

Implementasi AI Chatbot Sebagai Support Assistant Website Universitas Nurul Jadid Menggunakan Algoritma Long Short-Term Memory (LSTM)

M. Erfan Rianto¹, Maulidiansyah², Abu Tholib³

^{1,2,3} Fakultas Teknik, Universitas Nurul Jadid, Karanganyar, Paiton, Probolinggo, Jawa Timur, Indonesia

Article Info

Article history:

Diterima 25 April 2024

Revisi 27 April 2024

Diterbitkan 30 April 2024

Keywords:

Chatbot

Deep Learning

Long Short-Term Memory

Natural Language Processing

Universitas

ABSTRAK

Perkembangan teknologi semakin pesat, menciptakan perubahan besar dalam berbagai aspek kehidupan termasuk pada sektor pendidikan. Universitas Nurul Jadid merupakan lembaga pendidikan yang perlu mengadaptasi teknologi terkini untuk efisiensi dan pelayanan untuk menjawab peningkatan volume pertanyaan dan informasi yang dibutuhkan masyarakat/orang tua sebelum mendaftarkan putra-putrinya kuliah di Universitas Nurul Jadid. *Chatbot* merupakan bagian dari *Natural Language Processing (NLP)* berbasis *Artificial Intelligence (AI)* yang berfungsi melakukan percakapan dengan pengguna melalui teks atau ucapan yang memberikan layanan cepat dan akurat sepanjang waktu. *Long Short-Term Memory (LSTM)* yaitu algoritma *deep learning* untuk memprediksi serta klasifikasi data teks. Data penelitian terdiri dari *tag*, *pattern* dan *response* yang diperoleh secara manual dari referensi *website* Universitas Nurul Jadid kemudian di preprocessing guna membuat model. Bagian utama pada model *chatbot* ini yaitu lapisan *embedding* yang memberikan nilai vektor untuk setiap kata dalam data teks yang telah dimasukkan. Hasil training model menghasilkan akurasi sebesar 99.32% dan *loss* sebesar 12.57% Ini menandakan model sudah bagus dan tidak terjadi *overfitting* atau *underfitting* sehingga model layak untuk dilakukan pengujian dan *deployment*. Hasil ini mendukung penggunaan *chatbot LSTM* sebagai asisten virtual untuk membantu masyarakat/calon mahasiswa/mahasiswa mengakses informasi.

This is an open access article under the [CC BY-SA](#) license.



Corresponding Author:

Maulidiansyah

Teknologi Informasi, Universitas Nurul Jadid

Email: maulid@unuja.ac.id

1. PENDAHULUAN

Di zaman digital ini, perkembangan teknologi informasi semakin pesat dan cepat, sehingga menciptakan perubahan besar dalam berbagai aspek kehidupan [1]. Termasuk di dalam sektor pendidikan seperti Universitas Nurul Jadid yang merupakan perguruan tinggi berbasis pesantren perlu mengadaptasi teknologi terbaru untuk meningkatkan efisiensi dan pelayanan [2]. Seiring dengan meningkatnya jumlah calon mahasiswa yang ingin mendaftar setiap tahunnya, terjadi peningkatan volume pertanyaan dan kebutuhan informasi yang sangat cepat dan akurat [3]. Hal ini menjadi indikasi kebutuhan untuk solusi yang memberikan respon cepat dan akurat terhadap pertanyaan masyarakat/calon mahasiswa baru untuk memenuhi kebutuhan informasi tentang Universitas Nurul Jadid [2]. Pelayanan manual yang dilakukan oleh staf administrasi penerima siswa baru untuk menjawab pertanyaan memerlukan waktu dan tenaga yang signifikan sehingga dirasa kurang maksimal [4]. Masyarakat/calon mahasiswa baru biasanya bertanya kepada kontak admin staf akademik dan panitia penerimaan mahasiswa baru melalui chat pribadi seperti whatsapp, telegram dan email [5]. Hal ini dirasa kurang efisien karena keterbatasan jam respon sehingga ada keterlambatan memperoleh informasi [6].

