



**Analisa Kinerja RNN Menggunakan FastText Embedding  
terhadap Ulasan PeduliLindungi di Masa COVID-19**

*TUGAS AKHIR*



**PROGRAM STUDI TEKNIK INFORMATIKA  
FAKULTAS ILMU KOMPUTER  
UNIVERSITAS MERCU BUANA  
JAKARTA  
2022**



**Analisa Kinerja RNN Menggunakan FastText Embedding  
terhadap Ulasan PeduliLindungi di Masa COVID-19**

*Tugas Akhir*

Diajukan Untuk Melengkapi Salah Satu Syarat

Memperoleh Gelar Sarjana Komputer

UNIVERSITAS  
MERCU BUANA

Oleh:

ADINDA SITI SHALEHAH

41518010031

PROGRAM STUDI TEKNIK INFORMATIKA

FAKULTAS ILMU KOMPUTER

UNIVERSITAS MERCU BUANA

JAKARTA

2022

# LEMBAR PERNYATAAN ORISINALITAS

## LEMBAR PERNYATAAN ORISINALITAS

Yang bertanda tangan dibawah ini:

NIM : 41518010031

Nama : Adinda Siti Shalehah

Judul Tugas Akhir : Analisa Kinerja RNN Menggunakan FastText Embedding Terhadap Ulasan PeduliLindungi di Masa COVID-19

Menyatakan bahwa Laporan Tugas Akhir saya adalah hasil karya sendiri dan bukan plagiat. Apabila ternyata ditemukan didalam laporan Tugas Akhir saya terdapat unsur plagiat, maka saya siap untuk mendapatkan sanksi akademik yang terkait dengan hal tersebut.

Jakarta, 27 Juli 2022



Adinda Siti Shalehah



UNIVERSITAS  
MERCU BUANA

## SURAT PERNYATAAN PERSETUJUAN PUBLIKASI TUGAS AKHIR

### SURAT PERNYATAAN PERSETUJUAN PUBLIKASI TUGAS AKHIR

Sebagai mahasiswa Universitas Mercu Buana, saya yang bertanda tangan di bawah ini :

Nama Mahasiswa : Adinda Siti Shalehah  
NIM : 41518010031  
Judul Tugas Akhir : Analisa Kinerja RNN Menggunakan FastText  
Embedding Terhadap Ulasan PeduliLindungi di  
Masa COVID-19

Dengan ini memberikan izin dan menyetujui untuk memberikan kepada Universitas Mercu Buana **Hak Bebas Royalti Noneksklusif** (*None-exclusive Royalty Free Right*) atas karya ilmiah saya yang berjudul diatas beserta perangkat yang ada (jika diperlukan).

Dengan Hak Bebas Royalti/Noneksklusif ini Universitas Mercu Buana berhak menyimpan, mengalihmedia/formatkan, mengelola dalam bentuk pangkalan data (*database*), merawat dan mempublikasikan tugas akhir saya.

Selain itu, demi pengembangan ilmu pengetahuan di lingkungan Universitas Mercu Buana, saya memberikan izin kepada Peneliti di Lab Riset Fakultas Ilmu Komputer, Universitas Mercu Buana untuk menggunakan dan mengembangkan hasil riset yang ada dalam tugas akhir untuk kepentingan riset dan publikasi selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta.

Demikian pernyataan ini saya buat dengan sebenarnya.

UNIVERSITAS  
MERCU BUANA

Jakarta, 27 Juli 2022



Adinda Siti Shalehah

## SURAT PERNYATAAN LUARAN TUGAS AKHIR

### SURAT PERNYATAAN LUARAN TUGAS AKHIR

Sebagai mahasiswa Universitas Mercu Buana, saya yang bertanda tangan di bawah ini :

Nama Mahasiswa : Adinda Siti Shalehah  
NIM : 41518010031  
Judul Tugas Akhir : Analisa Kinerja RNN Menggunakan FastText  
Embedding Terhadap Ulasan PeduliLindungi di  
Masa COVID-19

Menyatakan bahwa :

1. Luaran Tugas Akhir saya adalah sebagai berikut :

No	Luaran	Jenis	Status
1	Publikasi Ilmiah	Jurnal Nasional Tidak Terakreditasi	
		Jurnal Nasional Terakreditasi	✓
		Jurnal International Tidak Bereputasi	
		Jurnal International Bereputasi	Diterima
	Disubmit/dipublikasikan di :	Nama Jurnal : Jurnal Informatika, Jurnal Pengembangan IT	
		ISSN : 2477-5126	
	Link Jurnal : <a href="https://ejournal.poltektegal.ac.id/index.php/informatika/index">https://ejournal.poltektegal.ac.id/index.php/informatika/index</a>		
	Link File Jurnal Jika Sudah di Publish		

2. Bersedia untuk menyelesaikan seluruh proses publikasi artikel mulai dari submit, revisi artikel sampai dengan dinyatakan dapat diterbitkan pada jurnal yang dituju.
3. Diminta untuk melampirkan scan KTP dan Surat Pernyataan (Lihat Lampiran Dokumen HKI), untuk kepentingan pendaftaran HKI apabila diperlukan

Demikian pernyataan ini saya buat dengan sebenarnya.

Jakarta, 27 Juli 2022

UNIVERSITAS  
MERCU BUANA



Adinda Siti Shalehah

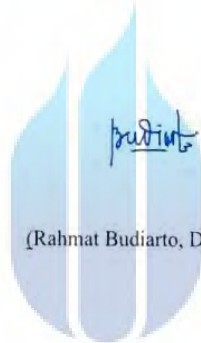
## LEMBAR PERSETUJUAN PENGUJI

### LEMBAR PERSETUJUAN PENGUJI

NIM : 41518010031  
Nama : Adinda Siti Shalehah  
Judul Tugas Akhir : Analisa Kinerja RNN Menggunakan FastText  
Embedding terhadap Ulasan PeduliLindungi di  
Masa COVID-19

Tugas Akhir ini telah diperiksa dan disidangkan sebagai salah satu persyaratan untuk memperoleh gelar Sarjana pada Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Mercu Buana.

Jakarta, 27 Juli 2022



(Rahmat Budiarto, Dr.Prof)

UNIVERSITAS  
MERCU BUANA


## LEMBAR PERSETUJUAN PENGUJI

### LEMBAR PERSETUJUAN PENGUJI

NIM : 41518010031  
Nama : Adinda Siti Shalehah  
Judul Tugas Akhir : Analisa Kinerja RNN Menggunakan FastText  
Embedding terhadap Ulasan PeduliLindungi di  
Masa COVID-19

Tugas Akhir ini telah diperiksa dan disidangkan sebagai salah satu persyaratan untuk memperoleh gelar Sarjana pada Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Mercu Buana.

Jakarta, 27 Juli 2022



UNIVERSITAS  
MERCUBUANA

*(Signature)*  
(Rushendra, S.Kom, MT)

## LEMBAR PERSETUJUAN PENGUJI

### LEMBAR PERSETUJUAN PENGUJI

NIM : 41518010031  
Nama : Adinda Siti Shalehah  
Judul Tugas Akhir : Analisa Kinerja RNN Menggunakan FastText  
Embedding terhadap Ulasan PeduliLindungi di  
Masa COVID-19

Tugas Akhir ini telah diperiksa dan disidangkan sebagai salah satu persyaratan untuk memperoleh gelar Sarjana pada Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Mercu Buana.

