

KLASIFIKASI INTENSI DENGAN METODE LONG SHORT-TERM MEMORY PADA CHATBOT BAHASA INDONESIA

Faiz Aulia Al Farisi¹, Rizal Setya Perdana², Putra Pandu Adikara³

^{1,2,3}Universitas Brawijaya, Malang
Email: ¹faizaulia727@gmail.com, ²rizalespe@ub.ac.id, ³adikara.putra@ub.ac.id
*Penulis Korespondensi

(Naskah masuk: 08 November 2023, diterima untuk diterbitkan: 30 Oktober 2024)

Abstrak

E-government merupakan sebuah konsep pemerintahan yang menyelenggarakan layanan publik secara digital yang didukung oleh teknologi informasi, sehingga menjadi lebih prima. Salah satu bentuk layanan publik digital yang umum ada di berbagai sektor adalah *helpdesk*. *Helpdesk* memungkinkan pengguna dapat bertanya atau melaporkan sesuatu untuk kemudian dijawab oleh staf. Permasalahan yang ada adalah ketersediaan staf yang hanya dapat menjawab pada jam kerja, sedangkan pengguna bisa jadi menemukan permasalahan yang *urgent* di luar jam kerja. Oleh karena itu, solusi yang ditawarkan adalah penerapan *chatbot*, sehingga dapat melayani kapan pun, sekaligus meringankan kerja dari staf. Penelitian ini mencoba untuk merancang salah satu komponen pada *chatbot*, yaitu model klasifikasi intensi, dengan metode Long Short-Term Memory. Data yang digunakan merupakan 501 teks riwayat pertanyaan dari *database Helpdesk TIK UB* yang termasuk ke dalam 7 kelas intensi yang ditentukan. Data akan melalui beberapa tahap prapemrosesan sebelum kemudian dilakukan pemodelan dan beberapa pengujian. Tahap pengujian meliputi pemilihan *embedding* yang digunakan, pemilihan teknik augmentasi data, dan penyetalan *hyperparameter*. Hasil dari keseluruhan pengujian, didapatkan model terbaik yang mampu menghasilkan akurasi sempurna untuk data latih dan data uji, serta *loss* 0,004 untuk data latih dan 0,044 untuk data uji.

Kata kunci: klasifikasi intensi, LSTM, embedding, augmentasi data

INTENT CLASSIFICATION WITH LONG SHORT-TERM MEMORY METHOD IN INDONESIAN LANGUAGE CHATBOT

Abstract

E-government is a government concept that organizes digital public services supported by information technology, so that they become more excellent. One form of digital public service that is common in various sectors is the *helpdesk*. *Helpdesk* allows users to ask questions or report something to be answered by staff. The problem that exists is the availability of staff who can only answer during working hours, while users may find urgent problems outside of working hours. Therefore, the solution offered is the implementation of a *chatbot*, so that it can serve at any time, while easing the work of staff. This study attempts to design one of the components of the *chatbot*, namely the intention classification model, using the Long Short-Term Memory method. The data used are 501 question history texts from the *Helpdesk TIK UB database* which fall into the 7 specified intensity classes. The data will go through several pre-processing stages before then being modeled and tested. The testing phase includes selecting the *embedding* used, data augmentation techniques selection, and *hyperparameter* tuning. The best model is obtained which can produce perfect accuracy for training data and test data, as well as a *loss* of 0.004 for training data and 0.044 for test data.

Keywords: intent classification, LSTM, embedding, data augmentation

1. PENDAHULUAN

Pada era kemajuan teknologi informasi saat ini, arus informasi berlangsung dengan sangat cepat. Semua pihak dituntut untuk mengikuti perubahan yang cepat apabila tidak ingin tertinggal dengan

zaman. Upaya untuk melakukan transformasi digital harus dilakukan oleh berbagai sektor. Demikian juga dengan sektor pemerintahan, berbagai layanan publik yang disediakan akan menjadi lebih prima ketika diselenggarakan secara digital. Konsep pemerintahan

dengan layanan publik yang diselenggarakan secara digital menggunakan teknologi informasi tersebut dikenal dengan istilah *e-government* (Rachmatullah & Purwani, 2022).

Digitalisasi layanan publik juga diperlukan pada pemerintahan di sektor pendidikan, salah satunya pada perguruan tinggi atau universitas. Sebagian besar universitas telah menerapkan teknologi informasi pada layanannya, seperti pengadaan sistem informasi akademik berbasis web (Guntoro et al., 2020). Pengadaan web tersebut dapat mempermudah baik bagi pihak universitas maupun civitas academica dalam melakukan kegiatan perkuliahan.

