan API Twitter dan mendapatkan data sebanyak 3904 data *tweet* yang akan dilakukan pelabelan oleh ahli Bahasa, mendapatkan 2847 data positif dan 1057 data negatif data tersebut akan dilakukan *preprocessing* untuk mendapatkan data yang bersih, setelah itu dilakukan pembagian data menjadi 2 bagian dengan rasio 80:20 yaitu data *training* dan *testing* dan dilakukan pembobotan kata menggunakan metode TF-IDF. Metode *Long short-term memory* membutuhkan nilai *epoch* dan nilai *learning rate* yang akan digunakan untuk mencari nilai akurasi dan nilai MSE, *error* pada data *training*. Penelitian ini telah menetapkan *learning rate* terbaik yaitu 0.04 menghasilkan nilai MSE sebesar 0.0031 dan *epoch* 200 yang memiliki nilai akurasi tertinggi serta *error* yang rendah yaitu nilai akurasi *training* sebesar 0.997% dan nilai *error* sebesar 0.0128%. Serta mendapat nilai akurasi *testing* sebesar 80%, nilai *Recall* sebesar 84%, nilai Presisi sebesar 90% dan nilai *F1-Score* sebesar 87%.

## ABSTRACT

Romadhoni, Yuliana. 2022. **Classification of Public Conversations Against the Covid-19 Pandemic on Twitter Using the *Long short-term memory* Method.** Thesis. Department of Informatics Engineering, Faculty of Science and Technology, State Islamic University of Maulana Malik Ibrahim Malang. Counselor: (I) Dr. Fachrul Kurniawan, M.MT. (II) Okta Qomaruddin Aziz, M.Kom.

---

Keywords: Text Classification, Social Media, Covid-19 Pandemic, Deep learning, Long short-term memory.

A new virus named Covid-19 was discovered on December 31, 2019 in the Wuhan area, China. The virus has been widely discussed by the public on social media. This public discussion can be used by the government to improve policies during a pandemic using text classification. This text classification requires the best method to measure the value of accuracy, precision, recall, and F1-Score. Long short-term memory is a deep learning method commonly used in the classification process with a high accuracy value. Therefore, this study uses this method to measure the accuracy value using the Python programming language. Before calculating using the LSTM method, the data to be experimented with was taken by crawling data on Twitter using the Twitter API and getting 3904 tweet data that would be labeled by linguists, getting 2847 positive data and 1057 negative data the data would be preprocessed to get clean data, after that, the data is divided into 2 parts with a ratio of 80:20, namely training and testing data and word weighting is carried out using the TF-IDF method. The Long short-term memory method requires an epoch value and a learning rate value that will be used to find the accuracy value and MSE value, errors in the training data. This study has determined the best learning rate of 0.04 resulting in an MSE value of 0.0031 and an epoch of 200 which has the highest accuracy value and the lowest error is the training accuracy value of 0.997% and the error value of 0.0128%. As well as getting a testing accuracy value of 80%, a recall value of 84%, a precision value of 90% and an F1-Score value of 87%.

## مستخلص البحث

رومادوني ، يوليانا .2022. تصنيف المحادثات العامة ضد جائحة كوفيد 19- على تويتر باستخدام طريقة الذاكرة طويلة المدى. فرضية. قسم هندسة المعلوماتية ، كلية العلوم والتكنولوجيا ، جامعة الولاية الإسلامية مولانا مالك إبراهيم مالانج .مستشار (I) :دكتور فخر كورنياوان (II) M.MT.IPM اوكتا قمر الدين عزيز , M.Kom.

الكلمة الرئيسية :تصنيف النص ، وسائل التواصل الاجتماعي ، جائحة كوفيد- ١٩ ، التعلم العميق ، الذاكرة طويلة المدى

قد وُجدت اكتشاف فيروس جديد يسمى كوفيد -١٩ في ٣١ ديسمبر ٢٠١٩ في منطقة وهان بالصين. تمت تحدثت ومناقشة هذه الفيروس على نطاق واسع من قبل الجمهور على وسائل التواصل الاجتماعي. يمكن استخدام هذه المناقشة العامة من قبل الحكومة لتحسين السياسات أثناء الوباء من خلال تصنيف النص. هذا تصنيف النص أفضل وتحتاج طريقة لقياس قيمة الدقة والدقة والتذكر ودرجة F1. الذاكرة قصيرة المدى طويلة المدى هي طريقة تعلم عميق تستخدم بشكل شائع في عملية التصنيف بقيم دقة عالية. لذلك تستخدم هذه الدراسة هذه الطريقة لقياس قيمة الدقة باستخدام لغة برمجة الثعبان. قبل الحساب باستخدام طريقة ذاكرة طويلة المدى ، يتم أخذ البيانات التي سيتم استخدامها كتنجربة من خلال الزحف إلى البيانات على تويتر باستخدام واجهة برمجة تطبيق تويتر والحصول على ٣٩٠٤ بيانات تغريدة سيتم تصنيفها من قبل اللغويين ، والحصول على ٢٨٤٧ بيانات إيجابية و ١٠٥٧ بيانات سلبية. تتم معالجتها أولاً لتنظيف البيانات ، بعد ذلك يتم تقسيم البيانات إلى جزأين بنسبة ٨٠:٢٠ ، أي بيانات التدريب والاختبار ويتم ترجيح الكلمات باستخدام طريقة تردد وثيقة معكوس التردد. تتطلب طريقة الذاكرة طويلة المدى قيم العصر وقيم معدل التعلم التي سيتم استخدامها للعثور على قيم الدقة وقيم متوسط مربع الخطأ والأخطاء في بيانات التدريب. حددت هذه الدراسة أفضل معدل تعليمي قدره ٠,٠٤ وذلك لإنتاج قيمة متوسط مربع الخطأ قدرها ٠,٠٠٣ وعصر ٢٠٠ والذي يحتوي على أعلى قيمة دقة وأقل خطأ ، وهي قيمة دقة التدريب ٠,٩٩٧٪ وقيمة الخطأ ٠,٠١٢٨٪ . واحصل على قيمة دقة اختبار ٨٠٪ وقيمة استدعاء ٨٤٪ وقيمة دقة ٩٠٪ وقيمة F1-نتيجة ٨٧٪.

# BAB I

## PENDAHULUAN

### 1.1 Latar Belakang

Kasus penyakit yang menyerupai *pneumonia* yaitu coronavirus 2019 (COVID-19) di Wuhan, Cina pada 31 Desember 2019 telah tersebar luas (Herliandry *et al.*, 2020). Penyakit tersebut telah mempengaruhi 220 negara dan wilayah di penjuru dunia dengan kasus positif sebanyak 191.148.056, dan lebih dari 4.109.303 kasus kematian per tanggal 21 juli 2021 (*WHO Coronavirus (COVID-19) Dashboard / WHO Coronavirus (COVID-19) Dashboard With Vaccination Data*, n.d.). Adanya virus yang berkembang ini negara merasakan dampak pada perekonomian memaksa seluruh negara di dunia untuk mengurungkan rencana strategis yang telah ditetapkan yang kemudian digantikan dengan kebijakan tanggap darurat dengan mengerahkan seluruh sumber daya (Kurniawan & Kurniawan, 2021). Salah satu contohnya perusahaan & peneliti di seluruh dunia telah melakukan penelitian untuk mengatasi tantangan virus ini, guna mengurangi penyebaran dengan mengembangkan obat untuk penyakit ini.

Masa pandemi yang membingungkan ini, sains dan teknologi telah memainkan peran penting, misalnya pada awal munculnya wabah ketika China mulai mengungkapkan kemunculan virus ini, China berfokus pada sistem yang menerapkan kecerdasan buatan dengan mengandalkan teknologi seperti kamera pengenalan wajah untuk melacak pasien yang terinfeksi dengan riwayat perjalanan, robot untuk mengirimkan makanan dan obat-obatan, drone berpatroli yang bertugas untuk mendisinfektan tempat-tempat umum, dan menyiarkan pesan

audio kepada masyarakat luas agar mereka tetap mengisolasi diri di rumah masing-masing (Kumar *et al.*, 2020). Menyiarkan pesan kepada masyarakat luas dengan menggunakan drone patroli memiliki beberapa kendala dikarenakan drone hanya dapat menyampaikan pesan dalam jangkauan lingkungan yang sempit. Adapun cara efektif yang telah banyak dilakukan masyarakat di era milenial dalam menyampaikan informasi kepada masyarakat luas, bahkan informasi yang disampaikan dapat dijangkau seluruh dunia yaitu menyampaikan pesan melalui media sosial.

Banyaknya pengguna media sosial dan bebasnya setiap individu dalam memberikan tanggapan atau informasi membuat banyak beredarnya berita yang tidak benar (Hoax). Agar menjadi manusia yang lebih baik Islam mengajarkan untuk tidak membawa berita hoax, karena membawa berita bohong hal yang tercela dan dilarang oleh Allah SWT. Melalui kisah terbebasnya Siti Aisyah dari fitnah atau berita hoax yang dibuat oleh kaum musrikin, munafik, dan kafir yang ingin merusak rumah tangga Rasulullah SAW (*Risalah Mutiara Tauhid: ASBABUN NUZUL SURAH 24 - AN NUR, AYAT 11, 4, 22, n.d.*). yang telah dijelaskan pada firman Allah SWT dalam surat An-Nur ayat 11:

إِنَّ الَّذِينَ جَاءُوا بِالْإِفْكِ عُصْبَةٌ مِّنْكُمْ لَا تَحْسَبُوهُ شَرًّا لَّكُم بَلْ هُوَ خَيْرٌ لَّكُمْ لِكُلِّ امْرِئٍ مِّنْهُمْ مَا اكْتَسَبَ مِنَ الْإِثْمِ وَالَّذِي تَوَلَّى كِبْرَهُ مِنْهُمْ لَهُ عَذَابٌ عَظِيمٌ

*“Sesungguhnya orang-orang yang membawa berita bohong itu adalah dari golongan kamu juga. Janganlah kamu kira bahwa berita bohong itu buruk bagi kamu bahkan ia adalah baik bagi kamu. Tiap-tiap seseorang dari mereka mendapat balasan dari dosa yang dikerjakannya..”* (Q.S An-Nur : 11)

Dalam tafsir Jalalayn Q.S An-Nur : 11, menyatakan bahwa dusta yang sangat buruk yang telah dituduhkan terhadap Siti Aisyah r.a. Umulmukminin, ia dituduh berbuat zina yakni dari segolongan dari kaum Mukmin. Siti Aisyah berkata, bahwa

mereka adalah Hissan bin Tsabit, Abdullah bin Ubay, Misthah dan Hamnah binti Jahsy. (Jangan kalian kira bahwa berita bohong itu) hai orang-orang beriman selain dari mereka yang melakukan tuduhan tersebut(buruk bagi kalian, tetapi hal itu mengandung kebaikan bagi kalian) dan Allah akan segera memberikan pahalanya kepada kalian. Kemudian Allah swt menampilkan kesucian dari seorang Siti Aisyah r.a. Seorang yang telah membantunya yaitu Shofwan. Berhubung dengan peristiwa ini Siti Aisyah r.a. Berccerita bahwa beliau mengikuti suatu peperangan Bersama nabi dan pada saat istirahat beliau melakukan buang hajat namun ternyata setelah itu rombongan nabi akan melanjutkan perjalanan dan siti Aisyah r.a menyadari bahwa kalungnya hilang dan memilih pergi ketempat beliau buang hajat, dan beliau tertinggal rombongan.

Setelah itu beliau menunggu di tempat awal istirahat berharap rombongan mengetahui keberadaannya, sampai beliau tertidur dan shofyan melihatnya kemudia beliau segera mengenalku, karena ia pernah bertemu dan melihat sebelum ayat hijab diturunkan dan mengucapkan kalimat Istirja' lalu siti Aisyah menutupi wajahnya dengan hijab Demi Allah, ia tidak berbicara denganku sepeatah kata pun, terkecuali hanya kalimat Istirja'. Lalu kami melanjutkan perjalanan kemudian tersiarlah berita bohong yang sangat keji itu, semoga diberikan kebinasaan mereka yang membuat berita tersebut. Sumber pertama yang telah menyiarkannya ialah Abdullah bin Ubay bin Salul."

Selanjutnya Allah telah berfirman, ("Tiap-tiap seseorang dari mereka) akan dibalas kepadanya (dari dosa yang dikerjakannya) mengenai berita bohong ini. (Dan siapa di antara mereka yang mengambil bagian yang terbesar dalam penyiaran berita bohong itu) maksudnya orang yang menjadi biang keladi dan berperan

penting dalam penyiaran berita bohong ini, yang dimaksud adalah Abdullah bin Ubay (baginya azab yang besar") yakni neraka kelak di akhirat (*Surat An-Nur Ayat 11 / Tafsirq.Com, n.d.*).

Pelajaran dari surah An-Nur ayat 11 mendorong kita untuk menggunakan media sosial sebaik mungkin. Salah satu media sosial yang populer digunakan hingga saat ini ialah Twitter. Dengan lebih dari 573 juta pengguna aktif setiap bulannya, Indonesia ialah salah satu negara yang menepati negara pengguna Twitter terbanyak keenam pada April tahun 2021. Twitter ialah layanan microblogging dan jejaring sosial di mana pengguna dapat memposting pesan untuk mengutarakan pendapat ataupun membagikan informasi menggunakan "*tweet*" yang dibatasi hingga 240 karakter. Selama lebih dari satu *decade* (Rosenberg *et al.*, 2020).

Platform Twitter memiliki fitur "*Trends for you*" yang digunakan untuk melihat apa yang sedang ramai diperbincangkan di media sosial Twitter pada hari ini. Pada masa pandemi ini trending pada Twitter sering membahas seputar Covid-19, pengguna Twitter dapat melihat tanggapan-tanggapan pengguna lainnya mengenai pandemi Covid-19 mulai dari tanggapan positif hingga tanggapan negatif.

Tanggapan yang dituliskan oleh masyarakat melalui platform Twitter ini bisa digunakan dalam melakukan klasifikasi teks. Klasifikasi teks ialah tugas mengelompokkan teks pada satu atau lebih kategori. Maka dari itu proses klasifikasi ini biasa digunakan dalam menentukan nilai positif dan negatif pada suatu pembahasan, pada penelitian ini peneliti akan melakukan klasifikasi pada pembahasan masyarakat mengenai Covid-19. Hal ini dapat dimanfaatkan oleh Lembaga-lembaga yang berkaitan, seperti pemerintahan Indonesia dalam

melakukan evaluasi untuk menangani Covid-19 di Indonesia. Selain digunakan sebagai evaluasi, klasifikasi tanggapan masyarakat ini juga dapat digunakan oleh media penyiaran berita untuk mengelompokkan berita positif atau berita negatif secara otomatis, klasifikasi tentang Covid-19 juga dapat digunakan oleh Lembaga Kesehatan untuk evaluasi pelayanan Kesehatan di Indonesia dalam menangani Covid-19 ini. Model klasifikasi terdapat dua kategori yaitu model *Machine Learning* dan model *Deep learning*.

Pada penelitian Giarsyani, Nuli (2020) telah membandingkan dua model klasifikasi yaitu model *Machine Learning* dan *Deep learning* pada *Machine Learning* penelitiannya menggunakan 4 algoritma yaitu *Support Vector Machine*, *Naïve Bayes*, *Random Forest* Dan *Decision Tree*. Model klasifikasi *Deep learning* dengan menggunakan 3 metode yaitu *Long short-term memory* (LSTM), *Convolution Neural Network* (CNN), dan *Gated Recurrent Units* (GRU). Hasil perbandingan 2 model tersebut menyatakan model *Deep learning* lebih unggul daripada *Machine Learning* dengan akurasi terbaik sebesar 0.999 menggunakan metode *Long short-term memory* (LSTM) dan *Gated Recurrent Units* (GRU).

Penerapan metode *Long short-term memory* merupakan salah satu metode yang banyak direkomendasikan pada penerapan teks mining, dengan alasan metode *Long short-term memory* telah menghasilkan nilai performa yang tinggi. Algoritma *Long Short Term Memory* dapat memecahkan permasalahan pada *exploding* dan *vanishing gradient* (Winda Kurnia Sari, 2020). LSTM memiliki kemampuan untuk mengingat dan menyimpan informasi yang didapatkan pada sel memori yang berfungsi untuk ruang yang besar. Dengan mempertimbangkan penerapan model klasifikasi melalui beberapa sumber dari jurnal maka peneliti akan melakukan

penelitian klasifikasi kalimat perbincangan masyarakat pada media sosial Twitter menggunakan metode *Long Short Term Memory* (LSTM).

Penelitian ini perlu melakukan evaluasi model klasifikasi untuk mengukur sejauh mana klasifikasi menggunakan model dengan menggunakan parameter nilai *confusion matrix* yang akan menghasilkan nilai akurasi, presisi, *Recall* dan *F1-Score* (Novaković et al., 2017). Akurasi pada pengukuran ialah tingkat kedekatan dalam pengukuran nilai kuantitas pada nilai yang sebenarnya. Nilai Presisi merupakan proses pengulangan pengukuran terhadap kondisi yang tidak berubah-ubah dalam mendapatkan hasil yang sama. *Recall* ialah rasio kelas prediksi positif yang akan dibandingkan dengan seluruh data yang positif. *F1-Score* merupakan perbandingan pada rata-rata *recall* dan presisi yang akan dibobotkan (Azhari et al., 2021).

## **1.2 Pernyataan Masalah**

Berdasarkan penjelasan dari latar belakang penelitian tersebut maka terdapat sebuah pernyataan masalah yaitu Bagaimana nilai akurasi, presisi, *Recall* dan *F1-Score* pada klasifikasi menggunakan metode *Long short-term memory* pada kalimat perbincangan masyarakat terhadap pandemi Covid-19.

## **1.3 Tujuan Penelitian**

Pada pernyataan masalah terdapat tujuan penelitian yaitu Mengukur tingkat akurasi, presisi, *Recall* dan *F1-Score* klasifikasi menggunakan metode *Long Short Term Memory* pada perbincangan masyarakat terhadap pandemi Covid-19.

#### **1.4 Batasan Masalah**

Batasan masalah penelitian digunakan untuk menghindari perluasan permasalahan, maka dirancang batasan masalah. Adapun batasan masalah pada penelitian tersebut yaitu:

- a. Bahasa yang digunakan berupa *tweet* yaitu Bahasa Indonesia.
- b. Data *tweet* yang diklasifikasikan akan dibatasi sebanyak 10.000 *tweet* yang akan diambil menggunakan API Twitter dengan cara *crawling* data menggunakan Bahasa pemrograman python.
- c. Waktu pengambilan data mulai dari September 2020 hingga September 2021.

#### **1.5 Manfaat Penelitian**

Manfaat dilakukan penelitian tersebut adalah:

1. Mempermudah pemerintahan dalam memperbaiki kebijakan mengenai Covid-19 dengan memperhatikan tanggapan masyarakat pada Twitter.
2. Penelitian ini juga dapat digunakan untuk media penyiaran dalam menganalisis perbincangan seputar Pandemi Covid-19.

## **BAB II**

### **STUDI PUSTAKA**

Studi pustaka akan membahas penelitian-penelitian terdahulu yang akan digunakan sebagai acuan untuk mempermudah peneliti dalam memahami suatu hal yang berhubungan dengan penelitian dan sebagai perbandingan terhadap penelitian yang akan dilakukan.

#### **2.1 Penelitian Terkait**

Penelitian yang dilakukan oleh Das & Kolya (2021), membahas tentang klasifikasi sentimen Covid-19 yang berasal dari beberapa sumber diantaranya yaitu Twitter, Reddit dan data teks dari Web lainnya, untuk mencapai nilai akurasi yang relative tinggi peneliti menggunakan model *deep neural networks* pada *tweet* siaran langsung tentang virus corona dan prediksi perkembangan corona diwaktu yang akan datang. Peneliti membagi dua data yaitu sebagai data uji dan data latih. Dengan melakukan penelitian tersebut mendapat nilai akurasi yang sangat tinggi yaitu 90,67% yang berarti prediksi kenaikan Covid-19 dimasa yang akan datang dengan menggunakan model deep neural network akan tinggi, seperti yang terlampir pada kurva matrik prediksi pertumbuhan Covid-19.

