

PREDIKSI KATA BERIKUTNYA UNTUK PENCARIAN JUDUL BUKU MENGGUNAKAN ALGORITMA BI-LSTM

Laporan ini dibuat untuk memenuhi persyaratan program Tugas Akhir



Universitas Logistik & Bisnis Internasional

Dibuat Oleh,

1.19.40.04 Alwizain Almas Trigreisian

**PROGRAM DIPLOMA IV TEKNIK INFORMATIKA
SEKOLAH VOKASI**

UNIVERSITAS LOGISTIK DAN BISNIS INTERNASIONAL

BANDUNG

2023

NEXT WORD PREDICTION FOR BOOK TITLE SEARCH USING BI-LSTM ALGORITHM

This report is made to fulfill the requirements of the Final Project program



Universitas Logistik & Bisnis Internasional

Created by,

1.19.40.04 Alwizain Almas Trigresian

**DIPLOMA IV PROGRAM IN INFORMATICS ENGINEERING
SEKOLAH VOKASI**

UNIVERSITAS LOGISTIK DAN BISNIS INTERNASIONAL

BANDUNG

2023

LEMBAR PENGESAHAN

PREDIKSI KATA BERIKUTNYA UNTUK PENCARIAN JUDUL BUKU MENGGUNAKAN ALGORITMA BI-LSTM

Alwizain Almas Trigreisian

1.19.40.04

Laporan Program Tugas Akhir ini telah diperiksa, disetujui dan disidangkan
di Bandung, 8 Juli 2023

Oleh:

Pembimbing 1

Pembimbing 2

Nisa Hanum Harani, S.Kom., M.T.

NIK: 215.89.158

Roni Andarsyah, S.T., M.Kom.

NIK: 115.88.193

Menyetujui,
Ketua Program Studi DIV Teknik Informatika

Roni Andarsyah, S.T., M.Kom.

NIK: 115.88.193

LEMBAR PENGESAHAN

PREDIKSI KATA BERIKUTNYA UNTUK PENCARIAN JUDUL BUKU MENGGUNAKAN ALGORITMA BI-LSTM

Alwizain Almas Trigreisian

1.19.40.04

Laporan Program Tugas Akhir ini telah diperiksa, disetujui dan disidangkan
di Bandung, 8 Juli 2023

Oleh:

Pembimbing 1

Pembimbing 2

Nisa Hanum Harani, S.Kom., M.T.

NIK: 215.89.158

Roni Andarsyah, S.T., M.Kom.

NIK: 115.88.193

Menyetujui,
Koordinator Program Tugas Akhir

Rolly Maulana Awangga, S.T., MT.

NIK: 117.86.219

Surat Pernyataan Tidak Melakukan Plagiarisme

Yang bertanda tangan dibawah ini:

Nama : Alwizain Almas Trigreisian
NPM : 1.19.40.04
Program Studi : D4 Teknik Informatika
Judul : Prediksi Kata Berikutnya untuk Pencarian Judul Buku
menggunakan Algoritma Bi-LSTM

Menyatakan bahwa:

1. Program Tugas Akhir saya ini adalah asli dan belum pernah diajukan untuk memenuhi kelulusan matakuliah Tugas Akhir pada Program Studi D4 Teknik Informatika baik di Universitas Logistik dan Bisnis Internasional maupun di Perguruan Tinggi lainnya.
2. Program Tugas Akhir ini adalah murni gagasan, rumusan, dan penelitian saya sendiri tanpa bantuan pihak lain, kecuali arahan pembimbing.
3. Dalam Program Tugas Akhir ini tidak terdapat karya atau pendapat yang telah ditulis atau dipublikasikan orang lain, kecuali secara tertulis dengan jelas dicantumkan sebagai acuan dalam naskah dengan disebutkan nama pengarang dan dicantumkan dalam daftar pustaka.
4. Pernyataan ini saya buat dengan sesungguhnya dan apabila di kemudian hari terdapat penyimpangan-penyimpangan dan ketidakbenaran dalam pernyataan ini, maka saya bersedia menerima sanksi akademik berupa pencabutan gelar yang telah diperoleh karena karya ini, serta sanksi lainnya sesuai dengan norma yang berlaku diperguruan tinggi lain.

Bandung, 8 Juli 2023

Yang Membuat Pernyataan,

Alwizain Almas Trigreisian
NPM. 1.19.40.04

Abstrak

Pencarian judul buku yang sesuai untuk saat ini masih terasa cukup sulit. Kita lebih sering untuk menebak sendiri judul buku apa yang kita inginkan, namun pada kenyataannya judul buku tersebut sering tidak tersedia. Penelitian ini bertujuan untuk mengatasi permasalahan tersebut dengan menghasilkan model prediksi yang akurat dan efisien dalam memprediksi kata-kata berikutnya pada pencarian judul buku dengan menggunakan algoritma *deep learning*, yaitu *Bidirectional Long Short Term Memory* (Bi-LSTM). Tahapan penelitian yang dilakukan terdiri dari *data collection*, *data preprocessing*, *data modeling*, *evaluation*, dan *implementation*. Penelitian ini menggunakan *dataset* judul buku berbahasa Indonesia yang diperoleh dari *website* toko buku *online* XYZ dengan data sebanyak 5618. Hasil penelitian menunjukkan bahwa model *deep learning* yang dihasilkan dapat memprediksi kata-kata berikutnya pada pencarian judul buku dengan akurasi mencapai 81.82%. Model tersebut diimplementasikan berupa aplikasi *website* dengan menggunakan *framework* Django, bahasa Python, dan basis data MySQL.

Kata Kunci: Prediksi Kata Berikutnya, *Deep Learning*, Bi-LSTM, Django, Python.

Abstract

Finding a suitable book title is still quite difficult at the moment. We often guess what book title we want, but in reality the book title is often not available. This research aims to overcome these problems by producing an accurate and efficient prediction model in predicting the next words in book title search using a deep learning algorithm, namely Bidirectional Long Short Term Memory (Bi-LSTM). The research stages consist of data collection, data preprocessing, data modeling, evaluation, and implementation. This research uses a dataset of Indonesian book titles obtained from the XYZ online bookstore website with 5618 data. The results show that the resulting deep learning model can predict the next words in the book title search with an accuracy of 81.82%. The model is implemented in the form of a web application using the Django framework, Python language, and MySQL database.

Keywords: *Next Word Prediction, Deep Learning, Bi-LSTM, Django, Python.*

Kata Pengantar

Assalamu'alaikum Wr. Wb.

Puji syukur kehadiran Allah SWT yang telah melimpahkan rahmat dan hidayah-Nya sehingga penyusunan laporan Tugas Akhir ini dengan judul “Prediksi Kata Berikutnya untuk Pencarian Judul Buku menggunakan Algoritma Bi-LSTM” dapat saya selesaikan dengan lancar tanpa suatu halangan yang berarti. Dalam penyusunan laporan Tugas Akhir ini, saya mengucapkan banyak terima kasih kepada seluruh pihak yang telah membantu. Oleh karena itu, saya mengucapkan terima kasih kepada:

1. Bapak Rolly Maulana Awangga, S.T., MT., CAIP, SFPC. selaku koordinator Tugas Akhir yang telah membantu memberikan arahan kepada penulis dalam penyusunan laporan Tugas Akhir ini.
2. Ibu Nisa Hanum Harani, S.Kom., M.T., SFPC. selaku dosen pembimbing pertama yang telah membantu dalam memberikan bantuan, bimbingan dan arahan dalam penyusunan laporan Tugas Akhir ini.
3. Bapak Roni Andarsyah, S.T., M.Kom., SFPC. selaku dosen pembimbing kedua yang juga telah memberikan banyak bantuan, bimbingan, serta arahan dalam Tugas Akhir ini.
4. Seluruh Dosen di Program Studi D4 Teknik Informatika ULBI, yang tidak bisa disebutkan satu-satu, atas ilmu dan bimbingannya selama penulis berkuliah disana.
5. Teristimewa untuk kedua orangtua penulis, yang telah memberikan kasih sayang, perhatian, dan dukungan moril maupun materil
6. Kedua kakak penulis yang telah banyak memberikan semangat dan dukungan berupa moril serta materil.
7. Teman-teman penulis yang tidak dapat disebutkan satu per satu, yang telah membantu dan rela dalam memberikan masukan serta bantuan dalam pembuatan laporan dan aplikasi ini.
8. Berbagai pihak yang secara langsung maupun tidak langsung telah membantu, baik dukungan moral maupun material.

Penulis menyadari bahwa dalam penyusunan laporan Tugas Akhir ini masih jauh dari kata sempurna. Oleh sebab itu, penulis mengharapkan kritik dan saran yang membangun dari seluruh pihak guna memperbaiki keterbatasan penulis.

Wassalamu'alaikum Wr. Wb.