Jakarta, 27 Juli 2022



(Dr. Harwikarya, MT)

UNIVERSITAS  
MERCU BUANA



# LEMBAR PENGESAHAN

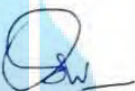
## LEMBAR PENGESAHAN

NIM : 41518010031  
Nama : Adinda Siti Shalehah  
Judul Tugas Akhir : Analisa Kinerja RNN Menggunakan FastText Embedding Terhadap Ulasan PeduliLindungi di Masa COVID-19


Tugas Akhir ini telah diperiksa dan disidangkan sebagai salah satu persyaratan untuk memperoleh gelar Sarjana pada Program Studi Teknik Informatika, Fakultas Ilmu Komputer, Universitas Mercu Buana.

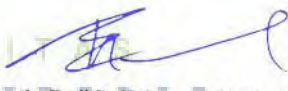
Jakarta, 27 Juli 2022

Menyetujui,

  
(Yava Sudarva Triana, Ph.D.)  
Dosen Pembimbing

Mengetahui,

  
(Wawan Ganawan, S.Kom, MT)  
Koord. Tugas Akhir Teknik Informatika

  
(Ir. Emil R. Kaburuan, Ph.D., IPM.)  
Ka. Prodi Teknik Informatika

## KATA PENGANTAR

Puji syukur kita panjatkan kehadiran Allah SWT, Tuhan yang Maha Esa, Maha Pengasih lagi Maha Penyayang. Alhamdulillah atas segala rahmat-Nya sehingga penulis dapat menyelesaikan skripsi yang berjudul “Analisa Kinerja RNN Menggunakan FastText Embedding terhadap Ulasan PeduliLindungi di Masa COVID-19“ dengan baik.

Penyusunan skripsi ini merupakan salah satu syarat untuk memperoleh gelar Sarjana Strata 1 Program Studi Teknik Informatika Fakultas Ilmu Komputer Universitas Mercu Buana Jakarta. Tidak hanya itu, penyusunan skripsi ini juga sebagai salah satu wujud implementasi dari ilmu yang telah didapat selama masa perkuliahan di Program Studi Teknik Informatika Fakultas Ilmu Komputer Universitas Mercu Buana Jakarta.

Penulis menyadari bahwa tanpa bantuan bimbingan, masukan, dan arahan dari berbagai pihak skripsi ini tidak akan berjalan lancar. Oleh karena itu, pada kesempatan ini penulis mengucapkan terima kasih kepada:

1. Bapak Yaya Sudarya Triana, M.Kom., Ph.D. selaku Dosen Pembimbing Tugas Akhir yang memberikan waktu, bimbingan dan masukan selama penyelesaian Tugas Akhir.
2. Bapak Emil Robert Kaburuan, PhD selaku Kepala Program Studi Teknik Informatika.
3. Bapak Wawan Gunawan, S.Kom, MT selaku Koordinator Tugas Akhir Jurusan Teknik Informatika.
4. Bapak Dr. Harwikarya, MT selaku dosen pembimbing akademik yang mengarahkan kegiatan akademik saya selama perkuliahan.
5. Kedua orang tua tercinta, kaka saya tercinta Annisa Fitri Nurdianti, serta kaka sepupu tercinta Alya Dipa A Yani yang telah mendoakan, memberikan dukungan dan memotivasi dalam menyelesaikan skripsi ini.
6. Sahabat-sahabat saya Tengku Salwa Nabila, Getsfy Mahardika, Nabilah Mumtaz, Anisa Tri Astuti, Rosanti Hutauruk, dan Rani Aprillya Putri yang selalu memberikan keceriaan, semangat, kasih sayang dan dukungan yang luar biasa sehingga dapat menyelesaikan skripsi ini.

7. Teman-teman dan semua pihak yang tidak dapat disebutkan namanya satu persatu yang telah membantu dalam penulisan skripsi ini.
8. Terakhir, terimakasih kepada saya sendiri telah percaya diri, mau berjuang dan bekerja keras untuk menyelesaikan skripsi ini.

Akhir kata, penulis menyadari bahwa penelitian ini jauh dari kesempurnaan. Oleh sebab itu, kritik maupun saran selalu penulis harapkan demi tercapainya hal terbaik dari penelitian ini. Besar harapan penulis, semoga penelitian ini dapat memberikan manfaat sekaligus menambah pengetahuan bagi berbagai pihak. Amin.

Jakarta, 30 Juni 2022

Adinda Siti Shalehah



## DAFTAR ISI

HALAMAN SAMPUL.....	i
HALAMAN JUDUL .....	i
LEMBAR PERNYATAAN ORISINALITAS .....	ii
SURAT PERNYATAAN PERSETUJUAN PUBLIKASI TUGAS AKHIR... ..	iii
SURAT PERNYATAAN LUARAN TUGAS AKHIR.....	iv
LEMBAR PERSETUJUAN PENGUJI.....	v
LEMBAR PENGESAHAN .....	viii
ABSTRAK .....	ix
ABSTRACT .....	x
KATA PENGANTAR.....	xi
DAFTAR ISI.....	xiii
NASKAH JURNAL .....	1
KERTAS KERJA.....	8
BAB 1. LITERATUR REVIEW .....	9
BAB 2. ANALISIS DAN PERANCANGAN.....	24
BAB 3. SOURCE CODE .....	33
BAB 4. DATASET.....	48
BAB 5. TAHAPAN EKSPERIMEN .....	57
BAB 6. HASIL SEMUA EKSPERIMEN.....	64
DAFTAR PUSTAKA .....	80
LAMPIRAN DOKUMEN HAKI.....	84
LAMPIRAN KORESPONDENSI .....	86

# Analisa Kinerja RNN Menggunakan FastText Embedding terhadap Ulasan PeduliLindungi di Masa COVID-19

Adinda Siti Shalehah<sup>1</sup>, Yaya Sudarya Triana<sup>2</sup>

<sup>1</sup>Jurusan Teknik Informatika, Fakultas Ilmu Komputer, Universitas Mercu Buana, Jakarta

<sup>2</sup>Jurusan Sistem Informasi, Fakultas Ilmu Komputer, Universitas Mercu Buana, Jakarta

<sup>1,2</sup> Jl. Raya, Meruya Sel., Kec. Kembangan, Jakarta, Daerah Khusus Ibukota Jakarta, 11650, Indonesia

email: <sup>1</sup>\*)yaya.sudarya@mercubuana.ac.id, <sup>2</sup>41518010031@student.mercubuana.ac.id