Penelitian yang dilakukan oleh Hanifa *et al.* (2021), membahas seputar klasifikasi berita palsu menggunakan Bahasa Indonesia dengan melakukan perbandingan model *Recurrent Neural Network* antara *Long short-term memory* (LSTM) dengan *Gated Recurrent Unit* (GRU). Untuk mendapatkan nilai kinerja yang maksimal atau optimal dilakukan beberapa percobaan, penelitian tersebut

menetapkan nilai parameter *epochs*: 15 sebagai nilai kinerja yang optimal dengan menggunakan fungsi *optimizer gradient descent* berupa *rmsprop*, dan *batch\_size*: 57. Menghasilkan nilai akurasi 73% pada metode LSTM dan nilai akurasi 57% pada metode GRU.

Munasatya & Novianto (2020) dalam sebuah penelitiannya melakukan analisis sentimen dengan menggunakan bantuan *Natural language Processing* (NLP) dalam melakukan olah data agar dapat difahami oleh mesin atau computer, olah data ini umumnya disebut dengan *preprocessing* yang biasanya dilakukan *cleaning text*. Tahapan menggunakan *Natural Language Processing* dalam proses *preprocessing* ini ialah sebelum melakukan *preprocessing* terlebih dahulu *crawling* data *tweet* secara realtime. Setelah itu melakukan labelling data atau menentukan kalimat positif, netral dan negatif dengan menggunakan *lexicon based* yang melakukan perhitungan polaritas dengan kamus kata. Kemudian menggunakan metode multi *layer* perceptron dalam proses klasifikasi agar mengetahui nilai akurasi pada *tweet* presiden Jokowi. Dengan menggunakan pendekatan *Natural Language Processing* (NLP) akan mendapatkan secara otomatis hasil teks dan label dengan nilai performa yang tinggi yaitu 93.26% untuk nilai akurasi, 95.36% nilai *F-measure*, dan nilai *Recall* sebesar 94.21%

Salah satu penelitian yang dilakukan menggunakan metode *Long short-term memory* ialah penelitian oleh Nurrohmat & SN (2019), yang melakukan analisis sentimen pada review novel menggunakan metode *Long Short Term Memory* dilakukan *embedding layer*. *Layer* ini mengubah indeks bilangan positif yang *diinput* menjadi vektor ukuran tetap berdasarkan dimensi vektor dari kamus kata berdasarkan model *word2vec*. Pada analisis sentimen penelitian ini melakukan

perbandingan menggunakan metode *Naïve Bayes*, dan hasilnya dengan menggunakan metode *Long short-term memory* lebih tinggi nilai performanya yaitu nilai akurasi 72.85%, presisi 73%, *Recall* 72%, dan *f-measure* 72%. Nilai performa menggunakan metode *Naïve Bayes* pada penelitian ini mendapat nilai akurasi sebesar 67.88%, presisi 69%, *Recall* 68%, dan *f-measure* 68%.

Penelitian yang dilakukan oleh Winda Kurnia Sari (2020) menyatakan bahwa penelitiannya menggunakan metode *Long Short-Term Memory* dengan menambahkan fitur Glove dan Word2Vec dan menghasilkan nilai akurasi dengan menambahkan Glove sebesar 95,17% dan mendapatkan nilai akurasi sebesar 95,38% dengan menggunakan fitur Word2Vec.

Tabel 2. 1 Perbedaan dengan penelitian terkait

Penelitian	Metode dan Konsep pada penelitian terkait	Metode dan Konsep penelitian ini
(Winda Kurnia Sari, 2020)	Menggunakan metode LSTM dengan fitur Word2Vec dan Glove	Menggunakan metode LSTM dengan fitur TF-IDF
(Nurrohmat & SN, 2019)	Menggunakan metode LSTM data yang digunakan Review Novel Menggunakan fitur Word Embedding	Menggunakan metode LSTM data yang digunakan teks <i>tweet</i> Menggunakan fitur TF-IDF
(Dwi et al., 2021)	Menggunakan 6 tahapan Preprocessing (case folding, removal punctuation, tokenizing, stopword removal, replacing acronym, stemming)	Menggunakan , 4 tahapan Preprocessing ( <i>Cleansing data, case folding, tokenizing dan stemming</i> )
(Muslikah, 2021)	Menggunakan Data SMS, 3 tahapan preprocessing ( <i>case folding, punctuation removal dan tokenizing</i> ) dan dengan fitur <i>Word Embedding</i>	Menggunakan data teks <i>tweet</i> , 4 tahapan Preprocessing ( <i>Cleansing data, case folding, tokenizing dan stemming</i> ) dengan fitur TF-IDF

Tabel 2.1 menyatakan bahwa penelitian tersebut belum ada yang melakukan sesuai dengan penelitian terdahulu, dikarenakan rata-rata penelitian terdahulu melakukan pembobotan kata menggunakan Word2Vec, *Word Embedding* dan melakukan tahapan *preprocessing* yang berbeda.

## 2.2 *Deep learning*

*Deep learning* merupakan metode pembelajaran dengan cara menirukan cara kerja sistem otak manusia yang ditiru oleh mesin. *Deep learning* merupakan bagian dari *Machine Learning* dan *Artificial Intelligence* (AI), yang terus memahami dalam berbagai level pada abstraksi yang berbeda. Namun *Deep learning* jelas berbeda dengan Teknik *Machine Learning* tradisional, *Deep learning* dapat merepresentasikan data seperti video, gambar dan teks secara otomatis. Beberapa penelitian menunjukkan keberhasilan pada *Deep learning* pada penelitian klasifikasi teks ini sangat bergantung pada efektivitas word embedding. Perbedaan pada *Deep learning* dan *Machine Learning* sendiri yaitu pada *Deep learning* berupa evolusi yang mengenai banyak *neuron* dalam jaringan, penghubung pada satu jaringan dengan jaringan-jaringan lainnya lebih kompleks, *Deep learning* mampu melakukan ekstraksi fitur, karena jaringannya lebih kompleks maka daya komputasi yang dimiliki ikut membesar, apabila *Machine Learning* jaringan saraf tiruannya hanya dapat digunakan pada klasifikasi\_ (Lecun *et al.*, 2015).

*Deep learning* yang terdapat tiga bagian, bagian pertama disebut *input layer*, bagian kedua disebut *hidden layer*, dan bagian terakhir ialah *Output layer*. Masing-masing bagian terdapat fungsi, *input layer* berfungsi sebagai nilai masukkan yang akan disimpan pada setiap node-node. *Hidden layer* digunakan sebagai meminimalisirkan *error* yang ada pada *output*, karena semakin banyak *layer* yang dihasilkan maka semakin kecil *error* yang didapatkan. Fungsi dari *Output layer* sendiri yaitu untuk menampilkan hasil dari perhitungan sistem yang telah dilakukannya fungsi aktivasi pada lapisan kedua yaitu lapisan *hidden layer* yang berasal dari *input* yang diterima. *Deep learning* memungkinkan dalam mempelajari

dan memahami representasi data dengan beberapa tingkatan abstraksi menggunakan model komputasi yang berasal dari beberapa processing *layer* (Lecun *et al.*, 2015).

*Deep learning* sendiri biasa digunakan dalam pengolahan teks, gambar, dan suara. Penggunaan *Deep learning* sangat membantu dalam mengklasifikasikan teks yang berasal dari *tweet* tanggapan masyarakat Indonesia tentang Covid-19 pada tahun 2020-2021. *Deep learning* dapat digunakan untuk klasifikasi atau analisis sentimen. Klasifikasi atau kategorisasi merupakan sebuah tipe analisis data yang biasa digunakan peneliti untuk menentukan kelas atau kelompok label dari sampel yang ingin dilakukan klasifikasi. Klasifikasi umumnya dibagi menjadi dua yaitu metode *supervised* dan metode *unsupervised*, metode *unsupervised* merupakan metode tak terbimbing yang berarti memngklasifikasikan otomatis dengan cara menentukan kelompok tanpa menggunakan sampel. Klasifikasi dengan metode *supervised learning* ialah metode yang berusaha menemukan hubungan antara atribut target dan atribut masukan. Tujuan pada klasifikasi sendiri ialah sebuah proses untuk meningkatkan kehandalan hasil yang diperoleh dari data yang ada (Pasaribu *et al.*, 2020).

Klasifikasi teks pada penelitian ini juga bisa disebut dengan analisis sentimen yaitu cara untuk mengelompokkan sesuai polaritas yang berasal dari kalimat dalam dokumen dan mendapatkan hasil pengelompokkan sesuai tingkat aspek apakah kalimat yang digunakan pada dokumen, umumnya dibagi menjadi 2 aspek sifat, positif, negatif. Melakukan klasifikasi sentimen negatif atau sentimen positif, computer sistem akan belajar dari ulasan-ulasan user atau konsumen yang mana dapat mengklasifikasikan ulasan user (Pasaribu *et al.*, 2020).

Umumnya model klasifikasi teks dibagi menjadi dua model yaitu model *Deep learning* dan *Machine Learning*. Klasifikasi teks telah banyak dikembangkan dengan menggunakan bermacam-macam metode yang umumnya terbagi menjadi 3 kelompok, kelompok yang pertama yaitu klasifikasi teks berbasis statistic misalnya seperti Naïve Bayes, Support Vector Machine, K-Nearest Neighbor, dll. Kelompok kedua merupakan klasifikasi teks yang berbasis koneksi, seperti Artificial Neural Network. Kelompok terakhir ialah rule-based atau kelompok yang berbasis aturan yaitu Decision Tree (Ridok & Latifah, 2015).

### **2.3 Natural Language Processing**

Penelitian yang dilakukan oleh Maitri & Sutopo (2019), bertujuan sebagai pengukur keberhasilan kerja sebuah *chatbot* yang digunakan sebagai penyampaian informasi untuk pengguna (*user*), dengan menggunakan pendekatan *Text Mining* dan *Natural Language Processing* sebagai penalarannya, pada penelitian ini menggunakan proses diantaranya *Tokenizing* sebagai pemecahan kata, *filtering* berguna sebagai penghilangan kata tidak penting dan *Analyzing* sebagai pencocokan dengan basis data. Didapatkan kesimpulan bahwa *customer service* dapat memberikan respon kepada pelanggan tanpa adanya batas waktu dengan menggunakan aplikasi *chatbot* tersebut. Pada penelitian yang dilakukan mendapatkan hasil bahwa dengan menggunakan metode *text mining* sebagai pemahan kalimat dan pendekatan *Natural Language Processing* (NLP) lebih memudahkan dalam pencarian *keyword*.

*Natural Language Processing* (NLP) salah satu cabang ilmu *artificial Intelligence* (AI) yang membahas seputar pengolahan Bahasa natural, Bahasa natural sendiri merupakan Bahasa yang umumnya digunakan untuk berkomunikasi

oleh manusia. Bahasa natural manusia tidak dapat diterima secara langsung oleh computer, maka dari itu harus terdapat proses dan pemahaman terlebih dahulu agar maksud dari manusia atau user dapat dipahami dengan baik dan benar oleh computer. NLP ini merupakan area integral yang berasal dari ilmu computer yang mana linguistic komputasi dan pembelajaran mesin digunakan dengan luas (Jain Aditya, Kulkarni Gandhar, 2018).

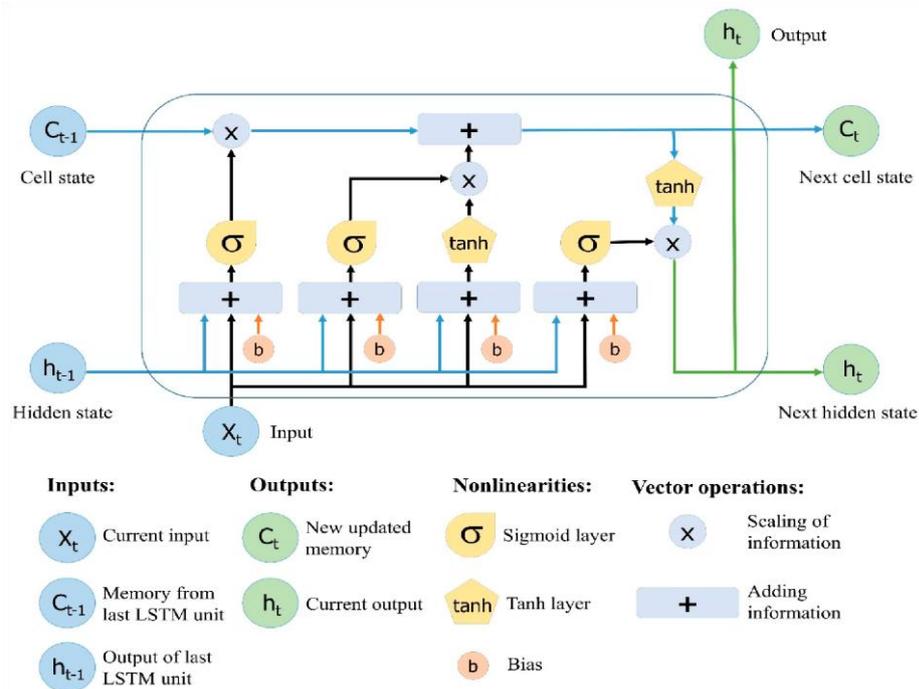
*Natural Language Processing* memiliki banyak kegunaan diantaranya yaitu, Analisis Sentimen, Pengkategorian atau Klasifikasi, Metode NLP baik digunakan pada saat klasifikasi teks, namun pada saat melakukan klasifikasi ulasan menggunakan NLP harus mendapatkan nilai akurasi yang tinggi, agar tidak ada kesalahan dalam penafsiran yang mengakibatkan kesalahan tindakan pada suatu perusahaan. Sistem NLP yang akan melakukan pengolahan kata/teks dimulai dengan adanya analisis morfologi. Analisis morfologi ini dilakukan dengan cara konversi teks dalam *query* atau dokumen guna mendapatkan beberapa macam morfologi kata-kata yang terlibat (Maitri & Sutopo, 2019).

#### **2.4 Long short-term memory (LSTM)**

*Long short-term memory* (LSTM) pertama kali dikenalkan oleh seorang Bernama Schmidhuber dan Hochreiter pada tahun 1997, hingga saat ini penggunaan metode LSTM banyak digunakan dalam bidang *forecasting* dan *recognition*. LSTM merupakan modifikasi dari arsitektur *Recurrent Neural Network* (RNN), RNN sendiri merupakan jenis arsitektur pada jaringan syaraf tiruan yang prosesnya dipanggil berkali-kali untuk memproses *input* yang berupa data sekuensial. Data sekuensial memiliki karakteristik pada sampel yang biasa diproses secara berurutan, suatu sampel pada data sekuensial ini saling memiliki hubungan erat

dengan satu sampel dan lainnya. Dalam tahap pemrosesan, *output* yang dikeluarkan berupa *state* internal yang mana hasil dari proses sampel setelahnya atau proses sampel sebelumnya. Modifikasi pada RNN menghasilkan model LSTM dengan cara menambahkan *memory cell* yang mampu menyimpan informasi dalam waktu yang lama, fungsi penambahan *memory cell* ini digunakan untuk menangani masalah saat adanya *vanishing gradient* pada RNN ketika dilakukan proses data *sequential* jangka panjang yang mengakibatkan lemahnya dalam mengetahui hubungan dari *sequence* yang jarak terpisahannya cukup jauh, didalam LSTM terdapat satu set gerbang yang digunakan untuk mengontrol ketika informasi akan masuk pada *memory*, yang akan memecahkan permasalahan *vanishing* atau *exploding* (Manaswi, 2018).

*Vanishing gradient* dapat mengakibatkan kegagalan pada RNN dalam menangkap adanya *long term dependencies*, yang mengakibatkan nilai akurasi berkurang pada suatu prediksi RNN (Zhao *et al.*, 2017). Hal ini mengakibatkan metode LSTM dapat sukses diimplementasikan pada sektor-sektor manapun, diantaranya yaitu pengenalan suara, pemahaman bahasa dari ucapan, dan pemodelan suatu bahasa. Banyaknya keunggulan-keunggulan yang dimiliki metode LSTM ini membuat metode ini ramai dilakukan penelitian dan pengembangan dalam klasifikasi teks. Model LSTM disusun dalam bentuk struktur rantai. Namun, modul berulang memiliki struktur yang berbeda. Berbeda dengan RNN perulangan pada jaringannya hanya dapat menggunakan satu *layer* yang sederhana yaitu *layer tahn*. LSTM sendiri memiliki 3 *gate* (*input gate*, *forget gate*, *output gate*) yang berinteraksi dengan metode komunikasi yang unik (Santoso *et al.*, 2021), berikut adalah gambar struktur jaringan saraf pada LSTM (Hanifa *et al.*, 2021).

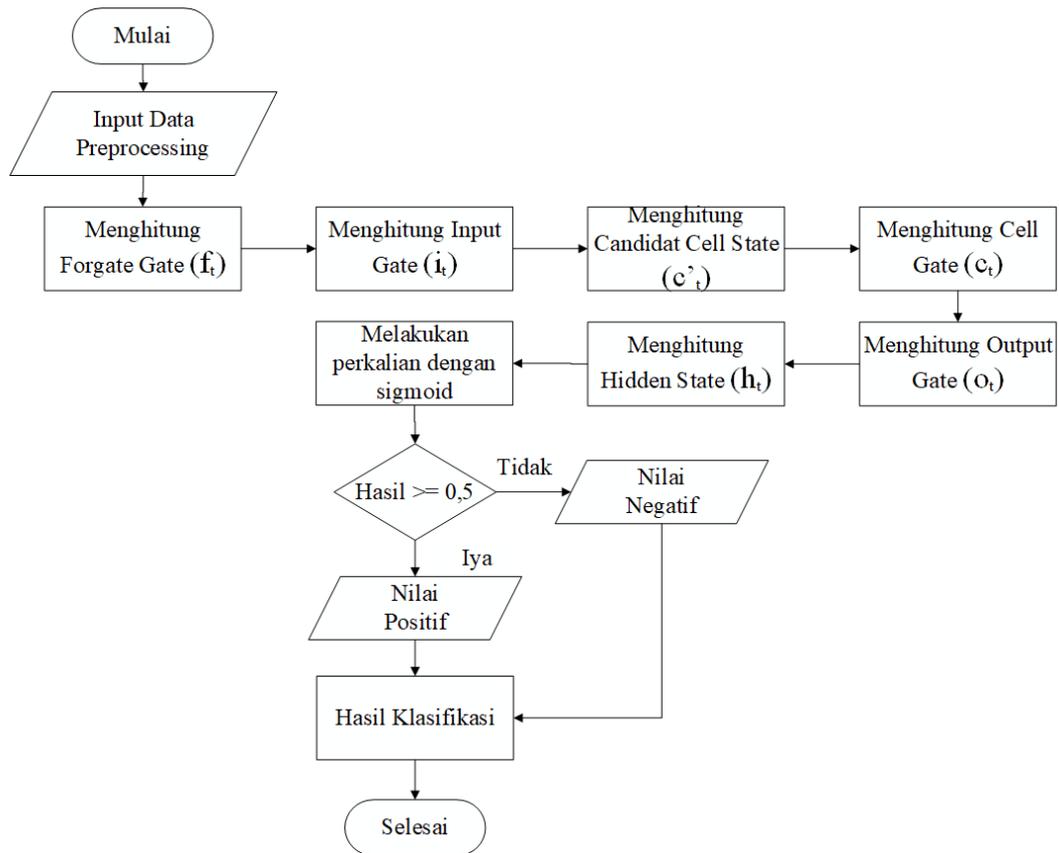


Gambar 2.1 Struktur jaringan saraf LSTM

Sumber: (Le *et al.*, 2019)

Gambar 2.1 merupakan penjelasan struktur jaringan saraf pada LSTM, Jaringan LSTM tipikal terdiri dari blok memori yang disebut sel. Dua state sedang ditransfer ke sel berikutnya, *cell state* dan *hidden state*. *Cell state* adalah rantai utama aliran data, yang memungkinkan data mengalir ke depan pada dasarnya tidak berubah. Namun, beberapa transformasi linier dapat terjadi. Data dapat ditambahkan atau dihapus dari *cell state* melalui gerbang *sigmoid*. Sebuah gerbang mirip dengan lapisan atau serangkaian operasi *matriks*, yang berisi bobot individu yang berbeda (Le *et al.*, 2019).