Bandung, 8 Juli 2023

Penulis

Daftar Isi

LEMBAR PENGESAHAN	iii
LEMBAR PENGESAHAN	iv
Surat Pernyataan Tidak Melakukan Plagiarisme	v
Abstrak	vi
Abstract	vii
Kata Pengantar	viii
Daftar Isi.....	x
Daftar Gambar.....	xii
Daftar Tabel	xiii
BAB I Pendahuluan	1
1.1 Latar Belakang	1
1.2 Identifikasi Masalah	2
1.3 Tujuan dan Manfaat.....	2
1.4 Ruang Lingkup Penelitian	3
1.5 Penelitian Sebelumnya	3
1.6 Sitematika Penulisan	4
BAB II Tinjauan Pustaka dan Landasan Teori	6
2.1 Tinjauan Pustaka	6
2.2 Landasan Teori	7
2.2.1 Prediksi	7
2.2.2 Deep Learning.....	7
2.2.3 RNN (Recurent Neural Network)	8
2.2.4 Bidirectional Long Short Term Memory (Bi-LSTM).....	8
2.2.5 Python	9
2.2.6 Django.....	9
2.2.7 MySQL	9
BAB III Metodologi Penelitian.....	11
3.1 Diagram Alur Metodologi Penelitian	11
3.2 Tahapan-Tahapan Diagram Alur Metodologi Penelitian	11

3.2.1 Data Collection	12
3.2.2 Data Preprocessing	12
3.2.3 Data Modeling	12
3.2.4 Evaluation	13
3.2.5 Implementation	14
BAB IV Hasil dan Pembahasan	16
4.1 Data Collection	16
4.2 Data Preprocessing	17
4.3 Data Modeling	20
4.4 Evaluation	21
4.5 Implementation	23
BAB V Penutup	25
5.1 Kesimpulan dan Saran	25
5.1.1 Kesimpulan	25
5.1.2 Saran	25
Lampiran	26
Daftar Pustaka	41

Daftar Gambar

Gambar 3.1 Alur Metodologi Penelitian.....	11
Gambar 3.2 Arsitektur Bi-LSTM.....	13
Gambar 3.3 Confusion Matrix	14
Gambar 4.1 Dataframe	16
Gambar 4.2 Wordcloud.....	17
Gambar 4.3 Hasil Huruf Kecil dan Penghapusan Tanda Baca	18
Gambar 4.4 Hasil Proses Tokenizing.....	18
Gambar 4.5 Hasil Proses N-gram	19
Gambar 4.6 Hasil Proses Padding.....	19
Gambar 4.7 Hasil Pembagian Data	20
Gambar 4.8 Struktur Layer Model.....	21
Gambar 4.9 Grafik Hasil Akurasi Pelatihan	22
Gambar 4.10 Grafik Hasil Loss Pelatihan	22
Gambar 4.11 Hasil Pengujian	23
Gambar 4.12 Tampilan Web Pencarian Judul Buku	23
Gambar 1 Kode Program Import Library	32
Gambar 2 Kode Program Koneksi Basis Data	32
Gambar 3 Kode Program Pengambilan Data	32
Gambar 4 Kode Program Import Library	33
Gambar 5 Kode Program Load Data.....	33
Gambar 6 Kode Program Wordcloud	33
Gambar 7 Kode Program Pengubahan Huruf Kecil.....	33
Gambar 8 Kode Program Penghapusan Tanda Baca	34
Gambar 9 Kode Program Tokenization	34
Gambar 10 Kode Program N-gram	34
Gambar 11 Kode Program Padding	34
Gambar 12 Kode Program Pembagian Data	35
Gambar 13 Kode Program Model.....	35
Gambar 14 Kode Program Pelatihan	35
Gambar 15 Kode Program Evaluasi.....	35
Gambar 16 Konfigurasi Basis Data settings.py	36
Gambar 17 Konfigurasi Route Project urls.py	36
Gambar 18 Tampilan HTML index.py	37
Gambar 19 Membuat Basis Data Buku model.py.....	37
Gambar 20 Konfigurasi Basis Data Buku admin.py	37
Gambar 21 Konfigurasi Views App views.py	38
Gambar 22 Konfigurasi Route App urls.py	38
Gambar 23 Konfigurasi Logika Model Prediksi model.py	39
Gambar 24 Konfigurasi Views App Search views.py	39
Gambar 25 Konfigurasi Route App Search urls.py	40

Daftar Tabel

Tabel 1.1 Penelitian Sebelumnya.....	3
Tabel 4.1 Pengujian Parameter Model.....	20
Tabel 4.2 Struktur Tabel Buku.....	24

BAB I

Pendahuluan

1.1 Latar Belakang

Seiring perkembangan zaman dan teknologi mengakibatkan pengaruh yang besar terhadap berbagai bidang, seperti halnya pada sistem perdagangan. Melalui penggunaan internet, suatu perusahaan dapat memanfaatkan perkembangan hal tersebut untuk mempromosikan dan menjual produknya dengan jangkauan yang lebih luas sebagai salah satu sarana dalam meningkatkan keuntungan [1]. Salah satu perusahaan yang memanfaatkan perkembangan teknologi seperti halnya internet dengan baik yaitu toko buku *online* XYZ. XYZ merupakan salah satu toko buku *online* yang menyediakan berbagai macam buku fiksi maupun non-fiksi yang berkualitas dari berbagai penerbit. Toko buku *online* tersebut memberikan kemudahan kepada pelanggannya dengan membeli buku tanpa harus pergi ke toko. Namun dibalik kemudahan yang diberikan, terdapat permasalahan berupa sistem pencarian judul buku yang kurang efektif.

Pencarian judul buku menjadi salah satu kegiatan yang kerap kali dilakukan oleh para pembaca dan akademisi dalam mencari informasi dalam buku tersebut. Dengan melakukan pencarian judul buku, seseorang dapat menemukan buku yang tepat dengan kebutuhan dan tujuannya. Namun, dalam melakukan pencarian judul buku tidak selamanya mudah. Para pencari buku lebih sering untuk menebak judul buku apa yang diinginkan, namun judul buku yang diinginkan sering tidak tersedia. Selain itu, kesulitan juga dialami ketika menentukan kata kunci apa yang tepat dalam melakukan pencarian judul buku yang diinginkan. Kemudian hasil pencarian pun juga terkadang tidak relevan dengan kata kunci yang di-*input*-kan.

Untuk mengatasi permasalahan tersebut, dikembangkanlah sebuah sistem prediksi kata berikutnya untuk pencarian judul buku yang dapat membantu pengguna dalam mencari judul buku yang diinginkan dengan lebih efektif dan efisien. Prediksi kata berikutnya merupakan sebuah proses dalam menebak kata yang muncul berikutnya dari sebuah kalimat atau teks. Prediksi memanfaatkan data pada masa lalu untuk memperkirakan yang terjadi dimasa mendatang secara pragmatis dan sistematis dengan harapan dapat memberikan objektivitas besar [2]. Kata berikutnya yang diprediksi berdasarkan kata sebelumnya yang telah di-*input*-kan. Sehingga, sistem yang dikembangkan akan dapat memberikan rekomendasi atau saran kata yang sesuai untuk melengkapi kata kunci atau teks dalam pencarian judul buku.

Sistem yang dikembangkan memanfaatkan metode *Deep Learning* dengan algoritma *Bidirectional Long Short Term Memory* (Bi-LSTM). Implementasi sistem dibangun berupa aplikasi berbasis *web* dengan menggunakan bahasa pemrograman Python. Kemudian terkait *backend* dalam pembuatan *web* pencarian judul buku menggunakan *framework* Django dan basis data MySQL.

1.2 Identifikasi Masalah

Berdasarkan latar belakang, maka dapat diidentifikasi masalah yang terjadi yaitu masih sulitnya mencari judul buku yang sesuai dan hasil pencarian judul buku terkadang tidak relevan dengan kata kunci yang diberikan.

1.3 Tujuan dan Manfaat

Berdasarkan identifikasi masalah yang diperoleh, maka didapatkanlah tujuan penelitian ini yaitu membantu proses pencarian judul buku secara mudah dan efisien dengan menampilkan hasil pencarian yang relevan dengan kata kunci yang diberikan.

1.4 Ruang Lingkup Penelitian

Ruang lingkup yang digunakan pada pembahasan ini agar tidak terlalu luas adalah sebagai berikut:

1. Sebagai syarat untuk memperoleh gelar sarjana terapan komputer.
2. Penelitian ini menggunakan metode *Bidirectional Long Short Term Memory* (Bi-LSTM).
3. Data yang digunakan berbahasa Indonesia bersumber dari toko buku *online XYZ*.
4. Implementasi yang dilakukan hanya sebatas *web* pada *localhost*.

1.5 Penelitian Sebelumnya

Berikut merupakan beberapa penelitian yang telah dilakukan sebelumnya dengan topik yang sama, yaitu:

Tabel 1.1 Penelitian Sebelumnya

No.	Peneliti	Topik	Model
1.	Sourabh Ambulgekar, Sanket Malewadikar, Raju Garande, Dr. Bharti Joshi	Next Words Prediction Using Recurrent Neural Networks [3]	LSTM
2.	Afika Rianti, Suprih Widodo, Atikah Dhani Ayuningtyas, Fadlan Bima Hermawan	Next Word Prediction Using LSTM [4]	LSTM
3.	Radhika Sharma, Nishtha Goel, Nishita Aggarwal, Prajyot Kaur, Chandra Prakash	Next Word Prediction in Hindi Using Deep Learning Techniques [5]	LSTM, Bi-LSTM
4.	Aejaz Farooq Ganai, Farida Khursheed	Predicting next Word using RNN and LSTM cells: Stastical Language Modeling [6]	LSTM
5.	Khrystyna Shakhovska, Iryna Dumyn, Natalia	An Approach for a Next-Word Prediction	LSTM, Markov Chains

	Kryvinska, Mohan Krishna Kagita	for Ukrainian Language [7]	
6.	Partha Pratim Barmana, Abhijit Boruah	A RNN based approach for next word prediction in assamese phonetic transcription [8]	LSTM
7.	Karma Wangchuk, Tandin Wangchuk and Tenzin Namgye	Dzongkha Next Words Prediction Using Bidirectional LSTM [9]	Bi-LSTM
8.	Dr. S. Rajakumar, Dr. V. Rameshbabu, Dr. D. Usha, Nikila K., Ramya Shree B., Sakthi Priya R.	EXO Next Word Prediction Using Machine Learning [10]	Bi-LSTM
9.	K. Chakradhar, K. Sai Kiran, K. Shanmukh, K. Sharath Kumar, K. Dinesh Sagar	Next Word Prediction Using Deep Learning [11]	LSTM, Bi- LSTM
10.	Milind Soam, Sanjeev Thakur	Next Word Prediction Using Deep Learning: A Comparative Study [12]	LSTM, Bi- LSTM
11.	S. Ramya, C.S. Kanimozhi Selvi	Recurrent Neural Network based Models for Word Prediction [13]	RNN, LSTM, Bi-LSTM

1.6 Sitematika Penulisan

Dalam laporan program Tugas Akhir ini terdiri dari bagian yang memiliki pembahasannya masing-masing. Berikut pemaparan dari setiap bagian tersebut:

BAB I PENDAHULUAN, berisi latar belakang, identifikasi masalah, tujuan dan manfaat, ruang lingkup penelitian, penelitian sebelumnya, dan sistematika penulisan.

