

**IMPLEMENTASI METODE *LONG SHORT-TERM MEMORY (LSTM)*
PADA KLASIFIKASI ULASAN APLIKASI MOBILE JKN**

(Skripsi)

Oleh

NENI YULIYANTI



**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG
2023**

ABSTRACT

IMPLEMENTATION OF THE LONG SHORT TERM MEMORY (LSTM) METHOD IN JKN MOBILE APPLICATION REVIEWS CLASSIFICATION

By

NENI YULIYANTI

Mobile JKN Application is an information channel and online service that is used to facilitate the process of administering the national health insurance program by a Public Legal Entity that reports directly to the president, namely the Social Security Management Agency (BPJS). Every participant who is registered with BPJS Kesehatan can provide services online through the Mobile JKN Application to simplify administrative processes and access information sources. The more users of the Mobile JKN application, the more diverse the reviews received by the Mobile JKN application, both in the form of positive and negative opinions. One of the text mining methods in sentiment analysis, namely Long Short Term Memory (LSTM) in this study classifies reviews into positive and negative sentiments. Word Embedding is used to present words from data that is owned in vector form. The data processed in this study is a collection of Mobile JKN application reviews taken from the Google Play Store totaling 5410 reviews. The results of the classification carried out using the Long Short Term Memory (LSTM) method showed very good results with an accuracy of 94.37%.

Keywords : BPJS Kesehatan, Classification, LSTM, Mobile JKN,
Text Mining, Word2Vec, Word Embedding

ABSTRAK

IMPLEMENTASI METODE *LONG SHORT TERM MEMORY* (LSTM) PADA KLASIFIKASI ULASAN APLIKASI MOBILE JKN

Oleh

NENI YULIYANTI

Aplikasi Mobile JKN adalah suatu kanal informasi dan pelayanan online yang digunakan untuk mempermudah proses penyelenggaraan program jaminan kesehatan nasional oleh Badan Hukum Publik yang bertanggung jawab langsung kepada presiden yaitu Badan Pengelola Jaminan Sosial (BPJS). Setiap peserta yang terdaftar BPJS Kesehatan dapat melakukan pelayanan secara online melalui Aplikasi Mobile JKN untuk mempermudah proses administrasi serta mengakses sumber informasi. Semakin banyaknya pengguna aplikasi Mobile JKN maka semakin beragam juga ulasan yang di terima oleh aplikasi Mobile JKN baik itu berupa opini positif maupun negatif. Salah satu metode *text mining* dalam analisis sentimen yaitu *Long Short Term Memory* (LSTM) dalam penelitian ini mengklasifikasikan ulasan kedalam sentimen positif dan negatif. *Word Embedding* digunakan untuk mempresentasikan kata dari data yang dimiliki kedalam bentuk vektor. Data yang diolah pada penelitian ini adalah kumpulan ulasan aplikasi Mobile JKN yang diambil dari Google Play Store sejumlah 5410 ulasan. Hasil klasifikasi yang dilakukan dengan metode *Long Short Term Memory* (LSTM) menunjukkan hasil yang sangat baik dengan akurasi sebesar 94,37%.

Kata Kunci : BPJS Kesehatan, Klasifikasi, LSTM, Mobile JKN, Teks Mining,
Word2Vec, Word Embedding

**IMPLEMENTASI METODE *LONG SHORT-TERM MEMORY (LSTM)*
PADA KLASIFIKASI ULASAN APLIKASI MOBILE JKN**

Oleh

Neni Yuliyanti

Skripsi

Sebagai Salah Satu Syarat untuk Mencapai Gelar
SARJANA MATEMATIKA

Pada

Jurusan Matematika
Fakultas Matematika dan Ilmu Pengetahuan Alam



**FAKULTAS MATEMATIKA DAN ILMU PENGETAHUAN ALAM
UNIVERSITAS LAMPUNG
BANDAR LAMPUNG
2023**

Judul Skripsi : **IMPLEMENTASI METODE LONG
SHORT TERM MEMORY (LSTM) PADA
KLASIFIKASI ULASAN APLIKASI MOBILE
JKN**

Nama Mahasiswa : **Neni Yuliyanti**

Nomor Pokok Mahasiswa : **1917031047**

Program Studi : **Matematika**

Fakultas : **Matematika dan Ilmu Pengetahuan Alam**



1. **Komisi Pembimbing**

Dr. Aang Nuryaman, S.Si., M.Si.
NIP 19740316 200501 1 001

Dr. Ahmad Faisol, S.Si., M.Sc.
NIP 19800206 200312 1 003

2. **Ketua Jurusan Matematika**

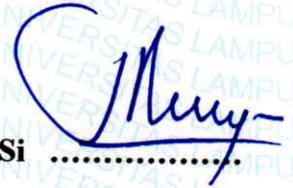
Dr. Aang Nuryaman, S.Si., M.Si.
NIP 19740316 200501 1 001

MENGESAHKAN

1. Tim Penguji

Ketua

: **Dr. Aang Nuryaman, S.Si., M.Si**



Sekretaris

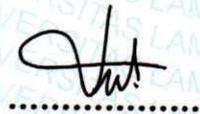
: **Dr. Ahmad Faisol, S.Si., M.Sc.**



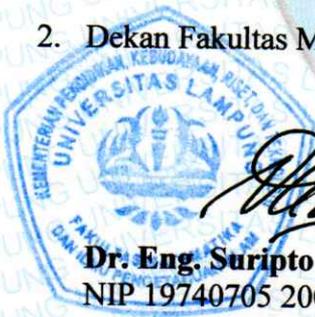
Penguji

Bukan Pembimbing

: **Drs. Nusyirwan, M.Si.**



2. Dekan Fakultas Matematika dan Ilmu Pengetahuan Alam



Dr. Eng. Surtpto Dwi Yuwono, S.Si., M.T.
NIP 19740705 2000031001

Tanggal Lulus Ujian Skripsi : 14 Februari 2023

PERNYATAAN SKRIPSI MAHASISWA

Yang bertanda tangan di bawah ini :

Nama : **Neni Yuliyanti**
Nomor Pokok Mahasiswa : **1917031047**
Jurusan : **Matematika**
Judul Skripsi : **IMPLEMENTASI METODE *LONG SHORT*
TERM MEMORY (LSTM) PADA
KLASIFIKASI ULASAN APLIKASI MOBILE
JKN**

Dengan ini menyatakan bahwa skripsi ini adalah hasil pekerjaan saya sendiri. Apabila kemudian hari terbukti bahwa skripsi ini merupakan hasil salinan atau dibuat oleh orang lain, maka saya bersedia menerima sanksi sesuai dengan ketentuan akademik yang berlaku.