Chatbot adalah cabang dari *Artificial Intelligence (AI)* domain *Natural Language Processing (NLP)* yang berfokus pada pengolahan bahasa alami berupa teks atau suara, dirancang agar bisa melakukan obrolan secara *online* layaknya manusia [7]. Walaupun dibuat agar bisa melakukan percakapan dengan manusia, sebenarnya tidak ada campur tangan *admin* pada percakapan itu, namun mesin yang bekerja sehingga

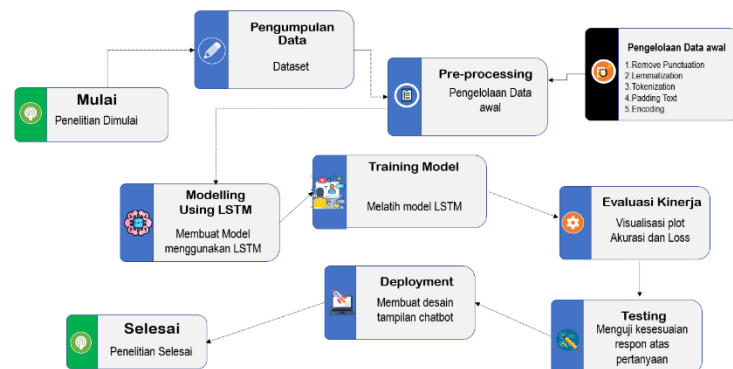
meminimalisir waktu dan biaya [8]. Survei membuktikan bahwa *chatbot* sangat efisien, menghemat keuangan, meminimalisir waktu dan memiliki akses 24 jam untuk memperoleh jawaban yang dibutuhkan [9].

Metode yang digunakan oleh peneliti sebelumnya pada pembuatan chatbot ini sangat beragam dengan hasil akurasi rata rata 80 - 90% [10]. Metode *Recurrent Neural Network (RNN)* lumayan banyak penggunaannya pada penelitian sebelumnya untuk membangun model chatbot dikarenakan mampu memprediksi dengan lebih akurat dari informasi terbaru [11]. Metode RNN sangat baik untuk memprediksi kata namun ada kekurangan yang serius yaitu tidak bisa melakukan prediksi kata yang tersimpan dalam memori dalam jangka yang panjang. Maka di rancanglah algoritma bernama LSTM (*Long Short-Time Memory*) untuk menyempurnakan algoritma RNN dengan tidak mengurangi kelebihan dari algoritma tersebut yang mampu memprediksi secara akurat dari informasi baru [12]. Dari data refrensi jurnal di atas, peneliti memilih metode *Long Short-Term Memory* sebagai metode untuk membangun model *chatbot*. Metode ini sangat memungkinkan untuk pengolahan bahasa alami dengan tujuan mepermudah *user* berkomunikasi dengan komputer [13].

Dari masalah di atas peneliti mendapatkan solusi untuk mengatasi masalah yaitu menghasilkan virtual asisten berupa chatbot yang berfungsi untuk memudahkan pengguna terkhusus bagi masyarakat/mahasiswa baru untuk mendapatkan informasi - informasi akademik secara *realtime*, cepat dan akurat dengan menggunakan model LSTM [14].

2. METODE

Pada gambar 1 menunjukkan alur penelitian dari awal hingga akhir yang di lakukan oleh penulis dimulai dari proses pengambilan data melalui *web*, *preprocessing* data, *remove punctuation*, *lemmatization*, *tokenization*, *padding text*, *encoding*, *modelling*, *training* untuk melatih model, evaluasi model untuk melihat apakah sudah *goodfitting*, *testing* untuk melihat kinerja model dan *deployment* untuk menerapkan pada aplikasi berbasis web [8], [15].



Gambar 1. Kerangka Alur Program

2.1 Pengumpulan Data

Data pada penelitian ini didapat secara manual melalui referensi website <https://www.unuja.ac.id/> disimpan dengan format json yang sangat cocok untuk untuk membangun model chatbot [16]. Data Json merupakan data yang termasuk dalam semi structural yang berisi data seperti *tag*, *pattern*, *context*, dan *respons* [17]. Pengambilan data dilakukan kurang lebih 1 bulan dengan malakukan penyesuaian pertanyaan dan jawaban. Data berisi label tag, pertanyaan dan jawaban kemudian di proses untuk dilakukan *training* sebanyak 200 kali untuk menghasilkan akurasi yang baik.