Pada LSTM terdapat empat fungsi aktivasi pada *input* setiap jaringan yang selanjutnya yang biasa disebut dengan *gates units* diantaranya: *input gate*, *forget gate*, *cell gate*, dan *output gate*. Berikut adalah algoritma atau penyelesaian dari metode *Long Short Term Memory*.



Gambar 2.2 Flowchart LSTM

Tahapan awal dalam membangun jaringan LSTM ialah mengidentifikasi suatu informasi yang tidak diperlukan dan akan dihapus dari sel. Pada tahap tersebut proses identifikasi dan hapus data ditentukan oleh fungsi aktivasi *sigmoid*, Fungsi dari *sigmoid* ini menentukan bagian dari *output* lama yang harus dihapus. Fungsi ini dilakukan pada gerbang yang disebut *Forget gate* ( $f_t$ ) dimana  $f_t$  adalah *vector* dengan nilai mulai dari nilai 0 hingga 1, sesuai dengan setiap angka dalam *cell state*, yang mana apabila hasil *output* 1 maka semua data dapat disimpan, apabila nilai *output* 0 maka data akan dihapus (Rais & Jondri, 2020) . Dalam rumusnya dituliskan sebagai berikut (Chung & Shin, 2018):

$$f_1 = \sigma(W_f x_1 + U_f h_{t-1} + b_f) \quad (2.1)$$

Langkah selanjutnya adalah memutuskan lalu menyimpan informasi dari *input* baru ( $x_t$ ) dalam *cell state* yang berguna untuk memperbarui *cell state*. Langkah ini berisi dua bagian, fungsi aktivasi *sigmoid* dan fungsi aktivasi *tanh*. Pertama, fungsi aktivasi *sigmoid* memutuskan apakah informasi baru harus diperbarui atau diabaikan (0 atau 1), dan kedua, fungsi aktivasi *tanh* memberi vektor nilai baru yang mana akan disimpan pada *memory cell* dengan memutuskan tingkat kepentingannya (-1 hingga 1). Kedua nilai dikalikan untuk memperbarui *cell state* (Chung & Shin, 2018). Dengan rumus yang dituliskan sebagai berikut:

$$i_1 = \sigma(W_i x_1 + U_i h_{t-1} + b_i) \quad (2.2)$$

$$c'_1 = \tanh(W_c x_1 + U_c h_{t-1} + b_c) \quad (2.3)$$

Selanjutnya *memory cell* baru akan ditambahkan ke *memory cell* lama ( $c_{t-1}$ ) dan mengakibatkan memori lama tergantikan dengan memori baru, sehingga menyebabkan ditemukannya rumus dari *Cell gates*( $C_t$ ). yang mana nilai didapatkan dari gabungan antara *input gate* dan *forget gate* (Chung & Shin, 2018).

$$C_1 = (f_1 * c_{t-1} + i_1 * c'_1) \quad (2.4)$$

Pada langkah terakhir, nilai keluaran dari *hidden state* ( $h_t$ ) didasarkan pada *output gate* ( $o_t$ ) tetapi merupakan versi yang telah difilter. Pertama, lapisan *sigmoid* memutuskan bagian mana dari *cell state* yang menghasilkan *output*. Lalu diletakkan nilai baru pada *memory cell* dengan menggunakan lapisan *tahn*, Selanjutnya, *output* dari gerbang *sigmoid* ( $o_t$ ) dikalikan dengan nilai baru yang dibuat oleh lapisan *tanh* dari *cell gates*( $c_t$ ), dengan nilai berkisar antara -1 dan 1 (Le et al., 2019).

$$o_1 = \sigma(W_o x_1 + U_o h_{t-1} + b_o) \quad (2.5)$$

$$h_1 = o_1 * \tanh(C_1) \quad (2.6)$$

Keterangan Rumus:

$f_t$  = Forget gate

$i_t$  = *Input gate*

$c_t$  = Cell gates

$o_t$  = *Output gate*

$h_t$  = Hidden State

$\check{c}_t$  = Cell Aktivasi

$b$  = Bias

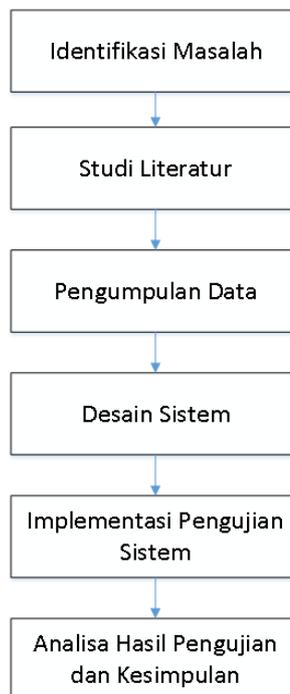
$W$  = Bobot

## BAB III

### METODE PENELITIAN

#### 3.1 Prosedur Penelitian

Prosedur penelitian merupakan rangkaian kegiatan yang akan dilakukan pada penelitian, agar penelitian yang dilakukan dapat berjalan secara sistematis dan terstruktur. Penelitian klasifikasi ini telah menggunakan pendekatan kuantitatif. Pendekatan ini biasa digunakan dalam penelitian sampel atau populasi dalam penelitiannya dibantu dengan instrument penelitian atau alat ukur, penelitian ini termasuk kedalam analisa data yang bersifat kuantitatif dengan tujuan menguji hipotesis yang telah didapatkan. Agar penelitian berjalan dengan lancar dan sesuai tujuan yang dicapai, maka dibuat gambaran prosedur penelitian yang berupa blok diagram penelitian.



Gambar 3.1 Diagram Blok Penelitian

Berdasarkan Gambar 3.1 Diagram Blok penelitian merupakan tahapan-tahapan agar sebuah penelitian dapat diselesaikan dengan terstruktur. Penjelasan dari Diagram blok penelitian.

### **3.1.1 Identifikasi Masalah**

Identifikasi masalah merupakan proses yang penting dalam sebuah penelitian selain latar belakang penelitian dan perumusan masalah, yang mana identifikasi masalah ini biasa dikenal dengan pengenalan masalah yang mendefinisikan bagaimana masalah pada penelitian ini terjadi.

### **3.1.2 Studi Literatur**

Studi literatur merupakan rangkaian kegiatan yang berhubungan dengan pengumpulan data pustaka yang bertujuan untuk dibaca, dicatat serta difahami. Studi literatur ini biasanya didapatkan dari penelitian yang telah dilakukan oleh peneliti sebelumnya yang biasa berbentuk buku, jurnal, dan majalah yang berkaitan dengan tujuan dan masalah sebuah penelitian. Literatur yang digunakan pada penelitian ini dengan topik yang membahas Klasifikasi, Natural Language Processing, *Long short-term memory*. Paper utama yang digunakan sebagai pedoman pada penelitian ini yaitu penelitian yang dilakukan oleh (Nurrohmat & SN, 2019) dan (Rais & Jondri, 2020).

### **3.1.3 Pengumpulan Data**

Pengumpulan data atau collecting data merupakan kegiatan mencari data dilapangan yang bertujuan untuk mendapatkan sebuah data baik berupa objek, teks, angka, tabel dll. Data tersebut akan dikelola dan diteliti pada penelitian agar dapat menjawab permasalahan pada penelitian.

### **3.1.4 Desain Sistem**

Desain sistem adalah perencanaan, penggambaran dan pengaturan atau pembuatan sketsa dari kumpulan elemen yang terpisah dan dikumpulkan agar menjadi suatu bentuk sistem yang harus diselesaikan.

### **3.1.5 Implementasi dan Pengujian Sistem**

Implementasi dan Pengujian Sistem tahapan ini dilakukan pengumpulan data dan desain sistem, maka dilakukan penerapan sebuah sistem dengan menggunakan data yang telah disediakan agar menjadi sebuah sistem yang dapat digunakan dengan baik, sebelum menggunakan sebuah sistem perlu melakukan pengujian sistem, agar mengetahui layak tidaknya sebuah sistem.

### **3.1.6 Analisa Hasil Pengujian dan Kesimpulan**

Analisa hasil pengujian dan kesimpulan ini merupakan penjelasan dari pengujian sistem yang telah dibuat.

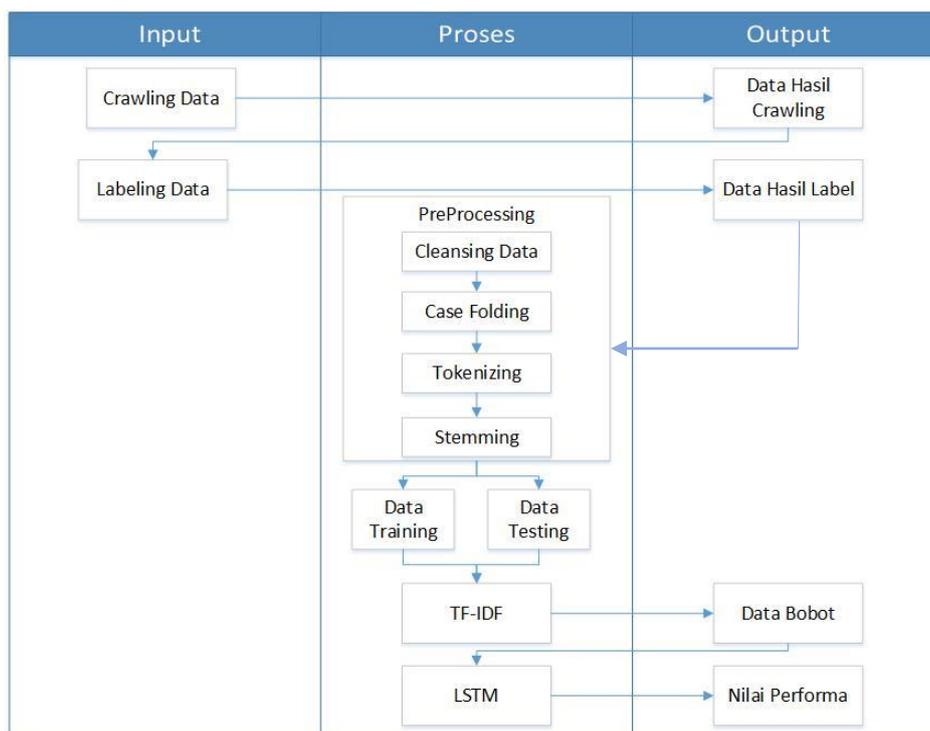
## **3.2 Desain Sistem**

Desain sistem pada penelitian tersebut menjelaskan bagaimana jalannya sistem yang akan dirancang mulai dari tahap pengumpulan data (*crawling data*), lalu dilanjut dengan proses *preprocessing* yang memiliki 4 tahap yaitu *Cleansing*, *Case folding*, *Tokenizing* dan *Stemming*. Selanjutnya terdapat tahap pembagian data, dan selanjutnya melakukan proses klasifikasi dan melakukan perhitungan

performa menggunakan metode LSTM. Desain sistem ini digambarkan dengan Flowchart sistem pada Gambar 3.2.

### 3.3 *Crawling Data*

Data yang akan digunakan pada penelitian ini termasuk kedalam data primer.



Gambar 3. 2 Desain Sistem

Data primer sendiri merupakan data yang perolehannya dilakukan secara langsung pada objek penelitian, yang biasanya berbentuk data mentah. Proses mendapatkan data primer biasanya lebih susah dibandingkan data sekunder, data primer biasa didapatkan melalui tahapan survei, wawancara atau pengumpulan data lainnya, namun pada penelitian ini data yang diperoleh dari hasil *crawling* data pada Twitter menggunakan Bahasa pemrograman python, pada proses *crawling* data ini peneliti menggunakan *keyword* “Pandemi Covid-19” pada *keyword* tersebut didapatkan data sebesar 9573 teks *tweet* yang didapat sekitar kurun waktu 1 tahun yaitu mulai dari September 2020 hingga September 2021.

### 3.4 Pelabelan Data

Setelah mendapatkan data tersebut dilakukan pelabelan data secara manual, pelabelan ini dilakukan oleh ahli yang telah menguasai dan memahami Bahasa dengan baik dan benar yang mana seorang ahli bahasa akan mengelompokkan kalimat positif dan kalimat negatif berdasarkan kata-kata yang telah tersusun pada teks *tweet* tersebut. Dataset yang akan diberi label ini yang akan dijadikan data *training* dan data *testing*. Berikut ini contoh dari teks *tweet* yang telah didapatkan dari Twitter dan dilakukan pelabelan manual oleh ahli Bahasa, ahli Bahasa yang akan melakukan pelabelan pada penelitian tersebut merupakan seorang guru Bahasa Indonesia SMP 2 Wlingi, Blitar atas nama Ibu Erni Wahyu Lindarti.

Tabel 3.1 Data Labeling

<i>Tweets</i>	Labeling Manual
RT @Ndukla2: Masyarakat yang tidak memiliki NIK saat ini bisa mendapatkan vaksinasi Covid 19. Kebijakan tersebut tertuang dalam Surat Edar...	Positif
RT @Ochi_Queen09_1: Terpuruk akibat adanya Pandemi, Pedagang Kaki 5 di Malioboro Kibarkan Bendera berwarna Putih <a href="https://t.co/PUsl8ZxXxy">https://t.co/PUsl8ZxXxy</a>	Negatif
Presiden Jokowi: Saya Mengajak Seluruh Elemen Bangsa Ikhtiar Melawan Pandemi COVID-19 <a href="https://t.co/zE4eJEnVFR...">https://t.co/zE4eJEnVFR...</a> <a href="https://t.co/YVIETRpbPy">https://t.co/YVIETRpbPy</a>	Positif
"Kadang ada juga orang yang ditangkap atau kriminalisasi dan juga dianggap Positif Covid-19. Padahal tidak terbukti. <a href="https://t.co/OGRD3SmFL5">https://t.co/OGRD3SmFL5</a>	Negatif
Sudah Rasakan Sendiri LaNyalla Optimis Vaksin Nusantara Mampu Atasi Pandemi Covid-19 <a href="https://t.co/EvVylyw6CV">https://t.co/EvVylyw6CV</a>	Positif
RT @Ndukla2: Pemerintah Penuhi Kebutuhan Oksigen dan Percepat Vaksinasi Covid-19 Berbasis Risiko Menurut Menkes, saat ini kebutuhan oksige...	Positif
RT @9iloveme_: 570 Dokter Meninggal akibat Terpapar Covid-19 Selama Pandemi <a href="https://t.co/3RMoO8ziqU">https://t.co/3RMoO8ziqU</a>	Negatif
30.168 Pasien Covid-19 Meninggal di Juli 2021, Tertinggi Selama Pandemi #LengkapCepatBeritanya #BeritaTerkini... <a href="https://t.co/txebqIdVcg">https://t.co/txebqIdVcg</a>	Negatif

### 3.5 Preprocessing

*Preprocessing* ini dilakukan untuk mempersiapkan sebuah data yang akan dilakukan proses klasifikasi, *preprocessing* sangat penting pada *text mining* dalam menghasilkan data yang lebih baik (Qosim *et al.*, 2021), hal ini juga mempengaruhi

hasil klasifikasi dan nilai performa, dikarenakan data yang dihasilkan *preprocessing* lebih baik daripada data yang belum dilakukan *preprocessing*, pada proses *preprocessing* ini dilakukan dengan menggunakan Bahasa python pada lampiran merupakan *Pseudocode Preprocessing*. Berikut tahapan-tahapan pada *preprocessing*.

### 3.5.1 *Cleansing*

Proses *Cleansing* ini digunakan untuk membersihkan kata yang tidak diperlukan atau yang tidak berpengaruh misalnya seperti menghilangkan tanda baca atau karakter symbol seperti (~!) @ & # \$ % ^ \* \_ + < > . , ? [ ] : { } | .), menghilangkan URL seperti ('bit.ly', 'Http'), menghapus angka yang ada, menghilangkan emoticon atau karakter seperti (😊 😊 😊), menghapus mention yang menggunakan awalan (@), menghapus hashtag yang menggunakan awalan (#), menghapus username/ *retweet* yang berawalan (rt) dan penghapusan kalimat yang sama (*remove duplicate*).

Tabel 3.2 *Cleansing data*

Sebelum <i>Cleansing</i>	RT @Ndukla2: Masyarakat yang tidak memiliki NIK saat ini bisa mendapatkan vaksinasi Covid-19. Kebijakan tersebut tertuang dalam Surat Edar...
Sesudah <i>Cleansing</i>	Masyarakat yang tidak memiliki NIK saat ini bisa mendapatkan vaksinasi Covid Kebijakan tersebut tertuang dalam Surat Edar

Tabel 3.2 merupakan hasil dari proses *Cleansing* data, yang mana masih terdapat *mention* dan nama *user* terdapat simbol titik (.), titik dua (:), dan dihasilkan kalimat tanpa *mention* nama *user* simbol yang tidak dibutuhkan.

### 3.5.2 Case folding

*Case folding* mempunyai fungsi sebagai pengubah dari huruf kapital menjadi huruf kecil, agar semua kata yang digunakan menjadi huruf kecil mulai dari awal kalimat, penamaan orang, penamaan kota dan lainnya. Berikut contoh dari penerapan *case folding*.

Tabel 3.3 Proses *Case folding*

Sebelum <i>Case folding</i>	Masyarakat yang tidak memiliki NIK saat ini bisa mendapatkan vaksinasi Covid Kebijakan tersebut tertuang dalam Surat Edar
Sesudah <i>Case folding</i>	masyarakat yang tidak memiliki nik saat ini bisa mendapatkan vaksinasi covid kebijakan tersebut tertuang dalam surat edar

### 3.5.3 Tokenizing

*Tokenizing* atau tokenisasi merupakan salah satu bagian penting dalam teks mining yang mana tokenisasi dilakukan sebagai pemisah sebuah kalimat menjadi kata kata. Acuan pemisah pada *tokenizing* ini berupa tanda baca dan spasi. Proses tokenisasi ini berguna agar mesin kita dapat memahami teks yang akan digunakan. Contoh hasil penerapan tokenisasi sebagai berikut.

Tabel 3.4 Proses *Tokenizing*

Sebelum <i>Tokenizing</i>	masyarakat yang tidak memiliki nik saat ini bisa mendapatkan vaksinasi covid kebijakan tersebut tertuang dalam surat edar
Sesudah <i>Tokenizing</i>	“masyarakat”, “yang”, “tidak”, “memiliki”, “nik”, “saat”, “ini”, “bisa”, “mendapatkan”, “vaksinasi”, “covid”, “kebijakan”, “tersebut”, “tertuang”, “dalam”, “surat”, “edar”.

### 3.5.4 Stemming

*Stemming* merupakan proses yang berguna untuk mengubah suatu kata menjadi kata dasar, dengan cara menghapus imbuhan yang ada pada suatu kata (affixes). Baik kata imbuhan yang ada diawal kata (prefixes), imbuhan yang ada ditengah atau sisipan (infixes), imbuhan yang ada diakhir kalimat (suffixes), maupun imbuhan kombinasi yang ada dari awalan dan imbuhan

akhiran (confixes). Berikut merupakan penerapan *stemming* pada penelitian tersebut.

Tabel 3.5 Proses *Stemming*

Sebelum <i>Stemming</i>	“masyarakat”, “yang”, “tidak”, “memiliki”, “nik”, “saat”, “ini”, “bisa”, “mendapatkan”, “vaksinasi”, “covid”, “kebijakan”, “tersebut”, “tertuang”, “dalam”, “surat”, “edar”.
Sesudah <i>Stemming</i>	“masyarakat”, “yang”, “tidak”, “milik”, “nik”, “saat”, “ini”, “bisa”, “dapat”, “vaksinasi”, “covid”, “bijak”, “tersebut”, “tuang”, “dalam”, “surat”, “edar”.