Universitas Brawijaya (UB) sebagai salah satu universitas ternama di Indonesia juga telah menerapkan digitalisasi ke berbagai layanannya. Dalam pengadaan layanan digital tersebut, tidak jarang mahasiswa, dosen, maupun pengguna lainnya mengalami suatu kesulitan atau permasalahan. Oleh karena itu, UB menyediakan Helpdesk TIK, yaitu sebuah layanan *helpdesk online* yang digunakan untuk memudahkan *civitas academica* UB yang memerlukan bantuan mengenai layanan IT di UB. *Civitas academica* UB dapat membuka sebuah tiket baru untuk melaporkan permasalahannya, kemudian staf TIK UB akan menindaklanjuti tiket tersebut setelah beberapa waktu (Helpdesk TIK, 2023). Staf TIK UB hanya dapat merespon pada hari dan jam kerja, sedangkan terkadang terdapat beberapa permasalahan yang mendesak ditemui di luar jam kerja.

Berdasarkan permasalahan tersebut, solusi yang dapat ditawarkan adalah penerapan *chatbot* pada Helpdesk TIK. *Chatbot* merupakan sebuah sistem dialog yang dibuat dengan kecerdasan buatan untuk dapat meniru dan melakukan percakapan dengan manusia (Fauzia et al., 2021). *Chatbot* dapat beroperasi 24 jam non-stop, sehingga dapat merespon kapan saja *civitas academica* UB membutuhkan bantuan. Selain itu, penggunaan *chatbot* juga akan meringankan beban kerja dari staf TIK UB.

Oleh karena itu, pada penelitian kali ini, penulis mencoba membuat sebuah sistem klasifikasi intensi pengguna untuk kemudian diterapkan pada *chatbot* Helpdesk TIK UB. Klasifikasi intensi pengguna merupakan sebuah proses penting pada *chatbot* agar maksud dari pengguna dapat dipahami dengan baik oleh *chatbot*. Pada penelitian ini, klasifikasi dilakukan dengan metode Long-Short Term Memory (LSTM), karena metode ini dapat mengingat/menyimpan informasi kontekstual yang penting. Hal ini sesuai dengan kebutuhan klasifikasi intensi, karena informasi sebelumnya dapat memengaruhi intensi dari pengguna (Di Gennaro et al., 2021). Selain LSTM, klasifikasi dibantu dengan penerapan embedding BERT yang berperan untuk merepresentasikan teks sesuai konteks. Dengan sistem klasifikasi intensi yang baik, diharapkan dapat menjadi tahap awal dari sistem *chatbot* yang handal

dan dapat menyelesaikan permasalahan keterbatasan ketersediaan layanan *helpdesk*.

2. LANDASAN KEPUSTAKAAN

2.1 Penelitian Terdahulu

Setyawan et al. (2018) membahas mengenai perbandingan penggunaan metode Naïve Bayes dan Logistic Regression dalam klasifikasi intensi pada *chatbot*. Data terdiri dari 55 data yang terbagi ke 6 kelas intensi. Hasil akurasi, *precision*, *recall* dan *f1-score* yang didapatkan adalah 0,63; 0,53; 0,64; 0,56 untuk model Naïve Bayes dan 0,72; 0,59; 0,73; 0,64 untuk model Logistic Regression. Rendahnya nilai-nilai tersebut diduga disebabkan oleh keterbatasan jumlah data dan persebaran yang tidak merata untuk jumlah data di tiap kelas intensinya.

Bhartiya et al. (2019) membuat sebuah *chatbot* yang mampu menjawab pertanyaan pengguna terkait konseling di bidang *engineering*. Penelitian tersebut menggunakan data latih sebanyak 1059 data yang tersebar ke 62 kelas intensi. Data latih yang telah dilakukan beberapa pemrosesan kemudian dilakukan pemodelan klasifikasi intensi menggunakan metode Feedforward Neural Network. Hasil evaluasi model dengan 100 data uji menghasilkan prediksi intensi yang benar untuk 78 data uji.

