ISSN: 1829-667X / E-ISSN: 2460-9021
DOI:10.31315/telematika.v22i2.14637

Implementation of Natural Language Processing with Deep Learning on Chatbot UKT (Uang Kuliah Tunggal) *University*

Implementasi *Natural Language Processing* dengan *Deep Learning* pada *Chatbot* UKT (Uang Kuliah Tunggal) Kampus

Ridha Evitafany[1], Rheza Ari Wibowo[2], Imam Adi Nata[3]

[1,2,3] Teknik Elektro, Mekatronika, dan Teknologi Informasi Universitas Tidar, Indonesia

[1*]ridha.evitafany@students.untidar.ac.id, [2]rhezaari@untidar.ac.id,
[3]imamadinata@untidar.ac.id

Informasi Artikel
Received: January 2025
Revised: April 2025
Accepted: June 2025
Published: June 2025

Abstract

Purpose: The research conducted is to create a chatbot that is able to classify question text based on intent/label more precisely, and help facilitate students in obtaining additional information related to UKT (Uang Kuliah Tunggal) services and tuition fees. This chatbot is able to overcome the limitations experienced by the Administration of the Faculty of Engineering (TU FT) of Tidar University, such as human delay in responding to emails or live chatting, limited service hours (office hours), and limited information on the campus website. The chatbot was developed using Natural Language Processing (NLP) approach and Deep Learning Bidirectional Long Short-Term Memory (BiLSTM) algorithm. The results of the chatbot system are integrated into the Telegram application to see the level of user satisfaction after interacting.

Design/methodology/approach: The process of this research begins with collecting datasets in the form of questions and answers that have been tagged or labelled in JSON file format. The dataset is subjected to text normalisation or Natural Language Processing (NLP) preprocessing, where this stage is carried out lower casing or case folding, punctuation removal, remove excess spaces, stop words and stemming with the Sastrawi library, tokenisation, and padding. Next, split the dataset process with a division of 80% training 20% testing, before processing into feature extraction, using FastText for word embedding. Next, the process of question text classification with the Bidirectional Long Short-Term Memory (BiLSTM) model and continued evaluation with confusion matrix. The last stage, is the integration of the chatbot into the Telegram bot, then conducting user testing of the chatbot and

ISSN: 1829-667X / E-ISSN: 2460-9021
DOI:10.31315/telematika.v22i2.14637

measuring the level of satisfaction with the Customer Satisfaction Score (CSAT) method.
Findings/result: The accuracy of the classification model produces a value of 96.05% and user testing by applying the Customer Satisfaction Score (CSAT) method provides an average satisfaction level in the range of 4 (Satisfied) and 5 (Very Satisfied) with a result of 88.86% based on the following points 'Satisfaction with the answers provided', 'Understanding questions and answers is easy to understand', and 'Performance as expected'.
Originality/value/state of the art: Research on question text classification on chatbot with Natural Language Processing (NLP) approach and Bidirectional Long Short-Term Memory (BiLSTM) model to handle the question and answer problem of UKT (Uang Kuliah Tunggal) service and tuition fee of Faculty of Engineering, Tidar University has never been done before.

Keywords: Artificial Intelligence (AI); Chatbot; Natural Language Processing (NLP); Bidirectional Long Short-Term Memory (BiLSTM); Customer Satisfaction Score (CSAT)

Kata kunci: Artificial Intelligence (AI); Chatbot; Natural Language Processing (NLP); Bidirectional Long Short-Term Memory (BiLSTM); Customer Satisfaction Score (CSAT)

Abstrak

Tujuan: Penelitian yang dilakukan yaitu untuk membuat *chatbot* yang mampu mengklasifikasikan teks pertanyaan berdasarkan *intent*/*label* dengan lebih tepat, dan membantu mempermudah mahasiswa dalam mendapatkan informasi tambahan terkait layanan UKT (Uang Kuliah Tunggal) maupun biaya kuliah. *Chatbot* ini mampu mengatasi keterbatasan yang dialami oleh Tata Usaha Fakultas Teknik (TU FT) Universitas Tidar, seperti *human delay* dalam merespons *email* atau *live chatting*, keterbatasan jam layanan (*office hour*), serta informasi yang terbatas di *website* kampus. Pembuatan *chatbot* menggunakan pendekatan *Natural Language Processing* (NLP) dan algoritma *Deep Learning Bidirectional Long Short-Term Memory* (BiLSTM). Hasil sistem *chatbot* diintegrasikan ke dalam aplikasi Telegram untuk melihat tingkat kepuasan pengguna setelah berinteraksi.
Perancangan/metode/pendekatan: Proses dari penelitian ini dimulai dengan pengumpulan *dataset* berupa pertanyaan dan jawaban yang telah diberi *tag* atau label dalam format JSON *file*. *Dataset* tersebut dilakukan proses *text normalization* atau *preprocessing Natural Language Processing* (NLP), di mana tahap ini dilakukan *lower casing* atau *case folding*, *punctuation removal*, hapus spasi berlebih, *stop word* dan *stemming* dengan *library* Sastrawi, tokenisasi, serta *padding*. Selanjutnya melakukan proses *split dataset* dengan pembagian 80% *training* 20% *testing*,

ISSN: 1829-667X / E-ISSN: 2460-9021
DOI:10.31315/telematika.v22i2.14637

sebelum diproses kedalam *feature extraction*, dengan menggunakan *FastText* untuk *word embedding*. Berikutnya proses klasifikasi teks pertanyaan dengan model *Bidirectional Long Short-Term Memory* (BiLSTM) dan dilanjutkan evaluasi dengan *confusion matrix*. Tahap terakhir, adalah integrasi *chatbot* ke dalam *bot Telegram*, kemudian melakukan pengujian *user* terhadap *chatbot* serta mengukur tingkat kepuasan dengan metode *Customer Satisfaction Score* (CSAT).