### 3.6 Term Frequency–Inverse Document Frequency (TF-IDF)

Data telah melalui tahapan *preprocessing*, selanjutnya data tersebut akan dilakukan proses klasifikasi menggunakan metode *Long short-term memory*, namun sebelum melakukan perhitungan dengan metode dilakukan proses pembobotan kata menggunakan TF-IDF.

Tahap TF-IDF ini dilakukan untuk sebuah data yang telah melewati tahapan *preprocessing* yang mana akan dilakukan perhitungan nilai frekuensi kemunculan sebuah kata atau mencari bobot pada setiap kata. Munculnya sebuah kata pada sebuah dokumen akan mempengaruhi besar kecilnya sebuah bobot, semakin jarang suatu kata yang muncul maka semakin besar nilai bobot atau IDF yang dihasilkan. Berikut merupakan rumus dan implementasi TF-IDF pada penelitian tersebut.

$$TF(w) = \frac{\text{Berapa kali kata } w \text{ muncul dalam dokumen } d}{\text{Jumlah total kata dalam dokumen } d} \quad 3.1$$

$$DF(w) = \log_e \frac{\text{Jumlah total dokumen}}{\text{Jumlah total dokumen dengan kata } w} \quad 3.2$$

$$TF - IDF(w) = TF(w) \times IDF(w) \quad 3.3$$

Tabel 3.6 Perhitungan TF

Dokumen	Kata	F.Kata	TF
D1	Pandemi	1	$1/3 = 0.33$
D2		1	$1/3 = 0.33$
D3		0	$0/3 = 0$

Tabel 3.6 merupakan data untuk melakukan perhitungan TF, apabila pada dokumen terdapat *keyword* yang telah ditentukan maka nilai frekuensi kata dihasilkan nilai 1 dan apabila tidak terdapat *keyword* maka frekuensi kata berjumlah 0 data. Setelah mendapatkan frekuensi kata dibagikan dengan banyak data. Setelah mendapatkan nilai *Term Frequency*. Maka dilakukan dengan melakukan perhitungan *Inverse Document Frequency* (IDF) seperti pada tabel 3.7.

Tabel 3.7 Perhitungan IDF

Dokumen	TF	Kata	IDF
D1	0.33	Pandemi	$\log \frac{3}{2} = 0,17609$
D2	0.33		
D3	0		

Tabel 3.7 merupakan tahapan dalam perhitungan IDF, melakukan perkalian log jumlah dokumen dibagi dengan total dokumen yang memiliki keyword.

Tabel 3.8 Perhitungan TF-IDF

Dokumen	Kata	TF	IDF	TF-IDF
D1	Pandemi	0.33	0,17609	0,0587
D2		0.33		0,0587
D3		0		0

Tabel 3.8 merupakan Langkah akhir pada perhitungan TF-IDF dengan melakukan perkalian perhitungan TF dengan perhitungan IDF dan menghasilkan nilai TF-IDF.

Tabel 3.9 Hasil Perhitungan TF-IDF

Token	TF-IDF		
	D1	D2	D3
pungkiri	0,159	0	0
pandemi	0,0587	0	0,0587
covid	0,0587	0,0587	0
dampak	0,159	0	0
buruk	0,159	0	0
kesehatan	0,159	0	0
mental	0,159	0	0
presiden	0	0,159	0
jokowi	0	0,159	0
saya	0	0,159	0
ajak	0	0,159	0
elemen	0	0,159	0
bangsa	0	0,159	0
ikhtiar	0	0,159	0
lawan	0	0,159	0
rasa	0	0	0,159
sendiri	0	0	0,159
optimis	0	0	0,159
vaksin	0	0	0,159
nusantara	0	0	0,159
atasi	0	0	0,159

Hasil dari Tabel 3.9 merupakan pembobotan kata yang menggunakan metode TF-IDF yang mana hasil *vector* yang dihasilkan dapat digunakan untuk perhitungan klasifikasi menggunakan metode LSTM. Nilai terendah dihasilkan oleh kata “Pandemi” dan kata “Covid” yang mana kata tersebut muncul pada 2 dokumen dari 3 dokumen yang ada dan menghasilkan nilai sebesar 0,117394.

### 3.7 Alur Perhitungan *Long Short Term Memory*

Proses pembobotan kata telah dilakukan, dan didapatkan hasil pembobotan kata yang ada pada ketiga dokumen yang diteliti. Selanjutnya melakukan proses

perhitungan menggunakan metode *Long Short Term Memory*. Perhitungan manual yang akan dilakukan ini telah menetapkan nilai bobot (W, U) dan nilai bias (b) dengan nilai yang telah diasumsikan oleh penelitian penelitian yang telah dilakukan, didapatkan nilai tersebut dari pembatasan bobot dan bias yang digunakan sebagai berikut:

$$\begin{bmatrix} W_i & U_i & b_i \\ W_c & U_c & b_c \\ W_f & U_f & b_f \\ W_o & U_o & b_o \end{bmatrix} = \begin{bmatrix} 0.5 & 0.25 & 0.01 \\ 0.3 & 0.4 & 0.05 \\ 0.03 & 0.06 & 0.002 \\ 0.02 & 0.04 & 0.025 \end{bmatrix}$$

Setelah itu kita telah memiliki nilai vector yang telah dihitung menggunakan TF-IDF. *Input* yang akan dilakukan perhitungan ini berupa kalimat “ikhtiar vaksin lawan pandemi covid” sebelum melakukan perhitungan maka dilakukan dengan menentukan *padding*, nilai *padding* ini akan digunakan pada awal *inputan* yang bernilai 0, pada penelitian tersebut telah menentukan nilai *padding* sebesar 10. Berikut *matriks* dari kalimat yang akan dilakukan proses klasifikasi.

$$\begin{bmatrix} -0.048577 \\ -0.048577 \\ -0.048577 \\ -0.048577 \\ -0.048577 \\ 0.159040 \\ 0.159040 \\ 0.159040 \\ 0.058697 \\ 0.058697 \end{bmatrix}$$

Matrik tersebut merupakan *input* X1 hingga X10. Yang mana nilai X1 hingga X5 merupakan tambahan nilai agar *padding* sebuah kalimat menjadi sebesar 10 *padding*. Proses perhitungan LSTM adalah sebagai berikut.

*Input:*

$$X1 = -0.048577$$

Setelah mengetahui nilai bobot, bias dan nilai  $X$  nya, perhitungan manual dimulai dari menentukan nilai *forget gate* dimana  $W_f$  bernilai 0.03,  $U_f$  bernilai 0.06,  $h_{t-1}$  bernilai 0, dan  $b_f$  bernilai 0.002, maka didapatkan nilai. 0.500150047 dengan persamaan berikut.

*Forget Gate :*

$$f_1 = \sigma(W_f x_1 + U_f h_{t-1} + b_f)$$

$$f_1 = \sigma(0.03 * -0.048577 + 0.06 * 0 + 0.002)$$

$$f_t = \sigma(0.0005406)$$

$$f_t = 0,500150047$$

Langkah selanjutnya yaitu menentukan nilai *input gate* dengan nilai  $W_i$  sebesar 0.5,  $U_i$  sebesar 0.25 dan  $b_i$  bernilai 0.01, dan didapatkan hasil 0.495009186 dengan persamaan sebagai berikut.

*Input gate :*

$$i_1 = \sigma(W_i x_1 + U_i h_{t-1} + b_i)$$

$$i_1 = \sigma(0.5 * -0.048577 + 0.25 * 0 + 0.01)$$

$$i_1 = \sigma(-0.0143257)$$

$$i_1 = 0,495009186$$

Setelah menemukan nilai *forget gate* dan *input gate* dilanjutkan untuk menghitung *Candidate cell state*, pada perhitungan *candidate cell state* ini sama dengan persamaan *forget gate* dan *input gate* namun pada *candidate cell state* menggunakan nilai  $W_c = 0.3$  dan  $b_c = 0.05$  maka didapatkan hasil sebagai berikut.

*Candidate cell state :*

$$C'_1 = \tanh(W_c x_1 + U_i h_{t-1} + b_c)$$

$$C'_1 = \tanh(0.3 * -0.048577 + 0.4 * 0 + 0.05)$$

$$C'_1 = \tanh(0.0574056)$$

$$C'_1 = 0.0574056$$

*Cell State* ini dilakukan setelah ketiga operasi perhitungan tersebut, karena pada *Cell gate* membutuhkan nilai *Forget gate*, *Input gate* dan *Candidate cell state* maka dihasilkan nilai berikut ini.

*Cell state* :

$$C_1 = (f_1 * C_{t-1} + i_1 * C'_1)$$

$$C_1 = (0.50015005 * 0 + 0,495009186 * 0.0574056)$$

$$C_1 = 0.0195303$$

Langkah terakhir dari perhitungan manual ini yaitu menemukan nilai *Output gate* dan *Hidden State*, pada *output gate* didapatkan nilai sebesar 0.4997568, dan pada *hidden state* didapatkan nilai sebesar 0.0089108. dengan persamaan berikut ini.

*Output gate* :

$$o_1 = \sigma (W_o x_1 + U_o h_{t-1} + b_o)$$

$$o_1 = \sigma(0.02 * -0.048577 + 0.04 * 0 + 0.025)$$

$$o_1 = \sigma(-0,0009729)$$

$$o_1 = 0.4997568$$

$$h_1 = o_1 * \tanh(C_1)$$

$$h_1 = 0.4997568 * \tanh(0.0195303)$$

$$h_1 = 0.0089108$$

Dari proses yang telah dilakukan menghasilkan nilai *hidden state* ( $h_1$ ) sebesar 0.0089108, setelah melakukan *hidden state*  $X1 = -0.048577$  ini dipakai sebagai *input* tambahan untuk proses selanjutnya. Proses perhitungan ini akan diulangi dengan mengganti *input*  $X1$  menjadi *input*  $X2$  hingga *input*  $X10$ . Hasil dari *forget gate*, *input gate*, *candidate cell state*, *cell state* dan *output gate* dari  $X1$  hingga  $X10$  ditampilkan pada Tabel 3.10 berikut ini

Tabel 3.10 Hasil perhitungan LSTM

<i>Input</i>	f	i	$C'$	C	o	h
-0,04865	0,50014	0,49572	0,05741	0,01758	0,49976	0,00895
-0,04865	0,50014	0,49572	0,05741	0,01758	0,49976	0,00895
-0,04865	0,50014	0,49572	0,05741	0,01758	0,49976	0,00895
-0,04865	0,50014	0,49572	0,05741	0,01758	0,49976	0,00895
-0,04865	0,50014	0,49572	0,05741	0,01758	0,49976	0,00895
0,15904	0,50169	0,57237	0,09771	0,05104	0,50080	0,02556
0,15904	0,50169	0,57237	0,09771	0,05104	0,50080	0,02556
0,15904	0,50169	0,57237	0,09771	0,05104	0,50080	0,02556
0,05870	0,50094	0,50984	0,06761	0,03447	0,50029	0,01724
0,05870	0,50094	0,50984	0,06761	0,03447	0,50029	0,01724

Dari perhitungan LSTM tersebut telah didapatkan hasil yaitu  $h_{10}$  dengan nilai 0,0140606. langkah selanjutnya merupakan *output layer* dengan aktivasi *sigmoid*. Nilai yang telah dihasilkan *layer LSTM* akan dilakukan perkalian dengan *sigmoid* untuk mendapatkan hasil nilai antara 0 atau 1. Nilai 0,01724 dikalikan dengan fungsi *sigmoid* mendapatkan hasil 0.500311. *Output layer* mengkategorikan hasil < 0.5 menjadi 0 (Negatif) dan  $\geq 0.5$  menjadi 1 (Positif). Hasil yang didapat lebih dari 0.5 maka kalimat yang digunakan akan diklasifikasikan sebagai sentimen positif.

### 3.8 Skenario Pengujian

Tahapan ini merupakan tahapan terakhir yang dilakukan yaitu tahapan menghitung nilai akurasi, *Recall*, Presisi, dan *F1-Score*. Pembagian data pada penelitian tersebut akan dibagi menjadi 2 bagian yaitu sebagai data *training* dan data *testing*, pembagian dataset ini menjadi 80% data *training* dan 20% data *testing*. Data *training* ini merupakan data yang digunakan untuk *training*/pelatihan model,

Data *testing* sendiri merupakan *testing* model yang akan dijadikan pertimbangan atau simulasi penggunaan model pada dunia nyata. Maka dari itu pembagian data *training* lebih besar daripada data *testing*, apabila data *training* terlalu sedikit atau pembagiannya terlalu kecil maka model tidak akan bisa memahami atau tidak bisa belajar. Data *training* dan data *testing* tersebut akan dilakukan proses *preprocessing*. Menghitung performa klasifikasi menggunakan *Confusion matrix*. *Confusion matrix* sendiri biasanya berbentuk table yang digunakan untuk menghitung performa dari sebuah proses atau model klasifikasi dimana true false telah diketahui. Berikut merupakan dan tabel *Confusion matrix*.

Tabel 3.11 *Confusion matrix*

Kelas Sebenarnya	Kelas Prediksi	
	Negatif	Positif
Negatif	<i>True Negative (TN)</i>	<i>False Positive (FP)</i>
Positif	<i>False Negative (FN)</i>	<i>True Positive (TP)</i>

Tabel 3.11 merupakan table *confusion matrix* yang digunakan untuk *binary classifier* yang berarti 2 *class*, pada *confusion matrix* ini juga bisa dilakukan penambahan *class*. Berikut merupakan penjelasan dari Tabel 3.10.

1. *True negative (TN)* merupakan jumlah data yang bernilai negatif dan telah diprediksi negative.
2. *False Positive (FP)* merupakan jumlah data yang bernilai negatif namun diprediksi positif.
3. *False negative (FN)* merupakan jumlah data yang memiliki nilai positif namun diprediksi sebagai negatif.
4. *True Positive (TP)* merupakan jumlah data yang memiliki nilai Positif dan diprediksi positif.

Persamaan *Confusion matrix* yang dapat digunakan sebagai perhitungan untuk menentukan performa sebagai berikut.

$$Akurasi (\%) = \frac{TP + TN}{TP + TN + FP + FN} \times 100\% \quad 3.3$$

$$Precision (\%) = \frac{TP}{TP + FP} \times 100\% \quad 3.4$$

$$Recall (\%) = \frac{TP}{TP + FN} \times 100\% \quad 3.5$$

$$F1 - Score(\%) = \frac{2 \times Precision \times Recall}{Precision + Recall} \times 100\% \quad 3.6$$

## BAB IV

### IMPLEMENTASI DAN PEMBAHASAN

Pada bab uji coba dan pembahasan ini memaparkan tentang implementasi uji coba dan pembahasan yang telah dilakukan peneliti. Setelah itu peneliti akan menjelaskan hasil dan skenario uji coba dengan integrasi islam pada penelitian ini.

#### 4.1 Implementasi

Implementasi merupakan sebuah alur yang akan dilakukan pengujian sebuah system. Pengujian yang akan diimplementasikan yaitu menghitung nilai Akurasi, Presisi, *Recall* dan *F1-Score* dalam melakukan klasifikasi kalimat perbincangan masyarakat pada Twitter dengan menggunakan tabel *confusion matrix*.

##### 4.1.1 Input Data

Data yang diambil pada media sosial Twitter dengan cara *crawling* pada *crawling* ini peneliti harus memiliki akun terlebih dahulu untuk dapat menghubungkan pada API Twitter, mendapatkan *consumer key* dan *consumer secret*, setelah mendapatkan kedua code tersebut selanjutnya masukkan pada program untuk dilakukan *crawling* data pada twitter selanjutnya memberikan *keyword* atau kata kunci yang telah ditentukan pada penelitian ini, peneliti memasukkan *keyword* “pandemi covid”. Setelah mendapatkan data *crawling* tersebut akan dilakukan pelabelan oleh ahli bahasa menghasilkan dataset sebanyak 3904 data yang telah terbagi menjadi 2 kelas yaitu positif dan negatif pada kelas negatif terdapat 1057 data, pada

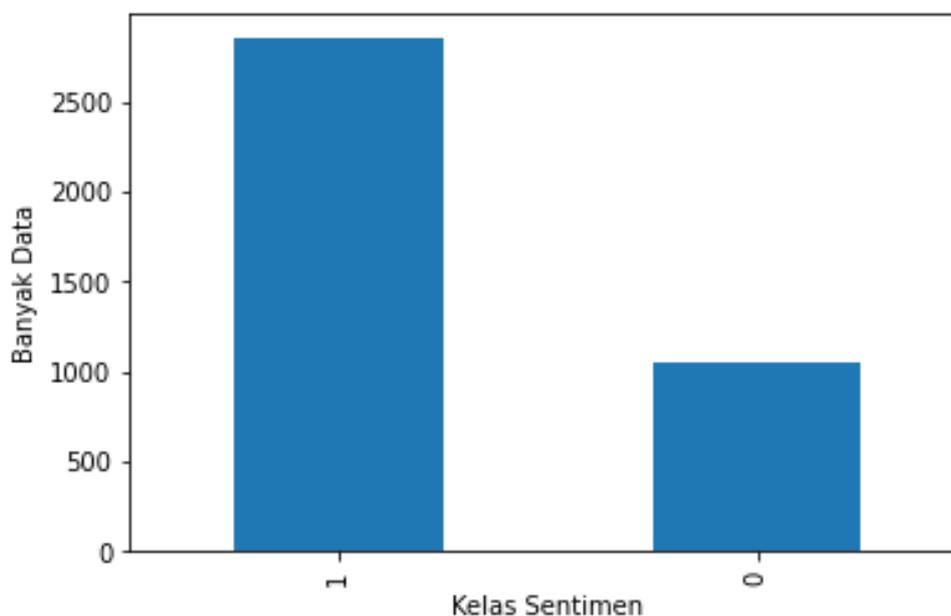
kelas positif terdapat 2847 data. Pada tabel 4.1 merupakan sampel dataset penelitian.