Bandar Lampung, 14 Februari 2023

Yang menyatakan,



Neni Yuliyanti

RIWAYAT HIDUP

Penulis bernama lengkap Neni Yuliyanti, anak pertama dari dua bersaudarayang lahir di Ngarip pada tanggal 17 Mei 2001 dari pasangan Bapak Haryanto dan Ibu Sri Hartinah. Penulis menyelesaikan pendidikan sekolah dasar di SD Negeri 1 Ngarip pada tahun 2007 s.d 2013, sekolah menengah pertama di SMP Negeri 1 Pringsewu pada tahun 2013 s.d 2016, dan sekolah menengah atas di SMA Negeri 1 Pringsewu pada tahun 2016 s.d 2019.

Pada tahun 2019 penulis diterima sebagai mahasiswa S1 di Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung melalui jalur SBMPTN. Selama aktif menjadi mahasiswa, penulisikut serta dalam Himpunan Mahasiswa Jurusan Matematika (HIMATIKA) sebagai anggota Bidang Keilmuan.

Pada tahun 2022, sebagai bentuk penerapan bidang ilmu di dunia kerja, penulis melaksanakan Kerja Praktek (KP) di Badan Pengelola Pajak dan Retribusi Daerah (BPPRD) Kota Bandar Lampung dan sebagai bentuk pengabdian kepada masyarakat penulis melaksanakan Kuliah Kerja Nyata (KKN) di Kelurahan Banyu Urip, Kabupaten Tanggamus.

KATA INSPIRASI

“Orang yang bersabar pasti akan meraih keberuntungan, meskipun itu diperoleh setelah waktu yang lama.”

(Ali bin Abi Thalib)

“Allah tidak membebani seorang hamba melainkan sesuai dengan kemampuannya”

(Q.S. Al-Insyirah: 5)

PERSEMBAHAN

Alhamdulillah rabbi'l'alamin,
Puji dan syukur tiada hentinya terhanturkan kepada Allah SWT
Kupersembahkan karya ini kepada:

Diri Sendiri

Terima kasih untuk tetap selalu berusaha di segala keadaan untuk
mengembangkan diri dan menjadi pribadi yang lebih baik.

Ayah, Ibu, dan Kakak

Orang tuaku tercinta, Bapak Haryanto dan Ibu Sri Hartinah serta adikku
tersayang, Ahmad Kurniawan yang selalu memberikan doa, dukungan dan kasih
sayang.

Dosen

Dosen-dosen pembimbing dan pembahas yang sangat berjasa dalam membimbing
dan memberikan masukan yang membangun serta menyampaikan ilmu kepadaku.

Sahabat-sahabatku

Para sahabat tersayang yang terus saling mendukung, menolong, serta
memberikan warna dalam hidupku.

Almamater kebanggaan, Universitas Lampung.

SANWACANA

Puji dan syukur penulis ucapkan kehadirat Allah SWT yang telah memberikan rahmat dan hidayah-Nya, sehingga penulis dapat menyelesaikan skripsi yang berjudul “Implementasi Metode *Long Short Term Memory* (LSTM) Pada Klasifikasi Ulasan Aplikasi Mobile JKN”.

Penulis menyadari bahwa dalam penulisan skripsi ini tidak terlepas dari bimbingan, dukungan, bantuan dan doa dari berbagai pihak. Oleh karena itu, dengan ketulusan hati penulis ingin menyampaikan terimakasih kepada:

1. Bapak Dr. Aang Nuryaman, S.Si., M.Si. selaku Dosen Pembimbing I yang senantiasa selalu membimbing dan memberikan arahan, kritik, dan saran serta dukungan kepada penulis selama proses perkuliahan dan pembuatan skripsi ini.
2. Bapak Dr. Ahmad Faisol, S.Si., M.Sc. selaku Dosen Pembimbing II yang telah memberikan bimbingan serta saran yang membantu kepada penulis dalam proses penyelesaian skripsi ini..
3. Bapak Drs. Nusyirwan, M.Si. selaku Dosen Pembahas atas ketersediaannya untuk membahas serta memberikan kritik dan saran serta evaluasi kepada penulis dalam penyelesaian skripsi ini.
4. Bapak Dr. Aang Nuryaman, S.Si., M.Si. selaku Ketua Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung.
5. Bapak Dr. Eng. Suropto Dwi Yuwono, M.T., selaku dekan Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung.
6. Seluruh dosen, staf, dan karyawan Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung.

7. Ayah, ibu, kakak dan seluruh keluarga besar yang selalu memberikan kasih sayang, dukungan, nasihat, motivasi serta doa kepada penulis.
8. Sahabat-sahabat seperjuangan yang telah saling mendoakan, mendukung, dan memberikan kenangan indah selama menjalani masa perkuliahan.
9. Teman-teman Matematika 2019, terima kasih atas kebersamaannya.
10. Almamater tercinta, Universitas Lampung.
11. Seluruh pihak yang telah membantu yang tidak dapat disebutkan satu persatu.

Bandar Lampung, 14 Februari 2023

Penulis

Neni Yuliyanti

DAFTAR ISI

	Halaman
DAFTAR TABEL	viii
DAFTAR GAMBAR	xi
I. PENDAHULUAN	1
1.1. Latar Belakang.....	1
1.2. Tujuan Penelitian.....	3
1.3. Manfaat Penelitian.....	4
II. TINJAUAN PUSTAKA	5
2.1. Analisis Sentimen	5
2.2. <i>Text Mining</i>	5
2.3. <i>Preprocessing Data</i>	6
2.3.1 <i>Case Folding</i>	6
2.3.2 <i>Filtering</i>	6
2.3.3 <i>Tokenize</i>	7
2.3.4 <i>Stopword Removal</i>	7
2.4. <i>Word Embedding</i>	7
2.5. Klasifikasi.....	8
2.6. <i>Long Short Term Memory (LSTM)</i>	10
2.7. Fungsi Aktivasi.....	14
2.8. Evaluasi Model.....	16
2.9. BPJS Kesehatan.....	17
III. METODOLOGI PENELITIAN	19
3.1. Waktu Penelitian	19
3.2. Data Penelitian	19
3.3. Metode Penelitian	19

IV. HASIL DAN PEMBAHASAN	21
4.1 Pengumpulan Data.....	21
4.2 Pelabelan Data	22
4.3 Pre-processing	23
4.4 Word embedding	26
4.5 Alur Perhitungan Model LSTM	29
4.6 Evaluasi Model.....	32
V. KESIMPULAN	35