Kedua penelitian di atas sama-sama melakukan klasifikasi intensi seperti yang dilakukan pada penelitian ini. Perbedaannya terletak pada algoritma klasifikasi yang digunakan, pada penelitian tersebut menggunakan Naïve Bayes, Logistic Regression, dan Feedforward Neural Network, sedangkan pada penelitian kali ini menggunakan LSTM. Selain itu, dataset yang digunakan juga berbeda.

Bilal et al. (2022) melakukan klasifikasi intensi dari *dataset Airline Travel Information System (ATIS)* yang diterjemahkan ke bahasa Indonesia. *Dataset* tersebut terdiri dari 4603 data latih dan 386 data uji yang terbagi ke 21 kelas intensi. Data kemudian dipetakan ke ruang vektor dengan bantuan *GloVe embedding* yang dilatih dengan korpus bahasa Indonesia. Baru kemudian data latih digunakan untuk melatih model yang menggunakan algoritma CNN, RNN, LSTM, dan BiLSTM. Hasilnya, akurasi tertinggi diraih oleh CNN dengan akurasi mencapai 95,84%; diikuti dengan BiLSTM dan LSTM pada angka 94% dan 93,8%.

Denny Prabowo et al. (2018) membandingkan penggunaan algoritma LSTM dan RNN pada *chatbot* bahasa Indonesia. *Chatbot* yang dibangun secara *end-to-end*, sehingga tidak terdapat proses klasifikasi intensi secara eksplisit. Data yang digunakan merupakan 50 data percakapan pada layanan pemesanan tiket. Hasilnya, algoritma LSTM mampu memberikan jawaban yang lebih akurat dibanding dengan RNN.

Kedua penelitian di atas telah menerapkan algoritma LSTM seperti yang digunakan pada penelitian kali ini. Perbedaannya, selain pada dataset

yang digunakan, pada penelitian yang dilakukan oleh Bilah et al. (2022) menggunakan *embedding* GloVe, sedangkan pada penelitian kali ini menggunakan BERT. Kemudian, pada penelitian yang dilakukan oleh Denny Prabowo et al. (2018), chatbot dibangun secara *end-to-end*, sehingga klasifikasi intensi tidak dilakukan secara eksplisit.

2.2 Klasifikasi Intensi

Klasifikasi intensi merupakan salah satu *task* yang ada dalam komponen utama dari *chatbot*, yaitu pemahaman bahasa alami atau *natural language understanding* (NLU) (Galitsky, 2019). Klasifikasi intensi berperan dalam memahami topik diskusi ketika chatbot berkomunikasi dengan manusia, sehingga secara tidak langsung juga berpengaruh pada penentuan jawaban yang benar dalam percakapan (Bilah et al., 2022).

2.3 Long Short-Term Memory

Long Short-Term Memory (LSTM) merupakan salah satu arsitektur jaringan saraf tiruan yang merupakan peningkatan dari generasi sebelumnya, yaitu Recurrent Neural Network (RNN). LSTM mampu mengatasi masalah yang ada pada RNN, yaitu hilangnya gradien seiring berjalannya input pada jaringan, dengan menambahkan sebuah unit yang bekerja sebagai ‘sel memori’ pada hidden layer. Dengan demikian, LSTM mampu membawa informasi kontekstual lebih lama (Di Gennaro et al., 2021).

Perhitungan dalam LSTM dibagi pada 4 gate, yaitu forget gate, input gate, cell state candidate, dan output gate (Di Gennaro et al., 2021). Pada setiap gate tersebut terdapat 3 parameter, yaitu bobot masukan, bobot hidden state, dan bias. Sehingga total terdapat 12 parameter untuk setiap unit LSTM yang diinisialisasi menggunakan teknik *xavier initialization*.

Setelah seluruh parameter diinisialisasi, maka dihitung keluaran dari setiap gate pada proses propagasi maju. Perhitungan tersebut dilakukan dengan Persamaan 1-6.