BAB II TINJAUAN PUSTAKA DAN LANDASAN TEORI, berisi penelitian terdahulu yang terkait dan uraian-uraian definisi yang digunakan dalam penelitian ini.

BAB III METODOLOGI PENELITIAN, berisi mengenai diagram alur metodologi penelitian dan tahapan-tahapan diagram alur metodologi penelitian yang akan digunakan.

BAB IV PENUTUP, berisi kesimpulan dari hasil penelitian, saran yang bersifat membangun dari kesimpulan hasil yang diperoleh, dan lampiran-lampiran yang diperlukan dalam penelitian ini.

BAB II

Tinjauan Pustaka dan Landasan Teori

2.1 Tinjauan Pustaka

Penelitian dengan topik yang sama telah dilakukan seperti penelitian yang memperoleh hasil prediksi kata berikutnya dengan akurasi sebesar 56% dengan menggunakan model LSTM oleh Sourabh Ambulgekar et al. [3]. Selanjutnya pada penelitian yang dilakukan oleh Aejez Farooq Ganai et al. [6] memperoleh hasil perbandingan dengan hasil *many-to-one model* memperoleh akurasi yang terbaik sebesar 48%, sedangkan untuk *many-to-many model* memperoleh akurasi sebesar 19%. Penelitian oleh Khrystyna Shakhovska et al. [7] mendapat hasil algoritma LSTM dan *Markov Chains* untuk prediksi kata dalam bahasa Ukraina dengan hasil akurasi sebesar 74.55%. Penelitian yang dilakukan oleh Partha Pratim Barman et al. [8] menghasilkan model RNN untuk prediksi kata berikutnya dengan bahasa Assamese menggunakan algoritma LSTM. Model tersebut memperoleh akurasi sebesar 72.10%. Penelitian lainnya yang dilakukan oleh Afika Rianti et al. [4] yang menggunakan algoritma LSTM dalam memprediksi kata berikutnya dengan data berbahasa Indonesia mengenai destinasi wisata di Indonesia memperoleh akurasi sebesar 75% dengan *epoch* 200.

Pada penelitian sebelumnya juga telah dilakukan dengan metode yang sama yaitu prediksi kata berikutnya dengan metode Bi-LSTM, seperti penelitian yang dilakukan oleh Radhika Sharma et al. [5] dengan model Bi-LSTM untuk prediksi kata berikutnya menggunakan bahasa Hindi memperoleh akurasi sebesar 79.54%. Penelitian dengan data berbahasa Inggris juga telah dilakukan oleh S. Rajakumar et al. [10] dan S. Ramya et al. [13] menggunakan algoritma Bi-LSTM sebagai model prediksi kata berikutnya dengan memperoleh akurasi masing-masing sebesar 81.07% dan 72%. Penelitian yang dilakukan oleh Karma Wangchuk et al. [9] menggunakan suku kata Dzongkha sebagai data *input*-nya dan algoritma Bi-

LSTM memperoleh akurasi sebesar 73.89%. Penelitian dengan menggunakan suku kata lain juga dilakukan oleh K. Chakradhar et al. [11] dengan menggunakan data suku kata Amharic dan model Bi-LSTM dapat memperoleh akurasi sebesar 76.1%. Penelitian lainnya dengan algoritma Bi-LSTM untuk prediksi kata berikutnya dilakukan oleh Milind Soam et al. [12] dengan perolehan akurasi sebesar 66.1%.

Dengan melihat hasil dari penelitian terdahulu, pada penelitian ini akan dikembangkan sistem prediksi kata berikutnya dengan menggunakan algoritma *Bidirectional Long Short Term Memory* (Bi-LSTM). Metode tersebut telah banyak dilakukan untuk prediksi kata berikutnya, namun pada penelitian ini diusulkan menggunakan data berbahasa Indonesia berupa judul buku dengan mengupayakan memperoleh akurasi terbaik. Sehingga model dan sistem dapat diterapkan untuk memudahkan para pencari buku dalam melakukan pencarian judul buku yang sesuai.

2.2 Landasan Teori

2.2.1 Prediksi

Prediksi merupakan proses memperkirakan dengan sistematis mengenai suatu hal yang mungkin akan terjadi di masa mendatang melalui informasi yang dimiliki dari masa lalu dan sekarang supaya kesalahannya dapat diminimalisir [14]. Prediksi memperkirakan apa yang akan terjadi pada masa mendatang berdasarkan data yang relevan pada masa lalu secara sistematis dan pragmatis, sehingga diharapkan mampu memiliki objektivitas yang besar [2].

2.2.2 Deep Learning

Deep Learning merupakan bagian dari *Machine Learning* yang merupakan Jaringan Syaraf Tiruan terinspirasi oleh sistem saraf pada otak manusia. *Deep learning* terdiri dari beberapa lapisan algoritma atau disebut dengan *neural network* yang digunakan untuk klasifikasi perintah berdasarkan *input*-an hingga menghasilkan suatu

output [15]. *Deep learning* memungkinkan suatu algoritma *neural network* dapat bekerja dengan sangat baik dalam membangun model prediksi pada masalah yang kompleks [16]. *Deep learning* dapat dikategorikan sebagai otomatisasi analitik prediktif yang melibatkan tingkat lebih tinggi mengenai presisi, matematika, dan komputasi. *Deep learning* telah menetapkan tolak ukur baru dalam bidang pengenalan suara, identifikasi gambar, sistem rekomendasi, pemrosesan bahasa alami, dan bioinformatika [17].

2.2.3 RNN (Recurrent Neural Network)

RNN menjadi sebuah metode atau algoritma yang sering digunakan untuk pengolahan data sekuensial seperti teks dan yang lainnya [18]. Arsitektur RNN terdiri dari beberapa *neuron* yang berbalik dari dirinya sendiri atau berbalik ke *neuron* pada *layer input* sehingga dapat menyimpan nilai yang dapat memberi pengaruh pada cara *input* guna memberikan hasil nilai sebelumnya dalam jaringan [19]. RNN membuat jaringan sebelumnya menjadi *input* untuk proses selanjutnya, sehingga jaringan sebelumnya akan menyimpan informasi dari proses awal ekstraksi fitur [20]. RNN menjadi salah satu bagian dari *Deep Learning* karena memiliki cara kerja dengan memproses data melalui beberapa lapisan [21].

2.2.4 Bidirectional Long Short Term Memory (Bi-LSTM)

Bi-LSTM merupakan pengembangan dari RNN dengan mengatasi permasalahan tentang *vanishing gradien* [22]. Algoritma tersebut terdiri dari dua jaringan LSTM yang memiliki fungsi untuk memproses urutan data dari arah depan dan sebaliknya (*forward* dan *backward*) [23]. LSTM sendiri akan mempelajari data yang harus dihilangkan dan disimpan pada setiap *neuron* [24]. Berdasarkan dua jaringan LSTM yang dimiliki oleh Bi-LSTM akan menghasilkan satu keluaran hasil dari penggabungan dua jaringan LSTM yang dimiliki

tersebut dan akan memperoleh informasi masa lalu dan masa mendatang secara bersamaan [25].

2.2.5 Python

Bahasa pemrograman Python sangat ideal untuk pengembangan dalam kecerdasan buatan, *machine learning*, dan *deep learning* [26]. Python memiliki keunggulan dalam kemudahan untuk pengembangan perangkat lunak, perangkat keras, dan aplikasi *website* karena memiliki keterbacaan kode yang baik [27]. Python dikategorikan sebagai bahasa pemrograman yang dinamis dan *high-level programming language*, yaitu bahasa pemrograman tingkat tinggi yang mendekati bahasa manusia. Selain itu, Python juga memiliki tingkat efisiensi yang tinggi pada struktur data *level* tinggi, kemudian mampu berjalan pada *multi-platform*, dan dapat diintegrasikan dengan bahasa pemrograman lain.

2.2.6 Django

Django merupakan sebuah *web framework* yang menggunakan bahasa pengembangan Python. Django memiliki kerangka pengembangan yang cepat, efisien, dan praktis dengan menawarkan keamanan, skabilitas, dan keserbagunaannya yang dapat membantu dalam membuat aplikasi *web* dengan mudah [28]. Selain itu, juga dapat membantu dalam merepresentasikan *Object Relational Mapper* (ORM) yang ketika terjadi perubahan pada basis data tidak perlu melakukan penyesuaian *query* kembali [29].

2.2.7 MySQL

MySQL merupakan sebuah program *server* basis data atau *database management system* dengan perintah SQL (*Structured Query Language*) yang dapat mengirim dan menerima data dengan cepat, *multithread*, dan *multi user* [30][31]. MySQL menjadi basis

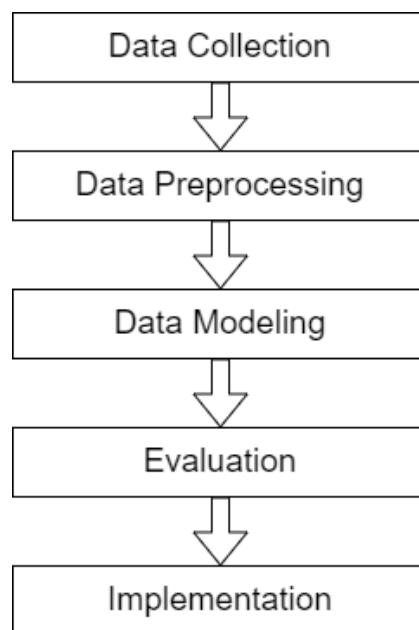
data yang populer dalam membangun aplikasi berupa *web* dengan basis data dalam mengelola data [32].