DAFTAR PUSTAKA

LAMPIRAN

DAFTAR TABEL

Gambar	Halaman
1. <i>Confusion Matrix</i>	16
2. Sampel dataset Ulasan Aplikasi Mobile-JKN.....	21
3. Sampel data labelling Ulasan Aplikasi Mobile-JKN	22
4. Proses <i>Case Folding</i>	23
5. Proses <i>cleansing</i>	24
6. Proses <i>Tokenizing</i>	24
7. Proses <i>Filtering</i>	25
8. Proses <i>Stemming</i>	25
9. Sampel Data Mentah.....	26
10. Indeks Perkata	26
11. Hasil dari <i>forget gate, input gate, candidate cell state, cell state dan output gate</i>	31
12. Hasil <i>Confusion matrix</i>	32
13. Hasil klasifikasi ulasan negatif	33

DAFTAR GAMBAR

Gambar	Halaman
1. Pengulangan RNN satu layer	10
2. Pengulangan LSTM empat layer.....	11
3. Ilustrasi <i>cell state</i>	11
4. <i>Flow chart</i> Analisis sentimen Mobile JKN	20

I. PENDAHULUAN

1.1 Latar Belakang

Data mining merupakan salah satu ilmu untuk menangani masalah penumpukan data yang belum dimanfaatkan. Salah satu cabang data mining dalam konteks data berupa teks adalah *text mining*. *Text mining* adalah proses menemukan informasi dalam koleksi dokumen, dan mengidentifikasi secara otomatis pola yang terbentuk, dan berhubungan dengan informasi yang didapat dari kumpulan data yang tidak terstruktur (Feldman & Sanger, 2007). Pengolahan teks mining diterapkan dalam suatu analisis sentimen. Menurut Liu (2010), Analisis Sentimen merupakan proses memahami, mengekstrak dan mengolah data berupa teks (baik itu dokumen, kalimat, dan entitas) untuk mendapatkan informasi sentimen yang terkandung cenderung bersifat positif, negatif atau netral terhadap suatu topik. Sebuah tulisan dapat diklasifikasikan ke dalam kelas positif jika tulisan tersebut bersifat baik dan setuju akan suatu hal, tulisan akan diklasifikasikan negatif jika tulisan tersebut tidak baik, tidak setuju atau menentang (Cahyadi, dkk., 2020).

Analisis sentimen memiliki manfaat untuk berbagai bidang mulai dari bidang pendidikan, ekonomi, maupun kesehatan. Analisis sentimen biasanya banyak digunakan untuk melihat kecenderungan penilaiannya terhadap barang, jasa maupun sistem dari suatu perusahaan. Proses dalam analisis sentimen dapat berupa pengklasifikasian ulasan atau penilaian suatu sistem dalam status positif dan negatif. Klasifikasi teks otomatis merupakan salah satu solusi untuk menangani masalah ini. Klasifikasi teks merupakan proses menemukan kesamaan

dalam dokumen, corpus, maupun kelompok-kelompok dari dokumen yang telah dilabeli sebelumnya (*supervised learning*), berdasarkan topik, tema yang ditunjukkan oleh koleksi dokumen (Chakraborty, dkk., 2013). sehingga hasil dari analisis bisa digunakan untuk perbaikan kebijakan selanjutnya.

Recurrent Neural Network (RNN) adalah salah satu metode pengolahan teks untuk melakukan analisis sentimen. Dengan menggunakan *word embedding* pada RNN data secara sekuensial seperti teks, suara dan video dapat memproses lebih lanjut. RNN memiliki kelemahan dalam mengelola data yang bersifat sekuensial panjang yakni mengalami *vanishing gradient*, maka dirancanglah *Long Short-Term Memory* (LSTM) untuk dapat menutupi kekurangan RNN tersebut (Hochreiter & Schmidhuber, 1997). Menurut Jelodar, dkk, (2020), LSTM adalah metode yang lebih baik dibandingkan dengan konvensional, metode LSTM sangat cocok untuk diterapkan pada analisis sentimen. Kemudian menurut Aprian & Nastiti (2020), bahwa *Long Short Term Memory* (LSTM) merupakan pengembangan dari *Recurrent Neural Network* (RNN) dengan penambahan berupa *memory cell* untuk menyimpan informasi dalam jangka waktu panjang algoritma ini terdiri atas *cell state*, *gate units*, dan *output gate*.

Setiap perusahaan atau instansi memerlukan analisis sentimen sebagai cara untuk mengetahui keluhan pelanggan dan selanjutnya dapat diterapkan untuk mengembangkan sistem supaya menjadi lebih baik. Analisis sentimen dapat diterapkan pada salah satu badan publik yang bergerak dibidang asuransi kesehatan yaitu Badan Penyelenggara Jaminan Sosial (BPJS) Kesehatan. BPJS Kesehatan menyediakan aplikasi bernama Mobile JKN yang ditujukan agar mempermudah pelayanan administrasi ataupun pemberian informasi bagi masyarakat indonesia secara online. Terdapat evaluasi yang bisa dinilai untuk mengetahui berbagai tanggapan masyarakat berupa kritik dan saran terhadap pelayanan BPJS Kesehatan. Berdasarkan ulasan pengguna aplikasi di Google Play Store dapat dinilai bagaimana masyarakat memanfaatkan fasilitas penyelenggaraan JKN-KIS melalui aplikasi ini. Ulasan ini pun dapat digunakan

untuk mengevaluasi kinerja aplikasi sehingga dapat membantu *developer* dari pihak BPJS Kesehatan untuk terus mengembangkan aplikasi tersebut.

Berdasarkan penelitian sebelumnya seperti yang dilakukan Astari, dkk. (2021), mengenai Analisis Sentimen Multi-Class pada Sosial Media menggunakan metode *Long Short-Term Memory (LSTM)* menunjukkan performa yang baik metode LSTM dengan nilai rata-rata akurasi sebesar 89.45%. Pada penelitian selanjutnya yaitu melakukan klasifikasi teks multilebel pada artikel berita menggunakan metode *Long Short-Term Memory (LSTM)*, mendapatkan hasil akurasi tertinggi model kelima dengan 95,38% (Sari, dkk., 2017). Dan penelitian oleh Farsiah, dkk. (2022), mengenai Analisis Sentimen Menggunakan Arsitektur *Long Short Term Memory (LSTM)* terhadap Fenomena *Citayam Fashion Week* dan diperoleh akurasi terbaik sebesar 88%. Oleh karena itu, peneliti tertarik untuk mengusulkan penelitian berupa Implementasi Metode *Long Short-Term Memory (LSTM)* untuk klasifikasi ulasan aplikasi mobile JKN. Diharapkan dengan adanya metode ini dapat melakukan Implementasi Metode *Long Short-Term Memory (LSTM)* Pada Klasifikasi ulasan aplikasi Mobile JKN menjadi kategori positif ataupun negatif.