Hasil: Akurasi model klasifikasi mengahasilkan nilai sebesar 96.05% dan pengujian *user* dengan menerapkan metode *Customer Satisfaction Score* (CSAT) memberikan tingkat kepuasan rata-rata pada *range* 4 (Puas) dan 5 (Sangat Puas) dengan hasil 88.86% berdasarkan *point* berikut 'Kepuasan jawaban yang diberikan', 'Memahami pertanyaan dan jawaban mudah dipahami', dan 'Kinerja sesuai harapan'.

Keaslian/*state of the art*: Penelitian mengenai klasifikasi teks pertanyaan pada *chatbot* dengan pendekatan *Natural Language Processing* (NLP) dan model *Bidirectional Long Short-Term Memory* (BiLSTM) untuk menangani permasalahan tanya jawab layanan UKT (Uang Kuliah Tunggal) maupun biaya kuliah Fakultas Teknik Universitas Tidar belum pernah dilakukan sebelumnya.

1. Pendahuluan

Chatbot merupakan salah satu contoh bentuk pengaplikasian dari *Artificial Intelligence* yang mampu melakukan interaksi antara manusia dengan mesin melalui percakapan teks atau suara [1]. Penggunaan *chatbot* sendiri telah banyak digunakan seperti contoh di perusahaan atau *e-Commerce* dengan tujuan membantu menjawab pertanyaan terkait informasi yang diinginkan oleh pelanggan [2]. Penerapan chatbot dalam sistem layanan akademik akan mempermudah pengguna yaitu mahasiswa, dalam mendapatkan informasi secara mudah. Adapun cabang ilmu kecerdasan buatan dalam membangun *chatbot* agar mampu berinteraksi secara natural atau alami yaitu *Natural Language Processing* (NLP). Penggunaan *Natural Language Processing* (NLP) bertujuan agar komputer dapat memahami, menganalisis serta menguraikan bahasa manusia sesuai dengan konteksnya [3]. NLP, *Machine Learning*, maupun *Deep Learning* dalam konteks klasifikasi teks dapat digunakan untuk mengkategorikan, mengklasifikasikan, serta memahami teks, *intent* atau maksud yang disampaikan dari pengguna ke dalam kategori tertentu, sehingga respon yang diberikan lebih akurat [4]. Terdapat beberapa penelitian sebelumnya terkait *chatbot* yang menerapkan pendekatan NLP, *Machine Learning* maupun *Deep Learning* seperti K-NN, ANN, *Naïve Bayes Classifier*, SVM, LSTM, dan lain-lain [5], [6], [7], [8], [9]. Tentu dalam penelitian sebelumnya, masih terdapat beberapa saran yang harus dikembangkan, seperti keterkaitan penggunaan kata atau bahasa tidak baku yang tidak ada dalam *dataset*, penambahan *dataset*, atau meningkatkan akurasi dengan model klasifikiasi lainnya [5], [6], [7]. *Natural Language Processing* (NLP) merupakan salah satu cabang dari

ISSN: 1829-667X / E-ISSN: 2460-9021
DOI:10.31315/telematika.v22i2.14637

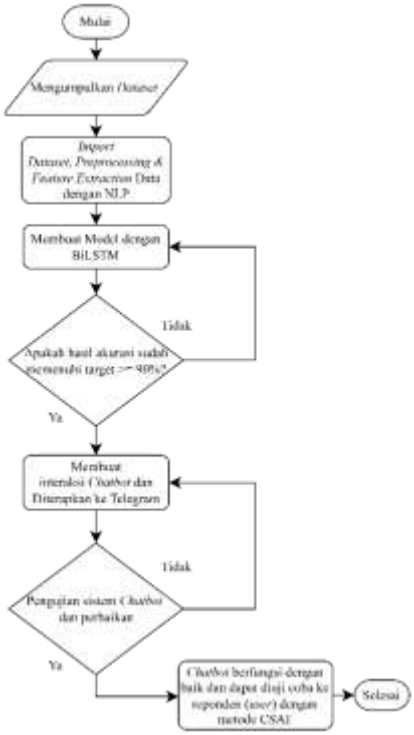
teknologi kecerdasan buatan (*Artificial Intelligence*) yang berfokus untuk memproses bahasa alami ataupun bahasa yang umumnya digunakan manusia untuk berkomunikasi dan berinteraksi dengan komputer [10], [11]. *Bidirectional Long Short-Term Memory* (BiLSTM) yang mana algoritma atau arsitektur ini berisi dua jaringan LSTM, umumnya digunakan untuk menangani permasalahan model sequence [12]. *Bidirectional Long Short-Term Memory* (BiLSTM) memanfaatkan konteks sebelum dan sesudah dari data dua arah kemudian memprosesnya dengan *hidden layer* terpisah [13].

