

**ANALISIS SENTIMEN PADA ULASAN APLIKASI  
TOKOPEDIA DI *GOOGLE PLAY STORE*  
MENGUNAKAN METODE *LONG SHORT TERM  
MEMORY***

**SKRIPSI**



Afifah Faadilah

11150940000007

**PROGRAM STUDI MATEMATIKA**

**FAKULTAS SAINS DAN TEKNOLOGI**

**UIN SYARIF HIDAYATULLAH JAKARTA**

**2020 M / 1441 H**

**ANALISIS SENTIMEN PADA ULASAN APLIKASI  
TOKOPEDIA DI *GOOGLE PLAY STORE* MENGGUNAKAN  
METODE *LONG SHORT TERM MEMORY***

**Skripsi**

Diajukan kepada

Universitas Islam Negeri Syarif Hidayatullah Jakarta

Fakultas Sains dan Teknologi

Untuk Memenuhi Salah Satu Persyaratan Dalam

Memperoleh Gelar Sarjana Matematika (S.Mat)

Oleh :

Afifah Faadilah

11150940000007

**PROGRAM STUDI MATEMATIKA**

**FAKULTAS SAINS DAN TEKNOLOGI**

**UIN SYARIF HIDAYATULLAH JAKARTA**

**2020 M / 1441 H**

## PERNYATAAN

DENGAN INI SAYA MENYATAKAN BAHWA SKRIPSI INI BENAR-BENAR HASIL KARYA SAYA SENDIRI YANG BELUM PERNAH DIAJUKAN SEBAGAI SKRIPSI ATAU KARYA ILMIAH PADA PERGURUAN TINGGI ATAU LEMBAGA MANAPUN.

Jakarta, Januari 2020



Afifah Faadilah

NIM. 11150940000007

The logo of Universitas Islam Negeri (UIN) is displayed in a large, stylized blue font. The logo is enclosed within a green rectangular border that has a decorative, slightly wavy top and bottom edge. The background of the page features a faint, light blue circular watermark with a grid pattern.

## LEMBAR PENGESAHAN

Skripsi ini berjudul “Analisis sentimen pada ulasan aplikasi Tokopedia di *google play store* menggunakan *Long Short Term Memory*” yang ditulis oleh Afifah Faadilah, NIM. 11150940000007 telah diuji dan dinyatakan lulus dalam sidang Munaqosah Fakultas Sains dan Teknologi Universitas Islam Negeri Syarif Hidayatullah Jakarta pada hari Selasa, tanggal 16 Januari 2020. Skripsi ini telah diterima untuk memenuhi salah satu persyaratan dalam memperoleh gelar sarjana strata satu (S1) Program Studi Matematika.

Menyetujui,

Pembimbing I

Dr. Taufik Sutanto, MScTech  
NIP. 197905302006041002

Pembimbing II

Dr. Nina Fitriyati, M.Kom  
NIP. 197604142006042001

Penguji I

Muhaza Liebenlito, M.Si  
NIDN. 200309802

Penguji II

Mahmudi, M.Si  
NIDN. 2029048801

Mengetahui,

Dekan Fakultas Sains dan Teknologi

Prof. Dr. Liliy Surayya Eka Putri, M.Env.Stud  
NIP. 196904042005012005

Ketua Program Studi  
Matematika

Dr. Suma'inna, M.Si  
NIP. 197912082007012015

## **PERSEMBAHAN DAN MOTTO**

### **PERSEMBAHAN**

**Kepada Allah SWT**

Yang maha pemberi keajaiban dikehidupan penulis

**Kepada Orang Tua, Kakak, Adik**

Terimakasih atas apa yang sudah kalian berikan kepada penulis. Maaf jika diri ini masih belum membuat kalian bangga.

### **MOTTO**

"Berbuat untuk sebuah harapan, yang tidak lagi dikeluhkan tetapi diperjuangkan."

-Najwa Syihab

## KATA PENGANTAR



Puji beserta syukur penulis ucapkan kehadiran Allah Yang Maha Esa atas segala rahmat dan *hidayah*-Nya sehingga penulis dapat menyelesaikan skripsi yang berjudul “**Analisis Sentimen Pada Ulasan Aplikasi Tokopedia Di Google Play Store Menggunakan Metode Long Short Term Memory**”.

Skripsi ini merupakan persyaratan bagi penulis untuk bisa memperoleh gelar sarjana dan semoga gelar ini dapat bermanfaat untuk orang banyak, keluarga maupun diri sendiri .

Tentu terbentuknya skripsi ini tidak akan berjalan dengan semestinya tanpa doa, arahan, kritikan, dan harapan dari beberapa pihak. Oleh karena itu, penulis ingin menyampaikan rasa terima kasih penulis kepada :

1. Ibu Prof. Dr. Lily Surayya Eka Putri, M.Env.Stud, selaku Dekan Fakultas Sains dan Teknologi Universitas Islam Negeri Syarif Hidayatullah Jakarta.
2. Ibu Dr. Suma'inna, M.Si, selaku Ketua Program Studi Matematika Fakultas Sains dan Teknologi UIN Jakarta.
3. Ibu Irma Fauziah, M.Sc, selaku Sekretaris Program Studi Matematika Fakultas Sains dan Teknologi UIN Jakarta dan sebagai penguji I yang telah memberikan masukan kepada penulis dalam menyelesaikan skripsi ini.
4. Bapak Dr. Taufik Sutanto, MScTech, selaku dosen pembimbing I yang telah memberikan banyak arahan, saran serta bantuan dengan cara yang bijaksana dan selalu membuat suasana setiap bimbingan terasa menyenangkan
5. Ibu Dr. Nina Fitriyati, M.Kom, sebagai dosen pembimbing akademik dan sebagai pembimbing II yang telah memberikan banyak motivasi, saran, dan bantuan kepada penulis selama kuliah dan dalam menyelesaikan skripsi ini.

6. Bapak Muhaza Liebenlito, M.Si, sebagai pembimbing II yang telah memberikan banyak sekali saran dan bantuan kepada penulis dalam menyelesaikan skripsi ini.
7. Bapak Mahmudi, M.Si, sebagai penguji II yang telah memberikan masukan kepada penulis dalam menyelesaikan skripsi ini.
8. Papah Suparno, Ibu Wiwik dan mama Henny Kuntarti selaku orang tua penulis yang selalu memberika perhatian, harapan dan doa untuk penulis untuk dapat menyelesaikan skripsi ini dengan baik.
9. Antony Abdulrahman Shaleh, Millatina Yahdiyani dan Hauzan Naufal Selaku Saudara Kandung yang telah memberikan dukungan dan semangat kepada penulis dalam menyelesaikan skripsi ini.
10. Yusran Syuja Farghani, S.Kom, selaku partner hidup yang sangat membantu dalam proses penyelesaian skripsi ini.
11. Para Sahabat penulis, Sabrah, Uu, Nunik, Silver, Refpici, Farah, Putre makasih guys kuliah gua yang haha hihi ini karna kalian.
12. Seluruh teman Matematika 2015, terutama Shinta, Khusnul, Hamid, Aldo yang telah berkontribusi atas perkuliahan dan pembuatan skripsi ini.
13. Seluruh pihak yang telah membantu penulis dalam menyelesaikan skripsi ini yang tidak bisa penulis sebutkan satu-persatu tanpa mengurangi rasa hormat.

Untuk segala kekurangan dimohon kritik dan saran yang membangun dari pembaca untuk perbaikan di masa yang akan datang. Terakhir, semoga skripsi ini bermanfaat bagi penulis dan pembaca sekalian.

Ciputat, Januari 2020

Penulis



**LEMBAR PERNYATAAN PERSETUJUAN  
PUBLIKASI KARYA ILMIAH UNTUK KEPENTINGAN AKADEMIS**

Yang bertanda tangan dibawah ini :

Nama : Afifah Faadilah

NIM : 11150940000007

Program Studi : Matematika Fakultas Sains dan Teknologi

Demi pengembangan ilmu pengetahuan, saya menyetujui untuk memberikan Hak Bebas Royalti Non-Eksklusif (*Non-Exclusive-Free Right*) kepada Program Studi Matematika Fakultas Sains dan Teknologi UIN Syarif Hidayatullah Jakarta atas karya ilmiah saya yang berjudul :

**“Analisis Sentimen Pada Ulasan Aplikasi Tokopedia Di *Google Play Store* Menggunakan Metode *Long Short Term Memory*”**

beserta perangkat yang diperlukan (bila ada). Dengan Hak Bebas Royalti Non-Eksklusif ini, Program Studi Matematika Fakultas Sains dan Teknologi UIN Syarif Hidayatullah Jakarta berhak menyimpan, mengalihmedia/formatkan, mengelolanya dalam bentuk pangkalan data (database), mendistribusikannya, dan menampilkan/mempublikasikannya di internet dan media lain untuk kepentingan akademis tanpa perlu meminta izin dari saya selama tetap mencantumkan nama saya sebagai penulis/pencipta dan sebagai pemilik Hak Cipta. Segala bentuk tuntutan hukum yang timbul atas pelanggaran Hak Cipta karya ilmiah ini menjadi tanggung jawab saya sebagai penulis.

Demikian pernyataan ini yang saya buat dengan sebenarnya.

Dibuat di Jakarta

Pada tanggal : 16 Januari 2020

Yang membuat pernyataan

METERAI  
TEMPEL

6A382AHF134420428

6000  
RUPIAH

(Afifah Faadilah)



## ABSTRAK

**Afifah Faadilah**, Analisis Sentimen Pada Ulasan Aplikasi Tokopedia Di *Google Play Store* Menggunakan Metode *Long Short Term Memory*, dibawah bimbingan **Dr. Taufik Sutanto, M.ScTech** dan **Dr. Nina Fitriyati, M.Kom**

Tujuan penelitian ini adalah untuk menganalisis sentimen pada ulasan Aplikasi Tokopedia. Jumlah ulasan yang akan dianalisis sebanyak 3067 ulasan pada periode Oktober 2018 sampai Mei 2019. Data dilakukan pelatihan untuk mendapatkan model yang optimal dan dapat memprediksi sentimen dengan model *Long Short Term Memory*. Untuk mendapatkan model yang terbaik dilakukan pengujian parameter seperti jumlah *neuron* dan fungsi aktivasi dengan menggunakan data training sebesar 50% dan validasi 30% dari dataset. Dari pengujian tersebut menghasilkan akurasi yang cukup baik pada parameter 400 jumlah *neuron* dan dengan fungsi aktivasi *sigmoid*. Dari parameter terbaik tersebut selanjutnya model diuji pada 20% dari dataset dan menghasilkan akurasi 93.32%, *presisi* 95.17% dan nilai *recall* 97.15%. Pada masing-masing kelas menghasilkan opini yang dapat dijadikan bahan evaluasi bagi Tokopedia diantaranya adalah pada kelas negatif didapatkan opini ketidakpuasan pelanggan terhadap pelayanan *customer service* dan ketidakbermanfaatan aplikasi saat di *update* ke versi yang baru dan pada kelas positif didapatkan opini kepuasan pelanggan terhadap kemudahan aplikasi saat digunakan.

**Kata kunci:** *Long Short Term Memory*, Tokopedia, Analisis Sentimen, Google Playstore

## ABSTRACT

**Afifah Faadilah**, Sentiment Analysis on Tokopedia Application Review on the Google Play Store Using the Long Short Term Memory Method, under guidance, Supervised by **Dr. Taufik Sutanto, M.ScTech** dan **Dr. Nina Fitriyati, M.Kom**

The purpose of this research is to sentiment analysis on the Tokopedia Application review with a total of 3067 reviews. and will be trained to get an optimal model for predicting sentiments on the Tokopedia review with the Short Term Memory model. To get the best model, three parameters were tested as much as the number of neurons, the value of learning rate and the activation function using training data for 80% and 30% validation of the dataset. From these tests have a good accuracy on the parameter 400 number of neurons and with sigmoid activation function. From the best parameters then the model was tested on 20% of the dataset and the result an accuracy of 93.32%, precision 95.17% and recall 97.15%. the best model has been found to predict sentiment class from Tokopedia reviews. In each class produce opinions that can be used as evaluation material for Tokopedia including those in the negative class opinions obtained customer dissatisfaction with customer service and application uselessness when updated to the new version and in positive classes obtained opinions on customer satisfaction regarding the ease of application when used

**Keywords :** *Long Short Term Memory*, Tokopedia, Sentiment Analysis, Google Playstore

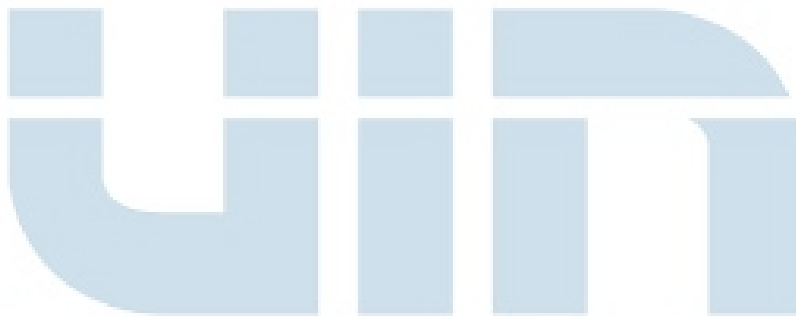
## DAFTAR ISI

<b>ANALISIS SENTIMEN PADA ULASAN APLIKASI TOKOPEDIA DI <i>GOOGLE PLAY STORE</i> MENGGUNAKAN METODE <i>LONG SHORT TERM MEMORY</i>.....</b>	<b>i</b>
<b>PERNYATAAN.....</b>	<b>Error! Bookmark not defined.</b>
<b>LEMBAR PENGESAHAN .....</b>	<b>ii</b>
<b>PERSEMBAHAN DAN MOTTO.....</b>	<b>iii</b>
<b>KATA PENGANTAR.....</b>	<b>v</b>
<b>LEMBAR PERNYATAAN PERSETUJUAN .....</b>	<b>Error! Bookmark not defined.</b>
<b>ABSTRAK .....</b>	<b>vii</b>
<b>ABSTRACT .....</b>	<b>ix</b>
<b>DAFTAR ISI.....</b>	<b>x</b>
<b>DAFTAR TABEL .....</b>	<b>xii</b>
<b>DAFTAR GAMBAR.....</b>	<b>xiii</b>
<b>BAB I PENDAHULUAN.....</b>	<b>1</b>
1.1. Latar Belakang .....	1
1.2. Rumusan Masalah .....	5
1.3. Batasan Masalah.....	5
1.4. Tujuan Penelitian.....	5
1.5. Manfaat Penelitian.....	6
<b>BAB II TINJAUAN PUSTAKA DAN LANDASAN TEORI.....</b>	<b>6</b>
2.2. Artificial Neural Network .....	6
2.3. Recurrent Neural Network .....	7
2.4. Fungsi Aktivasi.....	9
2.5. Loss Function .....	10
2.6. Optimasi Adam.....	11
2.7. Batch Size dan Epoch.....	12
<b>BAB III METODOLOGI PENELITIAN .....</b>	<b>14</b>
3.1. Sumber Data .....	14
3.2. Pelabelan .....	17
3.3. <i>Preprocessing</i> .....	18
3.4. Data Training dan Data Testing .....	20
3.5. Lapisan Word Embedding .....	20