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

$$i_t = \sigma(x_t \cdot W_i + h_{t-1} \cdot U_i + b_i) \quad (2)$$

$$g_t = \tanh(x_t \cdot W_g + h_{t-1} \cdot U_g + b_g) \quad (3)$$

$$c_t = f_t \cdot c_{t-1} + i_t \cdot g_t \quad (4)$$

$$o_t = \sigma(x_t \cdot W_o + h_{t-1} \cdot U_o + b_o) \quad (5)$$

$$h_t = o_t \cdot \tanh(c_t) \quad (6)$$

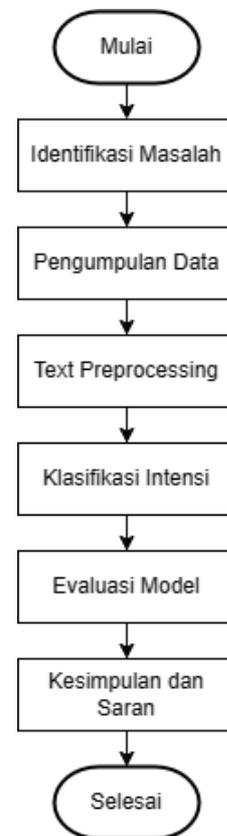
2.4 BERT

Bidirectional Encoder Representations from Transformers (BERT) merupakan sebuah model representasi bahasa yang didesain untuk merepresentasikan teks yang belum memiliki label dengan memperhatikan konteks yang ada di sekitarnya (*bidirectional*). BERT yang sudah

dilakukan *pre-trained* dapat dilakukan *fine-tuning* sehingga menjadi sebuah model yang mutakhir dalam menyelesaikan permasalahan pemrosesan bahasa alami, baik pada tingkat token maupun kalimat. Selain *fine-tuning*, BERT juga dapat dimanfaatkan dengan pendekatan *feature-based*, karena BERT dapat digunakan untuk mengekstraksi fitur dari input (Devlin et al., 2018).

3. METODE PENELITIAN

Penelitian ini dilaksanakan mengikuti alur yang ada pada Gambar 1. Dimulai dengan mengidentifikasi masalah yang ada di Helpdesk TIK UB, kemudian pengumpulan data riwayat percakapan pengguna, lalu pemrosesan dan pemodelan data, dilanjutkan dengan evaluasi performa model, dan diakhiri dengan penarikan kesimpulan dan saran.



Gambar 1 Strategi Penelitian

4. PERANCANGAN

4.1 Dataset

Data yang digunakan pada penelitian ini diambil dari *database* Helpdesk TIK UB, yang berisi riwayat percakapan antara pengguna dengan staf Helpdesk TIK UB. Data yang terekam sejumlah 91977 yang merupakan riwayat percakapan sejak Maret 2020 hingga Desember 2022. Kolom-kolom beserta contoh data yang ada di *database* tersebut dapat dilihat pada Tabel 1.

Tabel 1 Kolom dan contoh data Helpdesk TIK UB

Nama Kolom	Contoh Data
id	37773
pid	0
thread_id	12294
staff_id	0
user_id	9197
type	M
flags	65
poster	Faiz Ats Tsauri Syah
editor	NULL
editor_type	NULL
title	NULL
body	<p>Selamat pagi bapak/ibu, beberapa waktu lalu email ub saya telah dinonaktifkan. Apakah bisa saya untuk membuat email UB yang baru? terimakasih bapak/ibu</p>
format	html
ip_address	10.45.184.75
extra	NULL
recipients	NULL
created	2022-03-07 9:59:46
updated	0000-00-00 00:00:00

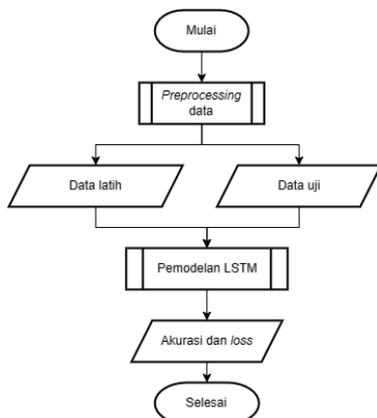
Berdasarkan data yang ada, ditentukan 7 intensi yang menjadi kelas target pada penelitian ini. Tujuh intensi yang menjadi kelas target tersebut ditentukan berdasarkan frekuensi kemunculan kategori permasalahan yang paling umum. Untuk setiap kelas intensi, diambil sekitar 70 data kolom *body* untuk dijadikan dataset penelitian, sehingga total didapatkan 501 data yang distribusinya dapat dilihat pada Tabel 2.