Berdasarkan penelitian sebelumnya yang berhubungan tentang *chatbot*, penelitian ini akan merancang *chatbot* dengan menerapkan metode *Natural Language Processing* (NLP), dan algoritma *Deep Learning* yaitu *Bidirectional Long Short-Term Memory* (BiLSTM). Penelitian yang dilakukan untuk mengetahui hasil performa akurasi kinerja dari metode yang digunakan terkait tugas klasifikasi teks pertanyaan. Selain itu penelitian terkait *chatbot* ini, diharapkan mampu menggantikan peran manusia (admin penyedia layanan informasi) dalam merespon pertanyaan mahasiswa tanpa perlu *visiting* atau *chat contact person* pihak TU FT, dan mengurangi masalah terkait *human delay* maupun *office hour* dalam memberikan balasan. *Chatbot* yang dirancang juga membantu ketika terdapat pertanyaan dari mahasiswa yang memiliki konteks serupa dalam waktu bersamaan. Berikutnya *chatbot* yang dirancang berfokus untuk membantu proses tanya jawab dengan mahasiswa terkait permasalahan informasi layanan UKT (Uang Kuliah Tunggal) atau biaya kuliah khususnya di Fakultas Teknik Universitas Tidar, dan diterapkan ke Telegram (*Bot Telegram*) agar mempermudah pengguna dalam mengakses layanan serta memberikan keleluasaan untuk bertanya. Adapun penerapan dari metode tersebut diharapkan *chatbot* mampu dalam memahami, mengklasifikasikan teks atau maksud percakapan dari pengguna lebih baik.

2. Metode dan Perancangan

Tahap selanjutnya yaitu metode penelitian yang mencakup beberapa tahapan seperti pengumpulan data untuk kebutuhan *dataset*, *preprocessing dataset* dengan pendekatan *Natural Language Processing* (NLP), *split dataset* untuk *training* dan *testing*, *feature extraction* dengan *FastText*, kemudian model untuk klasifikasi dengan *Bidirectional Long Short-Term Memory* (BiLSTM), terakhir evaluasi menggunakan *Confusion Matrix* dan pembuatan sistem interaksi *chatbot* untuk dilakukan pengujian tingkat kepuasan pengguna dengan metode *Customer Satisfaction Score* (CSAT). Proses rancangan *chatbot* dengan pendekatan *Natural Language Processing* (NLP) dan *Bidirectional Long Short-Term Memory* (BiLSTM) ditunjukkan pada **Gambar 1**. berikut ini.

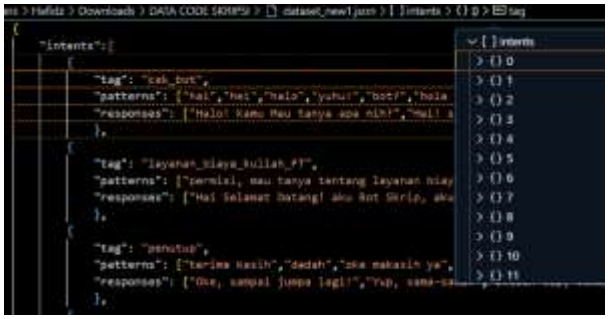
ISSN: 1829-667X / E-ISSN: 2460-9021
DOI:10.31315/telematika.v22i2.14637



Gambar 1. Diagram Alir Proses Pembuatan *Chatbot*

2.1. Pengumpulan *Dataset*

Pengambilan *dataset* ini dikumpulkan melalui berbagai sumber, seperti wawancara dengan pihak TU FT, membuat kuesioner untuk menampung pertanyaan terkait seputar UKT maupun biaya kuliah FT Untidar, mengakses *website* (https://um.untidar.ac.id/biaya perkuliahan/ dan https://s.id/untidarukt-ft) untuk mendapatkan informasi tambahan. Total keseluruhan dari pengumpulan *dataset* yaitu 1.140 pola pertanyaan (*patterns*), 190 jawaban (*responses*) dan 38 *tag*, yang mana masing-masing dari *tag* tersebut diwakili oleh 30 pertanyaan dan 5 jawaban. *Dataset* yang terkumpul tersebut merupakan data bertipe *string* dan dimasukkan ke dalam *file* dengan format JSON. Susunan *dataset* dalam format JSON terdiri dari '*intent*' dengan '*tag*', '*patterns*', dan '*responses*', untuk tampilan *dataset* yang terkumpul ditunjukkan pada **Gambar 2**. sebagai berikut ini.



Gambar 2. Tampilan *Dataset* Format JSON Pada *Visual Studio Code*

Sebagai contoh dari salah satu *dataset* untuk rancangan sistem *chatbot* dapat dilihat dalam **Tabel 1**. berikut di bawah ini, contoh yang digunakan berasal dari *tag* "info_angsuran_ukt".