3.6.	Long Short Term Memory .....	24
3.7.	Lapisan Fully Connected .....	26
3.8.	Evaluasi Confusion Matrix .....	27
3.9.	Alur Penelitian .....	29
<b>BAB IV HASIL DAN PEMBAHASAN .....</b>		<b>37</b>
4.1.	Hasil Preprosesing dan Text Analytics .....	37
4.2.	Arsitektur Jaringan .....	39
4.3.	Pengujian Parameter .....	40
4.3.1.	Pengujian Jumlah Neuron .....	40
4.3.2.	Pengujian Fungsi Aktivasi Pada Lapisan Output .....	41
4.4.	Hasil Training Data .....	41
4.5.	Evaluasi Data Uji .....	42
4.6.	Visualisasi Kata Setiap Kelas .....	44
4.7.	Perbandingan Prediksi Ulasan dengan Rating .....	45
4.8.	Perbandingan Model menggunakan Berbahasa Indonesia .....	46
<b>BAB V KESIMPULAN .....</b>		<b>48</b>
5.1.	Kesimpulan .....	48
5.2.	Saran .....	48
<b>REFERENSI .....</b>		<b>49</b>
<b>LAMPIRAN .....</b>		<b>53</b>

## DAFTAR TABEL

<b>Tabel 3.1.</b> Jumlah Ulasan Tiap Rating .....	15
<b>Tabel 3.2</b> Hasil Web Scraping Data Ulasan .....	16
<b>Tabel 3.3</b> Ulasan Sebelum dan Sesudah di Tokenizer .....	19
<b>Tabel 3.4</b> Hasil Pemberian Indeks pada Ulasan .....	21
<b>Tabel 3.5</b> Hasil Pad_Sequence .....	22
<b>Tabel 3.6</b> Hasil Word Embedding Layer .....	23
<b>Tabel 3.7.</b> Confusion Matrix .....	27
<b>Tabel 3.8.</b> Rumus Evaluasi Klasifikasi .....	28
<b>Tabel 4.1</b> Hasil Preprocessing dan Labeling .....	37
<b>Tabel 4.2</b> Total Parameter .....	39
<b>Tabel 4.3</b> Hasil Pengujian Jumlah Neuron .....	40
<b>Tabel 4.4</b> Hasil Pengujian Fungsi Aktivasi .....	41
<b>Tabel 4.5</b> Hasil Confusion Matrix .....	43
<b>Tabel 4.6</b> Evaluasi Confussion Matrix .....	43



## DAFTAR GAMBAR

<b>Gambar 1.1.</b> Presentasi Pengguna Mobile Operator System.....	2
<b>Gambar 2.1.</b> Arsitektur ANN[12].....	6
<b>Gambar 2.2</b> Arsitektur RNN[19].....	7
<b>Gambar 2.3</b> Ilustrasi Perhitungan Forward Propagation .....	8
<b>Gambar 3.1</b> Jumlah Ulasan Tokopedia Tiap Bulan.....	15
<b>Gambar 3.2</b> Jumlah Label Tiap Bulan.....	17
<b>Gambar 3.3</b> Word Embedding.....	21
<b>Gambar 3.4.</b> Struktur Jaringan Long Short Term Memory (LSTM).....	24
<b>Gambar 3.5</b> Lapisan Fully Connected[27] .....	27
<b>Gambar 3.6.</b> Alur Penelitian .....	29
<b>Gambar 4.1</b> Hasil Wordlink dari Label Positif.....	38
<b>Gambar 4.2</b> Hasil Wordlink dari Label Negatif .....	39
<b>Gambar 4.3</b> Nilai Loss Training dan Validasi (a), Akurasi Training dan Validasi(b).....	42
<b>Gambar 4.4</b> Wordcloud dikelas Positif (a), Wordlink dikelas positif(b).....	44
<b>Gambar 4.5</b> Wordcloud ud dikelas Negatif (a), Wordlink Dikelas Negatif(b) ...	45
<b>Gambar 4.6</b> Perbandingan Rating Negatif dengan Prediksi (a), Perbandingan Rating Positif dengan Prediksi (b) .....	46

## **BAB I**

### **PENDAHULUAN**

Peranan Tokopedia sebagai salah satu situs *e-commerce* di Indonesia semakin meningkat. Pada akhir 2018 Tokopedia telah mencapai jumlah pengunjung terbanyak setelah bukalapak dengan total kunjungan terhadap Tokopedia sebanyak 169 juta pengunjung [1]. Dengan jumlah pengunjung tersebut sangat variatif kegiatan yang dilakukan oleh pengunjung Tokopedia, seperti melakukan menjual produk, membeli produk dan memberikan ulasan produk pada Tokopedia. Peningkatan kunjungan Tokopedia tidak terlepas dari pemberian promo yang menarik oleh Tokopedia, sehingga menjadi sebuah faktor ketertarikan pengguna untuk melakukan transaksi jual beli produk di Tokopedia. Peningkatan pengguna dalam menggunakan aplikasi Tokopedia menuai ulasan berupa kritik, baik dari segi pelayanan maupun cara penggunaan aplikasi.

Pada bab ini akan dijelaskan terlebih dahulu tentang latar belakang yang mendasari penelitian ini, serta tujuan, rumusan masalah, batasan masalah dan manfaat penulisan sebagai tumpuan bagi penulis.

#### **1.1. Latar Belakang**

*“Padahal Allah telah menghalalkan jual beli dan mengharamkan riba”* (QS. Al Baqarah: 275)

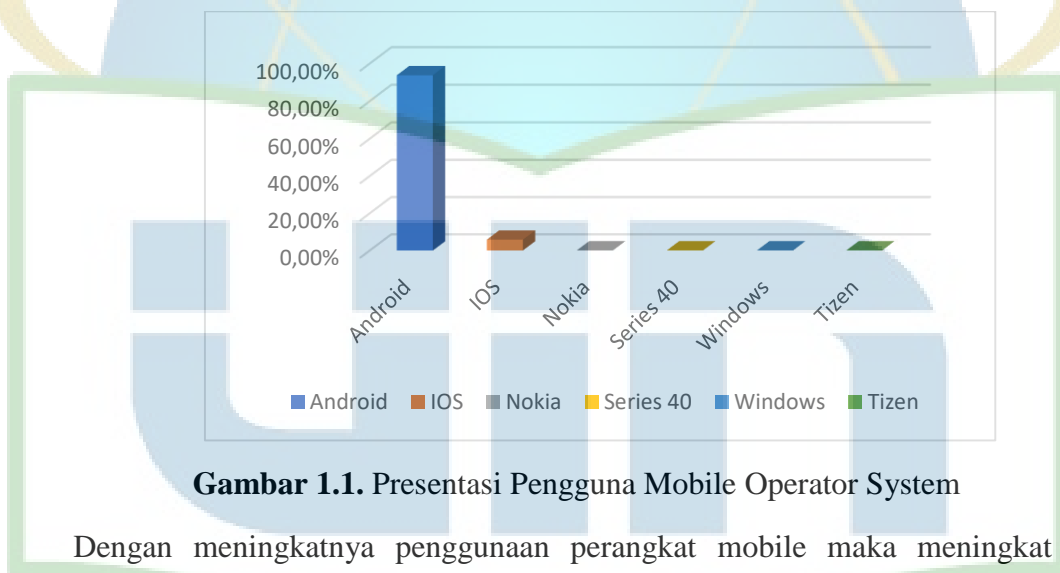
*“Janganlah kalian saling memakan harta sesamamu dengan jalan yang batil, kecuali dengan jalan perniagaan yang timbul dari kerelaan di antara kalian”* (Q.S An-Nissa: 29)

Dari terjemahan Q.S An-Nissa: 29 menceritakan kegiatan jual beli dengan cara yang baik dan melarang untuk melakukan kegiatan jual beli dengan cara yang batil yaitu dengan cara yang bertentangan dengan hukum islam. Seperti tidak menipu, tidak berbohong dan tidak saling merugikan satu sama lain. Pada zaman sekarang hal



tersebut sering terjadi pada kegiatan jual beli online atau perdagangan melalui media internet salah satunya adalah yang sedang populer dengan sebutan *e-commerce*[2]. Namun keberadaan *e-commerce* memudahkan pengguna dalam melakukan jual beli tanpa harus keluar rumah. Dengan demikian perlu adanya upaya perusahaan *e-commerce* dalam membangun kepercayaan pengguna dalam menggunakan aplikasi. Agar rantai jual beli berlangsung dengan baik sesuai syariat islam.

Pada Tabel 1.1 menggambarkan bahwa *mobile operating system* pada OS android menghasilkan presentasi tertinggi dibandingkan dengan penggunaan Operating System yang lainnya, yaitu dengan data pengguna *Android* 93,75%, pengguna *IOS* 5,82%, pengguna *Nokia* 0,08%, pengguna *Series 40* 0,07%, pengguna *Windows* 0,05% dan pengguna *Tizen* 0,07% [3].



**Gambar 1.1.** Presentasi Pengguna Mobile Operator System

Dengan meningkatnya penggunaan perangkat mobile maka meningkat pula kebutuhan layanan *e-commerce*. Perubahan yang terjadi dalam praktik komunikasi juga membawa perubahan bagi beberapa sektor industri, salah satunya adalah sektor industri perdagangan (*e-commerce*). Saat ini sektor industri perdagangan elektronik (*e-commerce*) berkembang dengan pesat. Di Indonesia, sektor *e-commerce* telah mendapat perhatian dan dukungan pemerintah. Hal ini dapat dilihat melalui diterbitkannya Peraturan Presiden (Perpres) Nomor 74 Tahun 2017. Peraturan ini

merupakan bentuk dukungan dan upaya pemerintah dalam mendorong percepatan dan pengembangan *system e-commerce* dan mengoptimalkan pemanfaatan potensi *e-commerce* di Indonesia.[4]

Salah satu situs *e-commerce* terbesar di Indonesia adalah Tokopedia. Tokopedia memungkinkan setiap individu, toko atau bisnis kecil dan bermerek untuk membuka dan mengelola toko online. Saat ini, Tokopedia telah menempatkan posisi ke-8 sebagai situs web yang paling banyak dikunjungi di Indonesia dan posisi teratas dalam kategori *e-commerce*. Tokopedia secara resmi diluncurkan ke publik pada 17 Agustus 2009 di bawah naungan PT. Tokopedia dimiliki oleh William Tanuwijaya dan Leontinus Alpha Edison pada 6 Februari 2009. Sejak diluncurkan secara resmi, PT. Tokopedia telah berhasil menjadi salah satu perusahaan internet di Indonesia dengan pertumbuhan yang sangat pesat.[5]

App markets seperti Google Playstore menyediakan mekanisme baru untuk distribusi perangkat lunak, dimana pengguna memperoleh aplikasi langsung dari pengembang dan juga proses ulasan pengguna aplikasi dapat langsung dilakukan melalui komentar. Ulasan pada Google Playstore memiliki rating dari 1 hingga 5. Namun sering kali pengguna memberikan rating tidak sesuai dengan ulasannya sehingga hal ini belum cukup menggambarkan kualitas dari aplikasi. Ulasan berbentuk kalimat lebih menggambarkan bagaimana tanggapan pengguna terhadap aplikasi. Hal ini dapat memungkinkan bahwa ulasan tersebut mempengaruhi pengunjung atau pengguna yang baru akan mengunduh aplikasi Tokopedia.[6]

Meningkatnya pengunjung dan pengguna baru yang akan mengunduh aplikasi Tokopedia berkaitan dengan ulasan pada aplikasi Tokopedia, ulasan tersebut berguna untuk meningkatkan kualitas layanan pada aplikasi Tokopedia yang bertujuan untuk membangun kepercayaan pengguna. Pada penelitian ini ulasan tersebut digunakan sebagai data penelitian, dengan menggunakan data ulasan bahasa Inggris maka penelitian ini mempunyai target sasaran yaitu para kaum milenial, dimana kaum

milennial 15 sampai 19 tahun, tercatat sebanyak 91 persen telah menggunakan internet. Kemudian rentang 20 hingga 24 tahun sebesar 88,5 persen, lalu rentang usia 25 sampai 29 tahun sebanyak 82,7 persen[7]. Hal ini juga ditunjukkan dengan hasil survey Ipsos bahwa Layanan e-commerce Tokopedia dan Shopee tercatat merupakan layanan e-commerce yang paling banyak dikunjungi responden milenial. Kaum milenial juga terkenal dengan kemampuan berbahasa asing. Dengan ini penelitian ini berguna bagi tokopedia meningkatkan kualitasnya berdasarkan ulasan bahasa inggris untuk mendapatkan pasar salah satunya pada kaum milenial[8].

Untuk memudahkan pengelolaan data ulasan, penelitian ini menggunakan analisis Sentimen. Analisis sentimen merupakan sebuah metode yang digunakan untuk memahami, mengekstrak data opini, dan mengolah data tekstual secara otomatis untuk mendapatkan sebuah sentiment yang terkandung dalam sebuah opini. Pada analisis sentiment dapat terdiri dari opini positif dan opini negatif sehingga dengan analisis sentimen pihak Tokopedia dapat mengetahui respon masyarakat terhadap suatu pelayanan atau produk, melalui feedback masyarakat atau pun para ahli[9].

Menurut penelitian sebelumnya analisis sentiment dapat diklasifikasikan dengan baik menggunakan salah satu model Deep Learning yaitu Long Short Term Memory[10]. LSTM pertama kali dikenal pada tahun 1997 [11] memiliki struktur yang disebut gerbang (gates) berguna untuk mengatur aliran informasi. Selain itu Gerbang ini dapat mempelajari data mana yang penting untuk disimpan dan dilupakan sehingga informasi yang relevan dapat diteruskan sepanjang urutan untuk menghasilkan prediksi.[12] LSTM juga menghasilkan akurasi yang cukup baik untuk prediksi sentiment dibandingkan dengan beberapa model lain seperti yang terdapat di Machine Learning[13]

Penelitian terkait analisis sentiment pada data ulasan juga sudah diteliti sebelumnya dari mulai data ulasan hotel[14], ulasan film[15], ulasan 3 aplikasi dari App Store dan 4 Google Playstore [13] serta ulasan Web JD.Com[16]. Dari beberapa penelitian

tersebut ada yang menggunakan model dari machine learning hingga deep learning dengan akurasi yang cukup baik. Maka penelitian kali ini akan melakukan hal yang serupa dengan penelitian sebelumnya yaitu melakukan analisis sentiment menggunakan model Long Short Term Memory namun dengan data berbeda yaitu tentang ulasan dari Tokopedia.

### **1.2. Rumusan Masalah**

Rumusan masalah yang akan dibahas dalam penelitian ini adalah :

1. Bagaimana mendapatkan model Long Short Term Memory yang optimal untuk mengklasifikasikan sentiment ulasan aplikasi Tokopedia?
2. Bagaimana Tokopedia meningkatkan kualitas layanannya berdasarkan ulasan dari Google Playstore?

### **1.3. Batasan Masalah**

Penelitian ini dibuat batasan masalah agar pembahasan tidak keluar dari pembahasan, Batasan masalah sebagai berikut:

1. Data diambil berasal dari ulasan aplikasi Tokopedia di Google Playstore dengan teknik Web Scaping pada periode Oktober 2018 sampai Mei 2019
2. Data yang diolah hanya data teks dari ulasan yang berbahasa Inggris
3. Kelas sentiment yang digunakan hanya kelas sentiment negatif dan positif
4. Parameter yang diuji untuk menentukan model terbaik adalah jumlah neuron dan fungsi aktivasi

### **1.4. Tujuan Penelitian**

Adapun tujuan dalam penelitian ini adalah :

1. Memperoleh model model Long Short Term Memory yang optimal untuk mengklasifikasikan sentiment ulasan aplikasi Tokopedia
2. Mendapatkan bahan evaluasi untuk Tokopedia meningkatkan kualitas layanan mereka dari ulasan Google Playstore

### 1.5. Manfaat Penelitian

Pada penelitian ini diharapkan bermanfaat bagi pembaca mengetahui model long short term memory bekerja untuk membangun model sehingga dapat memprediksi sentimen dari data tekstual khususnya ulasan aplikasi Tokopedia. Manfaat selanjutnya diharapkan bagi pihak Tokopedia untuk mengetahui opini setiap kelas pada ulasan di Google Playstore sehingga menjadikan bahan evaluasi agar Tokopedia dapat mempertahankan predikat *E-commerce* terbaik di Indonesia.