1.2 Tujuan Penelitian

Tujuan dari penelitian ini adalah sebagai berikut:

1. Menentukan mekanisme analisis sentimen pada klasifikasi ulasan menggunakan metode *Long Short-Term Memory (LSTM)* pada aplikasi Mobile JKN.
2. Mengetahui tingkat performa metode *Long Short-Term Memory (LSTM)* dalam melakukan klasifikasi data ulasan.
3. Mengetahui berbagai ulasan negatif mengenai Aplikasi Mobile JKN.

1.3 Manfaat Penelitian

Manfaat dalam penelitian ini adalah sebagai berikut:

1. Dapat mengetahui mekanisme analisis sentimen pada klasifikasi ulasan menggunakan metode *Long Short-Term Memory (LSTM)* pada aplikasi Mobile JKN.
2. Dapat mengetahui tingkat performa metode *Long Short-Term Memory (LSTM)* dalam melakukan klasifikasi data ulasan.
3. Dapat menambah referensi mengenai penerapan metode *Long Short-Term Memory (LSTM)*.
4. Dapat mengetahui strategi untuk menangani keluhan pengguna aplikasi Mobile JKN.

II. TINJAUAN PUSTAKA

2.1 Analisis Sentimen

Menurut Akbari, dkk. (2012), Analisis sentimen merupakan salah satu cabang ilmu dari *text mining*, *natural language program*, dan *artificial intelligence*. Proses yang dilakukan oleh analisis sentimen untuk memahami, mengekstrak, dan mengolah data teks secara otomatis sehingga menjadi suatu informasi yang bermanfaat. Analisis sentimen atau opini mining digunakan dalam pengelolaan bahasa alami, komputasi linguistik, dan *text mining*. Analisis sentimen bertujuan untuk menentukan perilaku ataupun opini yang diberikan oleh penulis pada topik tertentu. Perilaku tersebut dapat mengindikasikan penilaian serta alasan dan kondisi kecenderungan (Basari, dkk., 2013). Menurut Manalu (2014), Hasil yang diberikan oleh analisis sentimen bisa berupa teks bersifat positif, negatif, dan netral. Tidak hanya mengelompokkan teks secara positif, negatif, dan netral, analisis sentimen dapat menyatakan perasaan emosional, gembira, sedih, dan marah.

2.2 Text Mining

Text mining adalah ilmu yang bertujuan untuk memproses teks agar menjadi informasi yang diperoleh dari peramalan pola dan kecenderungan melalui pola statistik. Teks yang diolah bisa berupa teks terstruktur dan teks tidak terstruktur.

Text mining mengacu pada *information retrieval*, *data mining*, *machine learning*, statistik dan komputasi linguistik (Han, dkk., 2012).

2.3 Pre-Proceccing Data

Menurut Nurrohmat & Azhari (2019), *Text preprocessing* merupakan Langkah yang sangat penting dalam analisis sentimen, data yang telah dikumpulkan perlu dilakukan *preprocessing* sehingga menghasilkan *clean data* yang digunakan untuk membuat *word vectors* dan klasifikasi sentimen menjadi lebih akurat. Langkah langkah *preprocessing* sebagai berikut:

2.3.1 Case Folding

Case folding merupakan proses pengubahan semua huruf menjadi huruf kecil, hal ini diperlukan untuk menyeragamkan teks (Nurrohmat & Azhari, 2019).

2.3.2 Filtering

Menurut Nurrohmat & Azhari (2019) *filtering* adalah proses menghilangkan beberapa karakter spesial yang tidak dibutuhkan seperti titik(.), koma(,), tanda tanya(?), angka(0-9), dan apapun yang dianggap tidak diperlukan dalam proses *training* nantinya. Hal ini dikarenakan karakter karakter tersebut dianggap tidak memiliki efek yang signifikan dalam menentukan label.

2.3.3 *Stopword Removal*

langkah pre-processing selanjutnya yaitu penghapusan kata umum akan untuk mengurangi jumlah kata yang disimpan oleh sistem (Schutze, dkk., 2009). *Stopword* sendiri merupakan daftar kata umum yang tidak memiliki arti penting dan tidak digunakan.

2.3.4 *Tokenize*

Tokenize merupakan proses pemotongan kalimat menjadi kata per kata, hal ini dilakukan dengan cara memotong dengan acuan tiap spasi pada suatu kalimat (Nurrohmat & Azhari, 2019).

2.4 *Word Embedding*

Menurut Bengio, dkk. (2003), *Word embedding* adalah sebuah fungsi parameter yang memetakan setiap kata ke dalam vektor berdimensi tinggi. Keunggulan *word embedding* tidak membutuhkan anotasi, dapat langsung diturunkan dari korpus tak teranotasi. *Word Embedding* digunakan untuk representasi makna yang lebih baik karena adanya keterbatasan informasi pada teks pendek (Wang, dkk., 2018). *Word embedding* dapat dibuat langsung dari dataset yang dimiliki atau menggunakan *pre-trained dan trained word embedding* yang telah tersedia. Penggunaan *word embedding* ini harus disesuaikan dengan domain dari kasus yang dimiliki. Setelah seluruh kata memiliki token masing-masing, setiap data diubah menjadi *sequence* token berdasarkan *vocabulary* yang telah dibuat. Berikut merupakan beberapa algoritma yang dapat digunakan untuk *word embedding*.

1. Word2Vec

Word2vec merupakan salah satu algoritma word embedding yang memetakan setiap kata dalam teks ke dalam vektor. Algoritma word2vec ini diciptakan oleh Mikolov dkk. pada tahun 2013. Sejak kemunculannya, model word embedding ini banyak digunakan dalam penelitian NLP. Word2vec merepresentasikan kata ke dalam vektor yang dapat membawa makna semantik dari kata tersebut. Model word embedding ini merupakan salah satu aplikasi unsupervised learning menggunakan neural network yang terdiri dari sebuah hidden layer dan fully connected layer. Dimensi dari matriks bobot pada setiap layer adalah jumlah dengan kata dalam korpus dikalikan dengan jumlah hidden neuron pada hidden layer-nya. Matriks bobot pada hidden layer dari model yang telah dilatih digunakan untuk mentransformasikan kata ke dalam vektor. Matriks bobot ini merupakan bentuk *one hot encoded* pengkodean yang mewakili setiap kata pada kalimat tersebut dimana angka satu menunjukkan pada indeks keberapa kata tersebut berada dan indeks lain diisi dengan nol. Terdapat dua algoritma word2vec yaitu Continuous Bag-of-Words (CBOW) dan Skip-gram. Model ini menggunakan konteks untuk memprediksi target kata. CBOW memiliki waktu training lebih cepat dan memiliki akurasi yang sedikit lebih baik untuk frequent words. Skip-Gram Model ini menggunakan sebuah kata untuk memprediksi target konteks. Skip-Gram bekerja dengan baik dengan data pelatihan yang jumlahnya sedikit dan dapat merepresentasikan kata-kata yang dianggap langka.