Tabel 4.1 Sampel Data *Tweets*

<i>Tweets</i>	Labeling Manual
"Ancaman Kelaparan di Tengah Pandemi" Pandemi Covid-19 di Indonesia <a href="https://t.co/DI0">https://t.co/DI0</a>	Negatif
"Kadang juga ada orang yang tangkap atau kriminalisasi dan juga dianggap Positif Covid19. Padahal tidak ada bukti... <a href="https://t.co/OGRD3SmFL5">https://t.co/OGRD3SmFL5</a>	Negatif
Masa pandemi ini setiap hari timeline saya dan mungkin juga yang lainnya, selalu dihiasi oleh berita duka, kecewa, & sedih. Sedikit... <a href="https://t.co/BFzjcFoY0f">https://t.co/BFzjcFoY0f</a>	Negatif
Aktivitas fisik menjadi terbatas selama pandemi Covid-19. Alhasil, anak lebih sering menghabiskan lebih banyak waktu depan hp... <a href="https://t.co/7fwwYM2t4w">https://t.co/7fwwYM2t4w</a>	Negatif
Anak2 di Surabaya Butuh Perlindungan pada saat Pandemi Covid-19 <a href="https://t.co/IgrwMIMIZH">https://t.co/IgrwMIMIZH</a>	Negatif
Jumlah limbah medis melonjak berkali-kali lipat selama pandemi Covid-19. Lonjakan belum dimbangi ketersediaan fasilitas... <a href="https://t.co/jfZQxpgtp9">https://t.co/jfZQxpgtp9</a>	Negatif
Pandemi Covid-19 memaksa beberapa Satuan Pendidikan di bawah Kemenag, menutup sekolahnya. Hal ini terjadi se... <a href="https://t.co/d5XZultaF2">https://t.co/d5XZultaF2</a>	Negatif
Pandemi Covid 19 Melonjak, Indikator Perekonomian Kembali Memburuk <a href="https://t.co/gBn4Y9ZJHZ">https://t.co/gBn4Y9ZJHZ</a>	Negatif
Pandemi Covid-19 menyumbang permasalahan baru bagi lingkungan dengan sampah-sampah masker sekali pakai yang mulai m... <a href="https://t.co/cWNnKCLPUC">https://t.co/cWNnKCLPUC</a>	Negatif
Pandemi covid-19 gak cuma berdampak buruk bagi manusia. Tapi secara ga langsung perlahan membunuh para satwa. Ada... <a href="https://t.co/LhYSXAiF8H">https://t.co/LhYSXAiF8H</a>	Negatif
"Semoga penyerahan Alat Pelindung Diri (APD) kepada Satgas Covid-19 Provinsi Kepulauan Bangka Belitung oleh DPD PDI... <a href="https://t.co/RI32nw4i7E">https://t.co/RI32nw4i7E</a>	Positif
Ada banyak vaksin Covid-19 yang telah digunakan di seluruh dunia untuk mengendalikan pandemi. Simak lima vaksin den... <a href="https://t.co/tAnQzSY0wg">https://t.co/tAnQzSY0wg</a>	Positif
"Kita gugah semangat nasionalisme masyarakat di tengah pandemi Covid-19. Salah satunya dengan bergotong royong, sal... <a href="https://t.co/xofGNWV5P5">https://t.co/xofGNWV5P5</a>	Positif
...Peraturan Menteri Kesehatan (Permenkes) Nomor 10 Tahun 2021, tentang pelaksanaan vaksinasi dalam rangka penanggulangan pandemi Covid-19.	Positif
[Sonjo, Wujud Masyarakat Bergerak Atasi Pandemi lewat Gotong Royong] Selama lebih dari satu tahun, Indonesia tela... <a href="https://t.co/VqUsYNNcsB">https://t.co/VqUsYNNcsB</a>	Positif
Alhamdulillah SDH mulai ada penurunan yg positif Covid yg baru	Positif
DANA Serahkan Bantuan untuk Penanganan Pandemi Covid 19 di Bogor <a href="https://t.co/qzPciVsyo">https://t.co/qzPciVsyo</a>	Positif

<i>Tweets</i>	Labeling Manual
Di tengah pandemi, semester pertama 2021 PT Semen Indonesia (persero) Tbk (SIG) malah mencatatkan laba sebesar Rp... <a href="https://t.co/e0aAh8o3H5">https://t.co/e0aAh8o3H5</a>	Positif
Kasus Covid-19 membaik tapi belum aman untuk mencabut PPKM. Pemerintah memperpanjang PPKM Level 4 dari tanggal 3-9 A... <a href="https://t.co/YsC1VPV7w7">https://t.co/YsC1VPV7w7</a>	Positif
Kemenkeu mengklaim penyaluran bansos mampu mencegah 18 juta orang jatuh miskin akibat Pandemi Covid-19 pada 2020.... <a href="https://t.co/FZe1JUjB8Y">https://t.co/FZe1JUjB8Y</a>	Positif

Dataset tersebut dapat digambarkan grafik perbandingan antara kelas positif dengan kelas negatif pada gambar 4.1.



Gambar 4. 1 Grafik Perbandingan Kelas Sentimen

Kelas positif disimbolkan dengan angka 1 dan kelas negative disimbolkan dengan angka 0, yang mana sumbu Y grafik pada gambar 4.1 sebagai banyaknya data dan sumbu X sebagai kelas sentimen.

#### 4.1.2 *Preprocessing*

Pada implementasi *preprocessing* ini peneliti menggunakan 4 tahapan *preprocessing* yaitu *Cleansing*, *Case folding*, *Tokenizing* dan *Stemming*. Tahapan ini diambil dikarenakan tahapan tersebut merupakan langkah

penting dalam analisis sentimen. Data *tweets* dikumpulkan masuk ke proses *preprocessing* untuk mendapatkan data yang bersih sehingga proses pembuatan vektor kata dan klasifikasi sentimen lebih akurat (Nurrohmat & SN, 2019). Berikut ini merupakan implementasi dan penjelasan mengenai *preprocessing*.

#### **4.1.2.1 *Cleansing data***

Data yang telah dilabeli kemudian dimasukkan pada proses *Preprocessing* dan tahapan yang pertama pada *preprocessing* yaitu *Cleansing data*, agar data yang akan dilakukan proses *preprocessing* selanjutnya sudah bersih dari simbol-simbol, emoticon, hyperlink, angka, karakter-karakter yang tidak diperlukan dan kalimat yang duplikat, pada proses ini dilakukan dengan menginisialisasikan text yang berupa judul kolom pada data yang berisi data *tweet*. Proses *Cleansing data* ini banyak menghapus kata yang tidak diperlukan dan membuat data yang akan diproses lebih bersih.

#### **4.1.2.2 *Case folding***

Proses ini dilakukan setelah *Cleansing data*, proses *Case folding* melakukan panggil data dengan menuliskan df yang berisi data *load* yang telah diinputkan diawal, setelah itu memasukkan judul kolom yang berisi *tweet* data pada array ['text']. Proses ini bertujuan untuk merubah data *tweets* yang pada mulanya huruf kapital menjadi huruf kecil.

#### 4.1.2.3 *Tokenizing*

Proses ini proses yang sangat diperlukan dalam teks mining, untuk melakukan memecah kalimat menjadi unit kata. Proses *tokenizing* dilakukan dengan melihat setiap *space* yang ada di *tweet*. Berdasarkan spasi ini kata-kata dapat dipisahkan. Pada proses ini memerlukan *library* *nltk* agar program bisa dijalankan, dan juga memanggil data yang telah diinputkan selanjutnya data yang akan dilakukan *tokenizing* dimasukkan pada array.

#### 4.1.2.4 *Stemming*

*Stemming* ini dilakukan setelah *tokenizing*, proses ini bertujuan untuk menghilangkan imbuhan pada kata yang telah dilakukan proses *tokenizing*, maka dari itu akan dilakukan pemanggilan hasil *tokenizing* agar data dapat dilakukan *stemming*, pada proses ini program python menggunakan *library Porter Stemmer*, dan memasukkan judul baris yang berisi data *tweet* pada array.

#### 4.1.3 Pembagian Data

Pembagian data ini dilakukan dengan perbandingan menggunakan rasio 50% data *training* : 50% data *testing*, 60% data *training* : 40% data *testing*, 70% data *training* : 30% data *testing*, 80% data *training* : 20 data *testing* dan 90% data *training* : 10% data *testing*. Didapatkan hasil pada tabel 4.2.

Tabel 4. 2 Rasio Pembagian Data

Rasio	Akurasi	<i>Recall</i>	Presisi	<i>F1-Score</i>
50% : 50%	0.953	0.83	0.86	0.85
60% : 40%	0.790	0.82	0.90	0.86
70% : 30%	0.950	0.84	0.85	0.85
80% : 20%	0.833	0.90	0.87	0.89
90% : 10%	0.952	0.83	0.88	0.85

Data yang didapat menyatakan bahwa persentase sebesar 0.8 data *training* dan 0.2 data *testing* memiliki akurasi tertinggi. Maka diinisialisasikan dengan *test\_size* = 0.2, dengan keseluruhan data 3894 terbagi menjadi 3115 data *training* dan 779 data *testing*, pada pembagian data ini peneliti menggunakan *library* pada python berupa sklearn. Untuk melakukan validasi dan memeriksa data saat menjalankan program berkali-kali, Mengatur *random\_state* agar nilai akan menjamin urutan nomor yang acak sama hasilnya setiap dijalankan pada program, pada penelitian ini menginisialisasi *random\_state* = 1.

#### 4.1.4 Arsitektur Jaringan

Arsitektur jaringan dengan metode *Long short-term memory* pada penelitian ini terdapat 3 *layer*. *Layer* pertama berupa *Input layer* yang mana berisi node-node yang diisi dengan dataset yang telah dilakukan *preprocessing* dan pembobotan kata atau TF-IDF. Node yang ada pada *input layer* telah memiliki bobot masing-masing dan disimbolkan dengan *w*, kemudian nilai bobot tersebut dilakukan perkalian dengan *inputan* setelah mendapatkan nilai dari perkalian tersebut dilakukannya penambahan dengan

nilai bias, dan Langkah terakhir yaitu melakukan perkalian dengan fungsi *sigmoid*. Sebelum membangun model berikut ini peneliti telah menggunakan *library* Keras Tokenizer API dan data *input* sudah berkode integer. Setelah itu peneliti menginisialisasikan jumlah dimensi pembobotan penelitian tersebut menggunakan node 32 sebagai dimensi pembobotan. Setelah melakukan perhitungan *input layer* dilanjutkan dengan memasukkan nilai pada *hidden layer*. *Hidden layer* menggunakan nilai kelipatan berikut (misalnya 2, 16, 32 dll). Penelitian tersebut dibuatlah perbandingan untuk mendapatkan node *hidden layer* yang optimal.

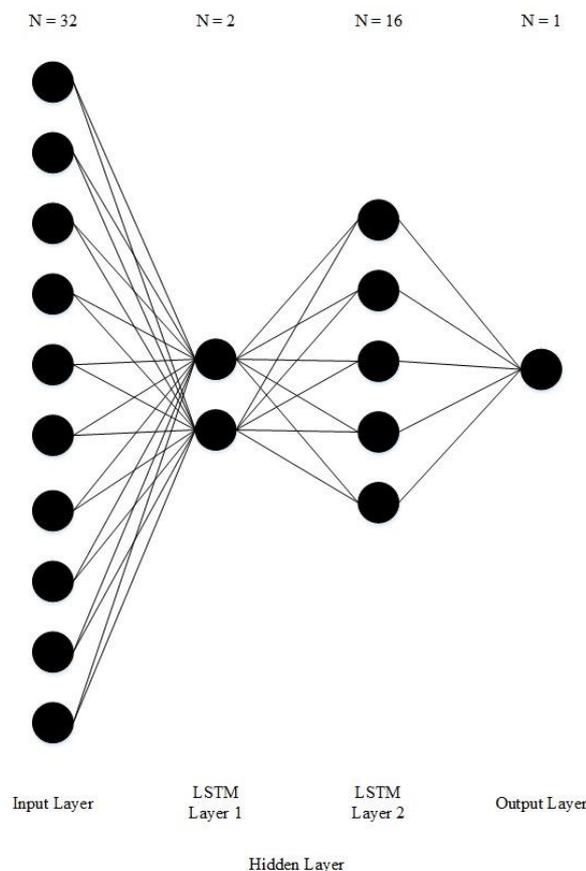
Tabel 4. 3 Perbandingan *Layer*

Node <i>Layer</i>	Hasil <i>Loss Training</i>	Hasil Akurasi <i>Training</i>
32-2-2-1	0.1655 %	0.9417%
32-2-16-1	0.1648 %	0.9973%
32-2-32-1	0.2754 %	0.8870%
32-16-2-1	0.2463 %	0.9123%
32-16-16-1	0.2525 %	0.9014%
32-16-32-1	0.2285%	0.9110%
32-32-2-1	0.2818 %	0.8960%
32-32-16-1	0.2858 %	0.8848%
32-32-32-1	0.2228 %	0.9145%

Tabel 4.3 menyatakan dengan menggunakan *hidden layer* sebesar 2 dan 16 mendapatkan nilai *loss* terendah dan akurasi tertinggi, pada *hidden layer* penelitian ini peneliti menggunakan node 2 untuk *hidden layer* pertama dan node 16 untuk *hidden layer* ke 2 dan *Layer* terakhir merupakan *Output layer* yang mana setelah mendapatkan hasil pada *hidden layer* dilakukan dengan menggunakan perkalian menggunakan fungsi *sigmoid*.

Setelah mendapatkan node pada *input layer*, *hidden layer* dan *output layer* maka didapatkan hasil nilai node 32 untuk *inputan* node 16 untuk *hidden*

layer pertama 32 untuk *hidden layer* kedua dan node 1 digunakan untuk *layer output*. Berikut merupakan gambar Arsitektur hasil pada LSTM.



Gambar 4. 2 Arsitektur LSTM

#### 4.1.5 Implementasi LSTM

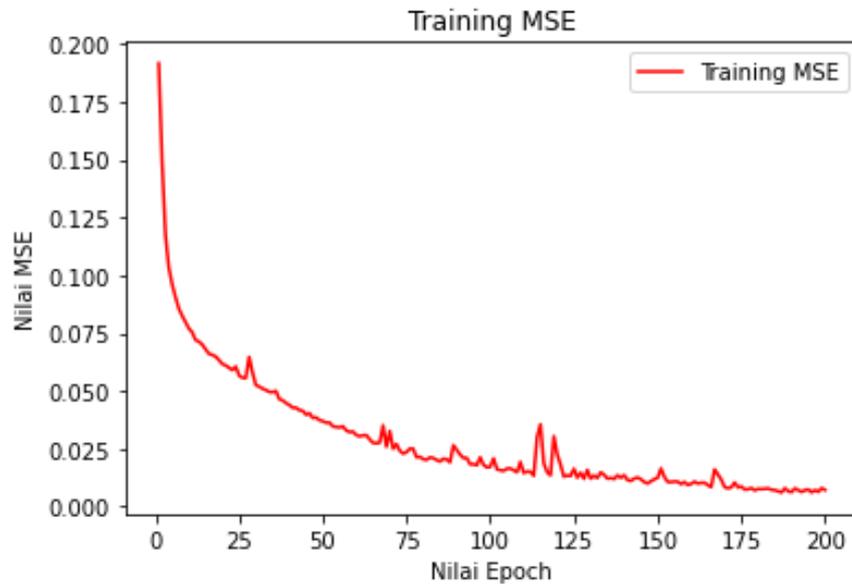
Pada arsitektur jaringan membutuhkan beberapa parameter mendasar diantaranya *epoch*, fungsi aktivasi dan fungsi optimasi. Fungsi aktivasi sendiri berfungsi untuk menghitung nilai *output* berdasarkan *input* dan bobot pada *neuron*. Fungsi optimasi yaitu untuk meminimumkan adanya *error* pada nilai yang telah dihasilkan *neuron* dengan nilai *output* itu sendiri. *Epoch* merupakan bobot yang diubah untuk melakukan beberapa iterasi agar menjadi konvergen. Untuk hasil klasifikasi yang optimal maka dilakukan pergantian-pergantian nilai ketiga parameter itu (Wibawa, 2016).

Pada proses *training* ini penerapan *epoch* akan mempengaruhi hasil perbedaan-perbedaan pada hasil klasifikasi, pada nilai *epoch* yang kecil maka penyesuaian parameternya belum selesai dikarenakan terbatas pada waktu iterasi, hal ini mengakibatkan nilai akurasi yang relative rendah. Maka disarankan agar melakukan peningkatan jumlah *epoch* agar kinerja klasifikasi mengalami kenaikan bertahap (Listyarini & Anggoro, 2021).

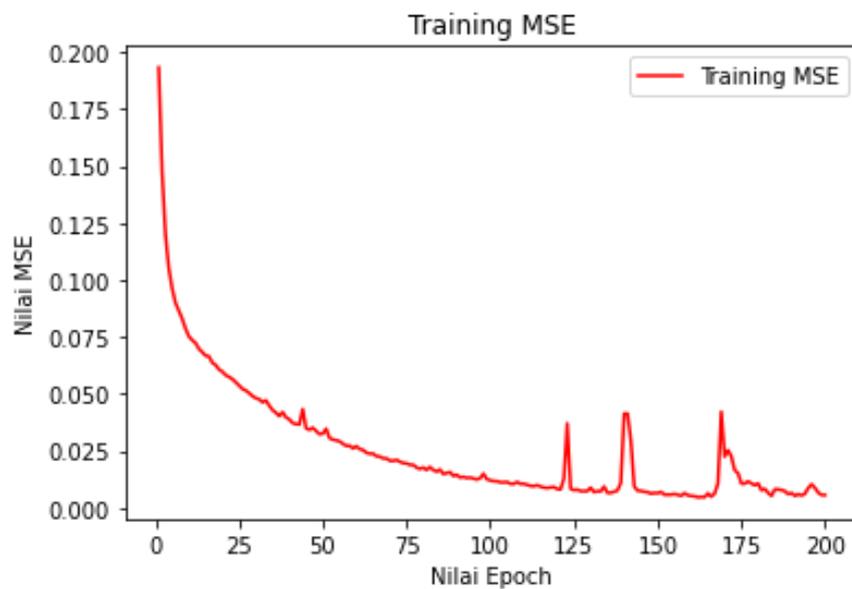
Penelitian tersebut menggunakan nilai *epoch* sebesar 50 agar mendapatkan nilai akurasi yang tinggi dan juga nilai *loss* yang rendah, nilai *epoch* juga dapat mengetahui nilai *Mean Squared Error* (MSE) yang berarti *metrics* pengukuran nilai *error* yang cukup sederhana dan banyak digunakan dalam melakukan evaluasi terhadap hasil prediksi. Nilai MSE memiliki jaminan tidak akan menghasilkan nilai yang negative, karna dalam prosesnya menggunakan kuadrat dari pengurangan data *actual* dengan data yang diprediksi. Melakukan perhitungan menggunakan *epoch* dengan cara menginisialisasikan *epoch* = 50, *batch size* = 10. Fungsi dari *batch size* sendiri untuk jumlah data *training* yang disebarkan menuju LSTM. Pada penelitian ini memiliki 3894 dan *batch size* penelitian ini merupakan 10 maka algoritma yang akan digunakan yaitu 10 data sampe pertama dari 3894 data yang dimiliki, lalu disebarkan atau dilakukan *training* oleh metode LSTM. Penelitian ini menggunakan *optimizer* adam yang merupakan sebuah algoritma pengoptimalan yang digunakan untuk perbaruai *weight network* secara iteratif yang berdasarkan data *training*.

Penelitian ini menggunakan perbandingan *Learning\_rate* sebesar 0.01, 0.02, 0.03, 0.04, 0.05, 0.06, 0.07, 0.08 dan 0.09 untuk mendapatkan nilai

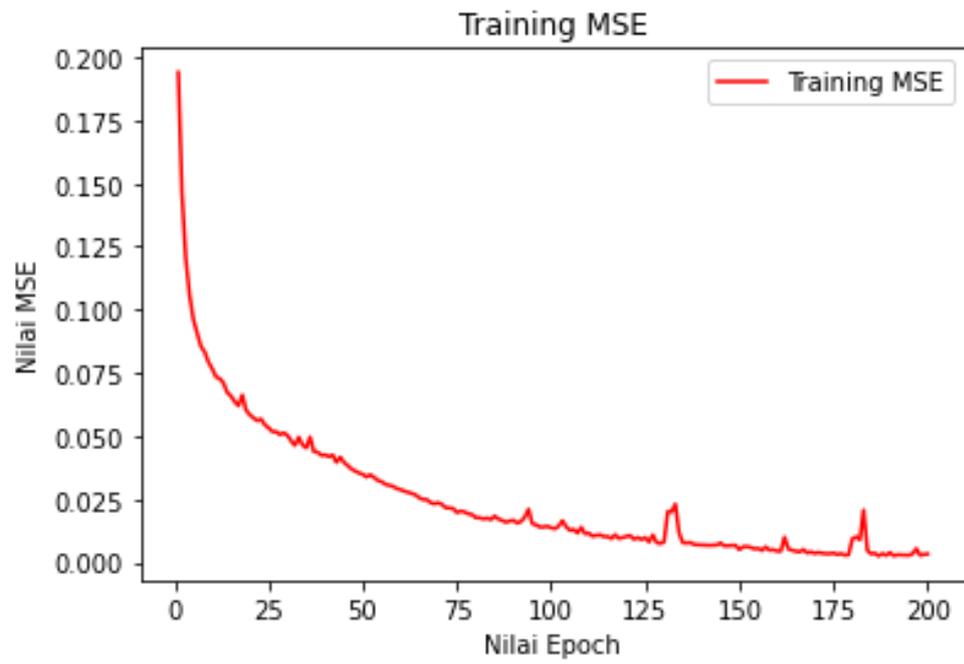
terkecil. dengan menggunakan fungsi *epoch*, optimizer Adam, dan meninisialisasikan `metrics = ['mse']`. berikut adalah perbandingan grafik dan nilai MSE yang telah lakukan dengan menggunakan nilai epoch sebesar 200, penggunaan epoch sebesar 200 ini dikarenakan grafik yang didapatkan telah konvergen dan waktu menjalankan program tidak terlalu lama.