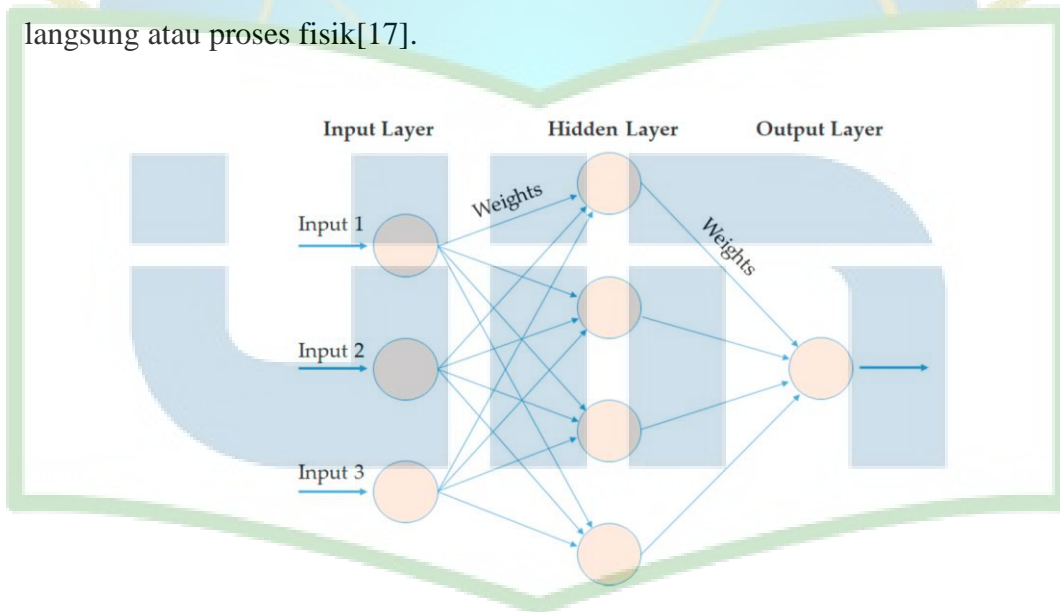
## BAB II

### TINJAUAN PUSTAKA DAN LANDASAN TEORI

Pada bab ini akan dijelaskan tinjauan pustaka dan landasan teori yang mendasari daripada penelitian ini. Teori yang dibahas antara lain mengenai *Artificial Neural Network* (ANN) dan beberapa istilah lain yang digunakan pada penelitian ini.

#### 2.2. Artificial Neural Network

*Artificial Neural Network* (ANN) merupakan sistem komputasi yang terinspirasi oleh sistem cara kerja otak manusia. Model ANN adalah model matematika berbasis data yang memiliki kemampuan untuk menyelesaikan masalah melalui neuron machine learning. Salah satu keuntungan dari ANN adalah mampu mengidentifikasi hubungan nonlinear yang kompleks antara input dan output tanpa memasukkan pengetahuan langsung atau proses fisik[17].



**Gambar 2.1.** Arsitektur ANN[12]

Jenis ANN yang paling umum terdiri dari tiga lapisan, yaitu: Lapisan input, lapisan tersembunyi, dan lapisan keluaran. Seperti yang terlihat pada Gambar 2.1 Arsitektur ANN memiliki lapisan data input dan output independen satu sama lain. Di

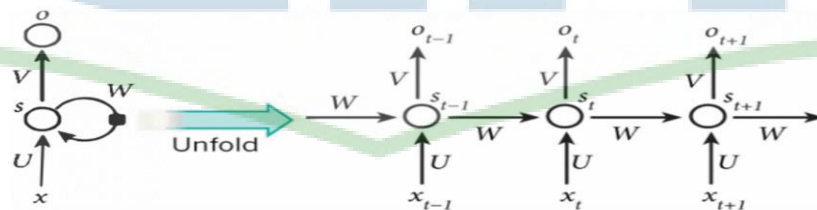
antara lapisan input dan output, satu atau lebih lapisan tersembunyi sering dihubungkan oleh matriks bobot, bias, dan beberapa fungsi aktivasi[12] atau dalam perhitungannya sebagai berikut:

$$s_t = \sum_{i=1}^n x_i \cdot w_{ij} + b_{ij}. \quad (2.1)$$

$$f(s_t) = \frac{1}{1 - e^{-s_t}}. \quad (2.2)$$

### 2.3. Recurrent Neural Network

*Recurrent Neural Network (RNN)* adalah jenis arsitektur Deep Learning yang pemrosesannya dipanggil berulang-ulang untuk memproses input yang biasanya adalah data sekuensial.[18]. RNN adalah model yang efisien untuk analisis sentimen. RNN menggunakan sel memori yang mampu menangkap informasi tentang urutan panjang[19]. Karakteristik data sekuensial adalah sampel diproses dengan suatu urutan (misalnya waktu), dan suatu sampel dalam urutan mempunyai hubungan erat satu dengan yang lain. RNN memproses input secara sekuensial, sampel per sampel. Dalam tiap pemrosesan, output yang dihasilkan tidak hanya merupakan fungsi dari sampel itu saja, tapi juga berdasarkan state internal yang merupakan hasil dari pemrosesan sampel-sampel sebelumnya (atau setelahnya, pada bidirectional RNN)[18]. Arsitektur RNN ditunjukkan pada Gambar 2.3.



**Gambar 2.2** Arsitektur RNN[19]

Jaringan memiliki lapisan input  $x$ , lapisan tersembunyi  $s$  (juga disebut lapisan konteks atau keadaan) dan lapisan keluaran  $y$ . Input ke jaringan dalam waktu  $t$  adalah



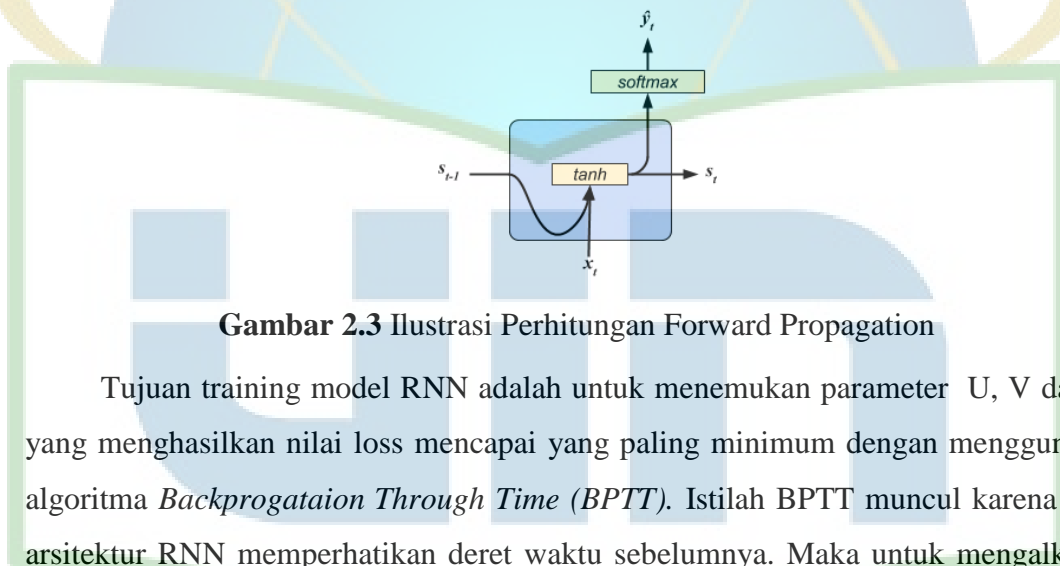
$x_t$ , output dilambangkan sebagai  $y_t$ , dan  $s_t$  adalah keadaan jaringan (lapisan tersembunyi). Input vektor  $x_t$  dibentuk dengan menyatukan vektor  $w$  yang mewakili kata saat ini, dan output dari neuron pada lapisan konteks pada saat  $t - 1$  [18]. Lapisan input, tersembunyi dan output kemudian dihitung sebagai berikut:

$$x_t = w_t + s_{t-1} \cdot \quad (2.3)$$

$$s_t = \tanh(U \cdot x_t + W \cdot s_{t-1}). \quad (2.4)$$

$$y_t = \text{softmax}(V \cdot s_t). \quad (2.5)$$

Tanh dan softmax merupakan fungsi aktivasi, dimana tanh adalah fungsi aktivasi pada lapisan  $s_t$  dan softmax pada lapisan  $y_t$ . Fungsi softmax di lapisan  $y_t$  digunakan untuk memastikan bahwa output membentuk distribusi probabilitas yang valid, yaitu semua output lebih besar dari 0 dan jumlahnya adalah 1[20]



**Gambar 2.3** Ilustrasi Perhitungan Forward Propagation

Tujuan training model RNN adalah untuk menemukan parameter  $U$ ,  $V$  dan  $W$  yang menghasilkan nilai loss mencapai yang paling minimum dengan menggunakan algoritma *Backpropagation Through Time (BPTT)*. Istilah BPTT muncul karena pada arsitektur RNN memperhatikan deret waktu sebelumnya. Maka untuk mengalkulasi gradien pada langkah waktu  $t$ , kita harus menghitung turunan pada langkah  $t-1$ ,  $t-1$ ,  $t-2$ ,  $t-3$  hingga berada pada saat  $t=1$ . Dengan demikian neural network mengingat informasi untuk beberapa langkah waktu dalam lapisan tersembunyi ketika dipelajari oleh BPTT

RNN memiliki permasalahan yaitu short-term memory. Pada sebuah sequence yang lebih panjang, RNN akan mengalami kesulitan membawa informasi dari time step

awal ke time step berikutnya. Pada proses BPPT, RNN mengalami permasalahan yaitu *vanishing gradient problem*. Gradien (gradient) adalah nilai yang digunakan untuk memperbarui bobot dari neural networks. *Vanishing gradient problem* merupakan masalah ketika gradien menyusut seiring proses back propagation melalui waktu. Jika nilai gradien menjadi sangat kecil, maka nilai tersebut tidak akan berkontribusi banyak dalam proses learning. Dalam RNN, lapisan (layer) yang mendapatkan pembaruan nilai gradien kecil akan berhenti melakukan proses learning[21]. Karena lapisan tersebut berhenti melakukan proses learning, RNN dapat melupakan informasi pada sequence yang lebih panjang, sehingga memiliki masalah short-term memory[13]

## 2.4. Fungsi Aktivasi

Fungsi aktivasi merupakan fungsi yang menggambarkan hubungan antara tingkat aktivitas internal (summation function) yang mungkin berbentuk linear ataupun non-linear. Fungsi ini bertujuan untuk menentukan apakah neuron diaktifkan atau tidak. Berikut adalah beberapa fungsi aktivasi yang digunakan:

### 1. Sigmoid

Fungsi aktivasi *sigmoid* merupakan fungsi nonlineear. Masukan untuk fungsi aktivasi ini berupa bilangan real dan output dari fungsi tersebut memiliki range antara 0 sampai 1[22][23]. Berikut ini perhitungan dari fungsi aktivasi sigmoid:

$$g(x) = \frac{1}{1 + e^{-x}}. \quad (2.6)$$

Fungsi sigmoid mentransformasi range nilai dari input  $x$  menjadi antara 0 dan 1. Jika masukannya sangat negatif, maka keluaran yang didapatkan adalah 0, sedangkan jika masukan sangat positif maka nilai keluaran yang didapatkan adalah 1. Fungsi ini memiliki kekurangan yaitu dapat mematikan gradient, ketika aktivasi dari neuron mengeluarkan nilai yang berada pada range 0 atau 1, dimana gradient di wilayah ini hampir bernilai 0. Kemudian output dari sigmoid tidak zero-centered[22][23]

### 2. Tanh

Fungsi aktivasi *Tanh* merupakan fungsi nonlonear. Masukan untuk fungsi aktivasi ini berupa bilangan real dan output dari fungsi tersebut memiliki range antara -1 sampai 1.

1. Berikut ini perhitungan dari fungsi aktivasi tanh:

$$g(x) = \frac{e^x - e^{-x}}{e^{-x} + e^{-x}}. \quad (2.7)$$

Sama seperti fungsi sigmoid, fungsi ini memiliki kekurangan yaitu dapat mematikan gradient, akan tetapi fungsi ini juga memiliki kelebihan yaitu output yang dimiliki fungsi Tanh merupakan zero-centered. Dalam pengaplikasiannya fungsi Tanh lebih menjadi pilihan jika dibandingkan dengan fungsi sigmoid. Fungsi Perlu diketahui fungsi tanh merupakan pengembangan dari fungsi Sigmoid[22].

### 3. Softmax

Fungsi aktivasi *softmax* adalah rentang probabilitas output dengan nilai 0 hingga 1 dan jumlah semua probabilitas akan sama dengan satu. Fungsi softmax yang digunakan untuk model multi-klasifikasi yang akan mengembalikan peluang dari masing-masing kelas dan kelas target akan memiliki probabilitas tinggi[24].

$$g(x) = \frac{e^{x_i}}{\sum_j e^{x_j}}. \quad (2.8)$$

### 2.5. Loss Function

Loss function adalah fungsi yang menghitung perbedaan antara output aktual dan output yang diprediksi. *Loss Function* merupakan fungsi yang menggambarkan kerugian yang terkait dengan semua kemungkinan yang dihasilkan oleh sebuah model. Loss Function ini bekerja ketika model pembelajaran memberikan kesalahan yang harus diperhatikan. Loss Function yang baik memberikan nilai error yang rendah. Pada penelitian ini akan menggunakan Loss Function yang disebut *Binary Cross Entropy*. Binary Cross Entropy adalah fungsi loss standar yang digunakan untuk masalah klasifikasi biner dimana nilai target berada di set  $\{0,1\}$ [25].

$$L = -\frac{1}{N} \sum_{i=1}^N y_i \log(h_{\theta}(x_i)) + (1 - y_i) \log(1 - h_{\theta}(x_i)). \quad (2.9)$$

## 2.6. Optimasi Adam

Optimasi merupakan algoritma yang digunakan untuk meminimalisir ataupun memaksimalkan sebuah fungsi objective (disebut juga error function), yang merupakan sebuah fungsi matematika yang bergantung pada parameter internal yang dapat dipelajari oleh model learning. Parameter internal tersebut digunakan dalam menghitung target value (y) dari sekumpulan predictor (x) yang digunakan dalam model. Sebagai contoh, nilai bobot dan bias dari neural network disebut sebagai parameter internal yang dapat dipelajari dan digunakan dalam menghitung nilai output. Kedua nilai tersebut dipelajari dan diperbarui ke arah solusi optimal dengan meminimalisir nilai loss dari proses training jaringan[11]

Adaptive Moment Estimation (Adam) adalah metode untuk optimasi stokastik efisien yang hanya memerlukan gradien orde pertama dengan kebutuhan memori yang sedikit. Metode ini menghitung tingkat pembelajaran adaptif individu untuk parameter yang berbeda dari perkiraan momen pertama dan kedua dari gradien. Beberapa kelebihan dari Adam adalah bahwa ukuran parameter dari parameter tunggal tidak berubah untuk penskalaan gradien, ukurannya kira-kira dibatasi oleh ukuran langkah hyperparameter, itu tidak memerlukan tujuan stasioner, ia bekerja dengan gradien yang jarang, dan ia secara alami melakukan bentuk ukuran step annealing[26]

Pada Adam, *Learning Rate* yang beradaptasi berdasarkan parameter  $\theta$ . Perubahan learning rate dihitung dari estimasi momentum. Momentum adalah nilai yang dihitung berdasarkan arah dari pelatihan sebelumnya. Dalam Adam, kuadrat gradien disimpan dan disimpan juga rata-rata gradien sebelumnya untuk perhitungan momentum[27].