2. Glove

Berbeda dengan word2vec yang hanya mengandalkan informasi lokal dari kata dengan local context window (CBOW dan Skip-gram), algoritma GloVe juga menggabungkan informasi co-occurrence kata atau statistik global untuk memperoleh hubungan semantik antarkata dalam korpus. GloVe menggunakan metode global matrix factorization, matriks yang mewakili kemunculan atau ketiadaan kata-kata dalam suatu dokumen. Pelatihan model GloVe bertujuan untuk mempelajari vektor kata sedemikian rupa sehingga dot product katakata tersebut sama dengan logaritma probabilitas katakata untuk muncul bersama atau probabilitas cooccurrence nya.

3. FastText

Menurut Bojanowski, dkk. (2017), FastText adalah metode word embedding yang merupakan pengembangan dari word2vec. Metode ini mempelajari representasi kata dengan mempertimbangkan informasi subword. Setiap kata direpresentasikan sebagai sekumpulan karakter ngram. Dengan demikian dapat membantu menangkap arti kata-kata yang lebih pendek dan memungkinkan embedding untuk memahami sufiks dan prefiks dari kata. Representasi vektor dikaitkan dengan setiap karakter ngram, sedangkan kata-kata direpresentasikan sebagai jumlah dari representasi vektor tersebut. Setelah kata direpresentasikan dengan karakter n-gram, model Skipgram dilatih untuk mempelajari embedding vektor dari kata. Pada umumnya model yang mempelajari representasi kata ke dalam vektor mengabaikan morfologi kata, setiap kata memiliki vektor yang berbeda. Hal ini menjadi keterbatasan untuk merepresentasikan kata dari bahasa dengan kosakata yang besar dan memiliki banyak katakata langka. FastText memiliki kinerja yang baik, dapat melatih model pada dataset yang besar dengan cepat dan dapat memberikan representasi kata yang tidak muncul dalam data latih. Jika kata tidak muncul selama pelatihan model, kata tersebut dapat dipecah menjadi n-gram untuk mendapatkan embedding vektornya.

2.5 Klasifikasi

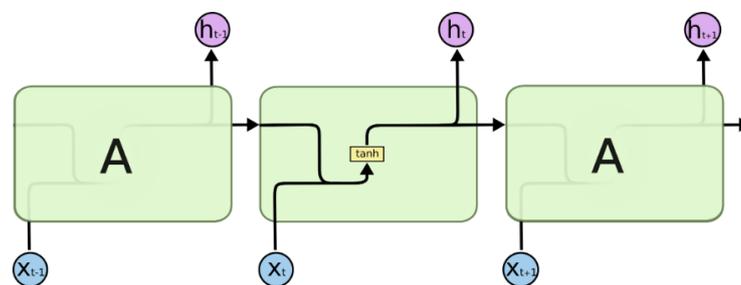
Menurut Han & Kamber (2006), Klasifikasi merupakan suatu proses pengelompokan data ataupun kumpulan fakta yang telah memenuhi suatu kriteria tertentu. Pendapat lain dari klasifikasi adalah model pada bidang ilmu data mining dimana *classifier* dikonstruksi untuk melakukan prediksi kategori atau kelas dari suatu data. Pengklasifikasian suatu data perlu melalui 2 proses terlebih dahulu. Proses awal yang perlu dilakukan adalah pelatihan atau *training* yang dilakukan untuk menganalisis data latih untuk menjadi model prediksi. Setelah proses pelatihan terpenuhi, baru dijalankan proses klasifikasi. Proses klasifikasi dilakukan untuk mengestimasi akurasi data yang didapat dari hasil model prediksi

yang diuji dengan data test atau uji. Jika akurasi yang didapat sesuai, maka model tersebut dapat digunakan untuk prediksi kelas atau kategori data yang belum diketahui.

2.6 Long Short Term Memory (LSTM)

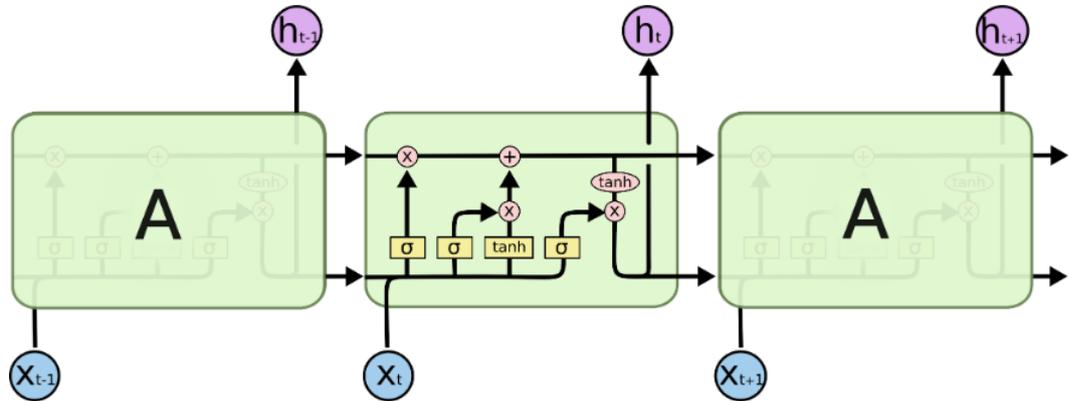
LSTM merupakan salah satu variasi dari RNN (Hochreiter & Schmidhuber, 1997). LSTM dapat menyelesaikan Permasalahan yang ada pada RNN yaitu ketidakmampuan dalam menyimpan informasi ketika proses *learning* apabila terlalu banyaknya informasi yang harus disimpan.

LSTM cocok digunakan untuk masalah yang memiliki ketergantungan jangka panjang. Mengingat informasi jangka panjang adalah perilaku bawaan LSTM. LSTM memiliki struktur rantai yang hampir sama dengan struktur RNN, perbedaannya terletak pada struktur modul pengulangannya. Seluruh Recurrent Neural Network memiliki bentuk rantai modul pengulangan jaringan syaraf tiruan. Dalam RNN standar, modul pengulangan ini akan memiliki struktur yang sangat sederhana, seperti lapisan *tanh* tunggal.