Tabel 2 Distribusi jumlah data tiap intensi

Intensi/kelas target	Jumlah data
buat_reset_webhosting	68
gagal_login	78
permintaan_lisensi_office	82
masalah_vpn	70
mengajukan_email	67
nilai_it_tidak_keluar	75
ucapan_terima_kasih	61

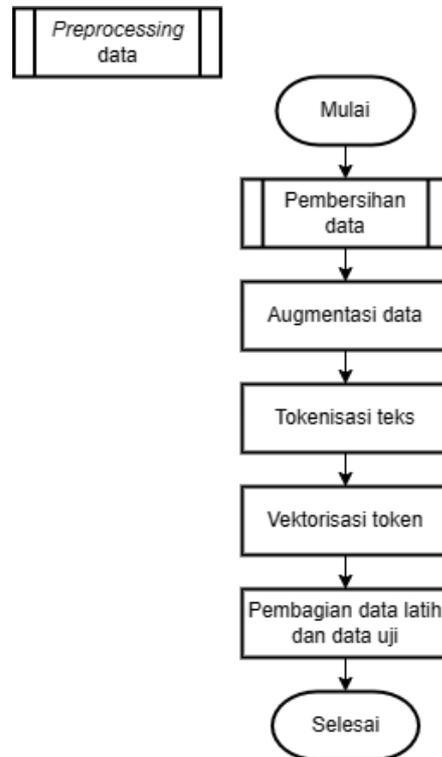
4.2 Perancangan Metode LSTM

Implementasi dari metode LSTM akan mengikuti diagram alir pada Gambar 2, yang terdiri dari 2 proses utama, yaitu *preprocessing* data dan pemodelan LSTM.



Gambar 2. Diagram alir perancangan metode LSTM

Preprocessing data terdiri dari beberapa sub-proses seperti pada Gambar 3, mulai dari pembersihan data dari *tag* HTML, tanda baca, karakter non-ASCII dan *stopword*, kemudian augmentasi data, tokenisasi teks, vektorisasi token, hingga pembagian data latih dan data uji.



Gambar 3. Diagram alir tahapan preprocessing

4.3 Skenario Pengujian

Beberapa pengujian akan diterapkan pada penelitian ini, baik terhadap data maupun terhadap model. Pengujian ini bertujuan untuk membandingkan dan mencari metode yang dapat menghasilkan kinerja terbaik dari model. Terdapat tiga pengujian yang akan dilakukan, yaitu pengujian *embedding*, pengujian teknik augmentasi data, dan pengujian *hyperparameter tuning*.

Pengujian *embedding* membandingkan kinerja model berdasarkan penggunaan 3 *embedding* yang berbeda. *Embedding* digunakan pada tahap augmentasi data, tokenisasi teks, serta vektorisasi token. Tiga variasi *embedding* yang diuji adalah ‘bert-base-uncased’, ‘bert-multilingual-base-cased’, dan ‘indolem/indobert-base-uncased’. Pengujian teknik augmentasi membandingkan kinerja model berdasarkan 3 jenis teknik augmentasi data, yaitu penambahan kata (*random word insertion*), substitusi kata (*random word replacement*), dan translasi balik (*backtranslation*). Pengujian terakhir, *hyperparameter tuning* mencari kombinasi *hyperparameter* terbaik yang menghasilkan model dengan nilai *validation loss* terkecil.

5. HASIL DAN ANALISIS

5.1 Pengujian *Embedding*

Hasil dari pelatihan menggunakan ketiga *embedding* yang berbeda dapat dilihat pada Tabel 3. Pengujian dinilai berdasarkan akurasi dan loss dari data latih dan data uji.

Tabel 3. Hasil pengujian *embedding*

<i>Embedding</i>	<i>Train acc</i>	<i>Train loss</i>	<i>Val acc</i>	<i>Val loss</i>
bert-base-uncased	0,977	0,073	0,925	0,149
bert-multilingual-base-cased	1,000	0,005	0,970	0,078
indolem/indobert-base-uncased	1,000	0,005	0,955	0,147

Berdasarkan hasil pengujian ketiga *embedding* terhadap kinerja model, yang dapat dilihat pada Tabel 3, *embedding* ‘bert-multilingual-base-cased’ menghasilkan kinerja model terbaik setelah pelatihan selama 20 *epoch*. *Embedding* ‘bert-multilingual-base-cased’ berhasil mengungguli dua *embedding* yang lain dari seluruh nilai metrik pengujian, yaitu akurasi dan *loss* pada pelatihan dan pengujian. Hal ini berarti *embedding* ‘bert-multilingual-base-cased’ dapat menghasilkan representasi kontekstual yang lebih baik terhadap data yang ada, dibandingkan dengan kedua *embedding* yang lain. Oleh karena itu, *embedding* ‘bert-multilingual-base-cased’ akan digunakan pada pengujian-pengujian selanjutnya.