*Abstract – PeduliLindungi is an application designed by the Ministry of Communication and Information in collaboration with the Ministry of Health and the Ministry of State-Owned Enterprises with the aim of monitoring the spread of COVID-19. In addition, it can be used to register and view vaccine status. The application can be downloaded through the market app, one of which is the Google Play Store. PeduliLindungi users are increasing day by day so there are pros and cons regarding the application system. Users submit their comments through the feedback feature on the Google Play Store in the form of reviews and ratings. The reviews given by users are quite varied, so a classification technique is needed to group reviews based on their class. Reviews are given two class labels, namely Positive and Negative. Recurrent Neural Network (RNN) is a proposed algorithm to handle text classification because its most famous architecture is used for text processing. This study examines the performance of RNN with three variants, namely SimpleRNN, Long Short Term Memory, and Bidirectional-LSTM with the addition of FastText embedding as input to improve algorithm performance. The results showed that text classification using RNN with the addition of FastText obtained good and quite accurate results. The highest accuracy obtained by the LSTM and Bi-LSTM models with the addition of FastText is 89% and SimpleRNN 88%.*

*Abstrak – PeduliLindungi merupakan aplikasi rancangan Kementerian Komunikasi dan Informatika bekerjasama dengan Kementerian Kesehatan dan Kementerian Badan Usaha Milik Negara dengan tujuan untuk memantau penyebaran COVID-19. Selain itu, dapat digunakan untuk mendaftar dan melihat status vaksin. Aplikasi tersebut dapat diunduh melalui market app salah satunya Google Play Store. Pengguna PeduliLindungi kian hari kian meningkat sehingga adanya pro dan kontra mengenai sistem aplikasi. Pengguna menyampaikan komentarnya melalui fitur feedback yang ada di Google Play Store berupa ulasan dan rating. Ulasan yang diberikan pengguna cukup bervariasi maka dibutuhkan teknik klasifikasi untuk mengelompokkan ulasan berdasarkan kelasnya. Ulasan diberi dua label kelas yaitu Positif dan Negatif. Recurrent Neural Network (RNN) merupakan algoritma yang diusulkan untuk menangani klasifikasi teks karena arsitekturnya yang paling terkenal digunakan untuk pemrosesan teks. Penelitian ini menguji kinerja RNN dengan tiga variannya yaitu SimpleRNN, Long Short Term Memory, dan Bidirectional-LSTM dengan penambahan FastText embedding sebagai input untuk meningkatkan kinerja algoritma. Hasil penelitian menunjukkan bahwa klasifikasi teks menggunakan RNN dengan penambahan FastText memperoleh hasil yang baik*

hasil yang baik dan cukup akurat. Akurasi tertinggi diperoleh oleh model LSTM dan Bi-LSTM dengan penambahan FastText adalah 89% dan SimpleRNN 88%.

**Kata Kunci – COVID-19, Klasifikasi, RNN, LSTM, Bi-LSTM, FastText.**

## I. PENDAHULUAN

Awal maret 2020 Indonesia terserang wabah COVID-19 yang disebabkan oleh virus SARS-CoV-2 [1], [2]. Virus ini menyerang tubuh manusia melalui sistem pernapasan ditandai dengan adanya gejala ringan seperti batuk, hilang rasa dan penciuman sampai dengan adanya gejala serius sesak napas [2], [3]. World Health Organization (WHO) telah menetapkan COVID-19 sebagai pandemi global. Penyebaran virus yang setiap harinya kian meningkat terhitung sampai dengan tanggal 26 Oktober 2020 lalu, ada 392 ribu lebih kasus positif COVID-19 dengan angka kematian sebanyak 13 ribu jiwa [4]. Berbagai upaya untuk memutus peyebaran COVID-19 telah dilakukan oleh pemerintah. Salah satunya adalah memanfaatkan teknologi informasi dengan menciptakan aplikasi *smartphone* bernama PeduliLindungi [5].

PeduliLindungi merupakan aplikasi hasil rancangan Kementerian Komunikasi dan Informatika (KOMINFO) bekerjasama dengan Kementerian Kesehatan (KEMENKES) dan Kementerian Badan Usaha Milik Negara (BUMN). Aplikasi tersebut dibuat untuk memudahkan pemerintah dalam memantau penyebaran COVID-19 serta partisipasi masyarakat untuk berbagi lokasi riwayat perjalanan supaya meminimalisir penyebaran virus. Selain itu, dapat digunakan untuk mendaftar dan melihat status vaksin. [5]. Pengguna PeduliLindungi semakin hari semakin meningkat sehingga menuai banyak pro dan kontra mengenai sistem aplikasi. Pengguna menyampaikan komentarnya melalui fitur *feedback* berupa ulasan dan *rating* yang ada pada *platform market app*. Salah satu *market app* yang banyak digunakan yaitu Google Play Store [6].

Ulasan yang diberikan oleh pengguna dapat menjadi sumber informasi yang efektif untuk mengetahui kualitas dari sebuah aplikasi. Ulasan pengguna cukup bervariasi dilihat dari kalimat serta *rating* yang diberikan, ada yang bersifat positif adapula yang negatif. [7]. Untuk mengetahui sifat dari ulasan yang bervariasi tidaklah cukup hanya dengan melihat beberapa

\*)penulis korespondensi: Yaya Sudarya Triana

Email: yaya.sudarya@mercubuana.ac.id

review dan rating saja. Perlu dilakukannya proses klasifikasi teks dari data ulasan, tujuannya untuk mempermudah dalam mengkategorikan ulasan sesuai dengan sifatnya. Namun, ulasan memiliki jumlah data yang besar sehingga akan berdampak pada proses klasifikasi yang cukup sulit. Untuk mengatasi permasalahan tersebut penelitian ini menggunakan model deep learning, menurut [8] deep learning merupakan salah satu model yang dapat mengatasi data dengan jumlah yang besar.

Pada penelitian ini, penulis melakukan analisa kinerja metode klasifikasi teks Deep Learning dengan penambahan *word embedding* FastText sebagai input nya. Algoritma yang digunakan pada penelitian ini mencakup tiga varian dari Recurrent Neural Network (RNN) yaitu SimpleRNN, Long Short Term Memory (LSTM), dan Bidirectional Long Short Term Memory (Bi-LSTM). Ketiga algoritma tersebut dibandingkan dengan penambahan FastText dan Non-FastText, bertujuan untuk melihat kinerja algoritma mana yang lebih akurat.

## II. PENELITIAN YANG TERKAIT

Recurrent Neural Network (RNN) adalah arsitektur pengembangan dari jaringan syaraf tiruan yang proses kerjanya secara berulang (*looping*) untuk memproses input data *sequence*. RNN memiliki arsitektur memori yang dapat mengenali dan menyimpan informasi dari masa lalu [9]. Selain itu, RNN juga mengembangkan pemodelannya menjadi beberapa varian yang dapat memaksimalkan efektivitas arsitektur diantaranya adalah Long Short Term Memory (LSTM) dan Bidirectional Long Short Term Memory (Bi-LSTM) [10]. Metode RNN telah banyak diterapkan untuk klasifikasi teks karena metode ini mempunyai “memori” yang dimana setiap sampel akan diproses secara sama dengan konsiderasi atas sampel-sampel sebelumnya.