Table 1. Contoh Data

Tag	Patterns	Responses
greeting	halo	Hai, aku bot siap membantumu
univ	Apa itu universitas nurul jadid?	Universitas Nurul Jadid adalah perguruan tinggi swasta favorit berbasis pesantren di jawa timur.
fakultas	Ada fakultas apa saja di sana?	Ada fakultas teknik, fakultas agama islam, fakultas sosial & humaniora dan fakultas kesehatan

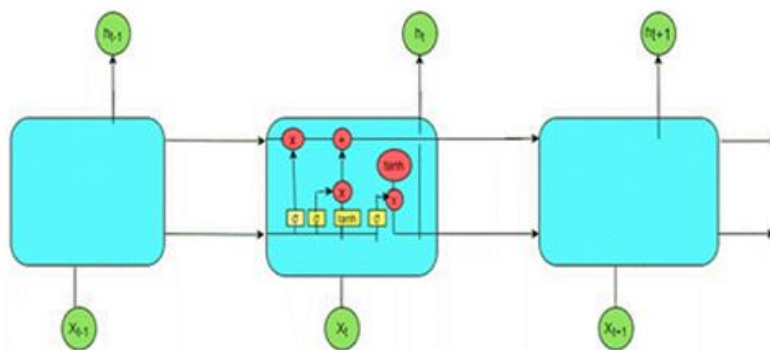
2.2 Preprocessing Data

Tahapan selanjutnya setelah data didapat dan diberikan label, maka dilakukan tahap-tahap preprocessing untuk membersihkan data agar mesin mudah untuk mengolah data tersebut. Oleh karena itu, data harus diseleksi terlebih dahulu melalui tahapan *remove punctuation*, *lemmatization*, *tokenization*, *padding text* dan *encoding*. Berikut adalah urutan tahapannya.

- Remove Punctuation* adalah tahap pemrosesan data teks dengan menghapus tanda seperti special character berupa tanda '?', '!', '.', ',', ';', dan tanda baca lainnya. Tahapan ini berfungsi mempermudah pemrosesan data teks.
- Lemmatization* adalah proses dimana merujuk pada melakukan sesuatu menggunakan *vocabulary* dan analisis morfologi kata guna menghilangkan inflectional endings only dan untuk mengembalikan bentuk *dictionary* dari suatu kata yang disebut sebagai lemmatizer. Contoh Lemmatisasi: Menggunakan (Kata Imbuhan) -> Guna (Kata Dasar).
- Tokenization* merupakan suatu proses memberikan urutan karakter dan sebuah unit dokumen terdefinisi. Tokenisasi bertugas untuk membagi kalimat menjadi beberapa bagian yang dinamakan Token dan menghilangkan bagian tertentu seperti tanda baca. Contohnya: Aku Pergi Ke Makassar -> 'Aku' 'Pergi' 'Ke' 'Makassar'.
- Padding Text* yaitu proses merubah panjang kata agar memiliki panjang sama dengan cara menambah nilai 0 secara suffiks atau secara prefiks sampai pada panjang maksimum. Hal ini sama saja seperti melakukan *resize*. Pada penggunaannya kita harus import library `pad_sequence` lalu memanggil fungsi `pad_sequence()` kemudian masukkan hasil token sebagai parameter.
- Encoding* merupakan pengkodean data kategori seperti huruf atau teks menjadi data angka atau numerik menggunakan bahasa biner komputer yaitu 0 dan 1 untuk mempermudah proses komputasi data teks.