5.2 Pengujian Teknik Augmentasi Data

Teknik augmentasi data yang berbeda akan menghasilkan data baru yang berbeda pula. Contoh data hasil augmentasi dari setiap teknik augmentasi dapat dilihat pada Tabel 4. Hasil dari pelatihan model menggunakan ketiga teknik augmentasi yang berbeda dapat dilihat pada

Tabel 5.

Tabel 4. Hasil augmentasi data

Teknik Augmentasi	Contoh Data
Data awal	Sebelum terkait password BAIS salah reset password ulang password masukan sesuai kirimkan email akses gapura Siam
<i>Random word insertion</i>	debut Sebelum debut terkait batas password berikut BAIS salah reset password berat ulang atau password masukan sesuai kemampuan kirimkan email akses SMS gapura lagu Siam
<i>Random word replacement</i>	Sebelum terkait password dari yang atau password terjadi dari yang sesuai dengan melalui akses gapura suara
<i>Backtranslation</i>	Sebelum mengasosiasikan sandi BAIS misreset kata sandi masukan menurut mengirim email akses Siam

Tabel 5. Hasil pengujian teknik augmentasi data

Teknik augmentasi	<i>Train acc</i>	<i>Train loss</i>	<i>Val acc</i>	<i>Val loss</i>
<i>Random word insertion</i>	0,977	0,073	0,925	0,149
<i>Random word replacement</i>	1,000	0,005	0,970	0,078
<i>Backtranslation</i>	1,000	0,005	0,955	0,147

Berdasarkan hasil pengujian ketiga teknik augmentasi data terhadap kinerja model, yang dapat dilihat pada

Tabel 5, teknik *random word insertion* menghasilkan kinerja model terbaik setelah menambahkan 168 data dan melakukan pelatihan model selama 20 *epoch*. Hasil pengujian untuk teknik *random word insertion* dan *backtranslation* hanya berbeda sedikit, *loss* yang dihasilkan mencapai nilai kurang dari 0,1 terutama untuk *loss* pengujian, sedangkan teknik *random word replacement* menghasilkan nilai yang relatif lebih buruk untuk akurasi dan *loss* pengujian.

Perbedaan hasil yang cukup signifikan ini mungkin disebabkan oleh perbedaan kualitas data yang dihasilkan. Pada *random word insertion*, data teks akan ditambahkan beberapa kata. Apabila kata yang ditambahkan tidak sesuai dengan konteks dari keseluruhan teks, konteks awal dari teks masih tetap terjaga. Demikian juga dengan *backtranslation*, teks baru yang dihasilkan mungkin akan berbeda, namun makna dari teks masih tetap terjaga karena hasil terjemahan tiap kata akan tetap memiliki makna yang sama. Namun, pada *random word replacement*, beberapa kata akan dihilangkan dari data teks, kemudian diganti dengan kata yang berhubungan dengan kata sekitarnya. Dengan menghilangkan beberapa kata, konteks awal dari teks dapat berubah, sehingga mengganggu kinerja model dalam mengklasifikasikan teks.

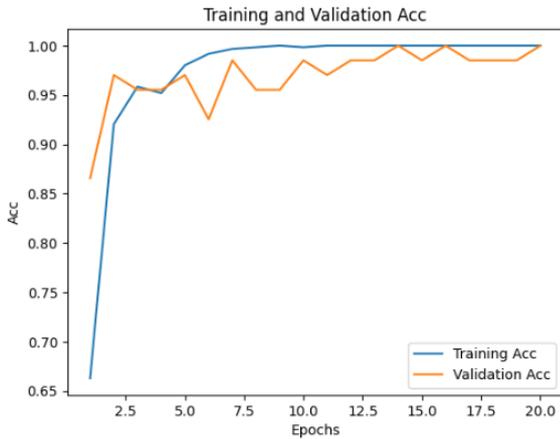
5.3 Pengujian *Hyperparameter Tuning*

Pada pengujian *hyperparameter tuning*, nilai *validation loss* dipilih sebagai acuan kinerja terbaik model. Pemilihan ini didasari oleh keterbatasan data dan arsitektur model yang sederhana, maka rawan untuk terjadinya *overfitting* terhadap data latih. Pengujian ini dilakukan sebanyak 100 kali percobaan. Dari 100 percobaan tersebut, didapatkan satu kombinasi *hyperparameter* yang menghasilkan nilai *validation loss* terendah yang tertera pada Tabel 6.