ISSN: 1829-667X / E-ISSN: 2460-9021
DOI:10.31315/telematika.v22i2.14637

Tabel 1. Salah Satu Isi *Tag Dataset*

Salah Satu Isi *Tag Dataset*
"tag": "info_angsuran_ukt", "patterns": [.....,"Berikan informasi mengenai UKT yang dapat diangsur pada fakultas teknik untidar!","saya perlu mengangsur UKT, apa saja yang perlu diperhatikan?","minta informasi tentang bantuan Angsuran UKT FT universitas tidar!","saya mau petunjuk terkait cicilan ukt di fakultas teknik Universitas Tidar!","Hal yang perlu diketahui sebelum mengajukan angsuran UKT di Fakultas Teknik untidar?", ……], "responses": ["Angsuran UKT yang tersedia di Fakultas Teknik Universitas Tidar merupakan salah satu bantuan cicilan UKT bagi mahasiswa dari keluarga kurang mampu. Adapun ketentuan pembayaran cicilan yang diperbolehkan sebanyak 2 (dua) kali angsuran/cicilan. Ketentuan pembayaran yang dilakukan yaitu 50% pembayaran awal (sebelum memasuki semester baru) dari besaran UKT yang diterima dan 50% sisanya dibayarkan paling lambat 1 minggu sebelum UTS (Ulangan Tengah Semester).\n\nPerlu dicatat, bahwa permohonan Angsuran UKT yang diajukan mahasiswa hanya berlaku 1 semester saja. Kemudian apabila di semester selanjutnya masih belum bisa melakukan pembayaran 100% maka perlu mengajukan kembali. Tidak seperti bantuan Banding UKT, Angsuran UKT bisa diajukan bagi Calon Mahasiswa Baru (CAMABA) maupun Mahasiswa Lama, namun yang membedakannya hanya di waktu, syarat berkas, dan pengiriman berkas secara online. Untuk informasi lebih lanjut kamu bisa akses di https://s.id/untidarukt-ft",.....]

2.2. *Preprocessing* atau *Text Normalization*

Dataset yang sudah terkumpul kemudian disimpan ke dalam *Google Drive* agar dapat di *mount* ketika proses koding program di *Google Colaboratory*. Sebelum diolah lebih lanjut, *dataset* perlu dilakukan *preprocessing* atau *text normalization* dengan tujuan agar mempermudah kinerja komputer, hasil dari algoritma proses *training model* berjalan lancar, dan mendapatkan akurasi sesuai yang diharapkan [14]. *Preprocessing* atau *Text Normalization dataset* yang dilakukan menerapkan pendekatan *Natural Language Processing* (NLP) dengan menggunakan beberapa *library python* untuk detail proses sebagai berikut [10]. Sebagai contoh perbedaan/perubahan yang terjadi, digunakan salah satu teks yang terdapat dalam *dataset* ("list lengkap nominal setiap golongan ukt S1 teknik elektro (TE)!") sebelum dilakukan *preprocessing*.

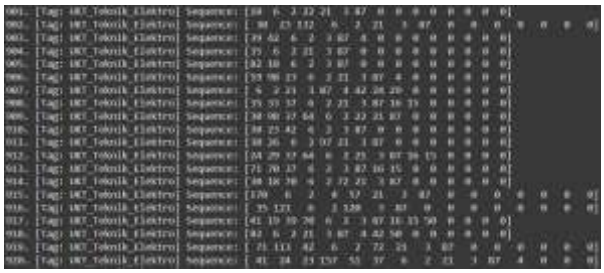
Case folding atau *lower casing* yaitu proses mengubah data teks yang memiliki huruf besar menjadi huruf kecil dengan tujuan agar data menjadi lebih seragam [15]. Melakukan pembersihan data dari spasi yang berlebihan, untuk proses ini digunakan *library RegEx*. Selanjutnya penghapusan tanda baca atau *punctuation removal* dengan *library RegEx* di mana menghapus semua tanda baca kecuali garis miring (/). *Stop word remove* dengan menggunakan *library Sastrawi*, dengan *library* yang sama dilakukan juga proses *stemming* Bahasa Indonesia [16]. **Gambar 3**. merupakan *code* program berserta dengan *output* yang dihasilkan.

ISSN: 1829-667X / E-ISSN: 2460-9021
DOI:10.31315/telematika.v22i2.14637

Gambar 3. *Code* Program *Preprocessing* dan *Output*

Proses terakhir dari *preprocessing dataset* yaitu tokenisasi (*Tokenizer*) dan *padding* dengan *library Tensorflow*. Tujuan dari tokenisasi untuk memecah kalimat menjadi kata, lalu setiap kata diberikan nomor urutan atau indeks kata berdasarkan frekuensinya [17]. Masuk tahap *padding* untuk menyamakan panjang kalimat agar menjadi sama rata ditandai dengan angka 0 atau 'nol'. **Gambar 4**. adalah *code* program pada proses tokenisasi dan *padding*, kemudian untuk hasil *output* dari proses tersebut dilampirkan pada **Gambar 5**.