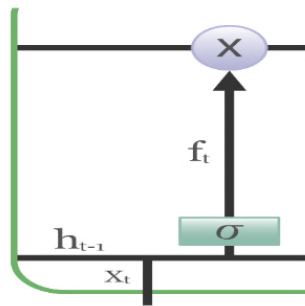
2.3 Membuat Model

Pada tahapan membangun model machine learning, data yang telah di preprocessing akan digunakan untuk pemodelan. Teknik yang digunakan untuk membuat model adalah *Long Short-Term Memory (LSTM)* yang merupakan perhitungan pembelajaran mendalam yang populer dan cocok untuk penerapan klasifikasi dan prediksi data waktu dan teks [18]. LSTM pengembangan dari algoritma RNN (*Recurrent Neural Network*) [19]. RNN bisa menghasilkan prediksi yang lebih akurat berdasarkan data yang tersedia namun tidak mampu memprediksi kata-kata yang tersimpan dalam memori jangka panjang. Untuk mengatasi kekurangan ini dan menjaga keunggulan metode RNN [20], algoritma LSTM dibuat guna menyimpan informasi dalam jangka waktu panjang [21]. Selanjutnya, dengan menggunakan data deret waktu, data tersebut digunakan untuk membangun model yang menganalisis, memperkirakan, dan mengkategorikan data [22]. Struktur LSTM terdiri atas jaringan syaraf dan beberapa memori blok berbeda yang dinamakan sel. Dari *cell state* dan *secret state* dilanjutkan ke sel berikut seperti pada gambar 2, bentuk persegi panjang berwarna biru merupakan penggambaran sel pada LSTM [14].



Gambar 2. Arsitektur LSTM

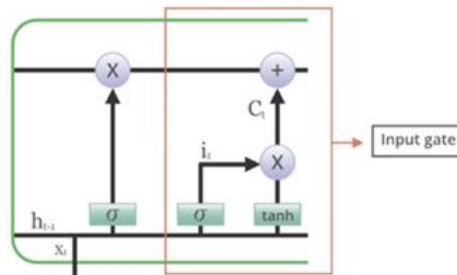
Data yang dikumpulkan melalui perhitungan LSTM disimpan oleh sel, sedangkan pengendalian memori dikerjakan oleh bagian yang bernama pintu (*gate*). Terdapat 3 macam pintu di LSTM, yaitu pintu pengabaian (*forget gate*), pintu masuk (*input gate*), dan pintu hasil (*output gate*) [8], [16].



Gambar 3. Forget Gate

$$F_1 = \sigma(W_f [h_{(t-1)}, x_t] + b_f) \tag{1}$$

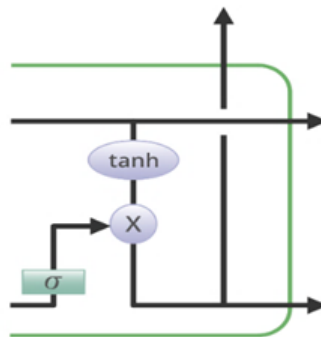
Persamaan (1) merupakan persamaan untuk menghitung *forget gate*. Persamaan ini akan menentukan seberapa banyak informasi yang akan diabaikan dari memori sebelumnya berdasarkan masukan $x(t)$ dan keluaran dari sel sebelumnya $h(t-1)$. F_1 adalah *output gate* untuk mengendalikan penghapusan, yang dihitung dengan cara mengkalikan vektor input dengan matriks bobot (W_f), menambahkan bias (b_f), kemudian menerapkan kedalam fungsi aktivasi sigmoid (σ). Nilai yang dihasilkan akan dilewatkan melalui fungsi aktivasi untuk menghasilkan *output* biner. Jika output nilainya 0, maka informasi akan dihapus dan apabila output nilainya 1 maka, informasi akan disimpan [23].



Gambar 4. Input Gate

$$i_t = \sigma(W_f \cdot [h_{t-1}, x_t] + b_i) \tag{2}$$

Persamaan (2) merupakan persamaan untuk menghitung *input gate*. Menambahkan data berguna ke kedalam *cell* dilakukan oleh information gate menggunakan kemampuan sigmoid dan memfilter *value* yang layak untuk disimpan, siklusnya seperti pada gambar 3 yang menggunakan masukan $h(t-1)$ dan $x(t)$. Selanjutnya dibuatlah *vektor* menggunakan tanh yang memberikan hasil - 1 sampai +1, yang berisi seluruh potensi keuntungan dari $h(t-1)$ dan $x(t)$. Akhirnya, *garis value* diduplikasi untuk mendapatkan data berharga [24].