Tabel 6. Kombinasi *hyperparameter* terbaik

<i>Hyperparameter</i>	Nilai
<i>Optimizer</i>	Adam
<i>Learning rate</i>	0,0160
<i>Weight decay</i>	0,0001
<i>Batch size</i>	32
<i>Hidden size</i>	96

Model kemudian diinisiasi ulang menggunakan kombinasi *hyperparameter* terbaik yang dihasilkan dari proses *hyperparameter tuning*. Model tersebut lalu dilatih selama 20 *epoch*. Hasil akhir dari pelatihan model hasil *hyperparameter tuning* dapat dilihat pada Gambar 4. *Loss* model *hyperparameter tuning*

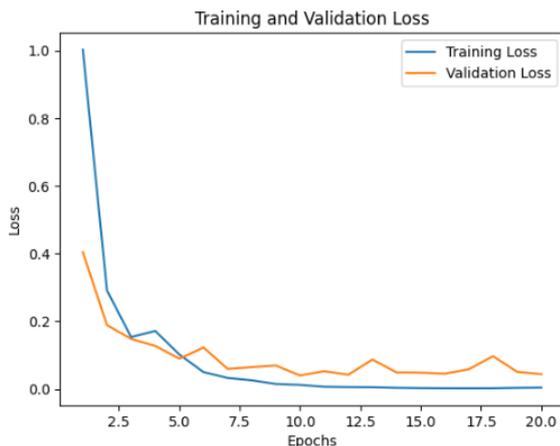


Gambar 5. Akurasi model hyperparameter tuning

, sedangkan untuk grafik riwayat akurasi dan loss dapat dilihat pada Gambar 4 dan Gambar 5.

Tabel 7. Kinerja model hyperparameter tuning

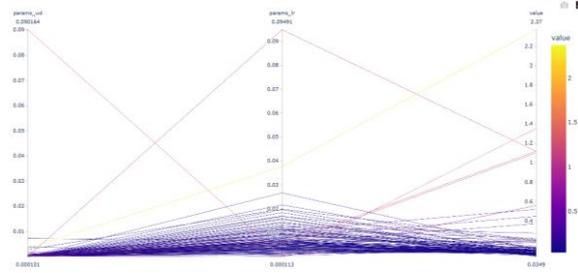
Train acc	Train loss	Val acc	Val loss
1,0000	0,0040	1,000	0,0439



Gambar 4. Loss model hyperparameter tuning



Gambar 5. Akurasi model hyperparameter tuning



Gambar 6. Parallel coordinates untuk hyperparameter weight decay dan learning rate terhadap objective value

Gambar 6 menjelaskan tentang persebaran nilai dari hyperparameter weight decay dan learning rate dan pengaruhnya terhadap objective value, yaitu nilai validation loss. Berdasarkan grafik tersebut, dapat dilihat bahwa ketika nilai dari kedua hyperparameter bernilai rendah atau di bawah 0,03, maka nilai validation loss yang dihasilkan juga relatif rendah. Akan tetapi, ketika salah satu dari kedua hyperparameter tersebut bernilai tinggi, maka berakibat pada buruknya nilai validation loss.

6. PENUTUP

Berdasarkan hasil pengujian, metode *preprocessing* yang diterapkan pada *dataset* Helpdesk TIK UB sebelum dilakukan pemodelan klasifikasi intensi menggunakan metode Long Short-Term Memory mencakup beberapa proses, seperti pembersihan data dari *tag* HTML, karakter non-ASCII, tanda baca, dan *stopword*, kemudian menambah data dengan melakukan augmentasi data dengan memanfaatkan *embedding*, terakhir adalah melakukan tokenisasi dan vektorisasi terhadap data teks menggunakan *embedding*. Model terbaik dihasilkan ketika menggunakan *embedding* ‘bert-multilingual-base-cased’ dan teknik augmentasi data *random word insertion*.

Berdasarkan hasil *hyperparameter tuning* sebanyak 100 kali percobaan, didapatkan kombinasi *hyperparameter* terbaik seperti pada Tabel 6. Nilai dari *hyperparameter* terbaik bisa jadi dapat berubah dikarenakan keterbatasan jumlah percobaan dan sensitifnya model terhadap data. Namun satu hal yang dapat disimpulkan dari percobaan ini adalah pengaruh *learning rate* dan *weight decay* lebih signifikan dibanding *hyperparameter* lainnya.