BAB III

Metodologi Penelitian

3.1 Diagram Alur Metodologi Penelitian

Metode penelitian yang digunakan pada penelitian ini terdiri dari beberapa tahap, yaitu dimulai dari *data collection*, *data preprocessing*, *data modeling*, *evaluation*, dan *implementation*. Tahapan dari metode penelitian dapat dilihat pada Gambar 3.1 [33].



Gambar 3.1 Alur Metodologi Penelitian

3.2 Tahapan-Tahapan Diagram Alur Metodologi Penelitian

Adapun tahapan-tahapan dalam diagram alur metodologi yang digunakan terdapat beberapa tahap, berikut merupakan penjelasan setiap tahap-tahapnya:

3.2.1 Data Collection

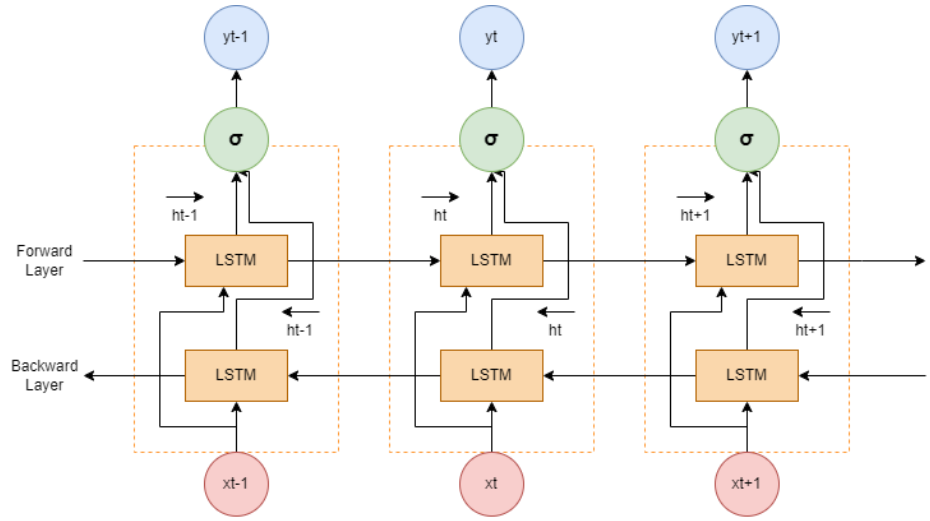
Pengumpulan data dilakukan dengan mengambil data judul buku pada *website* toko buku *online* XYZ menggunakan teknik *scraping*. *Scraping* dilakukan dengan menggunakan *framework web scraping* bahasa pemrograman Go. Data yang diperoleh pada penelitian ini sebanyak 5618 data dan disimpan dalam *file dataset* dengan format CSV. Kemudian hasil *scraping* data juga telah disimpan dalam basis data MySQL pada PHPMyAdmin untuk dapat terintegrasi dengan aplikasi.

3.2.2 Data Preprocessing

Tahapan *preprocessing data* dilakukan dengan membersihkan dan menyiapkan data sebelum digunakan pada tahapan pemodelan. Proses yang dilakukan pada tahap ini terdiri dari *case folding*, *remove punctuation*, *tokenizing*, *sequencing*, dan *padding*.

3.2.3 Data Modeling

Pemodelan data dilakukan untuk menerapkan algoritma yang tepat dalam menemukan pola pada sebuah data. Pada penelitian ini, mengimplementasikan *deep learning* dengan algoritma *Bidirectional Long Short Term Memory* (Bi-LSTM). Arsitektur pada algoritma Bi-LSTM dapat dilihat pada Gambar 3.2 [34].



Gambar 3.2 Arsitektur Bi-LSTM

Rumus perhitungan arsitektur Bi-LSTM disajikan dalam Formula 1.

$$y_t = W_{\overrightarrow{hy}} \overrightarrow{ht} + W_{\overleftarrow{hy}} \overleftarrow{ht} \quad (1)$$

Berdasarkan Formula 1, notasi y_t dapat diartikan sebagai nilai gerbang keluaran Bi-LSTM. Notasi $W_{\overrightarrow{hy}}$ diartikan sebagai nilai bobot dari gerbang keluaran LSTM *forward layer* dan notasi $W_{\overleftarrow{hy}}$ diartikan sebagai nilai bobot dari gerbang keluaran LSTM *backward layer*. Kemudian untuk notasi \overrightarrow{ht} diartikan sebagai nilai *output* dari LSTM *forward layer* dan notasi \overleftarrow{ht} diartikan sebagai nilai *output* dari LSTM *backward layer* [35].

3.2.4 Evaluation

Evaluasi digunakan untuk mengukur seberapa baik model yang telah dibuat. Evaluasi ini dilakukan untuk mengetahui kinerja model dalam memprediksi kata berikutnya. Metode pengukuran yang digunakan dalam evaluasi ini menggunakan *matrix* akurasi pada

metode *confusion matrix*. *Confusion matrix* merupakan pengukuran performa model *machine learning* dengan membandingkan nilai aktual dan nilai prediksi [36]. *Confusion matrix* dapat dilihat pada Gambar 3.3 [37].

		Actual Values	
		Positive	Negative
Predicted Values	Positive	True Positive (TP)	False Positive (FP)
	Negative	False Negative (FN)	True Negative (TN)

Gambar 3.3 Confusion Matrix

Formula akurasi dari *confusion matrix* dapat dilihat pada Formula 2.

$$Akurasi = \frac{(TP+TN)}{(TP+TN+FP+FN)} \times 100\% \quad (2)$$

Formula akurasi juga dapat didefinisikan seperti pada Formula 3.

$$Akurasi = \frac{Total\ prediksi\ benar}{Total\ prediksi} \times 100\% \quad (3)$$

3.2.5 Implementation

Implementasi merupakan langkah akhir dalam pengembangan pada penelitian ini. Implementasi ditujukan agar model yang telah

dibuat dapat diimplementasikan pada sistem dan dapat digunakan oleh pengguna untuk memudahkan dalam pencarian judul buku. Adapun implementasi yang dibuat berupa *website* pencarian judul buku yang dibangun menggunakan bahasa Python dan *framework* Django. Kemudian mengimplementasikan basis data MySQL sebagai tempat penyimpanan data buku.

BAB IV

Hasil dan Pembahasan

4.1 Data Collection

Data pada penelitian yang dilakukan diambil dari *website* toko buku *online XYZ* dengan menggunakan teknik *scraping*. Data tersebut belum pernah digunakan untuk pengujian pada penelitian-penelitian sebelumnya. Data yang diambil hanya berupa judul buku yang tersedia pada *website* tersebut yang disimpan dalam *file* berformat CSV dan juga tersimpan pada basis data MySQL. Berdasarkan hasil *scraping* diperoleh 10438 data yang kemudian dilakukan pembersihan data pada *dataset* CSV berupa duplikasi data dan hanya memilih data judul buku berbahasa Indonesia. Setelah dilakukan proses tersebut, diperoleh data judul buku berbahasa Indonesia sebanyak 5618 data. Kemudian data yang telah diperoleh diproses dengan diubah kedalam bentuk *dataframe* untuk memudahkan dalam tahapan *preprocessing* data seperti pada Gambar 4.1.

	judul;
0	Keanggotaan Investasi Reseller Bukukita;
1	Keanggotaan Investasi Dropship Bukukita;
2	Komik Dari Twit-nya Raditya Dika;
3	Rangkuman Tips Beriklan Selama 9 Tahun;
4	PROPERTI KOMODO ;
5	Panduan Menulis Siswa Tingkat Dasar;
6	Buku Saku Kumpulan Rumus Matematika SMA IPA;
7	Cahaya Di Penjuru HATI;
8	Catatan Ayahasi Ditambah-tambahin;
9	Al-Akram;
10	Buku Panduan Piala Dunia Brazil 2014;

Gambar 4.1 Dataframe

simbol. Proses tersebut dilakukan agar model yang akan dibuat memperoleh hasil kinerja yang baik. Hasil dari proses tersebut seperti pada Gambar 4.3.

```
0    keanggotaan investasi reseller bukukita
1    keanggotaan investasi dropship bukukita
2          komik dari twit-nya raditya dika
3    rangkuman tips beriklan selama 9 tahun
4          properti komodo
Name: judul;, dtype: object
```

Gambar 4.3 Hasil Huruf Kecil dan Penghapusan Tanda Baca

Kemudian data yang telah dibersihkan dilakukan *tokenizing* untuk memisahkan setiap kata (*token*) pada kalimat dalam data. Kemudian setiap kata akan memiliki urutan numeriknya masing-masing untuk memudahkan komputer dalam melakukan pengolahan. Hasil dari proses *tokenizing* dapat dilihat pada Gambar 4.4.

```
'dicintai': 539,
'jitu': 540,
'pemikiran': 541,
'modal': 542,
'batas': 543,
'mandiri': 544,
'tradisi': 545,
```

Gambar 4.4 Hasil Proses Tokenizing

Kata atau *token* yang telah dipisahkan dari kalimat dilakukan *sequencing* untuk diberikan urutan *n-gram* sesuai kata yang muncul bersamaan dalam kalimat. *N-gram* akan menghitung frekuensi kemunculan dari urutan kata yang muncul dalam *dataset*, lalu akan memperkirakan probabilitasnya [39]. Proses *n-gram* dapat dilihat pada Gambar 4.5.

```

[[2001, 581],
 [2001, 581, 3025],
 [2001, 581, 3025, 2002],
 [2001, 581],
 [2001, 581, 3026],
 [2001, 581, 3026, 2002],
 [64, 9],
 [64, 9, 3027],
 [64, 9, 3027, 1208],
 [64, 9, 3027, 1208, 3028],
 [64, 9, 3027, 1208, 3028, 3029],
 [725, 280],
 [725, 280, 3030],
 [725, 280, 3030, 1520],
 [725, 280, 3030, 1520, 264],
 [725, 280, 3030, 1520, 264, 110],

```

Gambar 4.5 Hasil Proses N-gram

Selanjutnya kata yang telah diberikan urutan dilakukan *padding* agar setiap kalimat atau *sequence* memiliki panjang data yang sama. Seperti yang terlihat pada Gambar 4.6, merupakan hasil dari proses *padding* yang dimana panjang dari *array* yang dimiliki adalah 17. Angka tersebut merupakan panjang maksimal kalimat pada keseluruhan data yang dimiliki. Apabila panjang salah satu data kurang dari 17, maka akan diisi angka 0 seperti pada Gambar 4.6.

```

array([ 0,  0,  0,  0,  0,  0,  0,  0,  0,  0,  0,
        0,  0,  0, 2001, 581, 3025], dtype=int32)

```

Gambar 4.6 Hasil Proses Padding

Data hasil proses *padding* berbentuk numerik agar dapat dilakukan komputasi pada tahapan pemodelan data dan akan dijadikan sebagai data *features*.