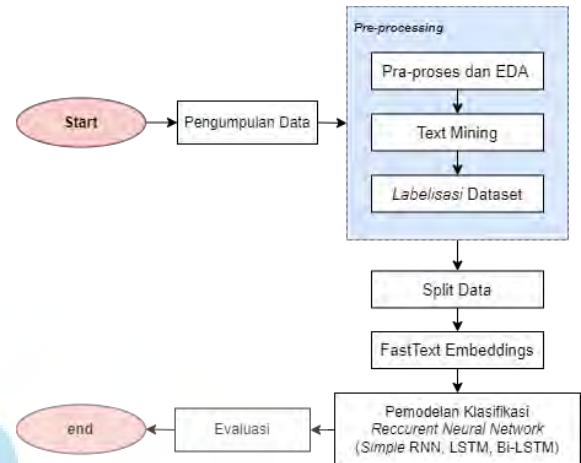
Terdapat beberapa penelitian yang telah dilakukan mengenai klasifikasi teks menggunakan RNN, diantaranya oleh [10] yang melakukan evaluasi terhadap lima variasi algoritma RNN yaitu LSTM, Group Long Short-Term Memory Networks, Gated Recurrent Unit, dan Update Recurrent Unit dengan penambahan tiga *word embedding* Word2Vec, Glove, dan FastText. Dari kelima varian RNN dan tiga *word embedding* membuktikan bahwa didapat hasil terbaik oleh algoritma GLRNN 93.75% dengan penambahan FastText untuk data unbalanced. Sedangkan untuk data balance hasil akhir terbaik 88.39% oleh algoritma LRNN dengan penambahan FastText. Penelitian lain juga dilakukan oleh [11] yang melakukan klasifikasi teks menggunakan LSTM dan RNN dengan penambahan Glove. Hasil dari penelitiannya menunjukkan bahwa LSTM dapat menghasilkan akurasi lebih baik yaitu 78% dan RNN hanya 76%. Penelitian [12] membuktikan bahwa klasifikasi teks data twitter menggunakan LSTM dengan *word embedding* FastText menunjukkan hasil akurasi sebesar 97.86%. Hal ini dikarenakan model LSTM memiliki keunggulan untuk memproses data *sequence* dan *word embedding* FastText dengan kinerjanya yang dapat memecah kata yang tidak terdeteksi selama training model menjadi n-gram.

Dari beberapa referensi diatas penulis tertarik untuk menggunakan Metode pengujian Algoritma Deep Learning

Reccurent Neural Network dengan varian SimpleRNN, LSTM, dan Bi-LSTM serta melakukan penambahan *word embedding* FastText terhadap data ulasan PeduliLindungi. Penelitian ini dilakukan untuk mengetahui kinerja dari masing-masing algoritma serta pengaruh penambahan *word embedding* pada implementasi klasifikasi teks.

## III. METODE PENELITIAN

Dalam penelitian ini terdiri dari beberapa tahapan proses. Dimulai dari Pengumpulan data, *Pre-processing*, *Split* data, FastText *embeddings*, Pemodelan klasifikasi, dan Evaluasi. Tahapan penelitian bisa dilihat pada gambar 1.



Gambar 1. Tahapan Penelitian

### A. Pengumpulan Data

Dataset yang digunakan adalah teks ulasan aplikasi PeduliLindungi yang diambil dari <https://play.google.com/> dengan teknik *scraping* menggunakan API yang telah disediakan oleh library google-play-scraper. Data diambil dengan rentan waktu satu tahun yaitu dari 22 Januari 2021 sampai 22 Januari 2022. Data yang terambil sebanyak 265.798 ulasan dan disimpan dengan format csv. Berikut merupakan hasil *scraping* data ulasan PeduliLindungi ditunjukkan pada Tabel I.

Tabel I. Ulasan Hasil *Scraping*

	review	rating	versi	date
0	Sertifikat tdk kunjung muncul,,di app meskipun...	1	NaN	2022-01-22 15:41:52
1	Kenapa y aku ko g bisa chek in klo mau ke mall...	1	4.0.8	2022-01-22 15:38:48
2		Good	5	4.0.8 2022-01-22 15:38:17
3		Mantap	5	4.0.2 2022-01-22 15:36:17
4	Hadeh error', mau kmn ? susah	2	NaN	2022-01-22 15:31:02
...	...	...	...	...
265793	Tingkatkan trs servisnya	5	3.1.2	2021-01-22 01:13:48
265794	Informasinya ngga akurat, di tempat saya banya...	3	3.1.3	2021-01-22 01:11:39
265795	Sudah dcek.. Tapi belum tersedia.. Diminta dat...	2	3.1.3	2021-01-22 00:58:55
265796	Aplikasi yang sangat berguna dan informatif	5	3.1.3	2021-01-22 00:47:18
265797	Maaf. Saya sudah di vaksin. Tapi belum dapat s...	4	3.1.3	2021-01-22 00:38:00

### B. Pre-processing

*Pre-processing* merupakan proses pembersihan dataset dari data *noise* dan untuk meningkatkan kualitas pengolahan data. Tahapan-tahapan dari *pre-processing* yaitu, sebagai berikut :

#### 1) Pra-proses dan EDA



Pra-proses terlebih dahulu dilakukan dengan menghapus kolom yang tidak dibutuhkan karena yang digunakan untuk proses klasifikasi teks hanyalah kolom review dan rating.

Selanjutnya, mengubah nama kolom content, score, dan date menjadi review, rating, dan date. Dan yang berikutnya adalah konversi kolom date menjadi bentuk datetime untuk mempermudah saat melakukan proses EDA. Setelah pra-proses selesai, dilakukanlah proses Exploratory Data Analysis (EDA) yaitu proses menganalisis data kedalam bentuk visualisasi untuk mempermudah memahami isi dari sebuah data dan dapat menjadi informasi yang berguna [13].

### 2) Text Mining

Text mining adalah proses merubah data teks dari *unstructured* menjadi teks *structured*. Proses text mining dapat dilakukan dengan teknik *Natural Language Processing* (NLP) yaitu komponen dari kecerdasan buatan yang dapat memproses dan memahami bahasa yang biasa digunakan manusia ketika berkomunikasi dalam bentuk teks. Berikut tahapan-tahapan dari teknik NLP :

- Filtering* merupakan proses menghilangkan tanda baca, emoji, angka, dan semua karakter yang bukan huruf [12].
- Case Folding* yaitu tahap mengubah semua huruf kapital yang ada pada dataset menjadi huruf kecil [14].
- Tokenizing*. Pada tahap ini dilakukan proses memisahkan kalimat menjadi beberapa bagian token berupa kata [12], [14].
- Normalized*. Tahap ini pada pengolahan teks dilakukan untuk mengubah kata non formal menjadi formal.
- Stopword Removal* adalah proses menghapus kata yang memiliki arti rendah seperti “ini, itu, yang, dan, di” dan memilih kata-kata yang dianggap penting pada saat proses *tokenizing*.
- Stemming*. Proses ini merubah semua kata ber-imbunan menjadi bentuk dasar memakai *library sastrawi* karena teks yang digunakan berbahasa Indonesia [12].

### 3) Labelisasi Dataset

Labelisasi adalah proses memberi kelas pada dataset. Pada penelitian ini dilakukan klasifikasi dengan membagi dataset menjadi dua kelas label yaitu kelas positif dan negatif. Kelas label dibuat secara manual berdasarkan rating pada review. Untuk rating 4-5 diberi label kelas positif dan rating 1-2 diberi label kelas negatif. Review dengan rating 3 tidak digunakan karena memiliki kecenderungan antara kelas positif atau negatif.