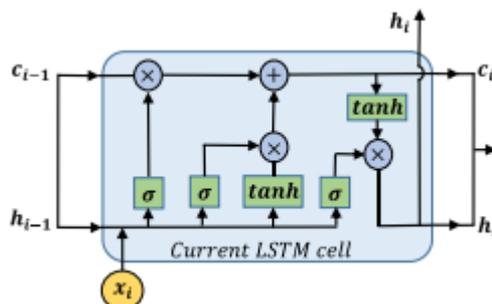
Gambar 1. Pengulangan RNN 1 layer

LSTM juga memiliki struktur seperti rantai, namun modul pengulangan memiliki struktur yang berbeda. Sebagai gantinya, LSTM memiliki empat lapisan jaringan syaraf tunggal yang berinteraksi dengan cara yang berbeda dengan RNN.



Gambar 2. Pengulangan LSTM empat layer

Pada Gambar 2 setiap baris membawa keseluruhan vektor, dari output satu simpul ke input lain. Lingkaran merah muda mewakili operasi pointwise, seperti penambahan vektor, sedangkan kotak kuning merupakan lapisan jaringan syaraf tiruan yang dipelajari. Penggabungan garis menunjukkan rangkaian (concatenation), sementara garis bercabang menunjukkan kontennya disalin dan salinannya masuk ke lokasi yang berbeda. Cell state atau garis horizontal yang melewati bagian atas diagram merupakan kunci pada algoritma LSTM. Modul pengulang dalam LSTM berisi empat layer seperti *conveyor belt*. yang berjalan ke seluruh rantai hanya dengan sedikit interaksi linear.



Gambar 3. Ilustrasi cell state

LSTM memiliki kemampuan untuk menambah dan menghapus informasi ke cell state, dan diatur secara teliti oleh struktur yang disebut gates. Gates adalah cara untuk melepas informasi yang lewat. Mereka terdiri dari lapisan jaringan saraf sigmoid dan operasi perkalian pointwise.

Berikut adalah langkah-langkah pada LSTM:

1. Menentukan informasi yang akan dibuang dari cell state. Informasi ini terletak pada layer sigmoid atau yang juga disebut forget gate. Sesuai dengan namanya, layer ini berperan untuk menyeleksi informasi yang akan dilupakan. Forget gate menghasilkan angka 0 dan 1 untuk cell state C_{t-1} . Angka 1 merepresentasikan untuk menjaga memori dan 0 representasi untuk melupakan memori.

$$f_t = \sigma((W_f \cdot x_t)(U_f \cdot h_{t-1}) + b_f) \quad (2.1)$$

Keterangan rumus :

- f_t : Forget gate
 x_t : input variabel x ke t
 σ : Fungsi sigmoid
 W_f : Bobot dari forget gate
 h_{t-1} : hidden state sebelumnya
 b_f : Bias forget state

2. Menentukan informasi yang akan masuk ke cell state

Pada tahap ini terdapat dua layer yaitu layer sigmoid dan layer *tanh*. Layer sigmoid dinamakan input layer yang memutuskan nilai mana yang akan diperbarui. Selanjutnya layer *tanh* berperan untuk membentuk vektor dari nilai kandidat baru \vec{C}_t yang dapat ditambahkan ke cell state. Lalu cell state yang lama, C_{t-1} di-update menjadi cell state baru, C_t .

$$i_t = \sigma((W_i \cdot x_t)(U_i \cdot h_{t-1}) + b_i) \quad (2.2)$$

$$\vec{C}_t = \tan h ((W_{\vec{C}_t} \cdot x_t)(U_{\vec{C}_t} \cdot h_{t-1}) + b_{\vec{C}_t}) \quad (2.3)$$

Keterangan rumus :

- i_t : Input gate

- \vec{C}_t : *cell* aktivasi
 σ : Fungsi sigmoid
 $\tan h$: Fungsi aktivasi
 x_t : input variabel x ke t
 W_i : Bobot dari *input gate*
 $W_{\vec{C}_t}$: Bobot dari *cell* aktivasi
 h_{t-1} : *hidden state* sebelumnya
 b_i : Bias *input state*
 $b_{\vec{C}_t}$: Bias *cell* aktivasi

3. Menambahkan informasi baru

Selanjutnya akan dieksekusi apa yang sudah diputuskan pada tahap sebelumnya.

$$C_t = f_t * C_{t-1} + i_t * \vec{C}_t \quad (2.4)$$

Keterangan rumus :

- f_t : *Forgate gate*
 i_t : *Input gate*
 C_t : *cell gate*
 \vec{C}_t : *cell* aktivasi
 C_{t-1} : *cell gate* sebelumnya

4. Menemukan output

Akan dijalankan layer sigmoid yang menentukan bagian dari *cell state* mana yang akan dijadikan output. Lalu masukkan *cell state* melewati *tanh* (untuk memaksa nilainya menjadi antara -1 dan 1) dan mengalikannya dengan hasil dari layer sigmoid agar kita hanya menghasilkan bagian yang telah ditentukan.

$$o_t = \sigma((W_o \cdot x_t)(U_o \cdot h_{t-1}) + b_o) \quad (2.5)$$

$$h_t = o_t * \tanh(C_t) \quad (2.6)$$

Keterangan rumus :

- o_t : *Output gate*

σ	: Fungsi sigmoid
W_o	: Bobot dari <i>output gate</i>
$\tan h$: Fungsi aktivasi
h_{t-1}	: <i>hidden state</i> sebelumnya
x_t	: input variabel x ke t
b_o	: Bias <i>output state</i>
C_t	: Bobot dari <i>cell gate</i>

2.7 Fungsi Aktivasi

a. Sigmoid

Menurut Putra (2016), masukan untuk fungsi aktivasi ini berupa bilangan real antara 0 sampai 1. Berikut perhitungan dari fungsi aktivasi sigmoid.

$$g(x) = \frac{1}{1 + e^{-x}} \quad (2.7)$$

Fungsi sigmoid mentransformasikan range nilai dari input x menjadi antara 0 dan 1. Jika input yang dimasukkannya sangat negatif maka keluaran yang didapatkan adalah 0 sedangkan, jika input sangat positif maka nilai keluaran yang didapatkan adalah 1.

b. Tan h

Fungsi Tan h merupakan fungsi nonlinear yang inputnya merupakan bilangan real dengan range antara -1 sampai 1. Berikut persamaan fungsi aktivasi tan h.

$$g(x) = \frac{e^x - e^{-x}}{e^x + e^{-x}} \quad (2.8)$$

Fungsi tan h merupakan pengembangan dari fungsi sigmoid yang memiliki output *zero centered* (Suhermi, dkk., 2018).

c. Softmax

Menurut Keen (2018), Fungsi softmax yang akan digunakan untuk model multi klasifikasi yang akan mengembalikan peluang dari masing masing kelas dan kelas target akan memiliki probabilitas tinggi. Dengan rentang probabilitas output antara nilai 0 sampai 1 dan jumlah semua probabilitasnya akan sama dengan satu.