Langkah algoritma optimasi Adam adalah akan dijelaskan dengan *pseudo code* berikut:

$a$  : Learning Rate  
 $\beta_1, \beta_2 \in [0,1)$  : Exponential decay rates for the moment estimates  
 $f(\theta)$  : Fungsi stokastik dengan parameter  $\theta$   
 $\theta_0$  : Inisial parameter vector  
 $m_0 \leftarrow 0$  (inisialisasi moment vector pertama)  
 $v_0 \leftarrow 0$  (inisialisasi moment vector kedua)  
 $t \leftarrow 0$  (inisialisasi timestep)  
**while**  $\theta_0$  not converged **do**  
     $t \leftarrow t + 1$   
     $g_t \leftarrow \nabla_{\theta} f_t(\theta_{t-1})$  (mendapatkan gradient dari fungsi stokastik pada timestep  $t$ )  
     $m_t \leftarrow \beta_1 \cdot m_{t-1} + (1 - \beta_1) \cdot g_t$  (update bias estimasi momen pertama)  
     $v_t \leftarrow \beta_2 \cdot v_{t-1} + (1 - \beta_2) \cdot g_t^2$  (update bias estimasi momen kedua)  
     $\hat{m}_t \leftarrow \frac{m_t}{1 - \beta_1^t}$  (menghitung bias yang dikoreksi estimasi momen pertama)  
     $\hat{v}_t \leftarrow \frac{v_t}{1 - \beta_2^t}$  (menghitung bias yang dikoreksi estimasi momen pertama)  
     $\theta_t \leftarrow \theta_{t-1} - a \cdot \hat{m}_t / (\sqrt{\hat{v}_t} + \epsilon)$  (update parameter)  
**End while**  
**Return**  $\theta_t$  (Hasil Parameter)

## 2.7. Batch Size dan Epoch

Epoch adalah ketika seluruh dataset sudah melalui proses training pada Neural Network sampai dikembalikan ke awal dalam satu putaran. Dalam Neural Network satu epoch itu terlalu besar dalam proses pelatihan karena seluruh data diikutkan kedalam proses training sehingga akan membutuhkan waktu cukup lama. Agar mempermudah dan mempercepat proses training, biasanya data rate dibagi per batch (Batch Size). Batch size merupakan jumlah sampel data yang akan disebar

dalam sebuah neural network. Batch size efisien secara komputasi ketika berhadapan dengan dataset yang besar. Penentuan nilai dari batch size biasanya tergantung peneliti dengan melihat banyak sampel[28].



## BAB III

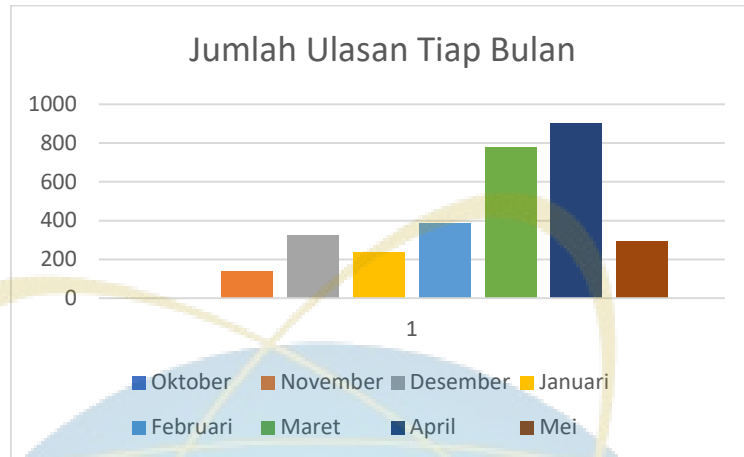
### METODOLOGI PENELITIAN

#### 3.1. Sumber Data

Data yang digunakan pada penelitian ini adalah berasal dari kolom ulasan aplikasi Tokopedia yang ada di website Google Play Store. Google Play Store adalah aplikasi resmi milik Google untuk perangkat yang menggunakan sistem baik operasi Android maupun Web. Bagi pengguna bisa mendapatkan aplikasi Tokopedia dapat dengan cara mengunduh aplikasi Tokopedia di Google Play Store. Pada Google Play Store juga terdapat kolom ulasan yang berisi ulasan para pengguna. Banyaknya dan beragam ulasan tersebut dapat dijadikan sebagai bahan penelitian di bidang data mining. Cara untuk mendapatkan ulasan tersebut dilakukan proses yang di namakan *Web Scraping*. Web Scraping adalah teknik untuk pengambilan sebuah dokumen dari internet umumnya berupa halaman Web yang dapat diambil datanya sesuai kebutuhan. Manfaat web scraping adalah memperoleh data melalui pengambilan dan ekstraksi data dengan ukuran data yang bervariasi[29]. Hasil dari web scraping dapat dilihat dari Tabel 3.2.

Pada penelitian ini pengambilan data ulasan Tokopedia menggunakan web scraping pada bulan Oktober 2018 sampai Mei 2019 dengan total 3067 ulasan dengan jumlah ulasan berbeda setiap bulannya seperti yang terlihat pada Gambar 3.1. Pada bulan Oktober hampir tidak ada yang memberikan ulasan melainkan didapatkan hanya 1 ulasan kemudian untuk bulan bulan selanjutnya mengalami relative penambahan setiap bulannya dan didapatkan ulasan terbanyak pada bulan April sampai sebanyak 905 ulasan dan mengalami penurunan kembali pada bulan selanjutnya yaitu Mei yang hanya terdapat 294 ulasan.





**Gambar 3.1** Jumlah Ulasan Tokopedia Tiap Bulan

Pada proses pengambilan data selain ulasan juga peneliti mengambil informasi Username, Date serta Rating pada setiap ulasannya. Sehingga didapatkan username sebanyak 3067 sama seperti jumlah ulasan yang didapat, hal ini menunjukkan bahwa yang memberikan ulasan adalah masing-masing dari pengguna yang berbeda. Sedangkan hasil yang didapatkan pada pengambilan rating ditunjukkan pada Tabel 3.1 masing-masing pengguna memberikan nilai rating yang berbeda-beda dimana jumlah rating tertinggi terdapat pada rating yang berbintang 5 dan jumlah rating terendah terdapat pada rating berbintang 2.

**Tabel 3.1.** Jumlah Ulasan Tiap Rating

Jumlah Ulasan	Rating				
	1	2	3	4	5
	169	56	84	266	2492

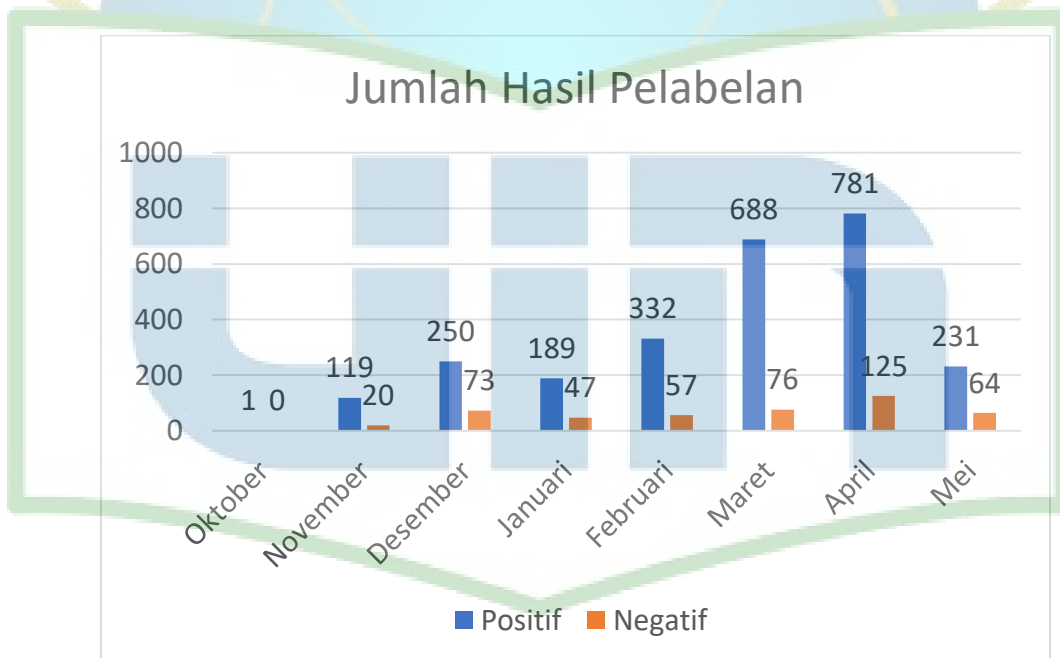
Namun jumlah rating pada data ini tidak selalu sesuai dengan isi ulasan. Seperti contoh terdapat pengguna memberikan ulasan ini “*unusable not responding long time access menu*” namun memberikan rating berbintang 5 dan masih banyak lagi ketidaksesuaian rating kepada isi ulasan.

**Tabel 3.2** Hasil Web Scraping Data Ulasan

User	Date	Rating	Review
Adi Prasetyo Raharjo	27 April 2019	4	A great market place! I have never found issues with their seller partner. Almost all the stuff I bought arrive within 3 hours. need some improvements in the menu though, i would like to see my most frequently purchases in a glance for repeat orders. And wish to edit the categories to suit my needs....Full Review
Dewi Rosalin	April 28, 2019	5	[UPDATE] The CS is more responsive and helpful in their Twitter account. Just contact them through their twitter and I got my cashback. Hopefully there are no more system failure that cause problem anymore
Johan Cahyadi	4 May 2019	1	Your promotions are useless, everytime I ent. it, it's always 'Terjadi kesalahan, silahkan tutup bla bla'. even if I have met the requirement.. what a big let down. You should care for an update that give less burden for users, esp. with heavy animations (breaking eggs take longer than before which ...Full Review

### 3.2. Pelabelan

Setelah data didapatkan dari tahan web scraping. Data kemudian dianalisis dengan memberikan label setiap ulasannya. Label pada penelitian kali ini menggunakan positif dan negatif. Label pada penelitian ini digunakan label 1 untuk kalimat yang bernilai sentiment positif misalnya kepuasan dalam menggunakan aplikasi Tokopedia sedangkan 0 untuk yang bernilai sentimen negatif misalnya tanggapan kekecewaan dan keluhan dari pengguna Tokopedia. Tahap ini bertujuan memberikan pembelajaran pada model yang akan dibentuk di tahap pelatihan data. Pada tahap pelabelan ini dilakukan oleh 3 orang bertujuan untuk menyamakan persepsi kalimat ulasan termasuk kedalam label yang mana. Jika ada 2 perbedaan maka orang ketiga melakukan penengah untuk akhirnya memutuskan kalimat tersebut masuk ke label positif atau negatif. Hal ini bertujuan untuk agar tidak terjadi subjektifitas pada penelitian ini.



**Gambar 3.2** Jumlah Label Tiap Bulan

Hasil dari pelabelan pada dataset dapat dilihat pada Gambar 3.3 bahwa setiap bulannya lebih banyak yang berlabel positif dari pada yang berlabel negatif yaitu

dengan jumlah masing-masing 2591 dan 476, label positif tertinggi didapatkan dibulan April 2019 dengan jumlah 781 ulasan yang mana pada bulan tersebut lebih banyak menanggapi terkait promo yang diberikan pada pihak Tokopedia.

### **3.3. Preprocessing**

Banyaknya data ulasan yang didapat banyak yang menggunakan kata yang tidak berstruktur seperti singkatan, emotikon, simbol dan angka hal ini diperlukan teknik preprocessing. Preprocessing berguna untuk mengekstrak informasi dari ulasan, mengubah kata-kata yang tidak terstruktur itu menjadi bentuk standar. Preprocessing sangat penting dan kritis dalam data mining[30]. Pada subab ini akan dibahas tentang perlakuan preprocessing yang dibutuhkan pada data yang digunakan pada penelitian ini

#### **1. Case Folding**

Case folding adalah proses untuk mengubah kata menjadi bentuk yang sama, misalnya huruf kecil atau besar. Pada data ini peneliti mengubah semua kata menjadi huruf kecil atau disebut lowercase[30]

#### **2. Hapus Simbol, Angka, dan Emoji**

Menghapus simbol, angka, string ASCII, dan tanda baca. Pesan ulasan biasanya berisi simbol, angka, dan tanda baca. Semua ini akan dihapus menggunakan sintaks ekspresi regular[30]

#### **3. Lemmatization**

Lemmatization adalah proses transformasi untuk menemukan bentuk normalisasi satu kata. Mengubah kata menjadi dalam bentuk kata dasarnya agar lebih mudah inti dari kata pada ulasan seperti yang ada pada dataset penelitian ini adalah kata “exelent” menjadi “excellent”, “recom” menjadi “recommended” dsb.[30]

#### **4. Stopword**

Stopword adalah menghilangkan kata-kata umum dan sering yang tidak memiliki pengaruh signifikan dalam kalimat. Dalam tugas pra-pemrosesan ini, peneliti menghapus stopwords pada data ulasan berdasarkan daftar kata stopwords yang berisi kata-kata seperti yang ada pada dataset penelitian ini adalah I, so, to, as, topads dsb[30]

## 5. Tokenizer

Tokenizer bertujuan untuk membagi kalimat menjadi beberapa bagian. Tokenisasi dapat dibentuk dalam kata-kata, frasa atau elemen bermakna lainnya[30]. Pada prinsipnya proses ini adalah memisahkan setiap kata yang menyusun suatu dokumen. umumnya setiap kata terpisahkan dengan kata yang lain oleh karakter spasi, sehingga proses tokenisasi mengandalkan karakter spasi pada dokumen untuk melakukan pemisahan kata. Setelah melalui proses tokenisasi maka kalimat tersebut menjadi sekumpulan array yang setiap selnya berisi kata-kata yang ada pada kalimat tersebut[31]. *Keras* menawarkan beberapa metode untuk kenyamanan preprocessing teks dan preprocessing urutan yang dapat kita terapkan untuk mempersiapkan teks. Contoh tokenisasi terlihat pada Tabel 3.2.