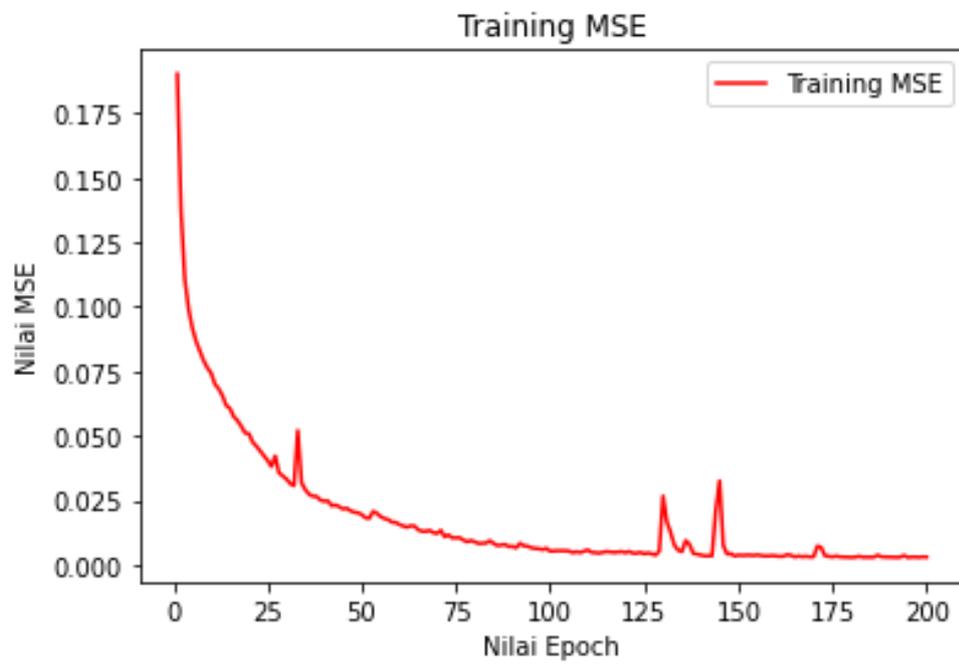
Gambar 4. 3 Grafik MSE pada Learning\_rate 0.01



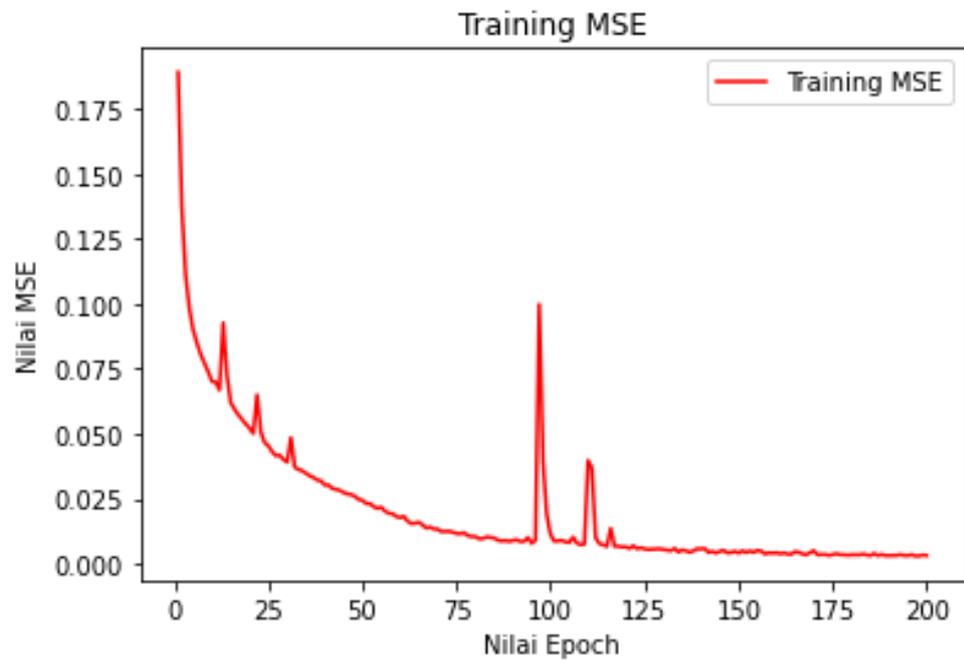
Gambar 4.4 Grafik MSE pada Learning\_rate 0.02



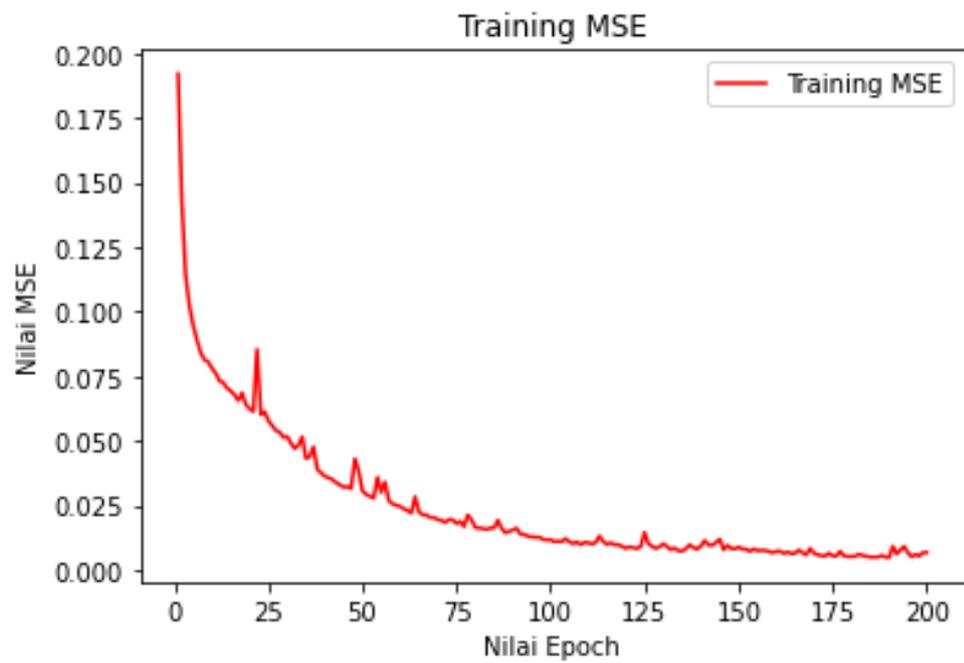
Gambar 4.5 Grafik MSE pada *Learning\_rate* 0.03



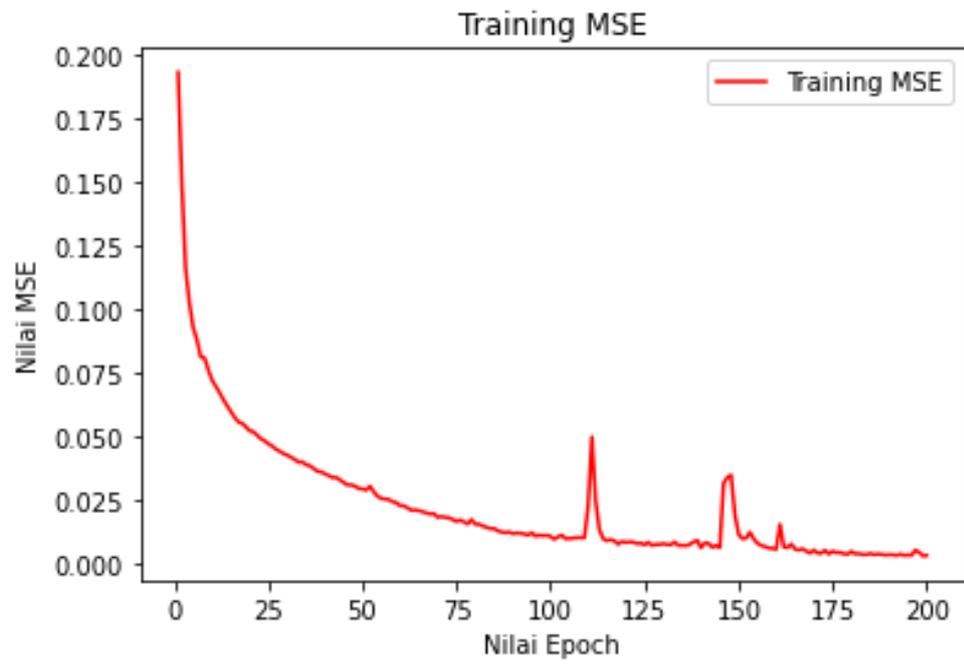
Gambar 4.6 Grafik MSE pada *Learning\_rate* 0.04



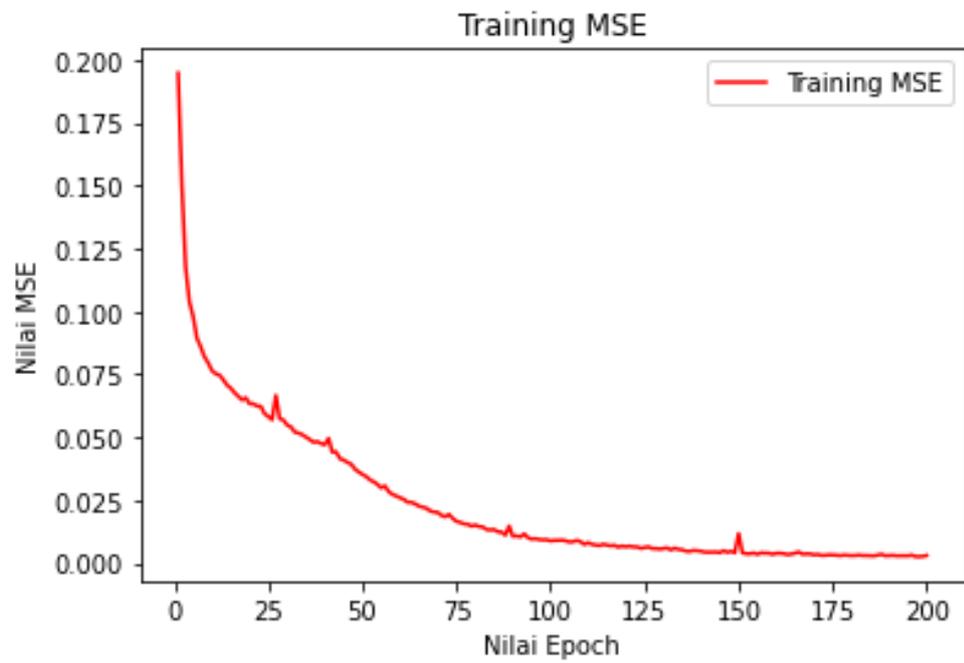
Gambar 4.7 Grafik MSE pada *Learning\_rate* 0.05



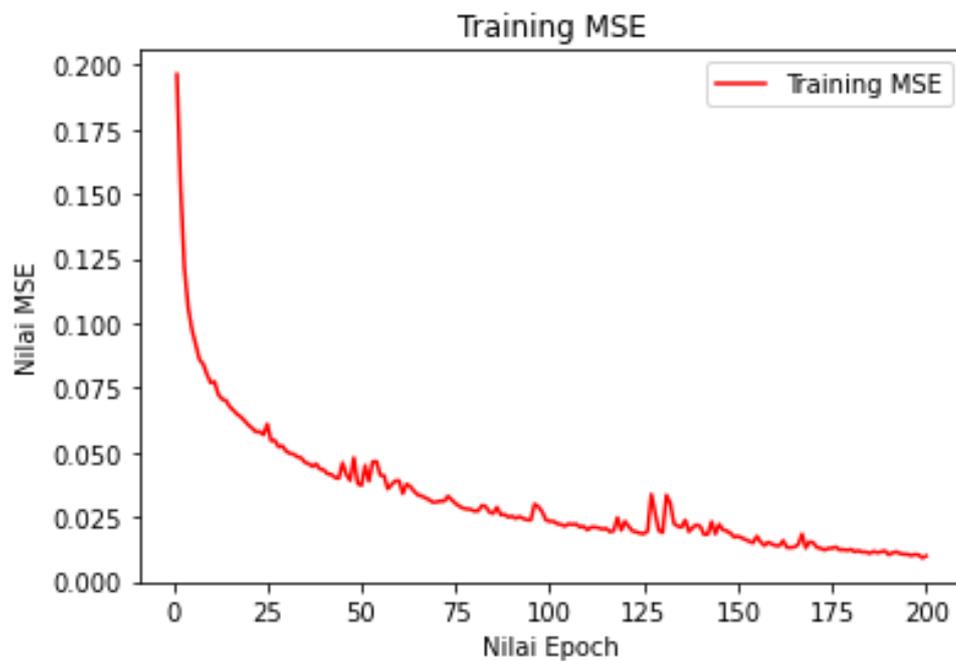
Gambar 4.8 Grafik MSE pada *Learning\_rate* 0.06



Gambar 4. 9 Grafik MSE pada Learning\_rate 0.07



Gambar 4. 10 Grafik MSE pada Learning\_rate 0.08



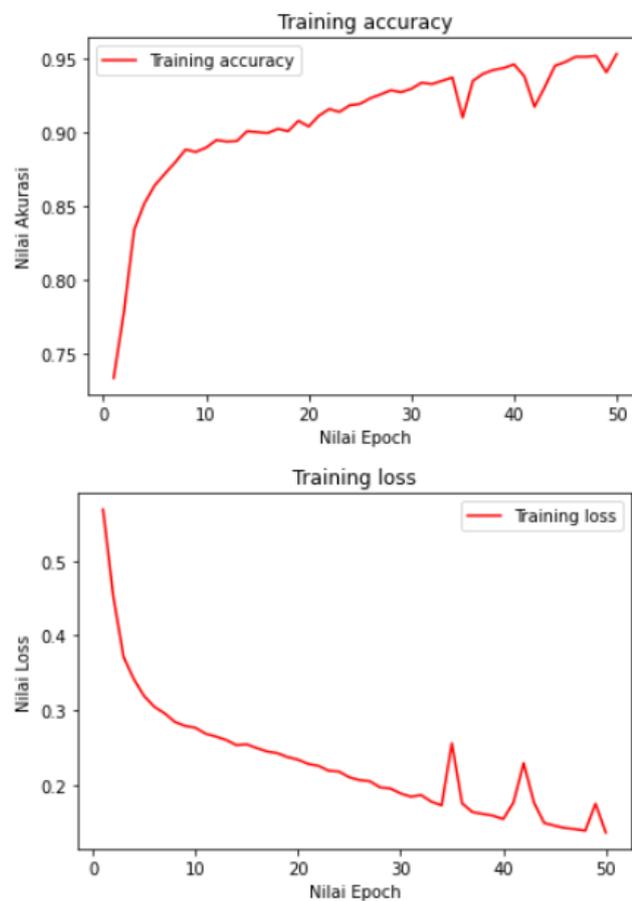
Gambar 4. 11 Grafik MSE pada Learning\_rate 0.09

Gambar percobaan untuk mendapatkan nilai MSE menggunakan nilai *Learning Rate* menghasilkan grafik dengan nilai sumbu Y merupakan nilai MSE dan sumbu X merupakan nilai *epoch*. Maka didapatkan perbandingan nilai MSE terbaik berikut ini

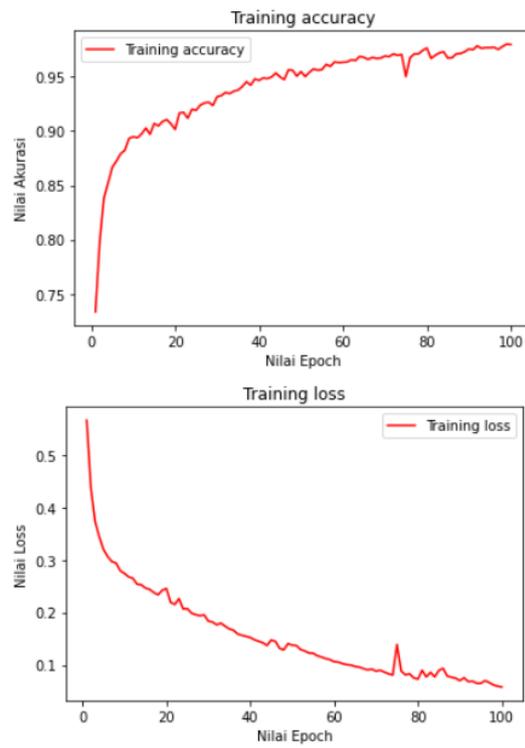
Tabel 4.4 Perbandingan MSE berdasarkan *learning rate*

<i>Learning rate</i>	Nilai MSE
0.01	0.0071
0.02	0.0056
0.03	0.0033
0.04	0.0031
0.05	0.0032
0.06	0.0068
0.07	0.0034
0.08	0.0032
0.09	0.0099

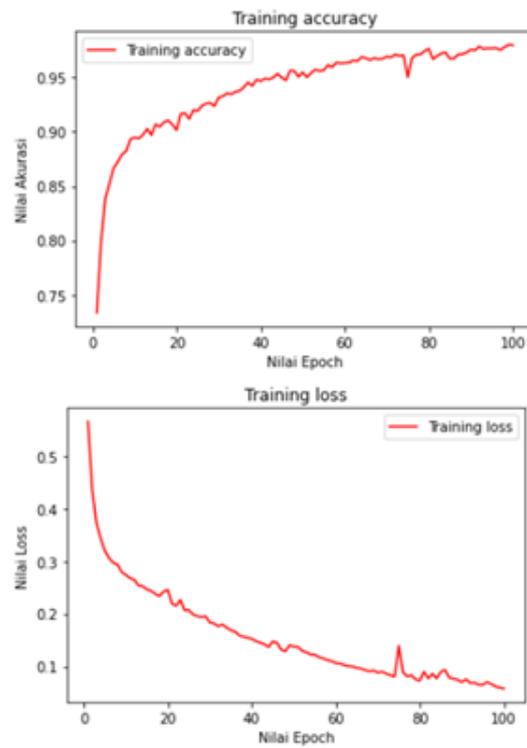
Tabel 4.4 menyatakan bahwa menggunakan *Learning rate* sebesar 0.04 dapat menghasilkan MSE optimum dengan nilai sebesar 0.0031. Setelah mendapatkan nilai *Learning rate* yang optimum selanjutnya akan melakukan uji coba menggunakan *epoch*. Perbandingan yang akan dilakukan pada penelitian ini yaitu menggunakan *epoch* sebesar 50, 100, 150, 200, 250. Berikut merupakan grafik Nilai akurasi pada data *training* dan nilai *loss* pada data *training*.