Berikut merupakan persamaan fungsi aktivasi softmax.

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

2.8 Evaluasi Model

Dalam evaluasi model dihitung nilai akurasi, *Recall*, Presisi, dan *F1-Score*. Pembagian data pada penelitian tersebut akan dibagi menjadi 2 bagian yaitu sebagai data *training* dan data *testing*, pembagian dataset ini menjadi 70% data *training* dan 30% data *testing*. Data *training* ini merupakan data yang digunakan untuk *training*/pelatihan model, Data *testing* sendiri merupakan *testing* model yang akan dijadikan pertimbangan atau simulasi penggunaan model pada dunia nyata. Data *training* dan data *testing* tersebut akan dilakukan proses *preprocessing*. Menghitung performa klasifikasi menggunakan *Confusion matrix*. *Confusion matrix* sendiri biasanya berbentuk table yang digunakan untuk menghitung performa dari sebuah proses atau model klasifikasi dimana true false

telah diketahui. Berikut merupakan dan tabel *Confusion matrix*.

Tabel 1. *Confusion matrix*

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

Tabel 1 merupakan table *confusion matrix* yang digunakan untuk *binary classifier* yang berarti 2 *class*.

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

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

$$Akurasi (\%) = \frac{TP + TN}{TP + TN + FP + FN} \times 100\% \quad 2.9$$

$$Precision (\%) = \frac{TP}{TP + FP} \times 100\% \quad 2.10$$

$$Recall (\%) = \frac{TP}{TP + FN} \times 100\% \quad 2.11$$

$$F1 - Score (\%) = \frac{2 \times precision \times recall}{precision + recall} \times 100 \quad 2.12$$

2.9 BPJS Kesehatan dan Ulasan Aplikasi Mobile JKN

Badan Penyelenggara Jaminan Sosial dan Kesehatan (BPJS) adalah badan hukum yang dibentuk untuk menyelenggarakan program jaminan kesehatan nasional. BPJS Kesehatan mempunyai visi dan misi untuk mewujudkan jaminan kesehatan yang berkualitas dan memberikan layanan terbaik kepada peserta dan masyarakat Indonesia. BPJS Kesehatan dengan Program Jaminan Kesehatan Indonesia Sehat (JKN-KIS) berupaya untuk memperluas cakupan kepesertaan dengan targetnya adalah dapat mencakup seluruh masyarakat Indonesia sehingga mencapai *Universal Health Coverage* (UHC). Dengan meningkatnya jumlah peserta, permasalahan yang sering dihadapi oleh peserta JKN-KIS selama ini diantaranya adalah lamanya antrian di pendaftaran dan pelayanan kesehatan lainnya. Hal ini mendorong BPJS Kesehatan menciptakan inovasi baru yaitu membuat aplikasi mobile-JKN. Harapannya aplikasi ini dapat membantu memperbaiki pelayanan BPJS Kesehatan. Aplikasi mobile-JKN pertama kali diluncurkan pada November 2017. Namun, sistem online dari Mobile – JKN belum sepenuhnya sempurna sehingga banyak penilaian mengenai aplikasi mulai dari komentar negatif hingga positif.

III. METODOLOGI PENELITIAN

3.1 Waktu dan Tempat Penelitian

Penelitian ini dilakukan pada Semester ganjil tahun akademik 2022/2023 di Jurusan Matematika Fakultas Matematika dan Ilmu Pengetahuan Alam Universitas Lampung.

3.2 Data Penelitian

Data yang digunakan pada Penelitian ini adalah ulasan yang diberikan oleh pengguna aplikasi Mobile JKN yang sumbernya dari kolom komentar Google Play Store mulai dari tanggal 12 September 2022 sampai 10 Oktober 2022 Data ulasan diambil dengan menggunakan metode scraping data dari aplikasi AppFollow. Selanjutnya data yang diperoleh dilakukan pelabelan sebanyak 5410 data yang terdiri dari data berlabel “Positif” yang disimbolkan dengan angka 0 dan data berlabel “Negatif” yang disimbolkan dengan angka 1.

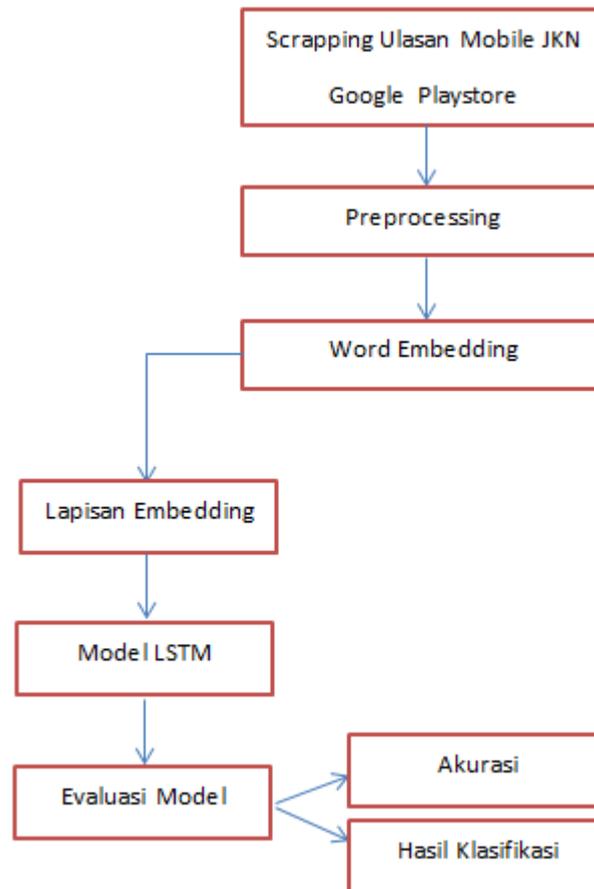
3.3 Metode Penelitian

Penelitian ini menggunakan metode *Long Short-Term Memory (LSTM)* untuk melihat performa model yang dibentuk dalam mengklasifikasikan ulasan aplikasi

Mobile JKN. Data diolah menggunakan bantuan bahasa pemrograman Python. Langkah-langkah yang dilakukan adalah sebagai berikut:

1. Melakukan proses *Scraping* data yang merupakan proses penambahan atau pengumpulan data teks ulasan aplikasi Mobile JKN dari Google Play Store Menggunakan bantuan aplikasi AppFollow.
2. Melakukan Pelabelan data menjadi 2 kategori label 0 untuk sentimen positif dan label 1 untuk sentimen negatif.
3. Melakukan Case folding untuk merubah teks dalam dokumen menjadi bentuk standar yakni lowercase.
4. Melakukan Stopword Removal untuk penghapusan kata berdasarkan kata yang terdapat dalam stoplist.
5. Melakukan *Filtering* untuk menguraikan kata menjadi bentuk kata dasar.
6. Melakukan Tokenize untuk pemecahan teks menjadi kata, dengan batasan tanda baca dan spasi.
7. Melakukan Word Embedding yang merupakan tahapan untuk mengkonversi data teks menjadi bentuk vektor
8. Pembagian data *training* dan *testing* dengan 70% *training* dan 30% data *testing* .
9. Membangun model untuk Klasifikasi dengan metode *Long Short-Term Memory (LSTM)*.
10. Evaluasi model dengan menentukan nilai akurasi pada *confusion matrix*.