**Tabel 3.3** Ulasan Sebelum dan Sesudah di Tokenizer

Ulasan	Ulasan yang di Tokenizer
['recommended', 'relied cheap mundane item pricy gadget extra care handling shipping delivered expected satisfied choice store courier user friendly design comprehensive', 'ovo shovel treatment dissapointed narrow money expect happy use point platform awful asinine use	['recommended'], ['relied', 'cheap', 'mundane', 'item', 'pricy', 'gadget', 'extra', 'care', 'handling', 'shipping', 'delivered', 'expected', 'satisfied', 'choice', 'store', 'courier', 'user', 'friendly', 'design', 'comprehensive'], ['ovo', 'shovel', 'treatment', 'dissapointed', 'narrow',

cashback turn cannae ongoing	'money', 'expect', 'happy',
payment purchase item put']	'use', 'point', 'platform',
	'awful', 'asinine', 'use',
	'cashback', 'turn', 'cannae',
	'ongoing', 'payment',
	'purchase', 'item', 'put']

### 3.4. Data Training dan Data Testing

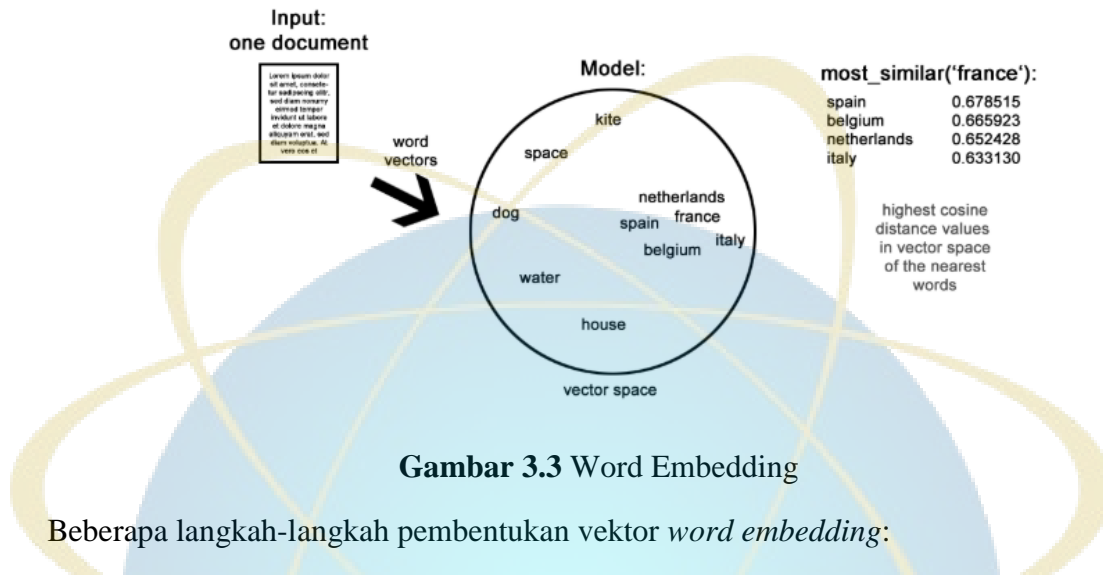
Setelah data dirubah menjadi bentuk vektor selanjutnya data dibagi menjadi data training, validasi dan test. Data Training bertujuan untuk melatih data yang akan membangun model akan dibentuk dengan ukuran 50%, kemudian model dilakukan validasi untuk menghindari over fitting dengan menggunakan 30% dari data training dan 20% sisanya menjadi data test untuk menguji peforma model yang sudah dilatih.

Maka jumlah data training yaitu 1717 ulasan, data validasi 736 ulasan dan 614 ulasan data testing.

### 3.5. Lapisan Word Embedding

*Word embedding* adalah sebuah pendekatan yang digunakan untuk merepresentasikan vector kata atau array bilangan riil. Word embedding merupakan pengembangan komputasi permodelan kata-kata yang sederhana seperti perhitungan menggunakan jumlah dan frekuensi kemunculan kata dalam sebuah dokumen. Hasil dari word embedding ini dapat digunakan ke dalam ruang geometris. Ruang geometris ini kemudian disebut ruang embedding. Kata-kata dengan makna yang serupa akan cenderung berada dalam satu area yang sama dalam bidang tersebut atau dengan kata lain katakata yang ada dalam satu area pada bidang tersebut dan mempunyai jarak kedekatan yang kecil cenderung mempunyai kesamaan semantik sebagai contoh dapat dilihat pada Gambar 3.4. Ada beberapa macam untuk melakukan proses word embedding[31]. Untuk proses word embedding pada penelitian ini adalah

melatih word embeddings selama men-training neural network. Pada proses word embedding ini menggunakan framework yang ada pada python yang bernama keras.



1. Setelah data dilakukan beberapa tahapan preprocessing dan menghasilkan ulasan dalam berbentuk list kata-kata yang sudah di tokenisasi. Selanjutnya adalah tahap memberikan indeks pada setiap kata di dataset. Pada tahap ini memerlukan parameter *num\_words* yang berguna untuk mengatur ukuran vocabulary yang ingin digunakan. Pada penelitian kali ini parameter *num\_word* menggunakan jumlah 2300 kata berdasarkan pembulatan dari jumlah kata yang umum pada pada dataset sebanyak 2290. Maka hasil dari memberikan indeks pada dataset terlihat pada Tabel 3.3.

**Tabel 3.4 Hasil Pemberian Indeks pada Ulasan**

Indeks Setiap Kata																			
[[34], [847, 130, 848, 41, 849, 850, 346, 289, 437, 123, 225, 582, 70, 80, 35, 139, 10, 13, 438, 583], [63, 851, 852, 853, 854, 65, 439, 101, 5, 98, 102, 855, 856, 5, 44, 249, 857, 858, 37, 103, 41, 859, 54]]																			



Terdapat juga pada keras.preprocessing.sequence untuk mengisi urutan kata dengan angka 0 secara otomatis. Sehingga jika terdapat kalimat yang pendek maka akan banyak pula angka 0 pada *feature vector* tersebut. Pada penelitian ini jumlah *input\_length* adalah 30. Sehingga didapatkan vektor pad sequence terlihat pada Tabel 3.4. Struktur data sudah dapat diolah untuk selanjutnya menjadi input pada lapisan *word embedding*

**Tabel 3.5 Hasil Pad\_Sequence**

Vektor Pad_Sequence
<code>array([[ 0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,         0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0, 34], [ 0,  0,  0,  0,  0,  0,  0,         0,  0, 847, 130, 848, 41, 849, 850, 346, 289, 437, 123, 225,         70, 80, 35, 139, 10, 13, 438, 583], [ 0,  0,  0,  0,  0,  0,  0,         851, 852, 853, 854, 65, 439, 101, 5, 98, 102, 855, 856, 5,         249, 857, 858, 37, 103, 41, 859, 54]])</code>

3. Pada tahap ini adalah tahap masuk lapisan pertama pada model yang akan dibangun pada penelitian ini. Lapisan pertama ini adalah lapisan word embedding atau *layar embedding*. Pada lapisan ini akan memasukan beberapa parameter yang sudah didapat dari langkah-langkah sebelumnya yaitu *input\_dim* adalah jumlah

## Vektor Pad\_Sequence

```
0, 0, 847, 130, 848, 41, 849, 850, 346, 289, 437, 123, 225,
70, 80, 35, 139, 10, 13, 438, 583], [ 0, 0, 0, 0, 0, 0, 0,
851, 852, 853, 854, 65, 439, 101, 5, 98, 102, 855, 856, 5
249, 857, 858, 37, 103, 41, 859, 54]])
```

3. Pada tahap ini adalah tahap masuk lapisan pertama pada model yang akan dibangun pada penelitian ini. Lapisan pertama ini adalah lapisan word embedding atau *word embedding*. Pada lapisan ini akan memasukan beberapa parameter yang sudah didapat dari langkah-langkah sebelumnya yaitu *input\_dim* adalah jumlah

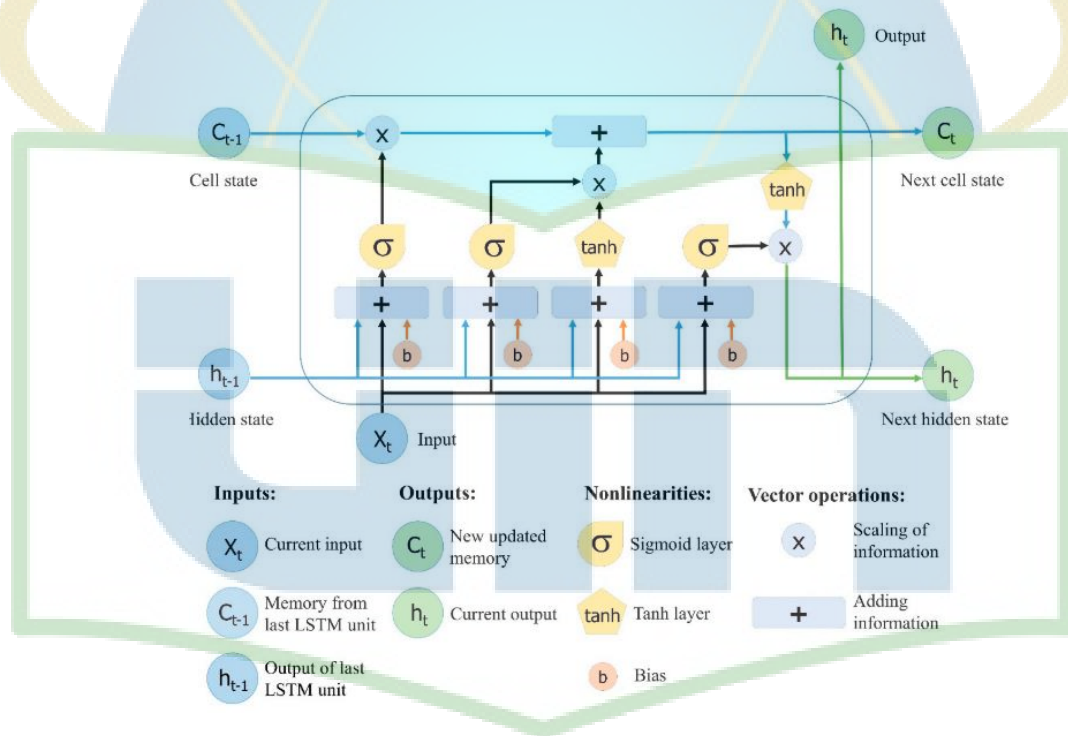
didapatkan variable input  $x_t$  berupa vektor word embedding tensor menjadi 3D dengan ukuran (3067, 30, 400) yang selanjutnya akan dimasukan ke lapisan LSTM

**Tabel 3.6** Hasil Word Embedding Layer

Vektor Word Embedding
<pre>array([[[ 0.03876838, -0.01809684, 0.03990138, ..., -0.00122931, 0.01254988, -0.00429102], [ 0.03876838, -0.01809684, 0.03990138, ..., -0.00122931, 0.01254988, -0.00429102], [ 0.03876838, - 0.01809684, 0.03990138, ..., -0.00122931, 0.01254988, - 0.00429102], ..., [ 0.03876838, -0.01809684, 0.03990138, ..., - 0.00122931, 0.01254988, -0.00429102], [ 0.03876838, -0.01809684, 0.03990138, ..., -0.00122931, 0.01254988, -0.00429102], [ 0.02450717, -0.00492223, -0.03573642, ..., 0.04055022, - 0.04313932, -0.0434492  ]], [[ 0.03876838, -0.01809684, 0.03990138, ..., -0.00122931, 0.01254988, -0.00429102], [ 0.03876838, -0.01809684, 0.03990138, ..., -0.00122931, 0.01254988, -0.00429102], [ 0.03876838, -0.01809684, 0.03990138, ..., -0.00122931, 0.01254988, -0.00429102], ..., [ 0.00706656, - 0.00257365, 0.04088905, ..., 0.01449505, -0.00464154, 0.04358189], [-0.00383022, -0.02674326, -0.01086992, ..., - 0.03303955, -0.02595526, -0.02655647], [-0.02707672, -0.0037173 , -0.01421509, ..., -0.02593316, -0.03787652, 0.02370617]], [[ 0.03876838, -0.01809684, 0.03990138, ..., -0.00122931, 0.01254988, -0.00429102], [ 0.03876838, -0.01809684, 0.03990138, ..., -0.00122931, 0.01254988, -0.00429102], [ 0.03876838, - 0.01809684, 0.03990138, ..., -0.00122931, 0.01254988, - 0.00429102], ..., [-0.02179501, -0.03831438, -0.04003658, ..., - 0.0396034 , 0.00798883, -0.02942543], [-0.02524986, -0.0248847 , -0.0374543 , ..., -0.0239279 , 0.04372492, -0.0353592  ], [- 0.02247027, 0.04684839, 0.01719972, ..., 0.03015948, -0.01121046, -0.00202378]]], dtype=float32)</pre>

### 3.6. Long Short Term Memory

LSTM dapat mengingat informasi jangka panjang. Pada RNN perulangan jaringan hanya menggunakan satu layer sederhana, yaitu layer  $\tanh$ . LSTM memiliki sel memori yang terdiri dari empat komponen utama: *input gate*, koneksi berulang, *forget gate* dan *output gate*. *Input gate* dapat memblokir atau memungkinkan sinyal masuk untuk mengubah kondisi sel. Demikian pula, *output gate* baik mencegah atau memungkinkan keadaan sel dari memiliki efek pada unit lain. *Forget gate* memungkinkan sel untuk mengingat atau melupakan keadaan sebelumnya dengan mengontrol koneksi berulang-ulang sel[12]



**Gambar 3.4.** Struktur Jaringan Long Short Term Memory (LSTM)[12]

LSTM memiliki dua bagian *cell* diantaranya adalah *cell state* dan *hidden state*. *Cell state* adalah jaringan utama aliran data yang memungkinkan data mengalir pada dasarnya tidak berubah. Namun, beberapa transformasi linear dapat terjadi. Data dapat

ditambahkan atau dihapus dari *cell state* melalui *sigmoid gates*. *Gates* tersebut mirip dengan lapisan atau serangkaian operasi matriks, yang berisi bobot individu yang berbeda. LSTM dirancang untuk menghindari masalah *long-term dependency*. [12]

Langkah pertama dalam membangun jaringan LSTM adalah mengidentifikasi informasi yang tidak diperlukan dan akan dihilangkan dari *cell* pada langkah itu. Proses mengidentifikasi dan mengecualikan data ini diputuskan oleh fungsi *sigmoid*, yang mengambil *output* dari LSTM ( $h_{t-1}$ ) pada waktu ( $t - 1$ ) dan disaat input ( $x_t$ ) pada waktu  $t$ . Selain itu, fungsi *sigmoid* menentukan bagian mana dari *output* lama yang harus dihilangkan. Fungsi itu disebut dengan *forget gate* di mana  $f_t$  adalah vektor dengan nilai mulai dari 0 hingga 1, sesuai dengan setiap angka dalam keadaan *cell* ( $C_t - 1$ ) [12]

$$\sigma(x) = \frac{1}{1 + e^{-x}} \quad (3.2)$$

Di sini,  $\sigma$  adalah fungsi sigmoid, Lapisan sigmoid ini akan menghasilkan angka antara 0 dan 1, yang menggambarkan seberapa banyak setiap komponen harus dilewati. Nilai 0 mengartikan tidak ada informasi yang akan diteruskan dan nilai 1 mengartikan semua informasi akan diteruskan. [12]

$$i_t = \sigma(W_i \cdot [h_{t-1}, x_t] + b_i) \quad (3.3)$$

$$f_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_f) \quad (3.4)$$

$$O_t = \sigma(W_o \cdot [h_{t-1}, x_t] + b_o) \quad (3.5)$$

Input gate mengatur nilai dari input untuk diperbarui pada *cell state*, dapat dihitung dengan Persamaan (3.3). *Forget gate* mengatur informasi mana yang akan dihapus dari *cell state*, dapat dihitung dengan Persamaan (3.4) dan *output gate* mengatur nilai yang akan dihasilkan output sesuai dengan input dan memori pada *cell state*, dapat dihitung dengan Persamaan (3.5).

Dimana  $i_t$  adalah input gate,  $f_t$  adalah forget gate, dan  $o_t$  adalah output gate.  $\sigma$  adalah fungsi sigmoid.  $W$  dan  $b$  adalah matriks bobot dan bias, masing-masing dari

keadaan sel. ( $h_{t-1}$ ) merupakan *hidden state* yaitu memori pada jaringan yang terdapat di *cell state* yang merupakan nilai output sebelum orde ke  $t$  dan  $x_t$  merupakan input pada orde ke  $t$ .

$$N_t = \tanh \cdot (W_n \cdot [h_{t-1}, x_t] + b_n) \quad (3.6)$$

$$C_t = C_{t-1} \cdot ft + N_t \cdot i_t \quad (3.7)$$

Fungsi tanh pada Persamaan (3.6) memberikan bobot pada nilai-nilai yang dilewati, memutuskan tingkat kepentingannya (-1 ke 1). Dua nilai dikalikan untuk memperbarui status sel baru. Memori baru ini kemudian ditambahkan ke memori lama ( $C_{t-1}$ ) yang menghasilkan ( $C_t$ ). Parameter ( $C_{t-1}$ ) dan ( $C_t$ ) adalah keadaan sel pada waktu ( $t - 1$ ) dan ( $t$ ), sedangkan  $W$  dan  $b$  adalah matriks bobot dan bias, masing-masing, dari keadaan sel.