Gambar 4. *Code* Program Tokenisasi dan *Padding*



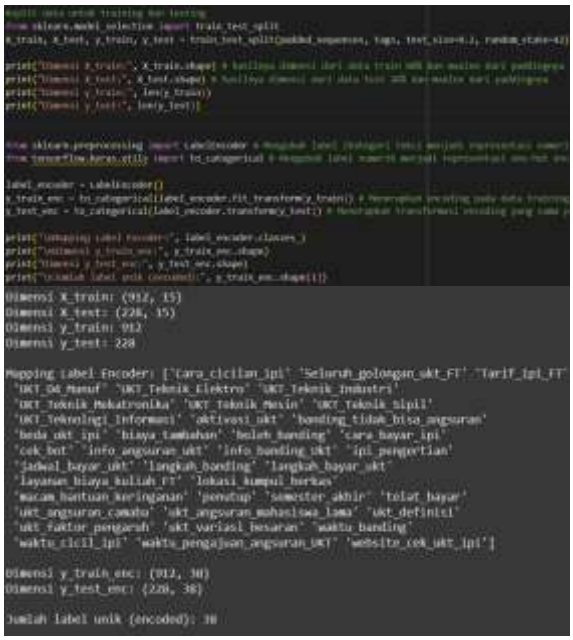
Gambar 5. Hasil *Output* Proses Tokenisasi dan *Padding*

2.3. *Split Data* dan *Label Encoder*

Dataset kemudian dilakukan proses *split data* dan *label encoder* sebelum masuk ke tahap *feature extraction Natural Language Processing* (NLP) dengan menggunakan *library Scikit-learn*. *Split data* dilakukan dengan tujuan untuk membagi *dataset* menjadi dua bagian, yaitu *training data* yang mana untuk melatih model lalu *testing data* untuk mengevaluasi performa

ISSN: 1829-667X / E-ISSN: 2460-9021
DOI:10.31315/telematika.v22i2.14637

model [18]. Perbandingan yang digunakan yaitu 80% untuk *training data* dan 20% *testing data*, dari hasil pembagian tersebut diperoleh 912 data untuk *training* dan 228 untuk *testing* dari jumlah total *dataset* yang terkumpul. Tahap yang dilakukan setelah *split data* yaitu proses *label encoder* yang bertujuan untuk konversi label yang sebelumnya berbentuk *string*/kategori teks menjadi angka atau *label numeric*, dan menghasilkan 38 label. Proses ini melakukan konversi menjadi *label numeric* kemudian diubah menjadi format *one-hot encoding* atau representasi biner agar membantu dalam klasifikasi multi kelas, **Gambar 6**. diberikan *code* program dan *output* dari proses berikut.



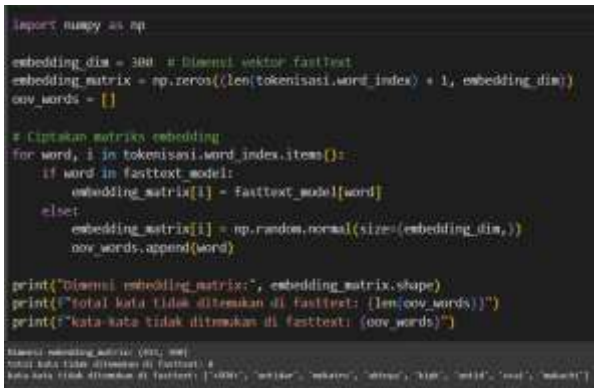
Gambar 6. *Code* Proses *Split Data* dan *Label Encoder* Data, serta *Output*

2.4. *Feature Extraction*

Proses selanjutnya masuk pada tahap *feature extraction* setelah *split dataset* dan *label encoder* dengan menggunakan *FastText* Bahasa Indonesia digunakan untuk membangun *vector embedding*. Penggunaan *FastText pre-trained* bertujuan untuk mendapatkan *vector embedding* yang sebelumnya sudah dilatih dan membantu dalam menangani kata-kata yang tidak ada dalam *Out of Vocabulary* (OOV) [19], [20]. *Website FastText* (https://fasttext.cc/) menyediakan model *word vector* untuk Bahasa Indonesia. **Gambar 7**. berikut ini menunjukkan proses untuk memuat model *FastText* dengan nama *file* "cc.id.300.vec.gz" yang sudah diunduh pada *website FastText* dengan melibatkan *library Gensim* dan *Numpy*. Langkah berikutnya memetakan kata-kata ke representasi *embedding* agar dapat digunakan dalam proses model *deep learning Bidirectional Long Short-Term Memory* (BiLSTM) yang ditampilkan pada **Gambar 8**. Proses kerja dari *code* ini pada setiap kata yang diproses yaitu, jika kata yang direpresentasikan ada di model *FastText*, maka vektornya akan ditambahkan ke dalam variabel "*embedding_matrix*", namun jika kata tidak ada atau disebut *Out of Vocabulary* maka akan dibuat vektor acak dan kata dicatat dalam daftar variabel "*oov_words*".