Gambar 5. Output Gate

$$O_t = \sigma(W_o \cdot [h_{(t-1)}, x_t] + b_o) \tag{3}$$

Persamaan (3) merupakan persamaan untuk menghitung output gate dengan mengekstrak informasi yang berguna dari cell state sebagai *output value* yang dikerjakan *output gate*. Awalnya, *vektor* dimulai menggunakan penerapan kemampuan tanh ke sel lalu, data dikoordinasikan dengan kemampuan *sigmoid* untuk memfilter nilai menggunakan input h_{t-1} dan x_t . Setelah itu, *vektor value* dikalikan dengan value yang dikirimkan sebagai hasil keluaran dan masukan ke sel selanjutnya [25]. LSTM bekerja dalam beberapa tahapan, pertama memilih yang data disimpan dan data yang dibuang dari *cell state*. Lapisan *sigmoid* bertugas memutuskan data baru mana yang harus dihilangkan dan menggantikan data tidak relevan. Kemampuan tanh dan *sigmoid* berperan penting dalam membedakan data penting dan tidak penting. Terakhir, *output* ditentukan dengan bantuan keadaan sel yang merupakan hasil filter dari kemampuan *sigmoid* dan *tanh* [3].

2.4 Training

Training adalah tahapan untuk melatih model agar mesin mengerti dan memahami pola dari model tersebut. Pemodelan yang akan dilakukan di latih sebanyak 200 kali untuk mengetahui hasil akurasi sehingga ketika dilakukan testing dengan apa yang diharapkan

2.5 Evaluasi

Evaluasi dilakukan untuk menganalisis kinerja model melalui *ploting* hasil akurasi dan *loss* untuk melihat kinerja model LSTM. Dengan melakukan analisis evaluasi dari tabel visualisasi kita bisa melihat apakah hasil model sudah baik atau tidak dan apakah terjadi model *overfitting* atau *underfitting* pada model tersebut. Sehingga, dengan mudah mengetahui apakah model ini layak dilakukan pengujian.

3. HASIL DAN PEMBAHASAN

Pengambilan data manual melalui referensi halaman website unuja.ac.id didapatkan total data teks sebanyak 4461 data. Kemudian data tersebut dilakukan pelebelan yakni “*tag*”, “*patterns*”, dan “*responses*” dengan rentang waktu pengambilan dari bulan Januari hingga februari tahun 2024.



Gambar 6. Website Universitas Nurul Jadid

Dari data yang telah diperoleh dan telah diklasifikasi ke dalam label maka akan di lakukan tahapan *preprocessing* data untuk menghasilkan data yang siap untuk di latih.

	patterns	tags
0	hallo	greeting
1	hai	greeting
2	halo	greeting
3	hei	greeting
4	hi	greeting
...
584	ada berapa program studi di s2 ?	Prodi_pasca_sarjana
585	ada berapa prodi di pascasarjana ?	Prodi_pasca_sarjana
586	apakah unuja ada program kuliah untuk s2 ?	pascasarjana
587	apakah menyediakan program kuliah untuk s2 ?	pascasarjana
588	apa ada program s2 di unuja ?	pascasarjana