4.3 Data Modeling

Setelah melalui berbagai proses pada data *preprocessing*, selanjutnya data akan dilakukan pemodelan dengan tahapan pertama yaitu pembagian data menjadi data x sebagai data *input*, data y sebagai variabel target, dan data label sebagai nilai target dari setiap urutan data *input*. Hasil dari salah satu pembagian data tersebut seperti pada Gambar 4.7.

```
[ ] print(x[5])
    print(labels[5])
    print(y[5][14])

[ 0 0 0 0 0 0 0 0 0 0 0 0 0 0 2001
 581 3026]
2002
0.0
```

Gambar 4.7 Hasil Pembagian Data

Tahapan selanjutnya adalah pemodelan data dengan menggunakan algoritma *Bidirectional Long Short Term Memory* (Bi-LSTM). Pemodelan data dengan algoritma Bi-LSTM tersebut telah dilakukan beberapa pengujian untuk menemukan parameter model yang terbaik seperti pada Tabel 4.1.

Tabel 4.1 Pengujian Parameter Model

No	Layer	Epoch	Loss	Akurasi
1	Bi-LSTM : 512	51	63.92%	80.29%
2	Bi-LSTM : 256	48	69.72%	79.80%
3	Bi-LSTM : 256	75	60.57%	80.68%
4	Bi-LSTM : 256	200	50.37%	81.38%
5	Bi-LSTM : 512	200	45.47%	81.72%

Berdasarkan hasil pengujian pada Tabel 4.1, diperoleh parameter model yang terbaik dengan menggunakan layer *Bidirectional* dengan *input* 1000 dan *Epoch* sebesar 200. Sehingga struktur *layer* pada model terdiri dari *layer embedding* yang berguna untuk menanamkan vektor kata dengan *input* 16 sebagai panjang maksimum urutan kata yang digunakan sebagai *input* dan vektor 10 sebagai dimensi ruang vektor dalam merepresentasikan kata. Kemudian *layer Bidirectional* dengan *input layer* 2000 yang merupakan hasil dari keterbacaan algoritma secara dua arah berdasarkan *input* 1000. Selanjutnya *layer Dense* dengan *input* berupa total dari jumlah kata yaitu sebanyak 6887 yang menggunakan *activation softmax*. Struktur *layer* pada model yang telah dibuat dapat dilihat pada Gambar 4.8.

Model: "sequential_1"

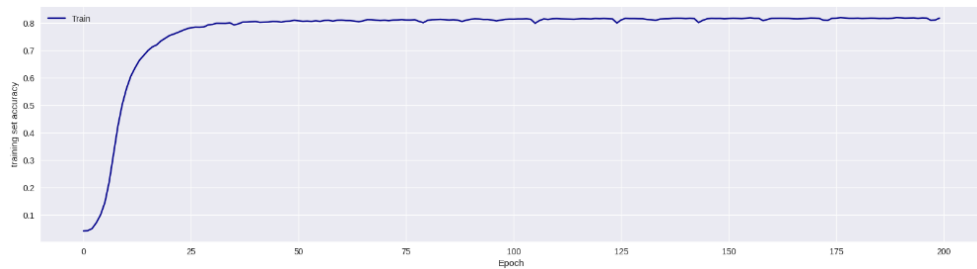
Layer (type)	Output Shape	Param #
embedding_1 (Embedding)	(None, 16, 10)	68870
bidirectional_1 (Bidirectional)	(None, 2000)	8088000
dense_1 (Dense)	(None, 6887)	13780887
Total params: 21,937,757		
Trainable params: 21,937,757		
Non-trainable params: 0		

Gambar 4.8 Struktur Layer Model

4.4 Evaluation

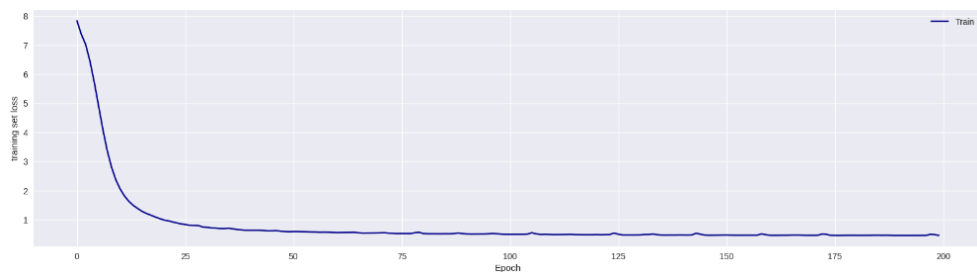
Pelatihan pada model yang telah dibuat menerapkan *loss* berupa *categorical crossentropy* dengan *optimizer adam* dan *metrics accuracy*. Model dilatih dengan menggunakan *epochs* sebesar 200. Hasil dari pelatihan model memperoleh nilai *loss* sebesar 0.4536 atau 45.36% dan nilai

akurasi sebesar 0.8182 atau 81.82%. Grafik hasil akurasi dari proses pelatihan model dapat dilihat pada Gambar 4.9.



Gambar 4.9 Grafik Hasil Akurasi Pelatihan

Berdasarkan Gambar 4.9, grafik akurasi menunjukkan kestabilan mulai dari *epoch* 50. Namun hasil akurasi terbaik yang diperoleh berada pada *epoch* 200 dengan hasil akurasi sebesar 81.82%. Kemudian grafik hasil *loss* dari proses pelatihan model dapat dilihat pada Gambar 4.10.



Gambar 4.10 Grafik Hasil Loss Pelatihan

Berdasarkan Gambar 4.10, grafik *loss* telah menunjukkan kestabilan mulai dari *epoch* 75. Namun hasil *loss* terbaik yang diperoleh berada pada *epoch* 200 dengan hasil *loss* sebesar 45.36%. Sehingga *epoch* 200 menjadi *epoch* terbaik untuk model prediksi kata berikutnya dalam pencarian judul buku berbahasa Indonesia.

Setelah mengetahui hasil evaluasi pelatihan, selanjutnya dilakukan pengujian dengan memprediksi dari kata yang di-*input*-kan. Pengujian

dilakukan dengan meng-*input*-kan kata dan memprediksi beberapa kata berikutnya. Hasil pengujian dapat dilihat pada Gambar 4.11.

```
Enter your line: nasihat
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 21ms/step
1/1 [=====] - 0s 19ms/step
1/1 [=====] - 0s 17ms/step
nasihat langit penentram jiwa 4
```

Gambar 4.11 Hasil Pengujian

4.5 Implementation

Model yang telah dilakukan pelatihan, selanjutnya diimplementasikan pada aplikasi. Implementasi ini ditujukan agar sistem prediksi kata berikutnya untuk pencarian judul buku dapat dinikmati atau digunakan oleh pengguna dalam mencari judul buku yang sesuai dengan mudah. Implementasi yang diterapkan berupa *website* pencarian judul buku yang telah terintegrasi dengan basis data kumpulan buku yang diperoleh dari *website XYZ*. *Website* pencarian judul buku tersebut dibangun dengan menggunakan *framework* Django, bahasa Python, dan basis data MySQL. Kemudian pada halaman *website* menggunakan HTML dan CSS. Pada Gambar 4.12, merupakan tampilan *website* pencarian judul buku yang telah dibuat.



Gambar 4.12 Tampilan Web Pencarian Judul Buku

Adapun basis data yang telah dibuat, digunakan untuk menyimpan data buku dan menampilkan hasil pencarian buku dari model prediksi kata berikutnya pada aplikasi *web*. Basis data tersebut terdiri dari satu tabel dengan struktur seperti pada Tabel 4.2.

Tabel 4.2 Struktur Tabel Buku

#	Nama	Tipe Data	Panjang
1	id_buku (pk)	Integer	11
2	cover_buku	Varchar	500
3	judul_buku	Varchar	255
4	stok	Varchar	20

BAB V

Penutup

5.1 Kesimpulan dan Saran

5.1.1 Kesimpulan

Berdasarkan penelitian yang telah dilakukan, sistem prediksi kata berikutnya untuk pencarian judul buku telah berhasil dibangun menggunakan algoritma *Bidirectional Long Short Term Memory* (Bi-LSTM) dan diperoleh hasil akurasi yang baik. Hal tersebut dibuktikan dengan hasil evaluasi model yang memperoleh *matrix* akurasi sebesar 81.72%. Model yang telah dibuat telah mampu memprediksi kata berikutnya untuk pencarian judul buku secara mudah dan tepat. Selain itu, model yang telah dibuat juga telah memberikan hasil yang relevan dengan kata kunci pencarian yang diberikan.