ISSN: 1829-667X / E-ISSN: 2460-9021
DOI:10.31315/telematika.v22i2.14637

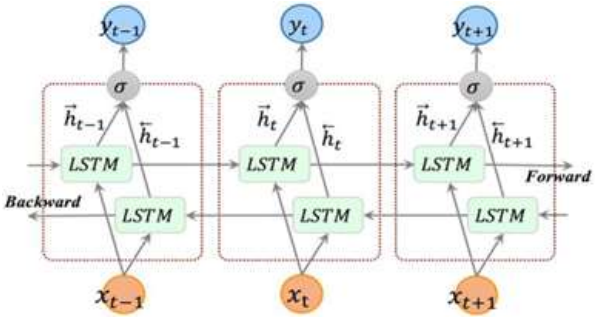
Gambar 7. *Code* Program Memuat Model *FastText* Bahasa Indonesia



Gambar 8. *Code* Program *Embedding Matrix* dan *Output*

Hasil dari *embedding matrix* berupa dimensi dengan ukuran (415, 300), kemudian untuk total kata yang tidak ditemukan dalam model *FastText* yaitu ('untidar', 'mekatro', 'uktnya', 'kipk', 'untid', 'ocai', 'makachi'). Kata yang tidak ada dalam model *FastText* terjadi karena kata tersebut adalah singkatan dari nama Universitas, program bantuan, maupun nama jurusan.

2.5. Model Klasifikasi BiLSTM



Gambar 9. Arsitektur *Bidirectional Long Short-Term Memory*

Tahap berikutnya membuat model untuk tugas klasifikasi teks/*intent* menggunakan algoritma *Bidirectional Long Short-Term Memory* (BiLSTM). *Code* program ini melibatkan *library Tensorflow Keras* untuk membuat model bertipe *sequential* [8]. **Gambar 9**. menunjukkan bentuk dari arsitektur *Bidirectional Long Short-Term Memory* (BiLSTM) di mana terdiri dari algoritma yang memiliki dua jaringan *Long Short-Term Memory* (LSTM), jaringan pertama untuk memproses urutan pada masukan (*input*) data ke arah depan/maju atau disebut *forward* dan lapisan kedua memiliki fungsi untuk memproses masukan data dari arah mundur/sebaliknya yang disebut *backward* [13]. Lalu untuk *output* dari kedua jaringan tersebut digabungkan pada setiap urutan waktu, dengan adanya penerapan dua jaringan *Long Short-Term Memory* (LSTM) ini model dapat mempelajari atau melihat informasi kontekstual dari dua arah, yaitu dari

ISSN: 1829-667X / E-ISSN: 2460-9021
DOI:10.31315/telematika.v22i2.14637

'Sebelum' maupun 'Sesudah' dari setiap *sequence input* [12], [13]. Karena berasal dari dua jaringan *Long Short-Term Memory* (LSTM), maka untuk proses persamaan rumus perhitungan dari *Bidirectional Long Short-Term Memory* (BiLSTM) tidak jauh berbeda. Berikut bentuk persamaan rumus dari *Long Short-Term Memory* (LSTM) dan *Bidirectional Long Short-Term Memory* (BiLSTM) dimulai dari **persamaan 1**. hingga **persamaan 10**.

a. *Forget Gate* yang berisi informasi tentang setiap data masukan (*input*) yang akan diproses dan data mana yang akan disimpan atau dihapus pada unit penyimpanan yang dipilih. Berikut persamaan perhitungan pada proses *forget gate*.

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

b. *Input Gate* berisi informasi dengan menggunakan **persamaan 2**. dan **persamaan 3**. Terdapat dua gerbang yang diterapkan, yaitu menentukan beberapa informasi nilai yang diperbarui sesuai *cell state* menggunakan fungsi aktivasi *sigmoid*. Selanjutnya membentuk kandidat vektor yang baru dengan fungsi aktivasi *tanh*, kemudian disimpan pada *memory cell*.

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

$$\tilde{C}_t = tanh(W_c \cdot [h_{t-1}, x_t] + b_c) \quad (3)$$

c. *Cell Gate* proses ini akan memperbarmultiui nilai dari *cell state* lama atau C_{t-1} dengan nilai *cell state* yang baru C_t. Hasil nilai tersebut didapatkan dari penggabungan nilai pada *forget gate* dan *input gate* dengan menggunakan **persamaan 4**. di bawah ini.

$$C_t = f_t * C_{t-1} + i_t * \tilde{C}_t \quad (4)$$

d. *Output Gate* memiliki dua gerbang yang diterapkan, yaitu memproses fungsi *sigmoid* untuk menghasilkan keluaran atau *output* pada *hidden state* serta mengatur *cell state* pada *tanh*. Setelah mampu menghasilkan nilai *output sigmoid* dan nilai *output tanh*, maka kedua gerbang tersebut dikalikan sehingga menghasilkan nilai yang menjadi *output*. Perhitungan ini dilakukan dengan **persamaan 5**. dan **persamaan 6**. berikut.