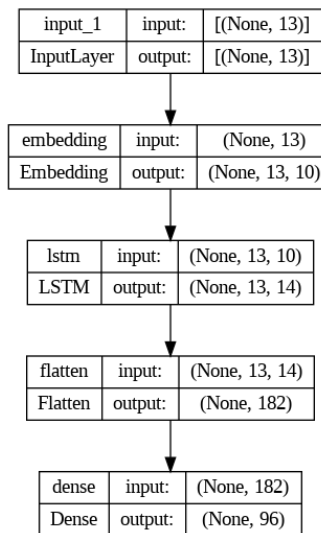
589 rows x 2 columns

Gambar 7. Data Frame

Tabel 2. Data preprocessing

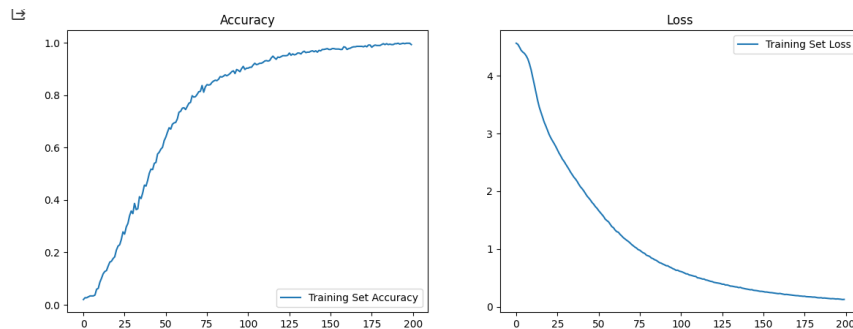
Sebelum	Metode	Sesudah
"Jelaskan tentang universitas nurul jadi!"	Remove Punctuation	["Jelaskan tentang universitas nurul jadi"]
	Lemmatization	["jelas", "tentang", "universitas", "universitas", "nurul", "jadi"]
	Tokenization	["jelaskan", "tentang", "universitas", "nurul", "jadi"]
"Fasilitas apa yang di sediakan?"	Remove Punctuation	["Fasilitas apa yang disediakan"]
	Lemmatization	["fasilitas", "apa", "sedia"]
	Tokenization	["fasilitas", "apa", "yang", "di", "sediakan"]

Pada gambar 7 hanya data pada label tag dan patterns saja yang akan di proses. Dari hasil preprocessing didapatkan jumlah data 4461 diperoleh informasi 227 kata unik dan 96 class. Pada tahapan preprocessing seperti tabel 2 hanya data berlabel "tag" dan "patterns" yang di proses dan akan di lakukan training pada model LSTM. Sebelum di training, harus membuat model dengan arsitektur lstm menggunakan library keras/tensorflow dengan jumlah layer sebanyak 14 [26]. Visualisasi model arsitektur lstm terlihat pada gambar 8 dibawah ini.



Gambar 8 Arsitektur Model

Setelah model terbentuk maka dilanjutkan dengan mengevaluasi model menggunakan plot model untuk mengetahui akurasi dan loss pada model seperti pada gambar 9.



Gambar 9. Evaluasi Model

Visualisasi hasil model lstm dapat disimpulkan bahwa dengan training model sebanyak 200 kali mendapatkan akurasi sebesar 99.32% dan loss sebesar 12.57%. Ini menandakan model sudah bagus dan tidak terjadi *overfitting* atau *underfitting* sehingga model layak untuk dilakukan pengujian dan *deployment*.



Gambar 10. Pengujian

Pada gambar 10 model chatbot LSTM menjawab dengan akurat dan sesuai dengan yang diharapkan sehingga, model bisa disimpan untuk dilakukan *deployment* berbasis *web python framework flask*. Menggunakan *front end* dasar untuk tampilan aplikasinya serta mengintegrasikan model h5 yang telah di simpan menggunakan bahasa *python*.



Gambar 11. Demo Chatbot

Gambar 11 adalah demo tampilan *chatbot* berbasis *website* dengan mengintegrasikan model LSTM untuk mengolah input guna menghasilkan output yang sesuai dengan pertanyaan. *Chatbot* ini responsif bisa di akses melalui *handphone* atau laptop, tidak hanya tampilan namun fitur input ditambahkan fitur *Speech Reqcognition* atau pengenalan suara sehingga *user* bisa memberikan input berupa suara dan input teks dengan di ketik.

4. KESIMPULAN

Penelitian ini telah menjawab permasalahan dengan memberikan solusi berupa aplikasi *chatbot* yang terintegrasi dengan kecerdasan buatan sehingga bisa menjawab pertanyaan dengan cepat dan akurat selama 24/7. Data yang diperoleh secara manual dari referensi web Universitas Nurul Jadid yang kemudian dilakukan pelabelan. Melalui tahap *preprocessing* data teks bisa digunakan untuk proses *training* melalui beberapa tahap hingga bisa diolah. Pemodelan menggunakan algoritma *Long Short-Term Memory* sangat bagus dalam klasifikasi teks, terbukti pada saat data di training sebantak 200 kali menghasilkan akurasi sebesar 99.32% dan

loss sebesar 12.57% sehingga pada saat visualisasi model tidak terlihat *overfitting* atau *underfitting* maka model layak untuk di uji. Saat di uji model dengan baik memahami pertanyaan dari user dan sehingga menghasilkan *output* yang sesuai dengan *input*. Kemudian hasil dari model disimpan dengan format h5 untuk di integrasikan kedalam *chatbot* berbasis website menggunakan *framework flask python* yang responsif dan ada *fitur speech recognition* atau pengenalan suara sehingga memungkinkan user memberikan input berupa suara dan teks. Penelitian ini menghasilkan aplikasi *chatbot* yang terintegrasi kecerdasan buatan dengan hasil cepat dan akurat terhadap pertanyaan. Dengan demikian hasil *deployment* dapat digunakan sebagai asisten virtual dalam membantu staff/panitia pmb untuk merespon pertanyaan.