### C. Split Data

Metode split data atau train test split adalah metode yang digunakan untuk membagi dataset menjadi dua bagian, yakni data testing dan data training. Dalam penelitian ini pembagian dataset untuk training sebesar 80% dan untuk data testing 20%. Data training dipergunakan untuk melatih model klasifikasi sedangkan data testing dipergunakan untuk menguji kinerja model dalam mengklasifikasikan data teks.

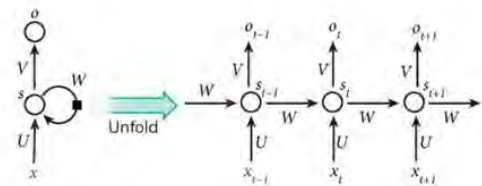
### D. FastText Embeddings

*Word embedding* ialah salah satu teknik merubah teks kedalam bentuk vektor [14]. *Word embedding* banyak macamnya salah satunya yaitu *FastText embeddings*. Pertama kali *FastText embeddings* dibuat oleh tim Facebook dengan tujuan untuk membantu proses kinerja klasifikasi teks dan

representasi kata supaya lebih efisien. *FastText* merupakan pengembangan dari *embeddings* *Word2Vec* yang sebelumnya telah lebih dulu digunakan sebagai *word embeddings*. Cara kerja *FastText* ialah dengan mewakili setiap kata menjadi beberapa *n-grams*. Maksudnya adalah dari satu kata akan mengandung beberapa kata atau karakter [15]. *FastText embeddings* juga memiliki kelebihan yang tidak dimiliki *word embedding* lain yaitu dapat mendeteksi kata yang tidak ada di kamus sebelumnya atau biasa disebut dengan *Out Of Vocabulary* (OOV).

### E. Recurrent Neural Network

*Recurrent Neural Network* adalah salah satu arsitektur pada jaringan syaraf tiruan yang proses kerjanya secara berulang kali untuk memproses data berurutan (sekuensial). Arsitektur jaringan RNN dapat dilihat pada gambar dibawah ini [9] :



Gambar 2. Arsitektur Jaringan RNN

Perulangan yang terjadi pada RNN berakibat menghasilkan keluaran di setiap waktunya. Proses tersebut diawali dari pengujian yang memiliki nilai masukan  $x$  lalu di proses kedalam RNN yang memiliki nilai *hidden layer*, dimana nilai yang terdapat dari *hidden layer* ini setiap kali dilakukan pembaruan pada saat RNN membaca masukan baru. Untuk menghitung layer pertama pada RNN digunakan rumus seperti persamaan 1:

$$h_t = f_w(h_{t-1}, x_t) \quad (1)$$

- $h_t$  = *hidden layer*
- $f$  = fungsi
- $w$  = bobot
- $h_{t-1}$  = *hidden layer* sebelumnya
- $x_t$  = nilai input

Untuk penambahan aktivasi fitur (sigmoid) pada layer berikutnya digunakan rumus seperti persamaan 2:

$$h_t = \text{sigmoid}(W_{hh}h_{t-1} + W_{xh}x_t) \quad (2)$$

$W_{hh}$  = matriks bobot lain

$W_{xh}$  = matriks bobot

Untuk menghitung semua performa layer RNN digunakan rumus persamaan 3:

$$Y_t = Why h_t \quad (3)$$

$Y_t$  = nilai output

$Why$  = matriks bobot nilai output

Pada penelitian ini selain menggunakan simple RNN diterapkan dua varian RNN lainnya, yaitu:

#### 1) Long Short Term Memory (LSTM)

LSTM merupakan peningkatan dari RNN dan diusulkan pertama kali oleh Hochreiter dan Schmidhuber. LSTM dianggap memiliki kinerja yang baik pada sejumlah kasus *Natural Language Processing* (NLP). Hal ini, karena LSTM memiliki sel memori dengan jangka panjang dan tiga unit gerbang yang dapat membaca, menyimpan, serta melakukan perbaruan informasi. Ketiga unit gerbang tersebut adalah *input*

gate, forget gate, dan output gate [9], [16], [17]. Alur kerja LSTM yang pertama dimulai dari *forget gate* ( $f_t$ ) untuk menentukan informasi yang tidak digunakan agar dikeluarkan dari *memory cell* ( $C_{t-1}$ ) menggunakan fungsi sigmoid [18], dirumuskan seperti persamaan 4:

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \quad (4)$$

$f_t$  = forget gate  
 $\sigma$  = fungsi aktivasi sigmoid  
 $W_f$  = bobot dari forget gates  
 $h_{t-1}$  = hidden layer sebelumnya  
 $x_t$  = input data  
 $b_f$  = nilai bias pada forget gate

Alur selanjutnya menetapkan informasi yang bakal diperbarui melalui *input gate* ( $i_t$ ) dan di waktu yang sama menghasilkan *memory cell* yang baru ( $\tilde{C}_t$ ) [18], proses ini ditunjukkan pada persamaan (5), (6), dan (7) :

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \quad (5)$$

$$\tilde{C}_t = \tanh(W_c \cdot [h_{t-1}, x_t] + b_c) \quad (6)$$

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \quad (7)$$

Kemudian alur terakhir yaitu menentukan perhitungan *output gates* dengan menjalankan aktivasi sigmoid dan tanh pada *cell state*, perhitungan *output gates* dapat dihitung dengan persamaan berikut:

$$O_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \quad (8)$$

$$h_t = o_t * \tanh(C_t) \quad (9)$$

## 2) Bidirectional-Long Short Term Memory

Pengembangan Bi-LSTM memanfaatkan konteks dari dua arah dengan *hidden layer* yang terpisah. Dua arah pada arsitektur Bi-LSTM yaitu *forward* dan *backward*, dimana kedua arah ini dapat meningkatkan banyaknya informasi baik ke jaringan ataupun konteks untuk algoritma [18]. Output dari Bi-LSTM ditunjukkan pada persamaan berikut:

$$y_t = W_{\rightarrow} \begin{matrix} \rightarrow \\ h_t \end{matrix} + W_{\leftarrow} \begin{matrix} \leftarrow \\ h_t \end{matrix} \quad (10)$$

## F. Evaluasi

Tahap terakhir dalam penelitian ini dilakukan evaluasi menggunakan *confusion matrix* yaitu metode yang digunakan untuk menilai kinerja model klasifikasi. Hasil dari *confusion matrix* diwakili dengan istilah TP (*True Positif*), TN (*True Negative*), FP (*False Positif*), dan FN (*False Negative*). Hasil tersebut digunakan untuk mendapat nilai *classification report* berupa *accuracy*, *precision*, *recall*, dan *f1-score* [17]–[19]. Dibawah ini merupakan persamaan untuk menghitung *classification report*:

$$Accuracy = \frac{TP + TN}{TP + TN + FP + FN} \quad (11)$$

$$Precision = \frac{TP}{TP + FN} \quad (12)$$

$$Recall = \frac{TP}{TP + FP} \quad (13)$$

$$F1-Score = \frac{2 * precision * recall}{precision + recall} \quad (14)$$

## IV. HASIL DAN PEMBAHASAN

### A. Hasil Pre-processing

Jumlah data awal ulasan PeduliLindungi yang terambil dari situs google play store sebanyak 265.798 baris data. Setelah dilakukan *pre-processing* dengan menghapus data null, menghapus beberapa kolom yang tidak digunakan, menghapus nilai 3 pada kolom rating, dan proses *text mining*. Dihilangkan data akhir sebanyak 201.307 baris data serta diambil dua kolom yaitu rating dan review\_clean untuk dilakukan klasifikasi. Proses klasifikasi dilakukan secara manual dengan memberi label kelas positif dan negatif. Setelah itu, dilakukan perubahan label kelas dari kategori menjadi numerik menggunakan proses One-Hot-Encode. Rincian jumlah dataset yang telah diberi label dua kelas dapat dilihat pada Tabel II.