$$o_t = \sigma(W_o[h_{t-1}, x_t] + b_o) \quad (5)$$

$$h_t = o_t * tanh(C_t) \quad (6)$$

Mengetahui persamaan rumus yang terdapat pada *Long Short-Term Memory* (LSTM), maka untuk persamaan *Bidirectional Long Short-Term Memory* (BiLSTM) sebagai berikut. **Persamaan 7**. untuk *forward layer* merepresentasikan konteks sebelumnya, dan **persamaan 8**. untuk *backward layer* merepresentasikan konteks setelahnya.

$$\overrightarrow{h_t} = LSTM(x_t, h_{t-1}) \quad (7)$$

$$\overleftarrow{h_t} = LSTM(x_t, h_{t+1}) \quad (8)$$

Hasil *output* dapat direpresentasikan pada **persamaan 9**. atau **persamaan 10**. Di mana kedua persamaan ini adalah hasil pertambahan atau total dari proses *forward layer* dan *backward layer*.

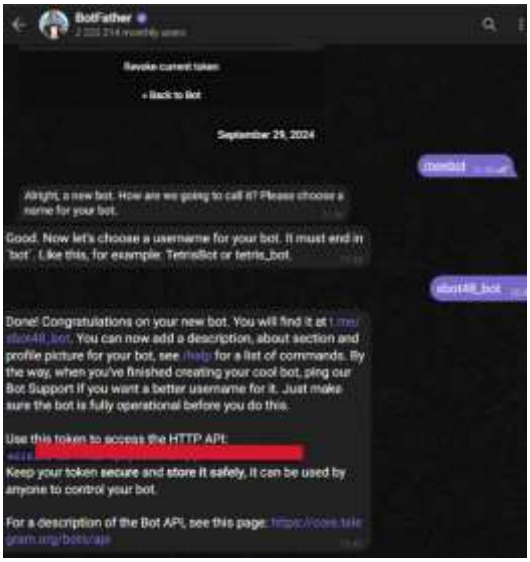
$$h_t = [\overrightarrow{h_t}, \overleftarrow{h_t}] \quad (9)$$

$$y_t = W_{\vec{h}y}\vec{h}_t + W_{\overleftarrow{h}y}\overleftarrow{h}_t \quad (10)$$

ISSN: 1829-667X / E-ISSN: 2460-9021
DOI:10.31315/telematika.v22i2.14637

2.6. Rancang Interaksi *Chatbot*

Merancang interaksi *chatbot* setelah proses *preprocessing*, *feature extraction*, dan model dengan menggunakan *Bidirectional Long Short-Term Memory* (BiLSTM), fungsi yang dirancang yaitu '*predict_intent*' dan '*chatbot_response*' sekaligus integrasi ke dalam *bot Telegram*. *Code* program untuk fungsi '*predict_intent*' maupun '*chatbot_response*' digunakan *threshold confidence* sebesar 0.7. Jika *input* pertanyaan tidak mencapai nilai *threshold confidence* tersebut maka respons yang dihasilkan berupa kalimat penolakan. *Chatbot* dapat terintegrasi dengan baik ke dalam *bot*, hal pertama yang harus dilakukan yaitu membuat *bot* dengan bantuan *BotFather* perintah "*/newbot*", setelah berhasil maka *BotFather* akan merespon dan memberikan token API untuk digunakan ke dalam *code* program [21], [22], [23]. **Gambar 10**. menunjukkan proses membuat *bot* pada *BotFather*, dan **Gambar 13**. merupakan hasil tampilan awal setelah *code* program berhasil diintegrasikan ke dalam *bot Telegram*.

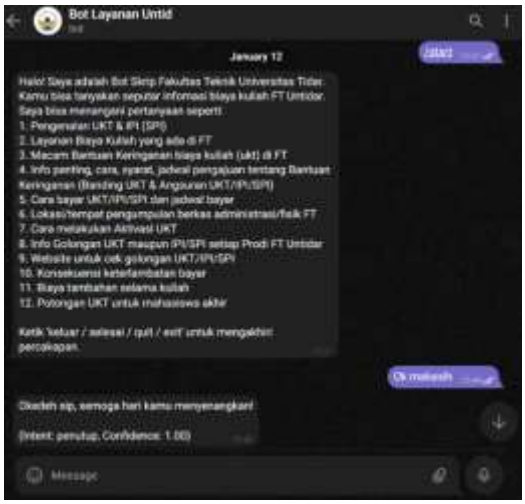


Gambar 10. Proses Pembuatan *Bot* di *BotFather* dan Hasil Token API

Gambar 11. *Code* Program Fungsi '*predict_intent*'

ISSN: 1829-667X / E-ISSN: 2460-9021
DOI:10.31315/telematika.v22i2.14637

Gambar 12. *Code* Program Fungsi '*chatbot_response*'



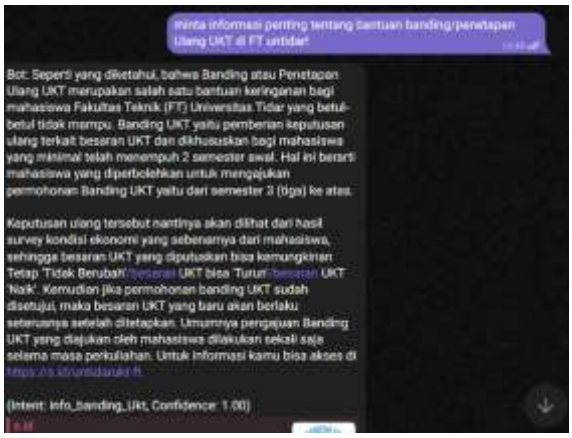
Gambar 13. Tampilan Awal Saat Memulai Interaksi dengan *Bot Telegram*