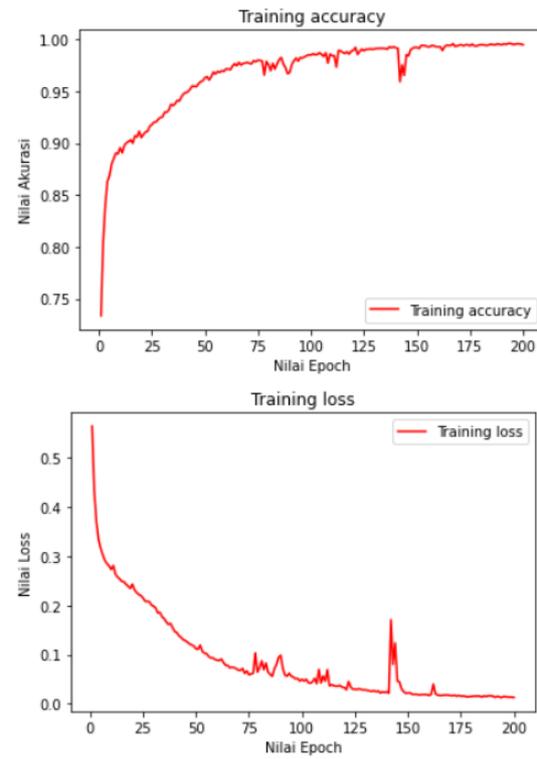
Gambar 4. 12 Grafik Akurasi dan *Loss* data *Training* pada *Epoch* 50



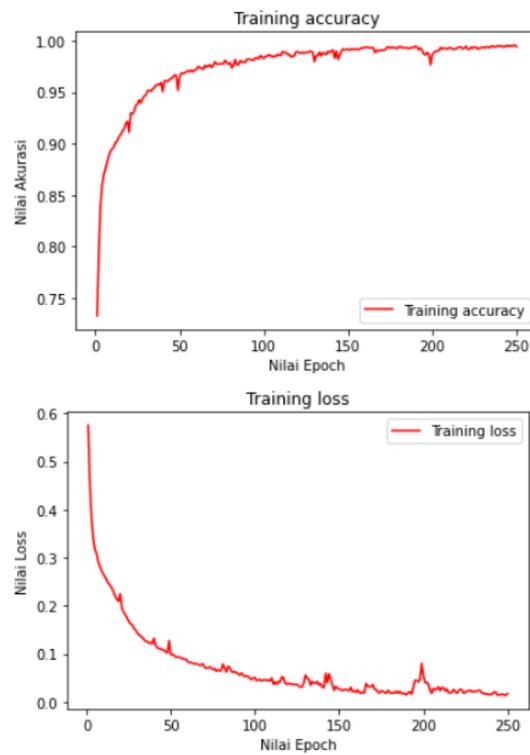
Gambar 4. 13 Grafik Akurasi dan *Loss* data *Training* pada *Epoch* 100



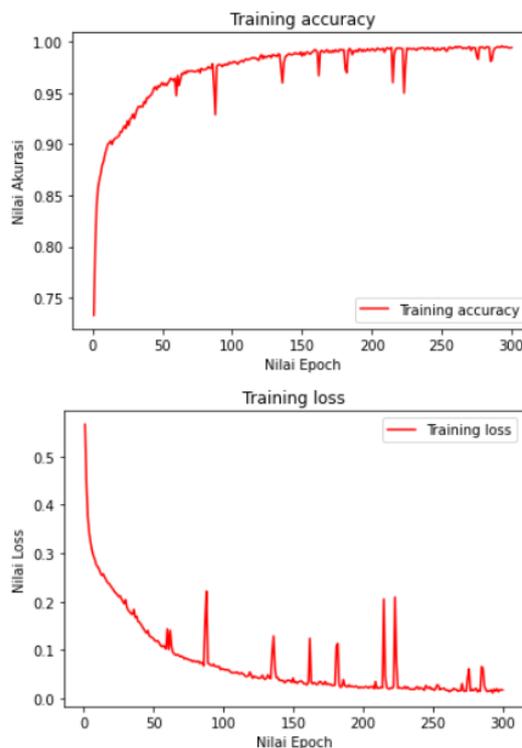
Gambar 4. 14 Grafik Akurasi dan *Loss* data *Training* pada *Epoch* 150



Gambar 4. 15 Grafik Akurasi dan *Loss* data *Training* pada *Epoch* 200



Gambar 4. 16 Grafik Akurasi dan *Loss* data *Training* pada *Epoch* 250



Gambar 4. 17 Grafik Akurasi dan Loss data Training pada Epoch 300

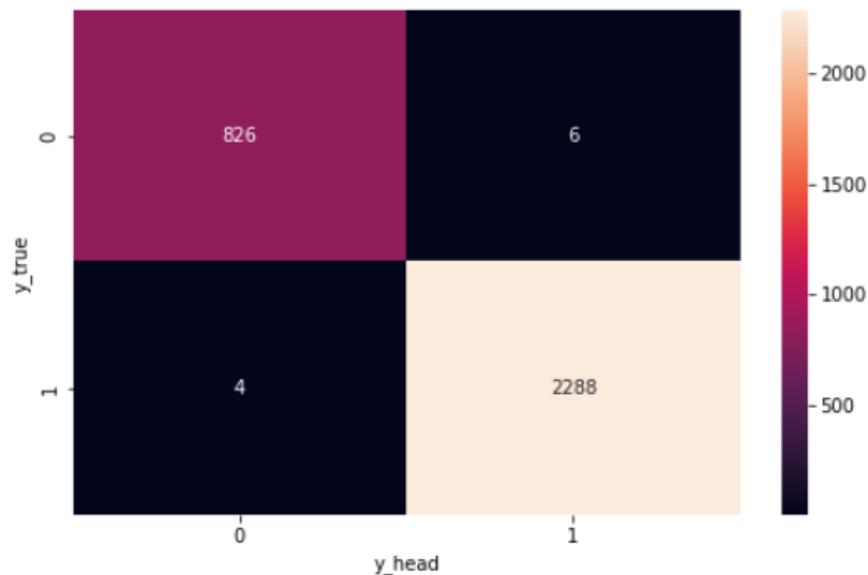
Gambar 4.12 hingga 4.17 merupakan grafik untuk penentuan nilai epoch terbaik dengan sumbu Y ialah nilai akurasi atau nilai *loss* dan sumbu X merupakan nilai *epoch* yang digunakan. Tabel 4.5 merupakan tabel perbandingan nilai akurasi pada data *training* dan nilai *loss* pada data *training* menggunakan nilai *epoch* 200 mendapatkan nilai akurasi optimum, maka dari itu penelitian tersebut menggunakan nilai *epoch* 200.

Tabel 4. 5 Menentukan Akurasi Optimum dengan *Epoch*

Nilai <i>Epoch</i>	Akurasi <i>Training</i>	<i>Loss Training</i>
50	0.9533	0.1360
100	0.9795	0.0579
150	0.9926	0.0204
200	0.9968	0.0128
250	0.9942	0.0172
300	0.9946	0.0177

Setelah mendapatkan akurasi dan *loss* pada *training* data, selanjutnya dilakukan penentuan *confusion matrix* pada data *training*, agar mendapatkan nilai *Recall*, presisi dan *F1-Score*. Berikut merupakan gambar *confusion matrix training* dan nilai akurasi, *Recall*, presisi dan *F1-Score*.

```
Accuracy of the model : 0.9967989756722151
recall_score: 0.997384481255449
precision-score: 0.9982547993019197
F1-score: 0.9978194505015264
Confusion matrix:
```



Gambar 4. 18 Confusion matrix data *training*

Tabel 4. 6 Penjabaran *Confusion matrix Training*

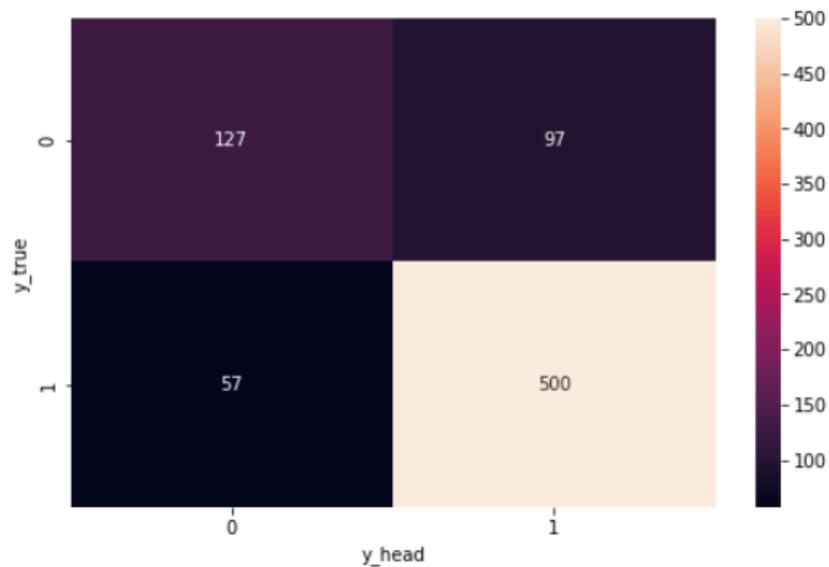
Kelas Sebenarnya	Kelas Prediksi	
	Negatif	Positif
Negatif	TN = <i>True negative</i> merupakan nilai sebenarnya dan nilai prediksi memiliki nilai negative, pada penelitian ini nilai TN sebesar 824	FP = <i>False positive</i> merupakan nilai ketika kelas sebenarnya bernilai negative dan kelas prediksi bernilai positif, pada penelitian ini FP bernilai 8
Positif	FN = <i>False positive</i> merupakan nilai ketika kelas prediksi bernilai negative dan kelas sebenarnya bernilai positif, FP penelitian ini bernilai 7	TP = <i>True Positive</i> merupakan nilai sebenarnya dan nilai prediksi memiliki nilai positif, pada penelitian ini nilai TP sebesar 2285

Maka dari itu didapatkan hasil akurasi sebesar 0.9968, nilai *Recall* sebesar 0.997, nilai presisi sebesar 0.997 dan nilai *F1-Score* sebesar 0.997. Nilai tersebut dapat dinyatakan optimum dikarenakan pada proses *training* menggunakan optimizer adam yang menyebabkan nilai akurasi yang optimal.dengan menggunakan *epoch* 200.

#### **4.2 Hasil Implementasi**

Setelah melalui proses implementasi tersebut selanjutnya dilakukan pemaparan tentang hasil uji coba yang telah dilakukan peneliti. Prediksi sistem akan dibandingkan dengan keadaan saat ini. Jika keluaran sistem memprediksi secara akurat untuk satu kelas, TN akan diset ke 1, yang menunjukkan bahwa ada satu kelas yang bukan kebenaran dasar tetapi diprediksi benar oleh sistem. Kolom TP akan menjadi 0 jika sistem menghasilkan prediksi yang salah karena tidak ada prediksi yang sesuai dengan *groundtruth*. FP memiliki nilai 1 karena ada satu kelas yang sistem harus prediksi dengan benar sebagai kelas pada pesan, sedangkan FN memiliki nilai 1 karena ada satu kelas yang sistem harus prediksi salah (Muslikah, 2021). Hasil dari penelitian ini telah tertulis pada Gambar 4.11 *confusion matrix* pada data *testing* seperti pada Gambar 4.11.

Accuracy of the model : 0.8028169014084507  
 recall\_score: 0.8375209380234506  
 precision-score: 0.8976660682226212  
 F1-score: 0.8665511265164644  
 Confusion matrix:



Gambar 4.17 Confusion matrix Testing

Tabel 4. 7 Confusion Matrix data testing

Kelas Sebenarnya	Kelas Prediksi	
	Negatif	Positif
Negatif	TN = <i>True negative</i> pada penelitian tersebut nilai TN sebesar 128	FP = <i>False positive</i> pada penelitian tersebut FP sebesar 95
Positif	FN = <i>False positive</i> FP penelitian tersebut bernilai 57	TP = <i>True Positive</i> TP penelitian tersebut sebesar 500

Dari Gambar 4.11 dapat dilakukan perhitungan Akurasi, Presisi, *Recall* dan *F1-Score*. Berikut merupakan hasil evaluasi dari klasifikasi kalimat perbincangan masyarakat terhadap pandemi Covid-19 dengan metode *Long short-term memory* sebagai berikut.

$$Akurasi (\%) = \frac{128 + 500}{128 + 500 + 95 + 57} \times 100\% = 80\% \quad 4.1$$

$$Precision (\%) = \frac{500}{500 + 57} \times 100\% = 90\% \quad 4.2$$

$$Recall (\%) = \frac{500}{500 + 95} \times 100\% = 84\% \quad 4.3$$

$$F1 - Score (\%) = \frac{2 \times 0.87 \times 0.906}{0.87 + 0.906} \times 100\% = 86\% \quad 4.4$$

Setelah mendapatkan nilai Akurasi, Presisi, *Recall* dan *F1-Score*, langkah selanjutnya merupakan implementasi klasifikasi pada dataset yang telah diberi label oleh ahli dan dibandingkan dengan klasifikasi menggunakan sistem pada metode LSTM dan menghasilkan nilai berikut ini.

Tabel 4.8 Hasil Klasifikasi

<i>Tweets</i>	Labeling Manual	Klasifikasi Sistem
"Ancaman Kelaparan di Tengah Pandemi" Pandemi Covid-19 di Indonesia	Negatif	Negatif
"Kadang juga ada orang yang tangkap atau kriminalisasi dan juga dianggap Positif Covid19. Padahal tidak ada bukti... <a href="https://t.co/OGRD3SmFL5">https://t.co/OGRD3SmFL5</a>	Negatif	Negatif
Masa pandemi ini setiap hari timeline saya dan mungkin juga yang lainnya, selalu dihiasi oleh berita duka, kecewa, sedih	Negatif	Negatif
Aktivitas fisik menjadi terbatas selama pandemi Covid-19. Alhasil, anak lebih sering menghabiskan lebih banyak waktu depan hp	Negatif	Negatif
Anak-Anak di Surabaya Butuh Perlindungan saat Pandemi Covid-19	Negatif	Negatif
Jumlah limbah medis melonjak berkali-kali lipat selama pandemi Covid-19. Lonjakan belum dimbangi ketersediaan fasilitas	Negatif	Negatif
Pandemi Covid-19 bahkan memaksa sejumlah Satuan Pendidikan di bawah Kemenag, menutup sekolahnya.	Negatif	Negatif
Pandemi Covid 19 Melonjak, Indikator Perekonomian Kembali Memburuk	Negatif	Negatif
Pandemi Covid-19 menyumbang permasalahan baru bagi lingkungan dengan sampah-sampah masker sekali pakai yang mulai	Negatif	Negatif
Pandemi covid-19 gak cuma berdampak buruk bagi manusia. Tapi secara ga langsung perlahan membunuh para satwa.	Negatif	Negatif
"Semoga penyerahan Alat Pelindung Diri (APD) kepada Satgas Covid-19 Provinsi Kepulauan Bangka Belitung oleh DPD PDI	Positif	Positif

<i>Tweets</i>	Labeling Manual	Klasifikasi Sistem
Ada banyak vaksin Covid-19 yang telah digunakan di seluruh dunia untuk mengendalikan pandemi. Simak lima vaksin	Positif	Negatif
Kita gugah semangat nasionalisme masyarakat di tengah pandemi Covid-19. Salah satunya dengan bergotong royong	Positif	Positif
Peraturan Menteri Kesehatan (Permenkes) Nomor 10 Tahun 2021, tentang pelaksanaan vaksinasi dalam rangka penanggulangan pandemi Covid-19.	Positif	Positif
[Sonjo, Wujud Masyarakat Bergerak Atasi Pandemi lewat Gotong Royong] Selama lebih dari satu tahun, Indonesia telah melakukan gotong royong	Positif	Positif
Alhamdulillah SDH mulai ada penurunan yg positif Covid yg baru	Positif	Positif
DANA Serahkan Bantuan untuk Penanganan Pandemi Covid 19 di Bogor	Positif	Negatif
Di tengah pandemi, semester pertama 2021 PT Semen Indonesia (persero) Tbk (SIG) malah mencatatkan laba	Positif	Positif
Kasus Covid-19 membaik tapi belum aman untuk mencabut PPKM. Pemerintah memperpanjang PPKM Level 4 dari tanggal 3-9	Positif	Negatif
Kemenkeu mengklaim penyaluran bansos mampu mencegah 18 juta orang jatuh miskin akibat Pandemi Covid-19 pada 2020	Positif	Positif

### 4.3 Pembahasan

Berdasarkan hasil implementasi yang dilakukan oleh peneliti dengan menggunakan metode *Long short-term memory* pada kalimat perbincangan masyarakat seputar pandemi covid-19 dengan rasio pembagian data sebesar 80% : 20%, data *training* sebanyak 3124 dan data *testing* sebanyak 951. Dataset yang digunakan terbagi menjadi 2 kelas yaitu kelas positif dan kelas negatif yang perbandingannya tidak imbang yakni pada kelas positif terdapat sebanyak 2847 data yang memiliki label positif dan sebanyak data 1057 yang berlabel negatif pada hal ini sentimen penelitian ini dapat dinyatakan kurang seimbang.

Setelah menentukan pembagian data dilakukan penentuan node *layer* yang akan digunakan pada metode LSTM yaitu membutuhkan *input layer*, *hidden layer* dan *output layer*, perbandingan nilai *hidden layer* yang dilakukan pada penelitian ini terdapat pada tabel 4.3 yang mana pada percobaan ini dilakukan dengan

memulai nilai yang terkecil pada *hidden layer* pertama dan ke-2, setelah itu menambahkan nilai yang lebih besar pada *hidden layer* ke-2 setelah nilai *hidden layer* sudah dilakukan pada nilai tertinggi selanjutnya *layer* pertama dilakukan penambahan angka. Setelah mengetahui node perbandingannya didapatkan hasil akurasi terbaik pada perbandingan kedua yakni 32 sebagai nilai *input*, 2 sebagai *hidden layer* pertama, 16 sebagai *hidden layer* kedua dan 1 sebagai nilai *output*, menghasilkan nilai akurasi sebesar 0.997 dan nilai *loss* sebesar 0.164.

Setelah itu dilakukan uji coba pada *Epoch* untuk menentukan *epoch* yang memiliki nilai *optimum*. Maka didapatkan data pada tabel 4.5. menyatakan bahwa banyaknya *epoch* yang digunakan memiliki pengaruh yang besar terhadap hasil akurasi. Apabila *epoch* yang digunakan terlalu rendah maka nilai akurasi yang akan dihasilkan terbilang sangat rendah dan mengakibatkan nilai *loss* atau *error* yang dihasilkan akan tinggi, namun jika jumlah *epoch* yang telah digunakan terlalu banyak maka hal itu akan mengakibatkan *overfitting* atau perbedaan nilai akurasi pada *training* dan nilai akurasi pada *testing* sangat jauh. Pada percobaan dengan menggunakan nilai epoch 200, didapatkan nilai akurasi tertinggi dan nilai *loss* yang terendah, sehingga besarnya nilai optimum untuk epoch adalah 200. Berdasarkan tabel 4.5 juga dapat diketahui bahwa nilai epoch yang lebih besar dari 200 tidak terlalu berpengaruh atau bahkan mendapatkan nilai akurasi dan nilai *loss* yang lebih buruk. Sehingga nilai epoch sebesar 200 dapat dikatakan sebagai nilai epoch yang teroptimum dalam menghasilkan akurasi terbaik pada data training. Hasil implementasi penelitian ini mendapatkan nilai akurasi yang relatif tinggi dan mendapatkan hasil *error* yang rendah pada *training* nya. Pada *epoch* ke 200 nilai *loss training* pada program sebesar 0.0128 dan nilai akurasinya sebesar 0.997.

Berdasarkan perhitungan pada data *training* menggunakan metode *Long short-term memory* mendapatkan nilai  $TN = 824$ ,  $FP = 8$ ,  $FN = 7$  dan  $TP = 2285$  mendapatkan nilai akurasi sebesar 0.997, nilai *Recall* sebesar 0.997, nilai presisi sebesar 0.998 dan nilai *F1-Score* sebesar 0.998. Data *testing* terdapat nilai *Confusion matrix*  $TN = 127$ , nilai  $FP = 97$ , nilai  $FN = 57$  dan nilai  $TP = 500$ , maka dihasilkan tingkat kedekatan dalam pengukuran nilai kuantitas pada nilai aktual (Akurasi) sebesar 80%, nilai pengulangan dalam pengukuran terhadap kondisi yang tidak berubah-ubah (Presisi) dihasilkan sebesar 90%, rasio kelas prediksi positif dan dibandingkan dengan seluruh data positif (*Recall*) yang dihasilkan sebesar 84%, dan nilai rata-rata *recall* dan presisi (*F1-Score*) yang dihasilkan sebesar 87%. Penelitian (Gorunescu, 2011) menyatakan bahwa nilai klasifikasi penelitian tersebut termasuk kedalam golongan klasifikasi yang baik, berikut merupakan tabel penentuan golongan klasifikasi.