Tabel II. Data dengan Label Kelas

LABEL	JUMLAH
Positif	108.564
Negatif	92.743

### B. Analisis Kinerja Algoritma

Seperti yang telah dijelaskan sebelumnya pada bagian pembahasan, penelitian ini akan melakukan analisa kinerja klasifikasi teks menggunakan tiga algoritma varian RNN yaitu SimpleRNN, LSTM, dan Bi-LSTM serta penambahan *FastText embeddings* pada input model. Untuk menguji kinerja tersebut dataset yang diperoleh dibagi menjadi dua bagian yaitu data train dan data test dengan perbandingan 80:20. Implementasi dilakukan menggunakan bahasa pemrograman python yang terintegrasi dengan library Keras, Tensorflow, dan Sklearn.

Hyper-parameter yang digunakan untuk model pelatihan terdiri dari *input layer* menggunakan *FastText embeddings* berukuran 300 dimensi vektor dan maksimum sekuens sebanyak 21, sehingga pada *input layer* akan menghasilkan matriks dengan ukuran 21 x 300. Layer kedua dari masing-masing model berukuran 128 filter dan layer berikutnya menggunakan 100 filter dengan fungsi aktivasi relu. Layer terakhir adalah *output layer* dengan jumlah *neuron* 2 disesuaikan berdasarkan jumlah label kelas.

Ke-tiga model varian RNN yang digunakan dilatih dengan *batch size* berukuran 1024 dan *epoch* berukuran 100, serta menggunakan optimasi Adam. Untuk mencegah *overfitting* pada layer sebelum *output layer* diaplikasikan teknik dropout dengan rate 0.4 yang secara random akan me-non aktifkan 40% *neuron* selama proses pelatihan berlangsung. Selain itu, digunakan juga fitur *early stopping*. Cara kerja dari *early stopping* yaitu dengan menggunakan fungsi *callback* dimana proses pelatihan akan berhenti ketika nilai training dan validasi sudah melewati batasan yang telah ditentukan sebelumnya.

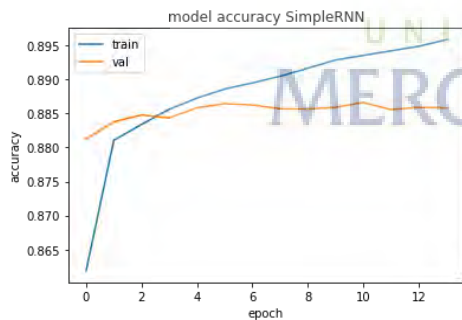


Tabel III. Hasil Pengujian

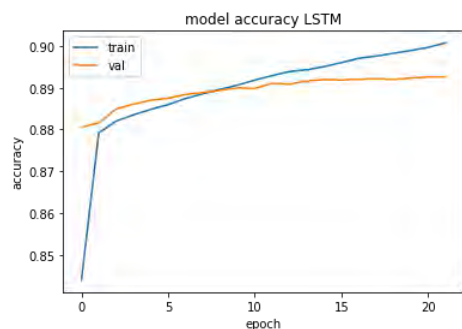
Models	Label	Classification Report			Accuracy
		Precision	Recall	F1-Score	
Simple RNN with FastText	Positif	0.91	0.87	0.88	0.88
	Negatif	0.86	0.9	0.89	
LSTM with FastText	Positif	0.93	0.86	0.90	0.89
	Negatif	0.85	0.93	0.89	
Bi-LSTM with FastText	Positif	0.92	0.87	0.90	0.89
	Negatif	0.86	0.91	0.89	
Simple RNN without FastText	Positif	0.89	0.87	0.88	0.87
	Negatif	0.85	0.88	0.86	
LSTM without FastText	Positif	0.88	0.87	0.88	0.87
	Negatif	0.85	0.86	0.86	
Bi-LSTM without FastText	Positif	0.90	0.87	0.88	0.88
	Negatif	0.85	0.89	0.87	

Berdasarkan pada tabel III menunjukkan kinerja dari setiap algoritma varian RNN dengan penambahan FastText dan tanpa penambahan FastText. Dievaluasi menggunakan *classification report* berupa nilai *precision*, *recall*, *f1-score*, dan *accuracy*. Terlihat bahwa model dengan hasil terbaik diperoleh pada model LSTM dan Bi-LSTM dengan penambahan FastText senilai 89%. Sedangkan model tanpa penambahan FastText mengalami penurunan akurasi 1-2%.

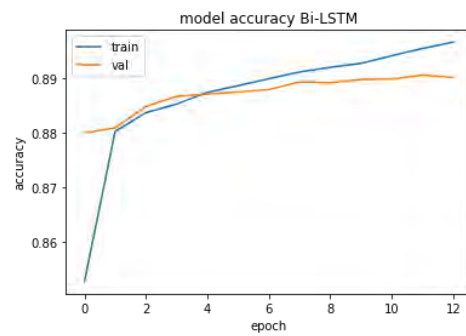
Pada gambar 3 sampai 5 menunjukkan akurasi training dan validasi akurasi ke tiga model dengan penambahan FastText. Akurasi training pada model SimpleRNN dimulai pada 0.86 dan berakhir di 0.89. Validasi tingkat keakuratan training SimpleRNN berada di tingkat yang sama yaitu 0.88. Untuk akurasi training model LSTM dimulai pada 0.84 sampai 0.90. Keakuratan validasi model training LSTM dimulai dari 0.88 sampai 0.89. Sedangkan, akurasi training untuk Bi-LSTM dimulai pada 0.85 dan berakhir pada 0.89 serta akurasi validasinya dimulai dari 0.87 sampai 0.89.