UCAPAN TERIMA KASIH

Terima kasih kepada rekan sesama peneliti yang telah berbagi ide, saran dan masukan. Terima kasih juga kepada dosen pembimbing yang telah memberikan arahan hingga selesai, hal ini merupakan pengalaman berharga yang tidak akan terlupakan. Terima kasih juga disampaikan kepada keluarga dan teman – teman yang telah mensupport hingga saat ini. Tanpa dukungan dan bantuan dari semua ini, pencapaian ini tidak akan pernah terwujud.

REFERENSI

- [1] A. B. U. THOLIB, "Buku Refrensi Implementasi Algoritma Machine Learning Berbasis Web dengan Framework Streamlit".
- [2] M. Maulidiansyah, "Bot Whatsapp Untuk Pelaporan Pelanggaran Siswa SMP Nurul Jadid," *COREAL: Jurnal Kecerdasan Buatan, Komputasi Dan Teknologi Informasi*, vol. 2, no. 1, 2021.
- [3] M. Agung Nugroho dkk., "PENGEMBANGAN APLIKASI QnA UNTUK PENDAFTARAN MAHASISWA BARU STMIK AKAKOM," 2021.
- [4] A. Zulhilmi dan R. Perdana, "Pengenalan Entitas Bernama Menggunakan Bi-LSTM pada Chatbot Bahasa Indonesia," *Jurnal Teknologi Informasi dan Ilmu Komputer*, vol. 10, no. 7, hlm. 1425–1430, 2023.
- [5] A. H. ISIK dan A. YAĞCI, "Sequence to Sequence LSTM Modeli ile Telegram Bot Uygulaması," *Gazi Mühendislik Bilimleri Dergisi*, vol. 6, no. 1, hlm. 32–39, 2020.
- [6] L. Anindyati, "Analisis dan Perancangan Aplikasi Chatbot Menggunakan Framework Rasa dan Sistem Informasi Pemeliharaan Aplikasi (Studi Kasus: Chatbot Penerimaan Mahasiswa Baru Politeknik Astra)," *Jurnal Teknologi Informasi Dan Ilmu Komputer (Jtiik)*, vol. 10, no. 2, hlm. 291–300, 2023.
- [7] S. D. Nithyanandam, S. Kasinathan, D. Radhakrishnan, dan J. Jebapandian, "NLP for Chatbot Application: Tools and Techniques Used for Chatbot Application, NLP Techniques for Chatbot, Implementation," dalam *Deep Natural Language Processing and AI Applications for Industry 5.0*, IGI Global, 2021, hlm. 142–168.
- [8] M. Solekhah, "Pemanfaatan Teknologi Artificial Intelligence Cubatbot (Culture Balinese Chatbot) Sebagai Informasi Kebudayaan Bali," *Kreativitas Pada Pengabdian Masyarakat (Krepa)*, vol. 1, no. 2, hlm. 90–101, 2023.
- [9] A.-C. Le, "A Deep Reinforcement Learning Model using Long Contexts for Chatbots," dalam *2021 International Conference on System Science and Engineering (ICSSE)*, IEEE, 2021, hlm. 83–87.
- [10] M. Dhyani dan R. Kumar, "An intelligent Chatbot using deep learning with Bidirectional RNN and attention model," *Mater Today Proc.*, vol. 34, hlm. 817–824, 2021.
- [11] P. Anki, A. Bustamam, H. S. Al-Ash, dan D. Sarwinda, "Intelligent chatbot adapted from question and answer system using RNN-LSTM model," dalam *Journal of Physics: Conference Series*, IOP Publishing, 2021, hlm. 012001.
- [12] P. Anki, A. Bustamam, H. S. Al-Ash, dan D. Sarwinda, "Intelligent chatbot adapted from question and answer system using RNN-LSTM model," dalam *Journal of Physics: Conference Series*, IOP Publishing, 2021, hlm. 012001.
- [13] D. Udayan, N. Krishna, T. Reddy, dan L. Dinesh, "Conversational Chatbot for College Management Using LSTM," dalam *Proceedings of the International Conference on Innovative Computing & Communication (ICICC)*, 2022.