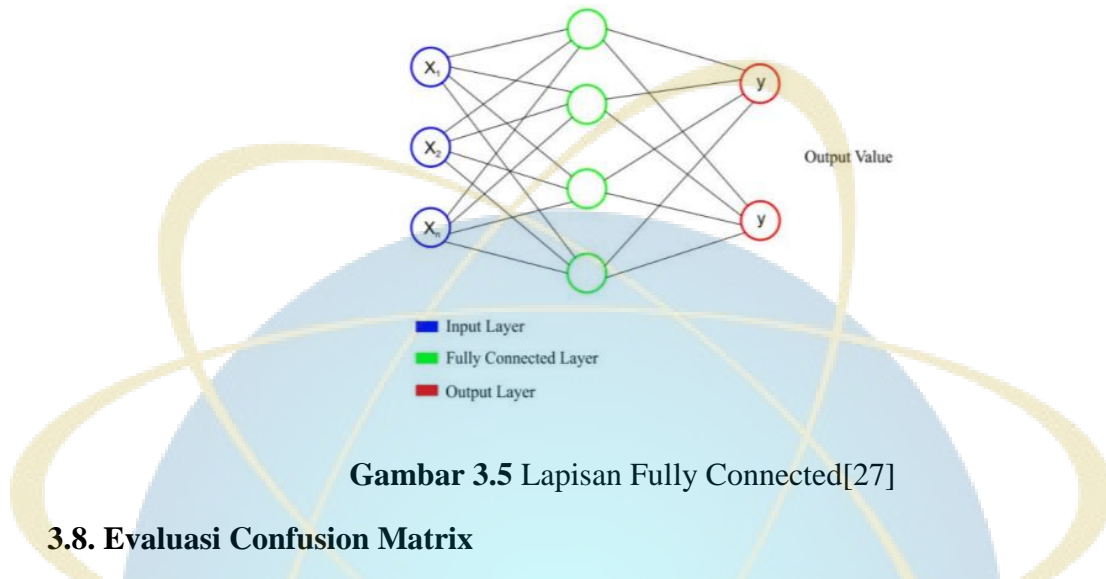
$$h_t = O_t \cdot \tanh(C_t) \quad (3.8)$$

Pada langkah terakhir terlihat pada Persamaan (3.8) nilai output ( $h_t$ ) didasarkan pada status sel output ( $O_t$ ) tetapi merupakan versi yang difilter. Pertama, lapisan sigmoid memutuskan bagian sel mana yang membuatnya menjadi output. Selanjutnya, output dari gerbang sigmoid ( $O_t$ ) dikalikan dengan nilai-nilai baru yang dibuat oleh lapisan tanh dari keadaan sel ( $C_t$ ), dengan nilai berkisar antara -1 dan 1.[12]

### 3.7. Lapisan Fully Connected

Lapisan *Fully Connected* ini adalah lapisan terakhir pada arsitektur Deep Learning. Lapisan Fully Connected adalah lapisan di mana semua neuron aktivasi dari lapisan sebelumnya terhubung semua dengan neuron di lapisan selanjutnya. Setiap aktivasi dari lapisan sebelumnya perlu diubah menjadi data satu dimensi sebelum dapat dihubungkan ke semua neuron [27] seperti yang terlihat pada Gambar 3.5. Lapisan Fully Connected memiliki beberapa parameter diantaranya adalah jumlah units fungsi aktivasi dan Loss Function. Pada penelitian ini digunakan jumlah unit 2 dikarenakan memiliki kelas positif dan negatif. Penelitian ini juga menguji 3 fungsi aktivasi seperti

sigmoid, tanh dan softmax dan menggunakan binary crossentropy sebagai loss function.



### 3.8. Evaluasi Confusion Matrix

*Confusion Matrix* digunakan untuk mengukur kinerja yang memiliki beberapa parameter. Confusion Matrix juga dikenal sebagai *error matrix*. Parameter yang digunakan adalah *TP* (*true positive*), *FN* (*false negative*), *TN* (*true negative*), dan *FP* (*false positive*) pada data uji[19]

**Tabel 3.7.** Confusion Matrix

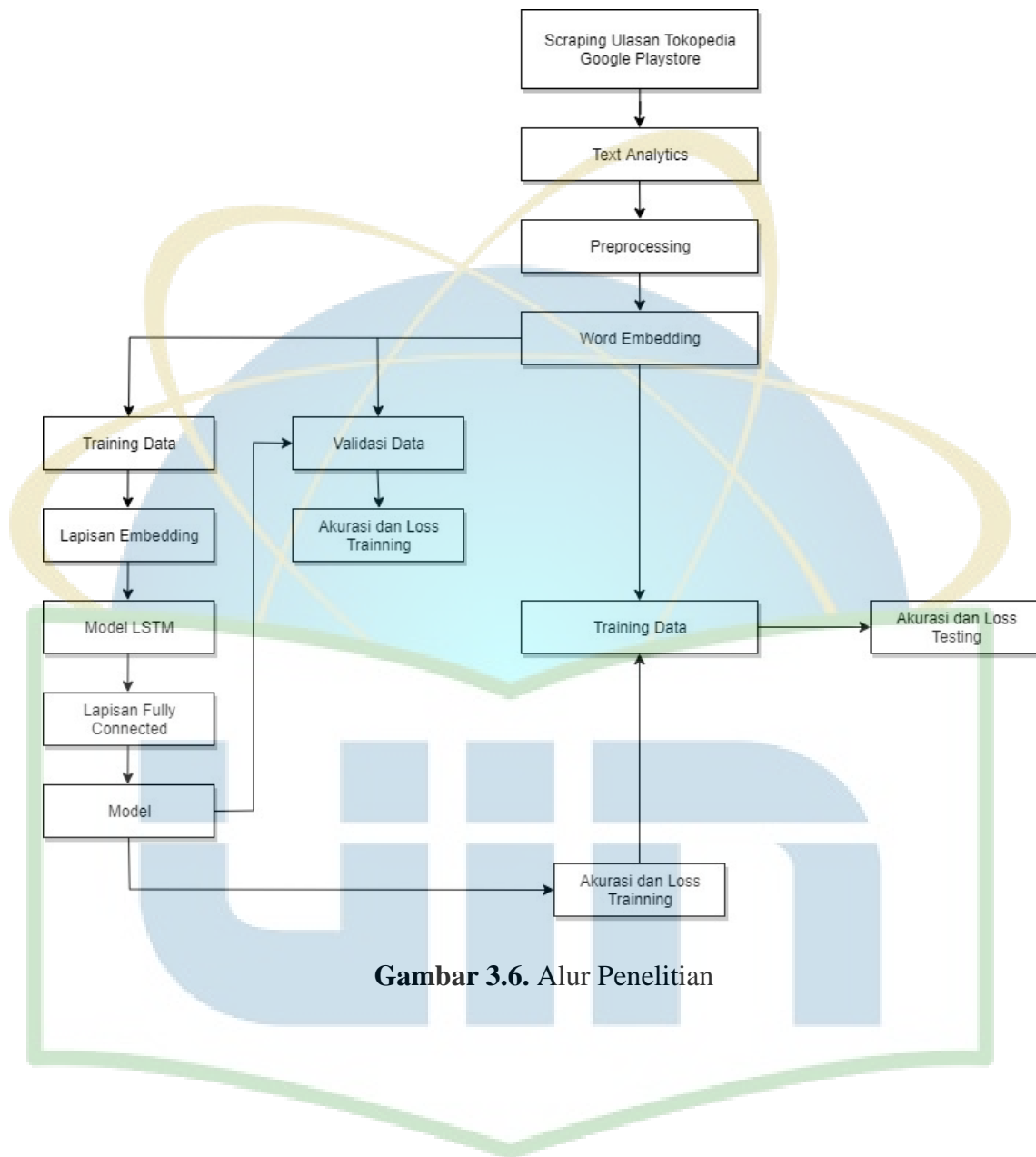
Aktual	Prediksi	
	Label Negatif	Label Positif
Label Negatif	TN	FN
Label Positif	FP	TP

Pada penelitian ini digunakan parameter pada Tabel 3.2 untuk membantu menghitung akurasi, presisi dan recall masing-masing memiliki kegunaan dan rumus sebagai berikut:

**Tabel 3.8.** Rumus Evaluasi Klasifikasi

<b>Akurasi</b>	$\frac{TP + TN}{TP + TN + FP + FN}$	Perbandingan antara kelas yang diprediksi dengan benar oleh sistem klasifikasi terhadap total data yang ada.
<b>Presisi</b>	$\frac{TP}{TP + FP}$	Perbandingan antara kelas yang diprediksi dengan benar oleh sistem terhadap total data yang terklasifikasi-kan suatu kelas
<b>Recall</b>	$\frac{TP}{TP + FN}$	Perbandingan antara kelas yang diprediksi dengan benar oleh sistem terhadap total data yang ada pada suatu kelas

### 3.9. Alur Penelitian



**Gambar 3.6.** Alur Penelitian



## BAB IV

### HASIL DAN PEMBAHASAN

Pada bab ini akan dijelaskan hasil dari pengambilan data dan menghasilkan data yang sudah dipreprocessing serta menunjukan wordlink pada setiap kelas dari hasil labeling. Serta akan dipaparkan grafik hasil training data dan menunjukkan hasil uji coba parameter untuk mendapatkan model yang optimal sampai mendapatkan evaluasi dengan menunjukan hasil dari confusion matrix.

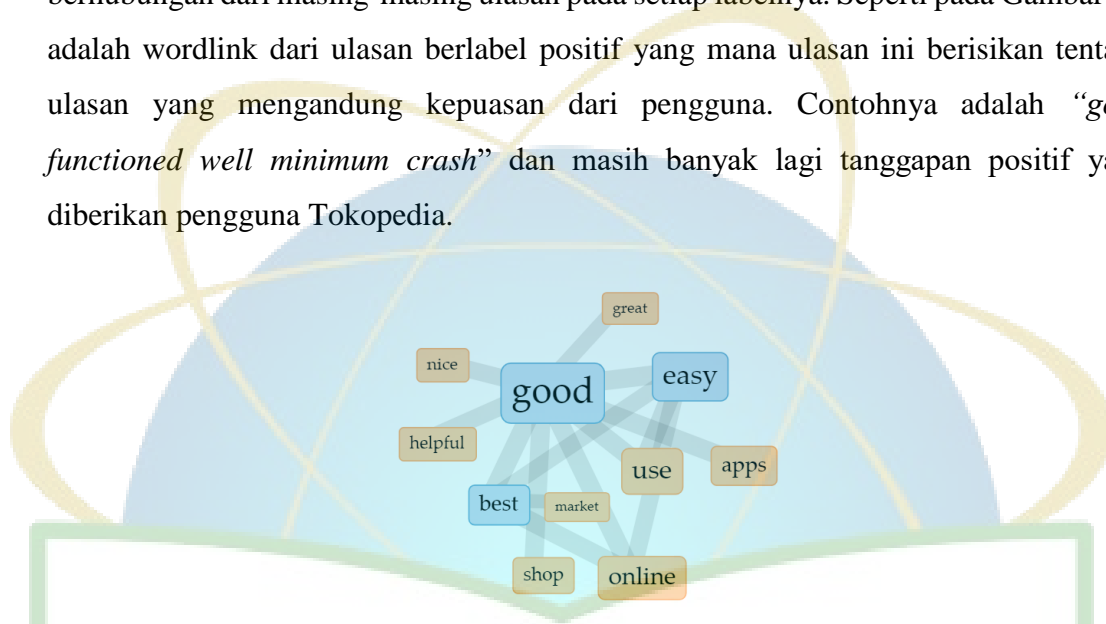
#### 4.1. Hasil Preprosesing dan Text Analytics

Data ulasan aplikasi Tokopedia yang didapatkan melalui web scraping dari website google play store selanjutnya di simpan dengan format CSV. Setelah data didapatkan dalam bentuk CSV kemudian data seluruhnya diberi label sentiment. Dikarenakan data tersebut merupakan data teks yang tidak terstruktur maka perlu dilakukan beberapa tahapan preprocessing, tahapan yang dilakukan diantaranya adalah case folding, remove number, remove emoticon, remove symbol, lemmatizer, remove stopword dan tokenization. Hasil dari tahap preprocessing dan pelabelan seperti pada Tabel 4.1

**Tabel 4.1** Hasil Preprocessing dan Labeling

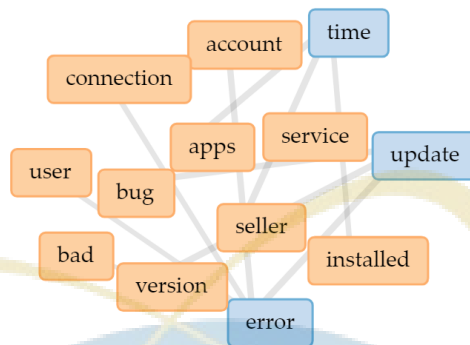
Ulasan	Ulasan Bersih	Label
I have relied on Tokopedia to obtain from cheap and mundane items to pricy gadgets that need extra care in handling and shipping. Everything were delivered as expected and I am satisfied (so far) with its choice of stores, couriers, user-friendly design, and comprehensive system.	relied cheap mundane item pricy gadget extra care handling shipping delivered expected satisfied choice store courier user friendly design comprehensive	1
Most Simple UI Marketplace i've ever used. Great Job Toped. Keep workin' on it !!!!	simple market place great job workin	1
Somwtimes there is still some server error that is annoying. But other than that good.	server error annoying good	0

Setelah didapatkan data ulasan bersih serta label disetiap ulasannya. Selanjutnya masing-masing label dari ulasan akan divisualisasi menggunakan *wordlink* dengan aplikasi *voyant tools*. Tujuannya agar dapat melihat kata-kata yang berhubungan dari masing-masing ulasan pada setiap labelnya. Seperti pada Gambar 4.1 adalah wordlink dari ulasan berlabel positif yang mana ulasan ini berisikan tentang ulasan yang mengandung kepuasan dari pengguna. Contohnya adalah “*good functioned well minimum crash*” dan masih banyak lagi tanggapan positif yang diberikan pengguna Tokopedia.



**Gambar 4.1** Hasil Wordlink dari Label Positif

Sebaliknya pada Gambar 4.2 menunjukan wordlink dari ulasan yang berlabel negatif yang mana ulasanya menunjukan kekecewaan dan keluhan dari pengguna. Dikarenakan Tokopedia merupakan e-commerce C2C dimana pengguna yang memberikan tanggapan di kolom komentar Google Play Store bisa jadi dari sisi pedagang maupun pembeli[4] sehingga pada tanggapan negatif ini sangat bervariasi ada yang memberikan tanggapan negatif untuk pembeli, penjual maupun ke perusahaan Tokopedia. Berikut adalah contoh ulasan pembeli kepada penjual “*search function slow seller not active terminate item minimize time search thing picture item dark reviewed poor uploading*”.