Gambar 3. Training dan Validasi Akurasi SimpleRNN

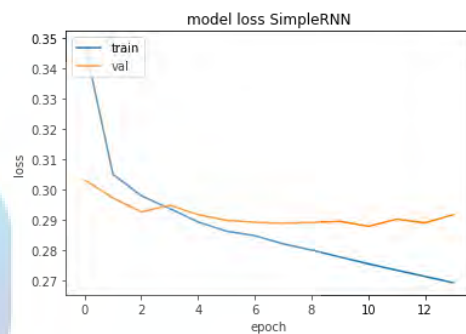


Gambar 4. Training dan Validasi Akurasi LSTM

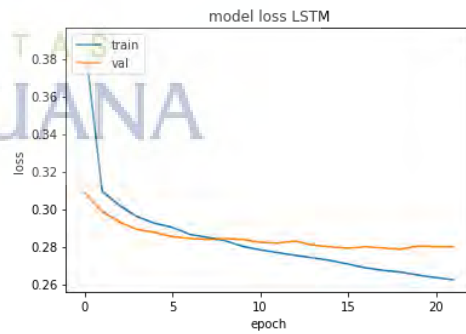


Gambar 5. Training dan Validasi Akurasi Bi-LSTM

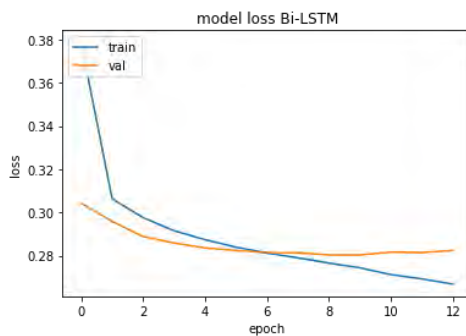
Validasi loss SimpleRNN dimulai pada 0.30 dan berakhir di nilai 0.29, sedangkan training loss nya dimulai pada 0.34 dan berakhir di nilai 0.26. Validasi loss model LSTM dimulai dari 0.30 dan berakhir pada 0.27 dan juga training loss nya mulai dari 0.38 dan berakhir pada 0.26. Untuk validasi loss model Bi-LSTM dimulai dari 0.30 sampai 0.28 serta training loss nya dimulai dari 0.37 dan berakhir di 0.26. Seperti yang ditunjukkan pada gambar 6 sampai 8.



Gambar 6. Training dan Validasi Loss SimpleRNN



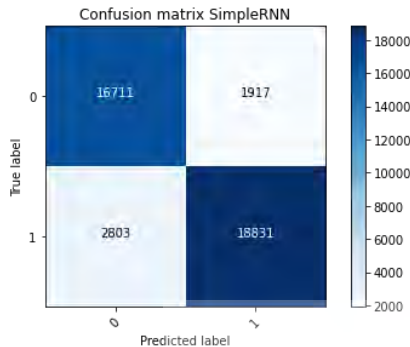
Gambar 7. Training dan Validasi Loss LSTM



Gambar 8. Training dan Validasi Loss Bi-LSTM

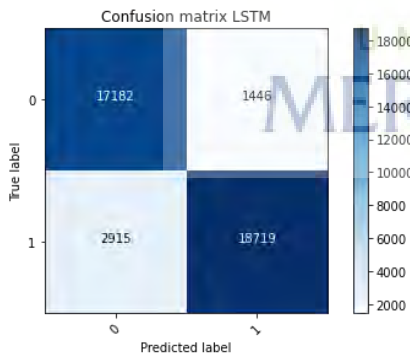
### C. Evaluasi

Setelah dilakukan pengujian menggunakan data latih, selanjutnya melakukan testing ke-tiga model varian RNN dengan penambahan FastText menggunakan hasil dari pengujian sebelumnya untuk bahan evaluasi. Evaluasi dilakukan menggunakan data test yang telah diberi dua label kelas yaitu positif dengan angka 1 dan negatif dengan angka 0. Label kelas akan dibandingkan jumlahnya menggunakan *Confusion Matrix* yang memiliki dua bagian sebagai bahan perbandingan yaitu total nilai prediksi dan total nilai benar. *Confusion Matrix* dari ke-tiga model varian RNN dapat dilihat pada gambar 9, 10, dan 11.



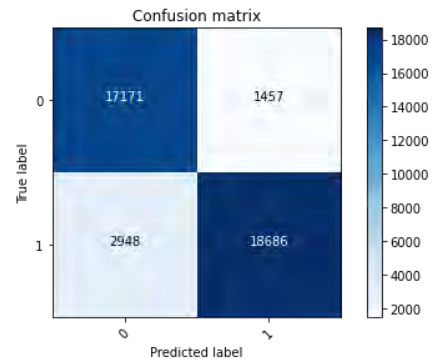
Gambar 9. Confusion Matrix SimpleRNN

Dari confusion matrix pada gambar 9 dapat dijelaskan bahwa model melakukan klasifikasi dengan benar sebanyak 16.711 data kelas positif (True Positive) dan 18.831 sebagai data kelas Negatif (True Negative). Model salah melakukan prediksi sebanyak 2.803 data ke dalam label kelas negatif yang seharusnya label kelas positif (False Negative), serta salah melakukan prediksi sebanyak 1.917 data ke dalam label kelas positif yang seharusnya label kelas negatif (False Positive).



Gambar 10. Confusion Matrix LSTM

Pada gambar 10 dapat dijelaskan model benar melakukan klasifikasi sebanyak 17.182 data sebagai kelas positif (TP) dan 18.719 data sebagai kelas negatif (TN). Selain itu, model melakukan kesalahan dalam memprediksi 2.915 data ke dalam label negatif yang seharusnya positif (FN), serta salah dalam melakukan prediksi sebanyak 1.446 data ke dalam label positif yang seharusnya negatif (FP)



Gambar 11. Confusion Matrix Bi-LSTM

Dapat dijelaskan mengenai gambar 11 bahwa ditunjukkan model benar melakukan klasifikasi sebanyak 17.171 data sebagai kelas positif (TP) dan 18.686 data sebagai kelas negatif (TN). Selain itu, model melakukan kesalahan dalam memprediksi 2.948 data ke dalam label negatif yang seharusnya positif (FN), serta salah dalam melakukan prediksi sebanyak 1.457 data ke dalam label positif yang seharusnya negatif (FP).

## V. KESIMPULAN

Berdasarkan hasil eksperimen yang telah dilakukan, penelitian ini berhasil menganalisis kinerja algoritma Recurrent Neural Network dengan tiga varian nya yaitu SimpleRNN, Long Short Term Memory, dan Bidirectional-LSTM dengan penambahan FastText embeddings terhadap data ulasan PeduliLindungi. Data ulasan diambil dari situs google play store rentan waktu satu tahun yaitu dari 22 Januari 2021 sampai 22 Januari 2022. Data yang dipakai sebanyak 201.307 ulasan setelah melalui *pre-processing* dengan diberi dua label kelas Positif dan Negatif. Pengguna aplikasi lebih banyak yang memberi ulasan positif sebanyak 108.564, sedangkan untuk ulasan negatif sebanyak 92.743.

Dari hasil pengujian klasifikasi menggunakan tiga model varian RNN, hasil akurasi terbaik didapat oleh model LSTM dan Bi-LSTM dengan penambahan FastText Embeddings sebesar 89%. Sedangkan model yang diuji tanpa FastText mengalami penurunan akurasi 1-2%. Terbukti bahwa penambahan *word embedding* FastText sebagai input pada pemodelan algoritma klasifikasi dapat menambah kinerja dari algoritma. Hal ini, karena FastText memiliki kelebihan yaitu dapat menerjemahkan kata yang tidak dikenali sebelumnya menjadi beberapa karakter kata.