5.1.2 Saran

Saran yang dapat disampaikan adalah model yang telah dibuat masih memiliki kekurangan berupa beberapa prediksi yang belum sesuai karena belum beragamnya data yang diperoleh. Kemudian model yang dibuat hanya menggunakan bahasa Indonesia yang dapat dikembangkan menjadi berbagai bahasa pada penelitian selanjutnya. Dengan demikian, penelitian lebih lanjut sangat diharapkan untuk pengembangan yang lebih baik.

Lampiran

A. Kartu Bimbingan



UNIVERSITAS LOGISTIK DAN BISNIS INTERNASIONAL
PROGRAM STUDI D4 TEKNIK INFORMATIKA
JL. SARIASIH NO. 54 BANDUNG 40151
Telp. 022-2009562, 2009570
Fax. 022-2009568
**FORMULIR KEGIATAN
INTERNSHIP II
TA. 2022/2023**

Nama : Alwizain Almas Trigreisian
Npm : 1194004
Judul : Studi Literatur: Prediksi Kata
Berikutnya dengan Metode
RNN
Pembimbing : Nisa Hanum Harani, S.Kom.,
M.T.,CDSP.,SFPC

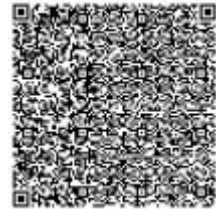


Pertemuan	Tanggal	Sudah Dikerjakan	Pekerjaan Selanjutnya	Nilai
1	15-03-2023	konsultasi terkait topik pengerjaan	mencari jurnal refrensi	100
2	04-04-2023	mencari jurnal refrensi	membuat dataset	100
3	12-04-2023	membuat dataset	membersihkan dataset dan membuat jurnal slr	100
4	05-05-2023	membersihkan dataset dan membuat jurnal slr	menambahkan sota pada laporan dan refrensi jurnal acuan	100
5	06-05-2023	menambahkan refrensi jurnal acuan	melanjutkan menambahkan sota pada laporan	100

6	08-05-2023	menambahkan sota pada laporan	submit jurnal dan pengumpulan laporan	100
			Rata-Rata:	100.00

Bandung, 08 Mei 2023

Pembimbing,



Nisa Hanum Harani,. S.Kom., M.T.,CDSP.,SFPC
NIDN. 0415048901

B. Curriculum Vitae

Alwizain Almas Trigreisian

Kecerdasan Buatan | Data Analyst

Bandung, Jawa Barat, 40151 | 081249851919 |
alwizainalmastrigreisian@gmail.com | [https://www.linkedin.com/in/alwizain-
almas-trigreisian-a44855119/](https://www.linkedin.com/in/alwizain-almas-trigreisian-a44855119/)

TENTANG SAYA

Mahasiswa Tingkat Akhir Jurusan Teknik Informatika pada Universitas Logistik dan Bisnis Internasional angkatan 2019. Memiliki pengalaman yang baik dalam bidang kecerdasan buatan dan mengolah, menganalisis serta menginterpretasikan suatu data. Memiliki keahlian pada bidang Bahasa Pemrograman Python dan tertarik pada bidang yang berkaitan dengan kecerdasan buatan dan data analyst. Memiliki keahlian softskills berupa berpikir kritis, pemecahan masalah, berorientasi pada detail, analisis, disiplin, konsisten, dan tanggung jawab.

PENDIDIKAN

Universitas Logistik dan Bisnis Internasional (Oktober 2019 – Sekarang)
D-IV Teknik Informatika (IPK 3.64)

KETERAMPILAN

Python, Data Analyst, Data Mining, Artificial Intelligence, Deep Learning, Machine Learning, Data Science, Data Visualization, MySQL, Web Development, Django.

PENGALAMAN

Synthesis Communications (Agustus – Desember 2022)
Artificial Intelligence For GenZ Jobseekers | Kampus Merdeka

- Memiliki pengetahuan dan wawasan tentang konsep dasar dan teknologi Artificial Intelligence.
- Mampu mengimplementasikan framework AI Project Cycle dalam mengembangkan produk Artificial Intelligence.

- Mampu menggunakan bahasa pemrograman Python untuk menyelesaikan masalah sederhana dan mengembangkan produk Artificial Intelligence.
- Mampu mengimplementasikan algoritma Artificial Intelligence yang sesuai dalam menyelesaikan suatu kasus atau permasalahan.
- Mampu menemukan potensi diri, merealisasikan ide, menciptakan peluang usaha, dan melakukan transformasi ide menjadi produk dan jasa.
- Mampu mengembangkan produk AI yang bermanfaat untuk pengembangan IPTEK dan sosial.

Indonesia AI - PT. Teknologi Artifisial Indonesia (Juni – Juli 2022)
Research & Development Group

- Memiliki kemampuan dalam melakukan pelabelan data (Tweet Annotator) pada dataset untuk kegiatan Sentiment Analysis.
-

PENGALAMAN ORGANISASI

Javanese Community – Komunitas (April 2021 – Juni 2022)
Mahasiswa Jawa
Bupati (Ketua Umum)

- Bertanggung jawab atas berjalannya komunitas mahasiswa Jawa.
 - Mengawasi program kerja yang sedang berjalan.
 - Memberikan arahan kepada anggota dalam menjalankan program kerja yang diberikan
-

PELATIHAN

ASEAN Data Science Explorers 2023 (Mei 2023)
Enablement Session – SAP Analytics Cloud
ASEAN Foundation
<https://bit.ly/SertifikatADSE2023>

Intro to Data Analytics (Maret 2023)
RevoU
<https://bit.ly/SertifikatDataAnalytics23>

HCIA-AI Course (September 2022)
Huawei
<https://bit.ly/SertifikatHCIA-AI>

Belajar Machine Learning untuk Pemula Dicoding Indonesia https://bit.ly/SertifikatMachineLearning	(Agustus 2022)
Belajar Dasar Visualisasi Data Dicoding Indonesia https://bit.ly/SertifikatVisualisasiData	(Juli 2022)
Memulai Pemrograman Dengan Python Dicoding Indonesia https://bit.ly/SertifikatPy	(Juli 2022)
Database Design Oracle https://bit.ly/SertifikatDatabaseDesign	(Juli 2020)

PROYEK

Chatbot Wawancara Kerja https://github.com/alwizain/chatbot	(Oktober 2022 – Februari 2023)
<p>Chatbot wawancara kerja pada platform Telegram dengan mengimplementasikan Deep Learning menggunakan algoritma Long Short Term Memory dan website manajemen data chatbot dengan mengimplementasikan CRUD dan Database MongoDB.</p> <p>Teknologi: Flask, Django, Python, Deep Learning, Natural Language Processing (NLP), MongoDB.</p>	
Aplikasi Web E-Repository https://github.com/alwizain/e-repository	(Oktober 2022 – Februari 2023)
<p>Aplikasi sederhana penyedia buku dan jurnal dengan fitur pembelian yang telah terintegrasi pembayaran Midtrans dan sistem rekomendasi (data science).</p> <p>Teknologi: Python, framework Django, MySQL, Data Science.</p>	
Aplikasi Web E-Repository https://github.com/alwizain/e-repository	(Oktober 2022 – Februari 2023)
<p>Aplikasi sederhana penyedia buku dan jurnal dengan fitur pembelian yang telah terintegrasi pembayaran Midtrans dan sistem rekomendasi (data science).</p> <p>Teknologi: Python, framework Django, MySQL, Data Science.</p>	

PUBLIKASI

**Telegram Bot Wawancara Kerja dengan Algoritma
Long Short Term Memory**

(Februari 2023)

Penerbit Buku Pedia

<https://books.google.co.id/books?id=lxuuEAAAQBAJ>

Publikasi berupa buku tutorial yang membahas mengenai tata cara pembuatan suatu sistem chatbot untuk menetralsir permasalahan yang terjadi antara pewawancara dengan pelamar saat proses wawancara kerja.

C. Kode Program Scraping Data

1. Import Library

```
1  package main
2
3  import (
4      "database/sql"
5      "log"
6
7      _ "github.com/go-sql-driver/mysql"
8      "github.com/gocolly/colly"
9  )
```

Gambar 1 Kode Program Import Library

2. Koneksi Basis Data

```
11 func main() {
12     // buat koneksi ke database MySQL
13     db, err := sql.Open("mysql", "root:@/buku")
14     if err != nil {
15         log.Fatal("Error connecting to MySQL database:", err)
16     }
17     defer db.Close()
```

Gambar 2 Kode Program Koneksi Basis Data

3. Pengambilan Data

```
19     // inisialisasi colly
20     c := colly.NewCollector()
21
22     // atur aksi ketika menemukan link buku
23     c.OnHTML("div.product-preview", func(e *colly.HTMLElement) {
24         // ambil informasi buku
25         cover := e.ChildAttr("div.product-preview__image img", "src")
26         title := e.ChildText("div.product-preview__info_title div.ellipsis a")
27         stock := e.ChildText("div.product-preview__info font.smaller.ready strong")
28
29         // simpan informasi buku ke database MySQL
30         stmt, err := db.Prepare("INSERT INTO buku (cover_buku, judul_buku, stok) VALUES (?, ?, ?)")
31         if err != nil {
32             log.Fatal("Error preparing MySQL statement:", err)
33         }
34         defer stmt.Close()
35
36         _, err = stmt.Exec(cover, title, stock)
37         if err != nil {
38             log.Fatal("Error executing MySQL statement:", err)
39         }
40     })
```