- [14] A. Silvani dan R. Subekti, "Aplikasi Chatbot Untuk Faq Akademik Di Ibi-K57 Dengan Lstm Dan Penyematan Kata," *JIKO (Jurnal Informatika dan Komputer)*, vol. 5, no. 1, hlm. 19–27, 2022.
- [15] K. B. Prakash, Y. V. R. Nagapawan, N. L. Kalyani, dan V. P. Kumar, "Chatterbot implementation using transfer learning and LSTM encoder-decoder architecture," *International Journal*, vol. 8, no. 5, 2020.
- [16] A. M. A. Sai, O. Balamurali, M. Karthikeya, dan S. Anand, "A Web-Based Chatbot for Indian Cities: A Comparison of CNN, ANN, and LSTM Models," dalam *2023 14th International Conference on Computing Communication and Networking Technologies (ICCCNT)*, IEEE, 2023, hlm. 1–6.
- [17] F. A. Al Farisi, R. S. Perdana, dan P. P. Adikara, "Klasifikasi Intensi dengan Metode Ling Short-Term Memory pada Chatbot Bahasa Indonesia," *Jurnal Teknologi Informasi dan Ilmu Komputer*, vol. 10, no. 7, hlm. 1511–1518, 2023.
- [18] E. D. Pratama, "Implementasi Model Long-Short Term Memory (LSTM) pada Klasifikasi Teks Data SMS Spam Berbahasa Indonesia," *The Journal on Machine Learning and Computational Intelligence (JMLCI)*, vol. 1, no. 2, 2022.
- [19] P. Anki, A. Bustamam, H. S. Al-Ash, dan D. Sarwinda, "Intelligent chatbot adapted from question and answer system using RNN-LSTM model," dalam *Journal of Physics: Conference Series*, IOP Publishing, 2021, hlm. 012001.
- [20] M. Dhyani dan R. Kumar, "An intelligent Chatbot using deep learning with Bidirectional RNN and attention model," *Mater Today Proc.*, vol. 34, hlm. 817–824, 2021.
- [21] M. Syarovy, A. P. Nugroho, L. Sutiarto, M. S. Muna, A. Wiratmoko, dan S. Primananda, "Prediction of Oil Palm Production Using Recurrent Neural Network Long Short-Term Memory (RNN-LSTM)," dalam *Proceedings of the 3rd International Conference on Smart and Innovative Agriculture (ICoSIA 2022)*, Springer Nature, 2023, hlm. 55.
- [22] Y.-L. Hsueh dan T.-L. Chou, "A Task-oriented Chatbot Based on LSTM and Reinforcement Learning," *ACM Transactions on Asian and Low-Resource Language Information Processing*, vol. 22, no. 1, hlm. 1–27, 2022.
- [23] N. Lhasiw, N. Sanglerdsinlapachai, dan T. Tanantong, "A bidirectional LSTM model for classifying Chatbot messages," dalam *2021 16th International Joint Symposium on Artificial Intelligence and Natural Language Processing (ISAInLP)*, IEEE, 2021, hlm. 1–6.

- [24] M. I. T. Khaqiqi, N. H. Harani, dan C. Prianto, "Performance Analysis and Development of QnA Chatbot Model Using LSTM in Answering Questions," *Indonesian Journal of Computer Science*, vol. 12, no. 3, 2023.
- [25] V. R. Prasetyo, N. Benarkah, dan V. J. Chrisintha, "Implementasi Natural Language Processing Dalam Pembuatan Chatbot Pada Program Information Technology Universitas Surabaya," *Teknika*, vol. 10, no. 2, hlm. 114–121, Jul 2021, doi: 10.34148/teknika.v10i2.370.
- [26] S. Patil, V. Mudaliar, dan P. Kamat, "LSTM based Ensemble Network to enhance the learning of Long-term Dependencies in Chatbot," *International Journal of Automation and Smart Technology*, vol. 12, no. 1, hlm. 2286, 2022.