**Gambar 4.2** Hasil Wordlink dari Label Negatif

#### 4.2. Arsitektur Jaringan

Arsitektur jaringan dibentuk untuk menghasilkan akurasi yang optimal. Setelah menerima outputan dari lapisan embedding selanjutnya adalah melakukan pelatihan model LSTM. Pada lapisan LSTM menggunakan jumlah neuron 400. Kemudian semua jaringan dihubungkan ke setiap neuron dengan lapisan Fully Connected dengan jumlah unit 2 berdasarkan banyaknya kelas pada penelitian ini. Tahap terakhir adalah menggunakan fungsi aktivasi sigmoid. Selanjutnya setelah semua dibangun model dikonfigurasi terlebih dahulu dengan menggunakan optimasi Adam dan Binary Cross Entropy untuk mengetahui nilai loss dari model yang sudah terbentuk. Parameter lainnya untuk membantu proses training adalah ukuran batch size dan epoch. Berdasarkan uraian parameter yang digunakan berikut adalah arsitektur jaringan pada model yang terbentuk:

**Tabel 4.2** Total Parameter

Layer (type)	Output Shape	Param #
embedding_1 (Embedding)	(None, 30, 400)	920400
lstm_1 (LSTM)	(None, 400)	1281600
dense_1 (Dense)	(None, 2)	802
Total params: 2,202,802		
Trainable params: 2,202,802		
Non-trainable params: 0		

### 4.3. Pengujian Parameter

Penentuan model yang optimal ditentukan berdasarkan parameter yang menghasilkan nilai yang terbaik. Pada penelitian ini penentuan model yang optimal dengan melakukan pengujian dari beberapa parameter diantaranya adalah pengujian jumlah neuron dan pengujian fungsi aktivasi. Berikut adalah hasil pengujian yang telah dilakukan:

#### 4.3.1. Pengujian Jumlah Neuron

Pengujian yang dilakukan pertama adalah mencoba jumlah neuron pada lapisan lapisan yaitu dengan mencoba pada lapisan berjumlah 100, 200, 300, 400 dan 500. Percobaan ini dilakukan untuk menentukan jumlah neuron berapa yang optimal untuk membangun model dengan menunjukan akurasi yang terbaik seperti yang ditunjukkan pada Tabel 4.3

**Tabel 4.3 Hasil Pengujian Jumlah Neuron**

Jumlah Neuron	Learning Rate	Test Accuracy	Test Loss
100	0.001	92.92%	0.25
200	0.001	93.16%	0.28
300	0.001	93.16%	0.25
400	0.001	93.32%	0.26
500	0.001	91.12%	0.37

Hasil dari pada pengujian beberapa jumlah neuron menunjukan pada jumlah neuron 400 memberikan hasil test akurasi yang terbesar diantara jumlah neuron lainnya sebesar 93.32% dan diiringi nilai test loss sebesar 0.27. Pada penelitian ini hasil akurasi disetiap jumlah neuron mengalami nilai yang *fluktuatif* dimana hal ini dapat terjadi karena pada setiap kasus tidak ada ketentuan khusus untuk memilih jumlah neuron berapa yang optimal maka hasil terbaik berdasarkan hasil dari trial and error.

#### 4.3.2. Pengujian Fungsi Aktivasi Pada Lapisan Output

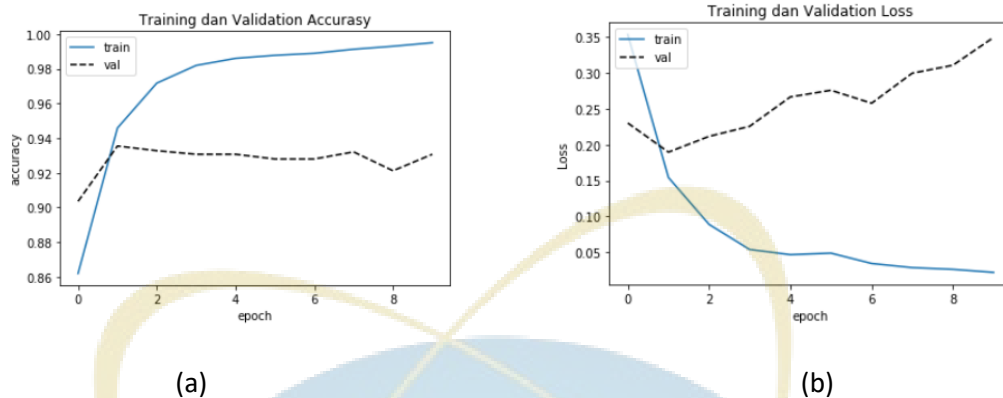
Pengujian terakhir adalah menguji beberapa fungsi aktivasi dengan menggunakan parameter yang terbaik sebelumnya yaitu menggunakan jumlah neuron 400 dan nilai learning rate 0.001. Pengujian ini bertujuan untuk menentukan fungsi aktivasi apa yang cocok pada data penelitian ini dengan melihat hasil akurasi yang terbaik. Pada pengujian ini dilakukan pengujian pada dua fungsi aktivasi yaitu fungsi sigmoid dan softmax. Berdasarkan hasil pengujian pada Tabel 4.5 menunjukkan bahwa fungsi aktivasi sigmoid memiliki akurasi test tertinggi yaitu sebesar 93.32% dibandingkan dengan fungsi aktivasi softmax yaitu sebesar 90.72%.

**Tabel 4.4** Hasil Pengujian Fungsi Aktivasi

Jumlah Neuron	Learning Rate	Aktivasi	Test Accuracy	Test Loss
400	0.001	Sigmoid	93.32%	0.26
400	0.001	Softmax	90.72%	0.27

#### 4.4. Hasil Training Data

Berdasarkan pengujian parameter yang telah dicoba, didapatkan model dengan nilai parameter terbaik pada jumlah neuron 400, nilai learning rate 0.001 dan menggunakan aktivasi fungsi sigmoid. Kemudian parameter ini selanjutnya ditraining bersamaan sehingga menghasilkan nilai akurasi dan loss pada data training dan validasi. Proses training menggunakan jumlah epoch sebanyak 10 dan batch size sebanyak 32. Dengan parameter yang sudah ditentukan akan dilihat berapa akurasi dari data training dan validasi dan melihat nilai loss terendah. Model akan menyimpan epoch yang optimal pada nilai loss yang terendah selama proses epoch berlangsung. Berikut adalah hasil training untuk membandung model:



**Gambar 4.3** Nilai Loss Training dan Validasi (a), Akurasi Training dan Validasi(b)

Berdasarkan Gambar 4.3 training menghasilkan nilai los terendah pada data validasi di epoch kedua dengan nilai sebesar 0.1895 dan nilai loss pada data training menghasilkan 0.1542 dan menghasilkan akurasi pada data training dan validasi masing-masing sebesar 94.58% dan 93.55%. Perbandingan terlihat bahwa model sudah cukup baik dengan menunjukkan akurasi data training dengan validasi tidak berbeda jauh sehingga tidak mengalami over fitting. Selanjutnya model akan diuji kepercayaannya dengan dilakukan pengujian dengan data testing.

#### 4.5. Evaluasi Data Uji

Tahap ini merupakan tahap pengujian model dengan data testing. Data yang akan diuji adalah berjumlah 614 ulasan dengan jumlah label di kelas negatif 86 ulasan dan 528 ulasan pada kelas positif. Selanjutnya setelah model menghasilkan prediksinya di masing-masing kelas dilakukan perhitungan tingkat kepercayaan model dengan melihat akurasi, presisi dan recall. Tujuannya untuk mengetahui seberapa dapat dipercaya model dalam memprediksi kelas.

Berdasarkan Tabel 4.6 menunjukkan hasil dari pengujian model. Dimana data test pada ulasan yang berlabel negatif benar diprediksi oleh model sebesar 59 dan salah memprediksi sebesar 16. Begitupula dengan ulasan yang berlabel positif dapat benar diprediksi oleh model sebesar 519 dan salah memprediksi sebesar 20 ulasan.

**Tabel 4.5** Hasil Confusion Matrix

Aktual	Prediksi	
	Negatif	Positif
Negatif	60	15
Positif	26	513

Evaluasi klasifikasi pada penelitian ini juga dapat dikatakan cukup baik dilihat dari nilai akurasi yang berarti model dapat dipercaya keakuratannya sebesar 93.32%. Begitupun dengan nilai presisi didapatkan sebesar 95.17% angka ini cukup baik untuk memprediksi sentiment positif yang merupakan sentiment terbanyak pada dataset. Kemudian nilai recall sebesar 97.15% juga memberikan hasil yang baik untuk mengetahui sensitivitas dari model yang didapat.

**Tabel 4.6** Evaluasi Confussion Matrix

Evaluasi	Hasil
Akurasi	93.32%
Presisi	95.17%
Recall	97.15%

Peforma model yang sudah diuji pada data testing menghasilkan akurasi yang tidak beda jauh dengan menggunakan model Machine Learning seperti Neural Network. Dimana dengan jumlah neuron yang sama pada data ini model Neural Netwok juga dapat dengan baik memprediksi dengan nilai akurasi sebesar 92%. Hal ini terjadi dikarenakan data yang digunakan mungkin tidak begitu rumit dipahami oleh model Neural Network.



Dari hasil model yang telah memprediksi ulasan pada data testing, selanjutnya adalah visualisasi hasil prediksi untuk melihat opini apa saja yang ada dimasing-masing kelas. Visualisasi dilihat dari *wordcloud* dan *wordlink* yang ada pada voyant tools. Wordcloud berguna untuk melihat keterkaitan kata yang paling banyak muncul pada suatu kelas, sedangkan wordlink berguna untuk melihat keberhubungan antar kata didalam suatu kelas. Sehingga hasil dari visualisasi ini dapat berguna untuk Tokopedia mengetahui upaya apa yang harus dilakukan untuk meningkatkan kualitas berdasarkan opini disetiap kelasnya.

*application & easy to use...many things I've bought from tokopedia...thanks a lot*

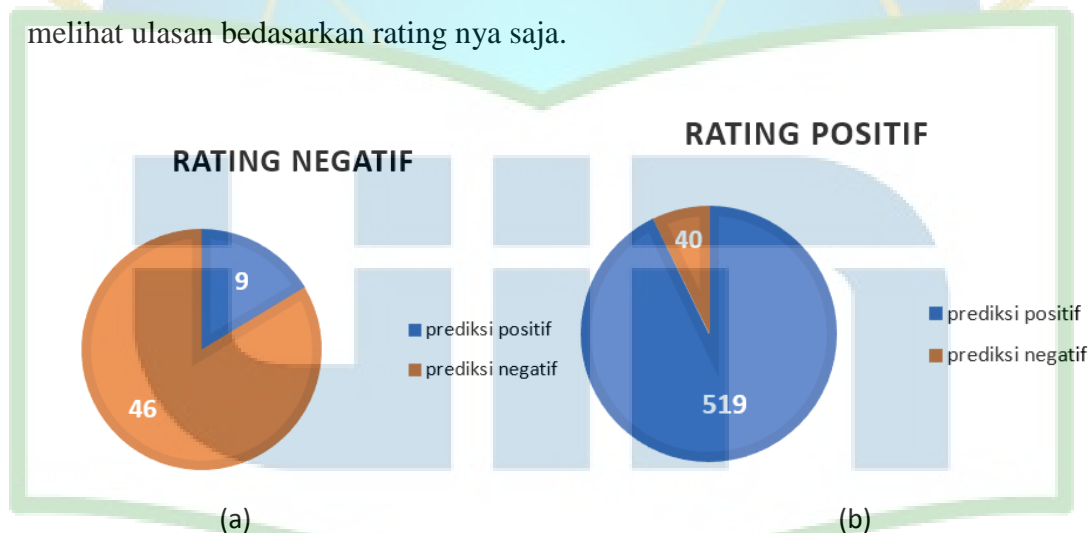
Pada visualisasi dikelas positif ini dapat dijadikan acuan untuk Tokopedia agar mempertahankan kualitasnya menjadi aplikasi yang mudah digunakan sehingga tetap menjadi pilihan pelanggan menggunakan aplikasi Tokopedia.

44





1 sampai 3 sedangkan rating positif pada ulasan yang memiliki rating 4 sampai 5. Tujuannya untuk menghitung kesesuaian antara rating dengan hasil prediksi yang dilakukan oleh model. Pada Gambar menunjukkan jumlah ulasan yang memiliki rating negatif sebanyak 55 ulasan dan model berhasil benar memprediksi bahwa ulasan tersebut pada kelas negatif lebih banyak dengan jumlah 46 ulasan daripada salah memprediksi sebanyak 9 ulasan. Untuk hasil perbandingan pada rating dikelas positif ditunjukkan pada Gambar 4.6. Pada rating positif memiliki jumlah ulasan sebanyak 559 ulasan dan model berhasil benar memprediksi dengan jumlah 519 ulasan sedangkan salah memprediksi dengan jumlah 40 ulasan. Hal dapat dikatakan perbandingan antara nilai rating yang diberikan oleh pengguna dengan hasil model memprediksi ulasan menunjukkan hasil yang tidak berbeda jauh karena untuk setiap kelas masing-masing menunjukkan lebih banyak memprediksi dengan benar sesuai kelas nya dibanding kesalahannya, maka dari itu pada kasus ini pengguna hanya dapat melihat ulasan berdasarkan rating nya saja.



**Gambar 4.6** Perbandingan Rating Negatif dengan Prediksi (a), Perbandingan Rating Positif dengan Prediksi (b)

#### 4.8. Perbandingan Model menggunakan Berbahasa Indonesia

Pada penelitian ini peneliti juga melakukan perbandingan model yang sama dengan mentraining model dengan menggunakan data berbahasa Indonesia. Dimana data yang digunakan berjumlah 5640, banyak kata umum berjumlah 8000 kata dan

maksimal panjang kata berjumlah 67. Dengan menggunakan hyperparameter yang sama dengan model berbahasa inggris yaitu jumlah neuron 400, nilai learning rate 0.001 dan fungsi aktivasi sigmoid. Pada data berbahasa Indonesia ini menghasilkan akurasi pada data testing yang lebih rendah yaitu sebesar 62.78%. Hal ini terjadi dikarenakan jumlah ulasan, jumlah kata umum dan panjang kata dalam dataset pada ulasan berbahasa Indonesia lebih besar daripada ulasan yang berbahasa Inggris sehingga dimensi yang digunakanpun lebih besar untuk membangun model. Seperti contoh yang ada pada data bahasa inggris yaitu kata "*unusable*" yang berarti "*tidak dapat digunakan*" kosa kata pada data berbahasa Indonesia dalam mengungkapkan arti yang sama di data bahasa Inggris menghasilkan panjang yang berbeda karena kata "tidak" dalam proses preprocessing untuk kasus analisis sentiment tidak dihapus. Maka hal ini yang membuat panjang vektor ulasan berbahasa Indonesia menjadi lebih panjang daripada ulasan berbahasa Inggris. Dengan demikian model tidak dapat digunakan secara umum. Untuk membangun model dengan ulasan berbahasa Indonesia perlu menggunakan penanganan yang lebih diantaranya dapat dengan menambahkan arsitektur yang lain seperti layer dropout atau regularization layer dan lainnya.

## **BAB V**

### **KESIMPULAN**

#### **5.1. Kesimpulan**

Ulasan aplikasi Tokopedia pada penelitian ini berjumlah 3067 dengan jumlah kelas negatif 2591 dan positif 476. Telah dilakukan proses preprocessing diantaranya adalah case folding, remove emotikon, angka, symbol, lemmatization dan tokenisasi dapat diimplementasikan pada analisis sentiment dengan menggunakan model LSTM. Model yang dibentuk merupakan hasil dari pelatihan dengan menggunakan hyperparameter terbaik diantaranya jumlah neuron 400, nilai learning rate 0.001, dan aktivasi fungsi sigmoid, ukuran batch size 20 dan epoch kedua. Pada hyperparameter tersebut model selanjutnya dilakukan pengujian dan model dapat memprediksi ulasan tokopedia dengan akurasi sebesar 93.32%, presisi 95.17% dan nilai recall 97.15%. Opini yang didapatkan pada setiap kelas menghasilkan diantaranya pada kelas negatif yaitu ketidakpuasan pelayanan pada Customer Service dan ketidakbermanfaat aplikasi setelah diupdate ke versi yang baru. Pada kelas Positif menghasilkan tanggapan pelanggan yang merasakan kepuasan pelanggan karena merasakan kemudahan saat menggunakan aplikasi.