Setelah melalui beberapa pengujian, hasil evaluasi terbaik dari model klasifikasi intensi dengan metode Long-Short Term Memory adalah nilai 1,000 untuk akurasi latih dan akurasi uji, serta 0,0040 untuk *loss* latih dan 0,0439 untuk *loss* uji. Nilai tersebut didapatkan dari pembagian data 9:1, distribusi data yang berbeda sangat mungkin untuk menghasilkan nilai yang berbeda pula.

Penambahan dan pemilihan data yang tepat untuk setiap intensinya dapat memberikan *knowledge* lebih banyak dan lebih baik pada model, sehingga seharusnya mampu menghasilkan model dengan kinerja lebih baik dan lebih global. Selain itu,

eksplorasi metode *preprocessing* dan kombinasi nilai *hyperparameter* lain mungkin juga dapat menghasilkan kinerja model yang lebih baik.

DAFTAR PUSTAKA

- BHARTIYA, N., JANGID, N., JANNU, S., SHUKLA, P., & CHAPANERI, R. (2019). Artificial Neural Network Based University Chatbot System. *2019 IEEE Bombay Section Signature Conference (IBSSC)*, 1–6. <https://doi.org/10.1109/IBSSC47189.2019.8973095>
- BILAH, C. O., ADJI, T. B., & SETIAWAN, N. A. (2022). Intent Detection on Indonesian Text Using Convolutional Neural Network. *2022 IEEE International Conference on Cybernetics and Computational Intelligence (CyberneticsCom)*, 174–178. <https://doi.org/10.1109/CyberneticsCom55287.2022.9865291>
- DENNY PRABOWO, Y., WARNARS, H. L. H. S., BUDIHARTO, W., KISTIANTORO, A. I., HERIYADI, Y., & LUKAS. (2018). Lstm And Simple Rnn Comparison In The Problem Of Sequence To Sequence On Conversation Data Using Bahasa Indonesia. *2018 Indonesian Association for Pattern Recognition International Conference (INAPR)*, 51–56. <https://doi.org/10.1109/INAPR.2018.8627029>
- DEVLIN, J., CHANG, M.-W., LEE, K., & TOUTANOVA, K. (2018). *BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding*.
- DI GENNARO, G., BUONANNO, A., DI GIROLAMO, A., OSPEDALE, A., & PALMIERI, F. A. N. (2021). Intent Classification in Question-Answering Using LSTM Architectures. In *Progresses in Artificial Intelligence and Neural Systems* (pp. 115–124). https://doi.org/10.1007/978-981-15-5093-5_11
- FAUZIA, L., HADIPRAKOSO, R. B., & GIRINOTO. (2021). Implementation of Chatbot on University Website Using RASA Framework. *2021 4th International Seminar on Research of Information Technology and Intelligent Systems (ISRITI)*, 373–378. <https://doi.org/10.1109/ISRITI54043.2021.9702821>
- GALITSKY, B. (2019). Chatbot Components and Architectures. In *Developing Enterprise Chatbots* (pp. 13–51). Springer International Publishing. https://doi.org/10.1007/978-3-030-04299-8_2
- GUNTORO, G., COSTANER, L., & LISNAWITA, L. (2020). Aplikasi Chatbot untuk Layanan Informasi dan Akademik Kampus Berbasis Artificial Intelligence Markup Language (AIML). *Digital Zone: Jurnal Teknologi Informasi Dan Komunikasi*, 11(2), 291–300. <https://doi.org/10.31849/digitalzone.v11i2.5049>
- HELPDESK TIK. (2023, February 22). *Helpdesk TIK*. <https://helpdesk-tik.ub.ac.id/>
- RACHMATULLAH, N., & PURWANI, F. (2022). Analisis Pentingnya Digitalisasi & Infrastruktur Teknologi Informasi Dalam Institusi Pemerintahan: E-Government. *JURNAL FASILKOM*, 12(1), 14–19. <https://doi.org/10.37859/jf.v12i1.3512>
- SETYAWAN, M. Y. H., AWANGGA, R. M., & EFENDI, S. R. (2018). Comparison Of Multinomial Naive Bayes Algorithm And Logistic Regression For Intent Classification In Chatbot. *2018 International Conference on Applied Engineering (ICAE)*, 1–5. <https://doi.org/10.1109/INCAE.2018.8579372>

Halaman ini sengaja dikosongkan.