Alur pengolahan data digambarkan pada diagram berikut ini.



Gambar 4. Alur Analisis sentimen Mobile JKN

V. KESIMPULAN

Berdasarkan penelitian yang dilakukan dengan dataset sebanyak 5410 yang terbagi menjadi 2 bagian yaitu 3989 data berlabel “Positif” dan 1421 berlabel “Negatif”, dengan perbandingan 30% data *testing* 70% data *training*. Pada data *training* menghasilkan nilai akurasi sebesar 94.37% setelah mendapatkan hasil akurasi didapatkan nilai *Recall* yang dihasilkan 90.64%, Presisi sebesar 94.71% dan nilai akurasi untuk data *testing* sebesar 76.85%. Dengan demikian dapat disimpulkan bahwa metode *Long short-term memory* (LSTM) layak digunakan untuk klasifikasi Ulasan pada Aplikasi Mobile JKN dengan sentimen negatif berupa keluhan pelanggan mengenai performa aplikasi yang sebagian besar mengeluh pada proses pendaftaran maupun login.

DAFTAR PUSTAKA

- Aprian, Y.A. & Nastiti, V.R.S. 2020. Prediksi pendapatan kargo menggunakan arsitektur Long Short Term Memory. *Jurnal Komputer Terapan (JKT)*. **6**(2): 148-157.
- Akbari, M.I.H.A.D., Novianty, A., & Setianingsih, C. 2012. Analisis Sentimen Menggunakan Metode Learning Vector Quantization. *Proceedings of Engineering*. **4**(2): 2283-2292.
- Astari, Y., Afiyati, A., & Rozaqi, S.W. 2021. Analisis Sentimen Multi-Class pada Sosial Media menggunakan metode *Long Short-Term Memory* (LSTM). *Jurnal Linguistik Komputasi*. **4**(1): 9-12.
- Basari, A.S.H., Hussin, B., Ananta, I.G.P., & Zeniarja, J. 2013. Opinion Mining of Movie Review using Hybrid Method of Support Vector Machine and Particle Swarm Optimization. *Procedia Engineerin*. **53**(2): 453-462.
- Bengio, Y., Ducharme, R., Vincent, P., & Jauvin, C. 2003. A Neural Probabilistic Language Model. *Journal of Machine Learning Research*. **19**(3): 11-15.
- Bojanowski, P., Grave, E., Joulin, A., & Mikolov, T. 2017. Enriching word vectors with subword information. *Transactions of the association for computational linguistics*. **5**: 135-146.
- Cahyadi, R., Damayanti, A., & Aryadani, D. 2020. Recurrent neural network (rnn) dengan long short term memory (lstm) untuk analisis sentimen data instagram. *J. Inform. dan Komput.* **5**(1): 1-9

- Chakraborty, G., Pagolu, M., & Garla, S. 2014. Text mining and analysis: practical methods, examples, and case studies using SAS. *SAS Institute*.
- Farsiah,R., Misbulla, A., & Hermawan, F. 2022. Analisis Sentimen Menggunakan Arsitekur Long Short Term Memory (LSTM) Terhadap Fenomena Citayam Fashion Week. *Jurnal Pendidikan Teknologi Informasi*. **6**(2): 86-94.
- Feldman, R. & Sanger, J. 2007. *The Text Mining Handbook*. Cambridge University Press, New York.
- Han, J., & Kamber, M. 2006. *Classification and prediction Data mining: Concepts and techniques Second Edition*. Morgan Kaufmann, Kanada.
- Han, J., Kamber, M. & Pei, J. 2012. *Data mining: Concepts and techniques Third Edition*. Morgan Kaufmann, Kanada.
- Hochreiter, S. & Schmidhuber, J. 1997. Long Short-Term Memory. *Neural Computation*. **9**(2): 1730-1780.
- Jelodar, H., Wang, Y., Orji, R., & Huang, S. 2020. Deep sentiment classification and topic discovery on novel coronavirus or COVID-19 online discussions: NLP using LSTM recurrent neural network approach. *IEEE Journal of Biomedical and Health Informatics*. **24**(10): 2733-2742.
- Keen, A.G. 2018. From Softmax to Sparsemax:A Sparse Model of Attention and Multi-Label Classification Andr'e. *Dynast Lycia*. **48**(2): 1-10.
- Liu,B. 2010. *Handbook of Natural Language Processing, chapter Sentiment Analysis and Analysis*. 2nd Edition. CRC Press, Prancis.
- Manalu, B.U. 2014. Analisis Sentimen Pada Twitter Menggunakan Text Mining. Doctoral dissertation, Universitas Sumatera Utara.
- Nurrohmat, M.A., & Azhari, S.N. 2019. Sentiment Analysis of Novel Review Using Long ShortTerm Memory Method. *Indonesian Journal of Computing and Cybernetics Systems*. **13**(3): 209–218.

- Putra, W.S.E. 2016. Klasifikasi citra menggunakan convolutional neural network (CNN) pada caltech 101. *Jurnal Teknik ITS*. **5**(1): A65-A69.
- Sari, W.K., Rini, D.P., Malik, R.F., & Azhar, I.S.B. 2017. Klasifikasi teks multilabel pada artikel berita menggunakan long short-term memory dengan Word2Vec. *Jurnal Rekayasa Sistem & Informasi*. **1**(10): 276–285.
- Schütze, H., Manning, C. D., & Raghavan, P. 2008. Introduction to information retrieval. *Cambridge University Press*. **39**: 234-265
- Suhermi, N., Suhartono, I., & Dwi, D. 2018. Pemeliharaan Arsitektur Terbaik pada Model Deep Learning Melalui Pendekatan Desain Eksperimen pada Peramalan Deret Waktu Nonlinier. *Jurnal Statistika*. **5**(1): 153-1
- Wang, J.H., Liu, T.W., Luo, X., & Wang, L. 2018. An LSTM Approach to Short Text Sentiment Classification with Word Embeddings. *In Proceedings of the 30th conference on computational linguistics and speech processing (ROCLING 2018)*. 214-223.