#### **5.2. Saran**

Pada penelitian ini terdapat beberapa hal yang dapat dikembangkan untuk penelitian selanjutnya yaitu dengan menambahkan jumlah dataset atau melakukan membangun model dengan data berbahasa Indonesia namun menambahkan arsitektur yang lain seperti dropout layer atau regularization layer. Selain itu saran dari hasil penelitian ini untuk Tokopedia agar dapat membuat sistem analisis sentiment yang nantinya akan berguna untuk memprediksi ulasan untuk bahan evaluasi tahun tahun berikutnya selain itu diharapkan pihak Tokopedia dapat lebih memperbaiki dan meningkatkan pelayanan berdasarkan hasil visualisasi dari masing-masing kelas.

## REFERENSI

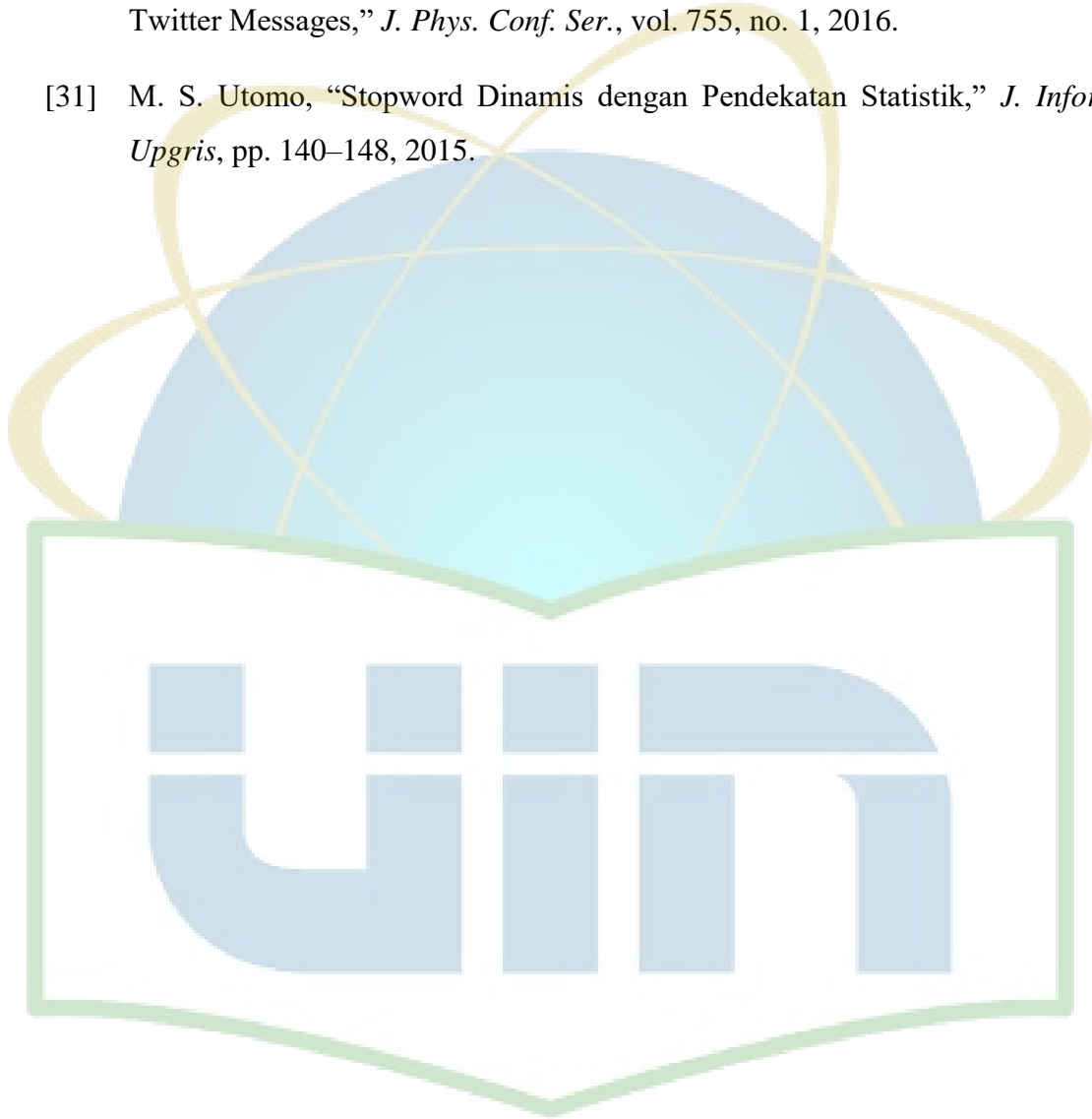
- [1] EcommerceIQ, “Indonesia C2C Ecommerce,” 2019. [Online]. Available: <https://ecommerceiq.asia/top-ecommerce-sites-indonesia/>.
- [2] R. H. & A. D. Aly, “Transaksi Jual Beli Online Perpektif Ekonomi Islam,” *Fak. Syari’ah dan Ekon. Islam IAIN Syekh Nurjati Cirebon*, vol. 3, no. 2, pp. 41–53, 2015.
- [3] S. Counter, “Mobile Operating System Market Share in Indonesia - December 2019,” 2019. [Online]. Available: <https://gs.statcounter.com/os-market-share/mobile/indonesia>. [Accessed: 01-Jan-2020].
- [4] L. Agustina and F. Kurniawan, “Sistem Reputasi Penjual dalam Proses Pengambilan Keputusan Pembelian di Platform C2C E-Commerce,” *J. Komun. Indones.*, vol. VII, pp. 28–42, 2018.
- [5] A. Dhiranty, B. Suharjo, and G. Suprayitno, “An Analysis on Customer Satisfaction, Trust and Loyalty Toward Online Shop (a Case Study of Tokopedia.Com),” *Indones. J. Bus. Entrep.*, vol. 3, no. 32, pp. 101–110, 2017.
- [6] R. Potharaju, M. Rahman, and B. Carbunar, “A Longitudinal Study of Google Play,” *Florida Int. Univ.*, pp. 1–14, 2018.
- [7] APJII, “Asosiasi Penyelenggara Jasa Internet Indonesia,” 2018. [Online]. Available: <https://www.apjii.or.id/>. [Accessed: 25-Jan-2020].
- [8] IPSOS, “E-commerce Outlook 2018.” [Online]. Available: <https://www.ipsos.com/en-id/e-commerce-outlook-2018>. [Accessed: 25-Jan-2020].
- [9] E. M. Sipayung, H. Maharani, and I. Zefanya, “Perancangan Sistem Analisis Sentimen Komentar Pelanggan Menggunakan Metode Naive Bayes Classifier,” *J. Sist. Inf.*, vol. 8, no. 1, pp. 958–965, 2016.

- [10] A. Rao and N. Spasojevic, "Actionable and Political Text Classification using Word Embeddings and LSTM," @lithium.com, vol. 2, pp. 1–9, 2016.
- [11] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Comput.*, vol. 9, no. 8, pp. 1735–1780, 1997.
- [12] X. H. Le, H. V. Ho, G. Lee, and S. Jung, "Application of Long Short-Term Memory (LSTM) neural network for flood forecasting," *Water (Switzerland)*, vol. 11, no. 7, 2019, doi: 10.3390/w11071387.
- [13] D. Li and J. Qian, "Text Sentiment Analysis Based on Long Short-Term Memory," *2016 1st IEEE Int. Conf. Comput. Commun. Internet, ICCCI 2016*, pp. 471–475, 2016.
- [14] H. X. Shi and X. J. Li, "A sentiment analysis model for hotel reviews based on supervised learning," *Proc. - Int. Conf. Mach. Learn. Cybern.*, vol. 3, pp. 950–954, 2011.
- [15] X. Ouyang, P. Zhou, C. H. Li, and L. Liu, "Sentiment analysis using convolutional neural network," *IEEE Int. Conf. Comput. Inf. Technol.*, pp. 2359–2364, 2015.
- [16] E. Guzman, M. El-Haliby, and B. Bruegge, "Ensemble Methods for App Review Classification: An Approach for Software Evolution," *IEEE/ACM Int. Conf. Autom. Softw. Eng.*, pp. 771–776, 2016.
- [17] I. Aichouri, A. Hani, N. Bougherira, L. Djabri, H. Chaffai, and S. Lallahem, "River Flow Model Using Artificial Neural Networks," *Energy Procedia*, vol. 74, pp. 1007–1014, 2015.
- [18] A. Ingolfsson and E. Sachs, "Recurrent neural network based language model," *J. Qual. Technol.*, vol. 25, no. 4, pp. 271–287, 1993.
- [19] A. Patel and A. K. Tiwari, "Sentiment Analysis by using Recurrent Neural

- Network,” *SSRN Electron. J.*, pp. 1–4, 2019.
- [20] T. Mikolov, S. Kombrink, L. Burget, J. Černocký, and S. Khudanpur, “Extensions of Recurrent Neural Network Language Model,” *ICASSP, IEEE Int. Conf. Acoust. Speech Signal Process. - Proc.*, pp. 5528–5531, 2011.
- [21] J. Rosindell and Y. Wong, “On The Difficulty of Training Recurrent Neural Networks,” *Phylogenetic Divers.*, no. 2, pp. 41–71, 2018.
- [22] N. Suhermi, I. M. Gde, M. Dana, and D. Dwi, “Pemilihan Arsitektur Terbaik pada Model Deep Learning Melalui Pendekatan Desain Eksperimen untuk Peramalan Deret Waktu Nonlinier,” *Dep. Stat.*, vol. 18, no. 2, pp. 153–159, 2018.
- [23] W. S. Eka Putra, “Klasifikasi Citra Menggunakan Convolutional Neural Network (CNN) pada Caltech 101,” *J. Tek. ITS*, vol. 5, no. 1, pp. A65–A69, 2016.
- [24] A. G. Keen, “From Softmax to Sparsemax: A Sparse Model of Attention and Multi-Label Classification Andr’e,” *Dynast. Lycia*, vol. 48, pp. 1–10, 2018.
- [25] K. Janocha and W. M. Czarnecki, “On loss functions for deep neural networks in classification,” *Schedae Informaticae*, vol. 25, pp. 49–59, 2016.
- [26] D. P. Kingma and J. L. Ba, “A Method For Stochastic Optimization,” *ICLR*, pp. 1–15, 2015.
- [27] A. Santoso and G. Ariyanto, “Implementasi Deep Learning Berbasis Keras Untuk Pengenalan Wajah,” *Emit. J. Tek. Elektro*, vol. 18, no. 01, pp. 15–21, 2018.
- [28] A. Nur and G. B. Hertantyo, “Implementasi Convolutional Neural Network untuk Klasifikasi Pembalap MotoGP Berbasis GPU,” *Inst. Teknol. Telkom Purwokerto*, pp. 50–55, 2018.



- [29] A. Josi *et al.*, “Penerapan Teknik Web Scraping Pada Mesin Pencari Artikel Ilmiah,” *Progr. Stud. Tek. Inform.*, pp. 159–164.
- [30] S. N. Kane, A. Mishra, and A. K. Dutta, “Pre-processing Tasks in Indonesian Twitter Messages,” *J. Phys. Conf. Ser.*, vol. 755, no. 1, 2016.
- [31] M. S. Utomo, “Stopword Dinamis dengan Pendekatan Statistik,” *J. Inform. Upgris*, pp. 140–148, 2015.





## LAMPIRAN

### Import Modul Yang Digunakan

```
import numpy as np # linear algebra
from sklearn.metrics import confusion_matrix, classification_report
import pandas as pd # data processing, CSV file I/O (e.g. pd.read_csv)
from sklearn.feature_extraction.text import CountVectorizer
from keras.preprocessing.text import Tokenizer
from keras.preprocessing.sequence import pad_sequences
from keras.models import Sequential
from keras.layers import Dense, Embedding, LSTM, SpatialDropout1D
from sklearn.model_selection import train_test_split
from keras.utils.np_utils import to_categorical
from sklearn.utils import resample
from sklearn.utils import shuffle
from sklearn.metrics import confusion_matrix, classification_report
import re
from keras.callbacks import ModelCheckpoint
import keras
from keras.layers import Dense, Dropout, Activation, LSTM
import matplotlib.pyplot as plt
```

### Memanggil Data Yang Digunakan

```
data=pd.read_csv('REVIEWTOKPED_fix.csv')
data.head()
listdata=data['REVIEW'].values.astype('U')
listdata = [d for d in listdata]
```

### Melihat Kata Umum Dari Data

```
text = [a for a in listdata]
sentence = ' '.join(text)
word = sentence.split()
len(np.unique(word))
```

### Mentokenisasi Data Ulasan

```
tokenizer = Tokenizer(num_words=2300, split=" ")
tokenizer.fit_on_texts(listdata)
X = tokenizer.texts_to_sequences(listdata)
X = pad_sequences(X) # padding our text vector so they all have the
same length
X[:5]
```

### Mengubah Data Label Menjadi Biner

```
y = pd.get_dummies(data['LABEL']).values
[print(data['LABEL'][i], y[i]) for i in range(0,5)]
```

### Membagi Data Train Dan Test

```
X_train, X_test, y_train, y_test, idx_train, idx_test =
train_test_split(X, y, index, test_size=0.2,
random_state=0)print(X_test.shape)
```

```
print(y_train.shape)
print(y_test.shape)
```

### **Membangun Arsitektur Model**

```
model = Sequential()
model.add(Embedding(2301, 400, input_length=X_train.shape[1]))
model.add(LSTM(400))
model.add(Dense(2, activation='sigmoid'))
adam = Adam(lr=0.001)
model.compile(loss='binary_crossentropy', optimizer='adam', metrics=['accuracy'])
model.summary()
```

### **Mentraining Model**

```
checkpointer=ModelCheckpoint(filepath='weights6.hdf5',monitor='val_loss',verbose=1,save_best_only=True)
batch_size = 32
epochs = 10
lstm = model.fit(X_train, y_train, validation_split=0.3, epochs=epochs, batch_size=batch_size, verbose=1, callbacks=[checkpointer])
```

### **Melihat Akurasi Training**

```
lstm.history
```

### **Membuat Grafik Dari Hasil Training**

```
plt.plot(lstm.history['acc'])
plt.plot(lstm.history['val_acc'])
plt.title('Training dan Validation Accuracy')
plt.ylabel('accuracy')
plt.xlabel('epoch')
plt.legend(['train', 'val'], loc='upper left')
plt.show()
plt.plot(lstm.history['loss'])
plt.plot(lstm.history['val_loss'])
plt.title('Training dan Validation Loss')
plt.ylabel('Loss')
plt.xlabel('epoch')
plt.legend(['train', 'val'], loc='upper left')
plt.show()
```

### **Predict Data Test**

```
predictions = model.predict(X_test)
[print(listdata[i], predictions[i], y_test[i]) for i in range(0, 5)]
predictions[:5]
y_DL, y_T = [], []
for x,y in zip(y_test,predictions):
    if x[0]>x[1]:
        y_T.append('Neg')
    else:
        y_T.append('Pos')
```

```
if y[0]>y[1]:
    y_DL.append('Neg')
else:
    y_DL.append('Pos')
print(y_DL[:5])
print(y_T[:5])
```

#### **Confussion Matrix**

```
print(confusion_matrix(y_T, y_DL))
print(classification_report(y_T, y_DL))
```

#### **Hasil Akurasi Data Test**

```
score = model.evaluate(X_test, y_test, verbose=0)
print("Test Loss: %.2f%%" % (score[0]))
print("Accuracy: %.2f%%" % (score[1]))
```

#### **Menyimpan Ulasan dengan Hasil Prediksi**

```
data_test = list(data['REVIEW'][idx_test].values)
rating_test = list(data['RATING'][idx_test].values)
sentimen_test = y_DL
dict_tst = {'test_data': data_test, 'hasil_sentimen':sentimen_test,
            'rating':rating_test}
prediksi_test = pd.DataFrame(dict_tst)
prediksi_test.to_csv('prediksi4_test.csv')
```