Gambar 4. 19 Golongan Klasifikasi

Nilai Akurasi	Golongan Klasifikasi
0.90 – 1.00	Klasifikasi Sangat Baik
0.80 – 0.90	Klasifikasi Baik
0.70 – 0.80	Klasifikasi Cukup
0.60 – 0.70	Klasifikasi Kurang Baik
0.50 – 0.60	Klasifikasi Gagal

Hasil klasifikasi yang dilakukan pada penelitian menggunakan 20 data pada dataset yang telah diberi label oleh ahli bahasa dan diambil secara random menunjukkan hasil 17 kali klasifikasi benar dan 3 kali klasifikasi salah, hal tersebut diakibatkan nilai akurasi yang digunakan tidak sempurna, maka dari itu terdapat kesalahan yang dihasilkan pada proses klasifikasi. Pada hasil yang didapatkan

penelitian ini masih terdapat klasifikasi yang salah, hal tersebut dikarenakan data yang dihasilkan pada *tweet* bukanlah data yang sempurna, masih banyak kata yang terdapat singkatan, kata gaul dan data pada twitter tidak bisa diambil secara utuh atau terdapat pemotongan kalimat yang mana biasa didapatkan pada kalimat yang terlalu panjang potongan kalimatnya berubah menjadi link, misalnya seperti kalimat “Kemendikbud Ristek akan menyalurkan Rp 745 miliar untuk lanjutan bantuan UKT bagi mahasiswa yang terdampak pandemi... <https://t.co/3jbOXKJ72x>” link tersebut akan menghilang apabila dilakukan proses *Cleansing data* pada *Preprocessing*, selain itu kurangnya tahapan *Preprocessing* dalam mengubah atau menghilangkan kata yang tidak tepat atau kata yang tidak baku misalnya seperti huruf yang diulang “capekkkk” “ngeri”, kata yang disingkat seperti “udh” , “rmhkan”, permasalahan lainnya yaitu tidak adanya spasi dalam suatu kalimat sehingga kalimat tersebut menjadi kata “menyesalkanPenangananPandemi”. Maka dari itu system tidak berjalan dengan lancar pada saat membaca kalimat tanggapan masyarakat. Pada penelitian ini peneliti menggunakan TF-IDF dalam melakukan pembobotan kata, pembobotan tersebut biasa digunakan untuk model *Machine Learning*, sedangkan penelitian ini menggunakan model *Deep learning* yang mana biasanya menggunakan metode pembobotan kata Word2Vec. Hal tersebut dapat mempengaruhi hasil pembobotan kata dan mempengaruhi proses klasifikasi menggunakan metode LSTM.

Penelitian ini dapat ditujukan oleh masyarakat, khususnya kepada kaum muslimin yang melakukan interaksi pada media sosial, yang mana media sosial ini berupa media yang bebas dalam menyampaikan pendapat ataupun suatu kabar, hal itu mempermudah dalam interaksi sosial dengan teman-teman, saudara, dan bahkan

orang yang tidak dikenal maka dari itu kita sebagai umat manusia harus bisa menjaga tutur kata kita dan menghindari omongan yang tidak benar agar kita tidak mendapatkan adzab yang sangat besar yang diberikan oleh Allah SWT, larangan agar tidak menyebarkan berita yang belum diketahui kebenaran informasinya telah disampaikan pada ayat Al-qur'an surat Al-Isra' ayat 36 yaitu :

مَسْئُولٌ عَنْهُ كَانَ أُولَئِكَ كُلاًّ وَالْفُؤَادَ وَالْبَصَرَ السَّمْعَ إِنَّ عِلْمَ بِهِ لَكَ لَيْسَ مَا تَقْفُ وَلَا

*“dan janganlah kamu mengikuti apa yang kamu tidak mempunyai pengetahuan tentangnya. Sesungguhnya pendengaran, penglihatan dan hati, semuanya itu akan diminta pertanggung jawabnya”.* (Al-Isra' : 36)

Pada tafsir At-habrani mengatakan bahwa makna dari *La Taqfu* pada ayat tersebut para ulama ahli bidang ta'wil mengartikan: 'janganlah kamu menyampaikan sesuatu yang tak kamu ketahui'. Sebagian ulama' lainnya menyatakan bahwa 'janganlah engkau menuduh orang lain atas sesuatu yang sebenarnya kamu tidak ketahui'. Kesimpulan pada beberapa pendapat ulama' menyatakan bahwa Allah SWT melarang kita umat muslim menyatakan sesuatu tanpa ilmu dan pengetahuan, bahkan Allah melarang pula menyatakan sesuatu yang berdasarkan dugaan (zan) yang bersumber pada ilusi sendiri atau sangkaan.

Maka dari itu kita harus menjaga lisan kita baik didunia nyata maupun media sosial, sebagai orang yang beriman kita juga lebih teliti dalam menemukan kebenaran suatu berita, seperti yang ada pada surat al-Hujurat ayat 6 yaitu :

مَا عَلَىٰ فَتُصْبِحُوا بِجَهَالَةٍ قَوْمًا نُصِيبُهُمْ أَنْ فَتَبَيَّنُوا بِنَبِيٍّ فَاسِقٍ جَاءَكُمْ إِنْ آمَنُوا الَّذِينَ يَأْتِيهَا  
نَدِيمِينَ فَعَلْتُمْ

*“Wahai orang-orang yang beriman! Jika seseorang yang fasik datang kepadamu membawa suatu berita, maka telitilah kebenarannya, agar kamu tidak mencelakakan suatu kaum karena kebodohan (kecerobohan), yang akhirnya kamu menyesali perbuatanmu itu.”* (Al-Hujurat : 6)

Tafsir Jalalain mengatakan bahwa : (Hai orang yag mukmin! Apabila kedatangan pada kalian seorang yang fasik dengan suatu berita yang dibawa) (maka telitilah terlebih dahulu dari kalian) kebenarannya pada beritanya tersebut, apakah ia benar atau melakukan dusta. Menurut sebuah qiraat dibaca Fatatsabbatuu yang berasal dari lafal Ats-Tsabaat, memiliki arti telitilah terlebih dahulu kebenaran (agar kalian tidak menumpukkan musibah kepada suatu kaum) menjadi Maf'ul dari lafal Fatabayyanu, yakni ditakutkan hal tersebut akan menimpa musibah pada suatu kaum (tanpa diketahui keadaannya) menjadi suatu hal atau kata keterangan keadaan dari Fa'il, yakni tanpa sepengetahuannya (yang menyebabkan kalian) membuat kalian (atas perbuatan kalian itu) ialah berbuat kesalahan terhadap kaum tersebut (menyesal) selanjutnya Rasulullah saw telah mengutus Khalid kepada mereka sesudah mereka kembali ke negerinya. Ternyata Khalid tiada menjumpai mereka melainkan hanya ketaatan dan kebaikan belaka, lalu ia menceritakan hal tersebut kepada Nabi saw (*Surat Al-Hujurat Ayat 6 / Tafsirq.Com, n.d.*).

Apabila kita tidak dapat memberikan kalimat positif maka lebih baik kita diam, telah dijelaskan pada sebuah hadis :

عَنْ أَبِي هُرَيْرَةَ رَضِيَ اللَّهُ عَنْهُ عَنِ رَسُولِ اللَّهِ صَلَّى اللَّهُ عَلَيْهِ وَسَلَّمَ قَالَ: «مَنْ كَانَ يُؤْمِنُ بِاللَّهِ وَالْيَوْمِ

الْآخِرِ، فَلْيُفْلِحْ خَيْرًا أَوْ لِيَصْنَمْتْ، وَمَنْ كَانَ يُؤْمِنُ بِاللَّهِ وَالْيَوْمِ الْآخِرِ، فَلْيُكْرِمْ جَارَهُ

*Dari Abu Hurairah RA bahwa Rasulullah SAW bersabda, "Barang siapa beriman kepada Allah dan Hari Akhir, hendaklah ia berkata baik atau diam."(HR. Bukhari dan Muslim).*

## BAB V

### SIMPULAN DAN SARAN

#### 5.1 Simpulan

Berdasarkan Implementasi dan Penjelasan pada bab 4 penelitian ini menggunakan dataset sebanyak 3904 yang terbagi menjadi 2 bagian yaitu 1057 sentimen negatif dan 2847 sentimen positif, dengan perbandingan 20% data *testing* : 80% data *training*. Pada data tersebut menghasilkan nilai MSE terendah pada *learning rate* 0.04 sebesar 0.0031, dan mendapatkan nilai *epoch* 200 sebagai *epoch* optimum dikarenakan mendapatkan nilai akurasi *training* yang tinggi sebesar 0.997 dan nilai *loss* sebesar 0.0128, setelah mendapatkan hasil akurasi pada data *training* didapatkan hasil Akurasi pada data *testing* sebesar 80% , nilai *Recall* yang dihasilkan 84%, nilai Presisi sebesar 90% dan nilai *F1-Score* sebesar 87%. Dengan demikian dapat disimpulkan bahwa metode *Long short-term memory* (LSTM) layak digunakan untuk klasifikasi kalimat perbincangan masyarakat terhadap Covid-19 pada Twitter.

#### 5.2 Saran

Berdasarkan implementasi penelitian ini masih adanya kekurangan yang harus diperbaiki oleh peneliti maka dari itu terdapat saran pada penelitian tersebut agar mendapatkan hasil yang memuaskan yaitu sebagai berikut :

1. Menggunakan metode yang berbeda untuk melakukan pembobotan kata seperti : Word2Vec, FastText dan pembobotan kata lainnya.
2. Pada penelitian ini apabila menggunakan model *Deep learning* maka data yang digunakan dapat dinyatakan bahwa berukuran yang kecil, sedangkan

pada model *Deep learning* memerlukan data yang besar, maka peneliti dapat menambahkan data *training* yang digunakan.

3. Data yang digunakan dapat dinyatakan bahwa perbandingan sentimen tidak seimbang positif jauh lebih banyak daripada sentimen negatif, maka dari itu dapat melakukan pertimbangan dalam mengambil dataset.

## DAFTAR PUSTAKA

- Azhari, M., Situmorang, Z., & Rosnelly, R. (2021). Perbandingan Akurasi, Recall, dan Presisi Klasifikasi pada Algoritma C4.5, Random Forest, SVM dan Naive Bayes. *Jurnal Media Informatika Budidarma*, 5(2), 640. <https://doi.org/10.30865/mib.v5i2.2937>
- Chung, H., & Shin, K. S. (2018). Genetic algorithm-optimized long short-term memory network for stock market prediction. *Sustainability (Switzerland)*, 10(10). <https://doi.org/10.3390/su10103765>
- Das, S., & Kolya, A. K. (2021). Predicting the pandemic: sentiment evaluation and predictive analysis from large-scale tweets on Covid-19 by deep convolutional neural network. *Evolutionary Intelligence*, 0123456789. <https://doi.org/10.1007/s12065-021-00598-7>
- Dwi, R., Santosa, W., Bijaksana, M. A., & Romadhony, A. (2021). Implementasi Algoritma Long Short-Term Memory ( LSTM ) untuk Mendeteksi Penggunaan Kalimat Abusive Pada Teks Bahasa Indonesia. *Jurnal Tugas Akhir Fakultas Informatika*, 8(1), 691–702.
- Giarsyani, N. (2020). Komparasi Algoritma Machine Learning dan Deep Learning untuk Named Entity Recognition : Studi Kasus Data Kebencanaan. *Indonesian Journal of Applied Informatics*, 4(2), 138. <https://doi.org/10.20961/ijai.v4i2.41317>
- Gorunescu, F. (2011). *Data Mining (Concepts, Models and Techniques)*. Springer Singapore
- Hanifa, A., Fauzan, S. A., Hikal, M., & ... (2021). Perbandingan Metode LSTM dan GRU (RNN) untuk Klasifikasi Berita Palsu Berbahasa Indonesia. *Dinamika Rekayasa*, 17(1), 33–39. <http://dinarek.unsoed.ac.id/jurnal/index.php/dinarek/article/view/436>
- Herliandry, L. D., Nurhasanah, N., Suban, M. E., & Kuswanto, H. (2020). Pembelajaran Pada Masa Pandemi Covid-19. *JTP - Jurnal Teknologi Pendidikan*, 22(1), 65–70. <https://doi.org/10.21009/jtp.v22i1.15286>
- Jain Aditya, Kulkarni Gandhar, S. (2018). Natural Language Processing. *Natural Language Processing*, 6(10), 628–632. <https://doi.org/10.26438/ijcse/v6i1.161167>
- Kumar, A., Gupta, P. K., & Srivastava, A. (2020). A review of modern technologies for tackling COVID-19 pandemic. *Diabetes and Metabolic Syndrome: Clinical Research and Reviews*, 14(4), 569–573. <https://doi.org/10.1016/j.dsx.2020.05.008>
- Kurniawan, A., & Kurniawan, F. (2021). Time Series Forecasting for the Spread of Covid-19 in Indonesia Using Curve Fitting. *3rd 2021 East Indonesia Conference on Computer and Information Technology, EIconCIT 2021*, 45–

48. <https://doi.org/10.1109/EIConCIT50028.2021.9431936>

Le, X. H., Ho, H. V., Lee, G., & Jung, S. (2019). Application of Long Short-Term Memory (LSTM) neural network for flood forecasting. *Water (Switzerland)*, *11*(7). <https://doi.org/10.3390/w11071387>

Lecun, Y., Bengio, Y., & Hinton, G. (2015). Deep learning. *Nature*, *521*(7553), 436–444. <https://doi.org/10.1038/nature14539>

Listyarini, S. N., & Anggoro, D. A. (2021). Analisis Sentimen Pilkada di Tengah Pandemi Covid-19 Menggunakan Convolution Neural Network (CNN). *Jurnal Pendidikan Dan Teknologi Indonesia*, *1*(7), 261–268. <https://doi.org/10.52436/1.jpti.60>

Maitri, A. L., & Sutopo, J. (2019). Rancangan Bangun Chatbot Sebagai Pusat Informasi Lembaga Kursus Dan Pelatihan Menggunakan Pendekatan Natural Language Processing. *Eprints.Uty.Ac.Id*, 1–9. <http://eprints.uty.ac.id/>

Manaswi, N. K. (2018). Deep Learning with Applications Using Python. *Deep Learning with Applications Using Python*. <https://doi.org/10.1007/978-1-4842-5006-4>

Munasatya, N., & Novianto, S. (2020). Natural Language Processing untuk Sentimen Analisis Presiden Jokowi Menggunakan Multi Layer Perceptron. *Techno.Com*, *19*(3), 237–244. <https://doi.org/10.33633/tc.v19i3.3630>

Muslikah, A. N. (2021). SMS Spam Detection Menggunakan Metode Long Short Term Memory. <Http://Etheses.Uin-Malang.Ac.Id/>, 6.

Novaković, J. D., Veljović, A., Ilić, S. S., Papić, Ž., & Milica, T. (2017). Evaluation of Classification Models in Machine Learning. *Theory and Applications of Mathematics & Computer Science*, *7*(1), Pages: 39 – 46. <https://uav.ro/applications/se/journal/index.php/TAMCS/article/view/158>

Nurrohmat, M. A., & SN, A. (2019). Sentiment Analysis of Novel Review Using Long Short-Term Memory Method. *IJCCS (Indonesian Journal of Computing and Cybernetics Systems)*, *13*(3), 209. <https://doi.org/10.22146/ijccs.41236>

Pasaribu, D. J. M., Kusriani, K., & Sudarmawan, S. (2020). Peningkatan Akurasi Klasifikasi Sentimen Ulasan Makanan Amazon dengan Bidirectional LSTM dan Bert Embedding. *Inspiration: Jurnal Teknologi Informasi Dan Komunikasi*, *10*(1), 9–20. <https://doi.org/10.57585/inspir.v10i1.2568>

Qosim, A. L., Kurniawan, F., Bahruddin, U., Mubaraq, Z., Suhartono, & Faisal, M. (2021). Analysis Classification Opinion of Policy Government Announces Cabinet Reshuffle on YouTube Comments Using 1D Convolutional Neural Networks. *3rd 2021 East Indonesia Conference on Computer and Information Technology*, *EIConCIT* 2021, 30–57. <https://doi.org/10.1109/EIConCIT50028.2021.9431884>

- Rais, I. L., & Jondri, J. (2020). Klasifikasi Data Kuesioner dengan Metode Recurrent Neural Network. *EProceedings of Engineering*, 7(1), 2817–2826.
- Ridok, A., & Latifah, R. (2015). Klasifikasi Teks Bahasa Indonesia Pada Corpus Tak Seimbang Menggunakan NWKNN. *Konferensi Nasional Sistem Dan Informatika 2015, Oktober*, 222–227.
- Risalah Mutiara Tauhid: ASBABUN NUZUL SURAH 24 - AN NUR, AYAT 11, 4, 22.* (n.d.). Retrieved August 5, 2021, from <http://risalahmutiaratauhid.blogspot.com/2015/03/asbabun-nuzul-surah-24-nur-ayat-11-4-22.html>
- Rosenberg, H., Syed, S., & Rezaie, S. (2020). The Twitter pandemic: The critical role of Twitter in the dissemination of medical information and misinformation during the COVID-19 pandemic. *Canadian Journal of Emergency Medicine*, 22(4), 418–421. <https://doi.org/10.1017/CEM.2020.361>
- Santoso, J., Setiawan, E. I., Purwanto, C. N., & Kurniawan, F. (2021). Indonesian Sentence Boundary Detection using Deep Learning Approaches. *Knowledge Engineering and Data Science*, 4(1), 38. <https://doi.org/10.17977/um018v4i12021p38-48>
- Surat Al-Hujurat Ayat 6 | Tafsiro.com.* (n.d.). Retrieved May 11, 2022, from <https://tafsiro.com/49-al-hujurat/ayat-6#tafsir-jalalayn>
- Surat An-Nur Ayat 11 | Tafsiro.com.* (n.d.). Retrieved March 18, 2022, from <https://tafsiro.com/24-an-nur/ayat-11#tafsir-jalalayn>
- WHO Coronavirus (COVID-19) Dashboard | WHO Coronavirus (COVID-19) Dashboard With Vaccination Data.* (n.d.). Retrieved July 24, 2021, from <https://covid19.who.int/>
- Wibawa, M. S. (2016). Pengaruh Fungsi Aktivasi, Optimisasi dan Jumlah Epoch Terhadap Performa Jaringan Saraf Tiruan. *Jurnal Sistem Dan Informatika*, 11(2), 1–8. <https://doi.org/10.13140/RG.2.2.21139.94241>
- Winda Kurnia Sari. (2020). *Klasifikasi Teks Menggunakan Long Short-Term Memory Dengan Fitur Word Embedding Glove Dan Word2Vec.*
- Zhao, Z., Chen, W., Wu, X., Chen, P. C. V., & Liu, J. (2017). LSTM network: A deep learning approach for short-term traffic forecast. *IET Image Processing*, 11(1), 68–75. <https://doi.org/10.1049/iet-its.2016.0208>

## LAMPIRAN

```
Function cleansing(text)
  for dokumen in text
    if (dokumen mengandung tanda baca)
      remove punctuation
    else if (dokumen mengandung karakter kosong)
      remove karakter
    else if (dokumen mengandung mention)
      remove mention
    else if (dokumen mengandung hyperlink)
      remove hyperlink/URL
    else if (dokumen mengandung hastag)
      remove hastag
    else if (dokumen mengandung karakter)
      remove karakter
    else if (dokumen mengandung spasi berlebih)
      remove spasi
    else if (dokumen mengandung kata rt)
      remove rt
    else if (dokumen mengandung angka)
      remove angka
    else if (dokumen mengandung duplicate)
      remove duplicate
  End if
End for
End function
```

Pseudocode Cleansing Data

```
Function list CaseFolding(tweet)
  for each text
    Casefoldinglist += text;
  End for
End function
```

Pseudocode Case Folding

```
Function list Tokenizing(tweet)
  for each text in tweet
    if(isword)(word)
      WordNew += word
    End if
  End for
End function
```

Pseudocode Tokenizing

```
Function list Stemming(tweet_tokenized)
  text = (porter_stemmer.stem(tweet) for text in Tokenizing)
  return text
End function
```

Pseudocode Stemming