Gambar 3 Kode Program Pengambilan Data

D. Kode Program Model Prediksi Kata Berikutnya

1. Import Library

```
[ ] import tensorflow as tf
    from tensorflow.keras.preprocessing.text import Tokenizer
    from tensorflow.keras.preprocessing.sequence import pad_sequences
    from tensorflow.keras.layers import Embedding, LSTM, Dense, Bidirectional
    from tensorflow.keras.models import Sequential
    from tensorflow.keras.utils import to_categorical
    from nltk.tokenize import word_tokenize
    from wordcloud import WordCloud
    import matplotlib.pyplot as plt
    import pickle
    import numpy as np
    import pandas as pd
    import os
    import nltk
    import re
```

Gambar 4 Kode Program Import Library

2. Load Data

```
[ ] # Load data
    df = pd.read_csv('buku.csv', encoding='ISO-8859-1', sep='delimiter', engine='python')
    df.head(30)
```

Gambar 5 Kode Program Load Data

3. Wordcloud

```
[ ] # Menggabungkan semua teks dari kolom yang ingin digunakan dalam satu string
    text = ' '.join(df['judul;'])

    # Membuat wordcloud dengan maksimal 100 kata dan warna background putih
    wordcloud = WordCloud(max_words=100, background_color="white").generate(str(text))
    plt.figure()
    plt.imshow(wordcloud, interpolation="bilinear")
    plt.axis("off")
    plt.show()
```

Gambar 6 Kode Program Wordcloud

4. Pengubahan Huruf Kecil

```
[ ] # Mengubah ke huruf kecil
    df['judul;'] = df['judul;'].apply(lambda x: " ".join(x.lower() for x in x.split()))
    df['judul;'].head()
```

Gambar 7 Kode Program Pengubahan Huruf Kecil

5. Penghapusan Tanda Baca

```
[ ] # Menghapus tanda baca
df['judul;'] = df['judul;'].apply(lambda x:re.sub('[!@#$.:;,&]', '', x.lower()))
df['judul;'].head()
```

Gambar 8 Kode Program Penghapusan Tanda Baca

6. Tokenization

```
[ ] # Proses tokenisasi
tokenizer = Tokenizer(oov_token=None)
tokenizer.fit_on_texts(df['judul;'])
total_words = len(tokenizer.word_index) + 1
```

Gambar 9 Kode Program Tokenization

7. N-gram

```
[ ] # Proses n-gram (sequencing)
input_sequences = []
for line in df['judul;']:
    token_list = tokenizer.texts_to_sequences([line])[0]
    #print(token_list)

    for i in range(1, len(token_list)):
        n_gram_sequence = token_list[:i+1]
        input_sequences.append(n_gram_sequence)
```

Gambar 10 Kode Program N-gram

8. Padding

```
[ ] # Proses padding
max_sequence_len = max([len(x) for x in input_sequences])
input_sequences = np.array(pad_sequences(input_sequences, maxlen=max_sequence_len, padding='pre'))
input_sequences[1]
```

Gambar 11 Kode Program Padding

9. Pembagian Data

```
[ ] # Membuat data features dan label
x, labels = input_sequences[:, :-1], input_sequences[:, -1]
y = to_categorical(labels, num_classes=total_words)
```

Gambar 12 Kode Program Pembagian Data

10. Model

```
[ ] # Pemodelan data
model = Sequential()
model.add(Embedding(total_words, 10, input_length=max_sequence_len-1))
model.add(Bidirectional(LSTM(100)))
model.add(Dense(total_words, activation='softmax'))
```

Gambar 13 Kode Program Model

11. Pelatihan

```
[ ] # Pelatihan
model.compile(loss='categorical_crossentropy', optimizer='adam',
              metrics=[tf.keras.metrics.Recall(), 'accuracy',
                      tf.keras.metrics.Precision()])
history = model.fit(x, y, epochs=200, verbose=1)
print(model)
```

Gambar 14 Kode Program Pelatihan

12. Evaluasi

```
[ ] # Membuat fungsi untuk menampilkan plot hasil akurasi
import matplotlib.style
import matplotlib as mpl
import matplotlib.pyplot as plt
def draw_plot(data, type_data):
    mpl.style.use('seaborn')
    plt.figure(figsize = (20, 5))
    plt.plot(data, 'darkblue', label='Train')
    plt.xlabel('Epoch')
    plt.ylabel(type_data)
    plt.legend()
```

Gambar 15 Kode Program Evaluasi

E. Kode Program Aplikasi Website

1. Nextwordpredictor Project

```
86 DATABASES = {
87     'default': {
88         'ENGINE': 'django.db.backends.mysql',
89         'NAME': 'buku',
90         'USER': 'root',
91         'PASSWORD': '',
92         'HOST': 'localhost',
93         'PORT': '3306',
94     }
95 }
```

Gambar 16 Konfigurasi Basis Data settings.py

```
16 from django.contrib import admin
17 from django.urls import path, include
18
19 from predictorapp import views
20 urlpatterns = [
21     path('admin/', admin.site.urls),
22     path('prediction/', views.predict),
23
24     # path app
25     path('', include('predictorapp.urls')),
26     path('search/', include('search.urls')),
27 ]
```

Gambar 17 Konfigurasi Route Project urls.py

2. Predictorapp App

```

235
236 
237 <!-- Image by jcomp on Freepik -->
238
239 <div class="container-fluid">
240   <div class="landingtext text-center" style="margin-top: -450pt; color: ■aliceblue;">
241     <h1>#CariBuku Tidak Sesulit Melupakan Masa Lalu,<br> Cari Judul Buku Anda Dengan Mudah</h1>
242   </div>
243
244   <br>
245   <main style="text-align: center;">
246     <div style="display: inline-block;">
247
248       <form action="{% url 'search:search' %}" autocomplete="off" method="GET">
249         <input type="search" id="textInput" onkeyup="callSearchAPI(event)" name="q"
250           placeholder="Tulis judul buku yang Anda cari..." value="{{search}}">
251       </form>
252
253       <!-- onkeyup="callSearchAPI()" -->
254       <div class = "dropdown" id="dropdown">
255
256       </div>
257     </div>
258   </main>
259 </div>

```

Gambar 18 Tampilan HTML index.py

```

1  from django.db import models
2
3  # Create your models here.
4  class Buku(models.Model):
5      id_buku = models.AutoField(primary_key=True)
6      cover_buku = models.CharField(max_length=500)
7      judul_buku = models.CharField(max_length=255)
8      stok = models.CharField(max_length=20)
9
10     def __str__(self):
11         return self.judul_buku

```

Gambar 19 Membuat Basis Data Buku model.py

```

1  from django.contrib import admin
2  from . models import Buku
3
4  # Register your models here.
5  admin.site.register(Buku)

```

Gambar 20 Konfigurasi Basis Data Buku admin.py

```

50 @api_view(['GET', 'POST'])
51 def predict(request):
52     if request.method == 'POST':
53         pretext = request.data['text']
54         predicted_words = model.generate_seq(max_sequence_len, pretext, 1)
55         return Response({"predicted_words": predicted_words})
56     return Response({"text": "<TEXT HERE>"})
57
58 def index(request):
59     bukus_ = Buku.objects.all()
60     paginator = Paginator(bukus_, 20)
61     page = request.GET.get('page')
62     books = paginator.get_page(page)
63
64     context = {
65         "data": books,
66     }
67     return render(request, 'index.html', context)

```

Gambar 21 Konfigurasi Views App views.py

```

1  from django.urls import path
2  from . import views
3
4
5  urlpatterns = [
6      path('', views.index, name='index'),
7
8  ]

```

Gambar 22 Konfigurasi Route App urls.py

3. Predictormodel App

```

4 class NextWordModel(object):
5     def __init__(self):
6         super(NextWordModel, self).__init__()
7
8     def load_model_tokenizer_and_padding(self, tokenizer, padding, model_path):
9         self.tokenizer = tokenizer
10        print('Loading model...')
11        self.padding = padding
12        print('Loading model...')
13        self.model = load_model(model_path, compile=False)
14        self.model.make_predict_function()
15        print('Model Loaded!')
16
17    def generate_seq(self, max_sequence_len, seed_text, n_words):
18        result = list()
19        in_text = seed_text
20        print(seed_text)
21
22        next_words = 1
23
24        for _ in range(next_words):
25            token_list = self.tokenizer.texts_to_sequences([in_text])[0]
26            # truncate sequences to a fixed length
27            token_list = pad_sequences([token_list], maxlen=max_sequence_len-1, padding='pre')
28            # predict probabilities for each word
29            predicted_l = list(tuple(enumerate(self.model.predict(token_list)[0])))
30            top_2 = sorted(predicted_l, key=lambda x: x[1], reverse=True)[:2]
31            print(top_2)
32            # map predicted word index to word
33            predicted_words = []

```

Gambar 23 Konfigurasi Logika Model Prediksi model.py

4. Search App

```

1 from django.shortcuts import render
2 from django.db.models import Q
3 from predictorapp.models import Buku
4
5 def search(request):
6     search = request.GET.get('q')
7     books = Buku.objects.all()
8
9     if search:
10        books = books.filter(
11            Q(judul_buku__icontains=search)
12        )
13
14    context = {
15        "data": books,
16        "search": search,
17    }
18
19    return render(request, 'index.html', context)

```

Gambar 24 Konfigurasi Views App Search views.py

```
1  from django.urls import path
2  from . import views
3
4  app_name = 'search'
5
6  urlpatterns = [
7      path('', views.search, name='search'),
8  ]
```