## DAFTAR PUSTAKA

- [1] R. Tosepu, D. S. Effendy, and L. O. A. I. Ahmad, "The First Confirmed Cases Of COVID-19 In Indonesian Citizens," *Public Heal. Indones.*, vol. 6, no. 2, pp. 70–71, 2020, doi: 10.3991/ijoe.v16i04.13531.
- [2] G. Peretto, S. Sala, and A. L. P. Caforio, "The origin, transmission and clinical therapies on coronavirus disease 2019 (COVID-19) outbreak – an update on the status," *Eur. Heart J.*, vol. 7, no. 11, pp. 1–10, 2020, doi: 10.1093/eurheartj/ehaa396.
- [3] R. N. Putri, "Indonesia dalam Menghadapi Pandemi Covid-19," *J. Ilm. Univ. Batanghari Jambi*, vol. 20, no. 2, pp. 705–709, 2020, doi: 10.33087/jiubj.v20i2.1010.
- [4] F. F. Rachman and S. Pramana, "Analisis Sentimen Pro dan Kontra Masyarakat Indonesia tentang Vaksin COVID-19 pada Media

Sosial Twitter,” *Indones. Heal. Inf. Manag. J.*, vol. 8, no. 2, pp. 100–109, 2020, [Online]. Available: <https://inohim.esaunggul.ac.id/index.php/INO/article/view/223/175>

- [5] C. E. Putri and R. E. Hamzah, “Aplikasi Pedulilindungi Mitigasi Bencana COVID-19 Di Indonesia,” *J. Pustaka Komun.*, vol. 4, no. 1, pp. 66–78, 2021, doi: 10.32509/pustakom.v4i1.1321.
- [6] A. Mustopa, Hermanto, Anna, E. B. Pratama, A. Hendini, and D. Risdiansyah, “Analysis of User Reviews for the PeduliLindungi Application on Google Play Using the Support Vector Machine and Naive Bayes Algorithm Based on Particle Swarm Optimization,” in *International Conference on Informatics and Computing (ICIC)*, 2020, vol. 2, doi: 10.1109/ICIC50835.2020.9288655.
- [7] M. Rezki, D. N. Kholifah, M. Faisal, Priyono, and R. Suryadithia, “Analisis Review Pengguna Google Meet dan Zoom Cloud Meeting Menggunakan Algoritma Naïve Bayes,” *J. Infortech*, vol. 2, no. 2, pp. 264–270, 2020, doi: 10.31294/infortech.v2i2.9286.
- [8] F. P. Rachman and H. Santoso, “Perbandingan Model Deep Learning untuk Klasifikasi Sentiment Analysis dengan Teknik Natural Language Processing,” *J. Teknol. dan Manaj. Inform.*, vol. 7, no. 2, pp. 113–121, 2021, doi: 10.26905/jtmi.v7i2.6506.
- [9] F. Faturohman, B. Irawan, and C. Setianingsih, “Analisis Sentimen Pada Bpjs Kesehatan Menggunakan Recurrent Neural Network,” in *e-Proceeding of Engineering*, 2020, vol. 7, no. 2, pp. 4545–4552.
- [10] N. M. Alharbi, N. S. Alghamdi, E. H. Alkhamash, and J. F. Al Amri, “Evaluation of Sentiment Analysis via Word Embedding and RNN Variants for Amazon Online Reviews,” *Hindawi*, vol. 2021, no. 5536560, pp. 1–10, 2021, doi: 10.1155/2021/5536560.
- [11] A. C. Pandey, M. Garg, and S. Rajput, “Enhancing Text Mining Using Deep Learning Models,” in *International Conference on Contemporary Computing*, 2019, pp. 1–5, doi: 10.1109/IC3.2019.8844895.
- [12] N. A. Hasanah, N. Suciati, and D. Purwitasari, “Pemantauan Perhatian Publik terhadap Pandemi COVID-19 melalui Klasifikasi Teks dengan Deep Learning,” *J. RESTI Rekayasa Sist. dan Teknol. Inf.*, vol. 5, no. 3, pp. 193–202, 2021.
- [13] A. E. Sari, S. Widowati, and K. M. Lhaksmana, “Klasifikasi Ulasan Pengguna Aplikasi Mandiri Online di Google Play Store dengan Menggunakan Metode Information Gain dan Naive Bayes Classifier,” in *e-Proceeding of Engineering*, 2019, vol. 6, no. 2, pp. 9143–9157.
- [14] M. Lestandy, A. Abdurrahim, and L. Syafa’ah, “Analisis Sentimen Tweet Vaksin COVID-19 Menggunakan Recurrent Neural Network dan Naive Bayes,” *J. RESTI (Rekayasa Sist. dan Teknol. Informasi)*, vol. 5, no. 4, pp. 802–808, 2021.
- [15] P. Asmi and M. S. Sanaj, “Toxic Speech Classification via Deep Learning using Combined Features from BERT & FastText Embedding,” in *International Journal of Engineering Research & Technology (IJERT), ICCIDT - 2021 Conference Proceedings*, 2021, vol. 9, no. 7, pp. 68–71.
- [16] M. Y. Day and Y. Da Lin, “Deep learning for sentiment analysis on google play consumer review,” in *Proceedings - 2017 IEEE International Conference on Information Reuse and Integration, IRI 2017*, 2017, vol. 2017-Janua, pp. 382–388, doi: 10.1109/IRI.2017.79.
- [17] M. A. Riza and N. Charibaldi, “Emotion Detection in Twitter Social Media Using Long Short-Term Memory (LSTM) and Fast Text,” *Int. J. Artif. Intell. Robot.*, vol. 3, no. 1, pp. 15–26, 2021, doi: 10.25139/ijair.v3i1.3827.
- [18] E. I. Setiawan and I. Lestari, “Stance Classification Pada Berita Berbahasa Indonesia Berbasis Bidirectional LSTM,” *J. Intell. Syst. Comput.*, vol. 3, no. 1, pp. 41–48, 2021, doi: 10.52985/insyst.v3i1.148.
- [19] G. Borah, D. Nimje, G. JananiSri, K. P. Bharath, and M. R. Kumar, “Sentiment Analysis of Text Classification Using RNN Algorithm,” in *Proceedings of International Conference on Communication and Computational Technologies (ICCCT)*, 2021, pp. 561–571, doi: 10.1007/978-981-16-3246-4\_44.

## KERTAS KERJA

### Ringkasan

Kertas kerja ini merupakan material kelengkapan artikel jurnal dengan judul “Analisa Kinerja RNN dengan FastText Embedding terhadap Ulasan PeduliLindungi di Masa COVID-19”. Kertas kerja ini berisi semua material hasil penelitan Tugas Akhir. Di dalam kertas kerja ini disajikan beberapa bagian yang terdiri dari literature review, dataset yang digunakan, tahapan eksperimen, dan hasil eksperimen secara keseluruhan.

Bagian I membahas mengenai literature review yang berisi artikel jurnal yang menjadi dasar atau landasan dalam penelitian ini. Bagian II menjelaskan mengenai source code yang digunakan pada penelitian ini. Bagian III menjelaskan mengenai dataset yang digunakan, meliputi penjelasan dan sumber data. Bagian IV memuat tahapan eksperimen yang disajikan dalam gambar dengan penjelasan dari setiap tahapan. Bagian V merupakan bagian terakhir dari kertas kerja ini yang menjelaskan hasil keseluruhan dari eksperimen yang telah dilakukan, meliputi penjelasannya.