Gambar 25 Konfigurasi Route App Search urls.py

Daftar Pustaka

- [1] N. Mulyani and J. Hutahaeen, “Rancang Bangun Website E-Commerce Toko Buku Online (Ol Book),” *Seminar Nasional Teknologi Komputer & Sains (SAINTEKS)*, pp. 373–382, 2020.
- [2] I. Admirani, “Penerapan Metode Fuzzy Time Series Untuk Prediksi Laba Pada Perusahaan,” *Jurnal JUPITER*, vol. 10, no. 1, pp. 19–31, 2018.
- [3] S. Ambulgekar, S. Malewadikar, R. Garande, and B. Joshi, “Next Words Prediction Using Recurrent Neural Networks,” *ITM Web of Conferences*, vol. 40, p. 03034, Aug. 2021, doi: 10.1051/itmconf/20214003034.
- [4] A. Rianti, S. Widodo, A. D. Ayuningtyas, and F. B. Hermawan, “Next Word Prediction Using LSTM,” *Journal of Information Technology and Its Utilization*, vol. 5, no. 1, 2022.
- [5] R. Sharma, N. Gael, N. Aggarwal, and C. Prakash, “Next Word Prediction in Hindi Using Deep Learning Techniques,” 2019.
- [6] A. F. Ganai and F. Khursheed, “Predicting next Word using RNN and LSTM cells: Stastical Language Modeling,” 2019.
- [7] K. Shakhovska, I. Dumyn, N. Kryvinska, and M. K. Kagita, “An Approach for a Next-Word Prediction for Ukrainian Language,” *Wirel Commun Mob Comput*, vol. 2021, 2021, doi: 10.1155/2021/5886119.
- [8] P. P. Barman and A. Boruah, “A RNN based approach for next word prediction in assamese phonetic transcription,” in *Procedia Computer Science*, Elsevier B.V., 2018, pp. 117–123. doi: 10.1016/j.procs.2018.10.359.
- [9] K. Wangchuk, T. Wangchuk, and T. Namgyel, “Dzongkha Next Words Prediction Using Bidirectional LSTM,” *Bhutan Journal of Research and Development*, no. 2, Feb. 2023, doi: 10.17102/bjrd.rub.se2.038.
- [10] S. Rajakumar, V. Rameshbabu, D. Usha, N. K. R. Shree, and S. Priya, “EXO Next Word Prediction Using Machine Learning,” *J Surv Fish Sci*, vol. 10, no. 3S, pp. 4112–4118, 2023.
- [11] K. Chakradhar, K. S. Kiran, K. Shanmukh, K. Sharath Kumar, K. Dinesh Sagar, and] Gmr, “Next Word Prediction Using Deep Learning,” 2022. [Online]. Available: www.ijrpr.com
- [12] M. Soam and S. Thakur, “Next Word Prediction Using Deep Learning: A Comparative Study,” in *2022 12th International Conference on Cloud*

Computing, Data Science & Engineering (Confluence), IEEE, Jan. 2022, pp. 653–658. doi: 10.1109/Confluence52989.2022.9734151.

- [13] Ms. S. Ramya and Dr. C. S. K. Selvi, “Recurrent Neural Network based Models for Word Prediction,” *International Journal of Recent Technology and Engineering (IJRTE)*, vol. 8, no. 4, pp. 7433–7437, Nov. 2019, doi: 10.35940/ijrte.D5313.118419.
- [14] M. Kafil, “Penerapan Metode K-Nearest Neighbors untuk Prediksi Penjualan Berbasis Web pada Boutiq Dealove BONDOWOSO,” 2019.
- [15] R. D. Nurfitri, “Implementasi Deep Learning Berbasis Tensorflow untuk Pengenalan Sidik Jari,” 2018.
- [16] B. Ekaba, *Building Machine Learning and Deep Learning Models on Google Cloud Platform (A Comprehensive Guide for Beginners-Apress)*. 2019. doi: 9781484244708.
- [17] B. Georgios Filippou, “Assets’ Study With Machine and Deep Learning Methods,” 2022.
- [18] Y. Liu, D. Zhang, L. Du, Z. Gu, J. Qiu, and Q. Tan, “A Simple But Effective Way To Improve The Performance Of RNN-based Encoder In Neural Machine Translation Task,” in *Proceedings - 2019 IEEE 4th International Conference on Data Science in Cyberspace, DSC 2019*, Institute of Electrical and Electronics Engineers Inc., Jun. 2019, pp. 416–421. doi: 10.1109/DSC.2019.00069.
- [19] Y. Fauziyah *et al.*, “Mesin Penterjemah Bahasa Indonesia-Bahasa Sunda menggunakan Recurrent Neural Networks,” *JURNAL TEKNOINFO*, vol. 16, no. 2, 2022, [Online]. Available: <https://ejurnal.teknokrat.ac.id/index.php/teknoinfo/index>
- [20] A. Bazwir, C. Setianingsih, and M. A. Murti, “Prediksi Penggunaan Energi Listrik Menggunakan Algoritma Recurrent Neural Network,” *e-Proceeding of Engineering*, vol. 10, no. 1, 2023.
- [21] M. R. Firmansyah, R. Ilyas, and F. Kasyidi, “Klasifikasi Kalimat Ilmiah Menggunakan Recurrent Neural Network,” *Industrial Research Workshop and National Seminar*, 2020.
- [22] I. Nyoman, K. Wardana, N. Jawas, I. Komang, and A. A. Aryanto, “Prediksi Penggunaan Energi Listrik pada Rumah Hunian Menggunakan Long Short-Term Memory,” 2020. [Online]. Available: <http://journal.undiknas.ac.id/index.php/tiers>
- [23] D. I. Puteri, “Implementasi Long Short Term Memory (LSTM) dan Bidirectional Long Short Term Memory (BiLSTM) Dalam Prediksi Harga

Saham Syariah,” *Euler : Jurnal Ilmiah Matematika, Sains dan Teknologi*, vol. 11, no. 1, pp. 35–43, May 2023, doi: 10.34312/euler.v11i1.19791.

- [24] A. A. Ningrum *et al.*, “Algoritma Deep Learning-LSTM untuk Memprediksi Umur Transformator,” vol. 8, no. 3, pp. 539–548, 2021, doi: 10.25126/jtiik.202184587.
- [25] H. Nurrohmah, “Klasifikasi Berita Hoax Berbahasa Indonesia Menggunakan Bidirectional Long Short Term Memory (Bi-LSTM),” Jakarta, Aug. 2022.
- [26] N. Thaker and A. Shukla, “Python as Multi Paradigm Programming Language,” *Int J Comput Appl*, vol. 177, no. 31, pp. 38–42, Jan. 2020, doi: 10.5120/ijca2020919775.
- [27] T. M. Kadarina and M. H. Ibnu Fajar, “Pengenalan Bahasa Pemrograman Python menggunakan Aplikasi Games untuk Siswa/i di Wilayah Kembangan Utara,” *Jurnal Abdi Masyarakat (JAM)*, vol. 5, no. 1, p. 11, Jul. 2019, doi: 10.22441/jam.2019.v5.i1.003.
- [28] J. Kavander, “Developing Kanban board backend by using Django web framework,” Laurea University of Applied Sciences, 2022.
- [29] K. Duisebekova, in Physics, A. Professor, R. Khabirov, master student, and A. Zholzhan, “Django as Secure Web-framework in Practice,” *The Bulletin of KazATC Вестник КазАТК*, vol. 1, no. 116, pp. 275–281, 2021, doi: 10.52167/1609-1817-2020-116-1-275-281.
- [30] M. Ahmadar, P. Perwito, and C. Taufik, “Perancangan Sistem Informasi Penjualan Berbasis Web pada Rahayu Photo Copy dengan Database MySQL,” *Dharmakarya: Jurnal Aplikasi Ipteks untuk Masyarakat*, vol. 10, no. 4, pp. 284–289, Dec. 2021, doi: 10.24198/dharmakarya.v10i4.35873.
- [31] H. Dhika, N. Isnain, and M. Tofan, “Manajemen Villa Menggunakan Java Netbeans dan Mysql,” *Jurnal IKRA-ITH Informatika*, vol. 3, no. 2, 2019.
- [32] A. M. Saputra, “Naskah Publikasi Sistem Prediksi Persediaan Obat pada Apotek menggunakan Metode Naive Bayes (Studi Kasus : Apotek Seger Waras, Cianjur),” University of Technology Yogyakarta, Yogyakarta, 2020. Accessed: Jun. 18, 2023. [Online]. Available: <http://eprints.uty.ac.id/4874/>
- [33] N. Afrianto, D. H. Fudholi, and S. Rani, “Prediksi Harga Saham Menggunakan BiLSTM dengan Faktor Sentimen Publik,” *Jurnal RESTI (Rekayasa Sistem dan Teknologi Informasi)*, vol. 6, no. 1, pp. 41–46, Feb. 2022, doi: 10.29207/resti.v6i1.3676.
- [34] D. Naik and C. D. Jaidhar, “A novel Multi-Layer Attention Framework for visual description prediction using bidirectional LSTM,” *J Big Data*, vol. 9, no. 1, p. 104, Nov. 2022, doi: 10.1186/s40537-022-00664-6.

- [35] M. G. Rizky, “Analisis Perbandingan Metode LSTM dan BiLSTM untuk Klasifikasi Sinyal Jantung Phonocardiogram,” Surabaya, Aug. 2021.
- [36] M. S. Anggreany, “Confusion Matrix,” *School of Computer Science BINUS University*, Nov. 01, 2020. <https://socs.binus.ac.id/2020/11/01/confusion-matrix/> (accessed Jun. 16, 2023).
- [37] R. M. Sagala, “Prediksi Kelulusan Mahasiswa Menggunakan Data Mining Algoritma K-Means,” *TeIKa*, vol. 11, no. 2, pp. 131–142, Oct. 2021, doi: 10.36342/teika.v11i2.2610.
- [38] F. K. Wardani, “Analisis Sentimen untuk Pemeringkatan Popularitas Situs Belanja Online di Indonesia menggunakan Metode Naive Bayes (Studi Kasus Data Sekunder),” Institut Bisnis dan Informatika STIKOM Surabaya, Surabaya, 2019.
- [39] P. Niharika and S. J. J. Thangaraj, “Long Short Term Memory model-based automatic next word generation for text-based applications In contrast to the N-gram model,” *J Surv Fish Sci*, vol. 10, no. 1S, 2023.