2.7. Penerapan *Chatbot* ke Aplikasi Telegram

Pengujian terhadap rancangan *chatbot* yang berhasil diintegrasikan ke *bot Telegram*, hal ini dilakukan untuk mengetahui apakah *bot* bekerja dengan baik dan mampu menjawab pertanyaan *user* dengan benar. Uji coba pertama yaitu mengetikkan kalimat sapaan atau pertanyaan untuk tes *bot*, dengan hasil sesuai konteks seperti yang ditunjukkan pada **Gambar 14**. Uji coba berikutnya dengan mengetikkan pertanyaan yang memiliki konteks 'info penting tentang banding UKT FT', menghasilkan respon atau jawaban terkait 'info penting banding' seperti yang ditampilkan **Gambar 15**.

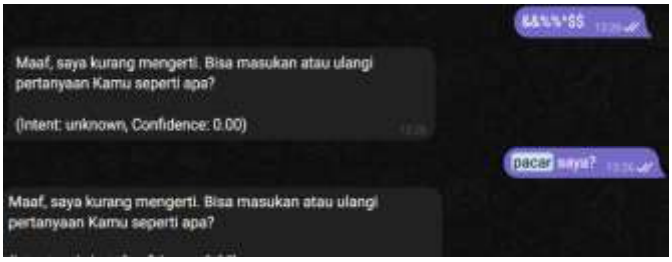
Gambar 14. Tes *Bot* dengan Pertanyaan Sapaan

ISSN: 1829-667X / E-ISSN: 2460-9021
DOI:10.31315/telematika.v22i2.14637



Gambar 15. Pertanyaan tentang Info Penting Banding

Uji coba selanjutnya dengan mengetikkan karakter atau pertanyaan yang tidak memiliki konteks sama sekali, untuk contoh pertanyaan yang diuji digunakan karakter berikut "&&%%^$$" dan pertanyaan "pacar saya?". Hasil dari kedua input tersebut adalah kalimat penolakan "Maaf, saya kurang mengerti. Bisa masukan atau ulangi pertanyaan Kamu seperti apa?", untuk hasil ditampilkan pada **Gambar 16**. berikut ini. Hasil dari gambar tersebut sesuai dengan harapan yang diinginkan karena dari kedua pertanyaan/masukan tersebut tidak termasuk dalam konteks, sehingga hasil yang dikeluarkan adalah kalimat penolakan.



Gambar 16. Uji Coba dengan *Input* Karakter dan Pertanyaan Tidak Valid

3. Hasil dan Pembahasan

Hasil dan pembahasan pada penelitian yang dilakukan terdiri dari hasil akurasi model *Bidirectional Long Short-Term Memory* (BiLSTM), evaluasi *confusion matrix*, dan pengukuran tingkat kepuasan *chatbot* terhadap *user* dengan metode *Customer Satisfaction Score* (CSAT) setelah melakukan interaksi dengan *bot user* mengisi kuesioner yang sudah disediakan.

3.1. Akurasi Model Klasifikasi BiLSTM

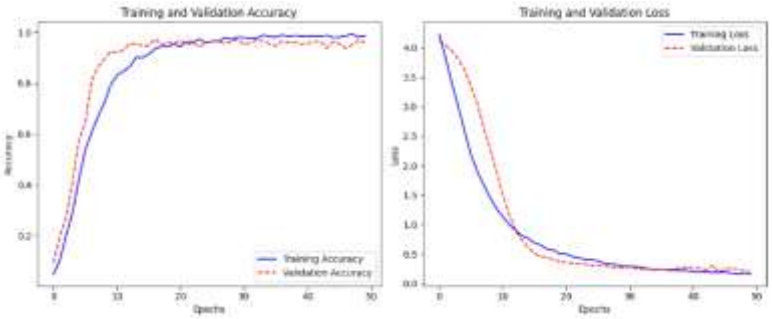
Hasil akurasi dari model BiLSTM tersebut diperoleh sebesar 98.15% untuk *training* dan 96.05% untuk *testing*/*validation*. **Gambar 17**. dan **Gambar 18**. berikut ini adalah hasil grafik akurasi dan *loss* model *Bidirectional Long Short-Term Memory* (BiLSTM), dengan menggunakan *epochs* dengan jumlah 50 dan *batch size* 32. Hasil akurasi tersebut dapat dibuktikan dengan rumus akurasi pada **persamaan 11**. dan hasil perhitungan akurasi model sebagai berikut:

$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (11)$$

ISSN: 1829-667X / E-ISSN: 2460-9021
DOI:10.31315/telematika.v22i2.14637

Hasil perhitungan akurasi model BiLSTM dengan TP = 219, FP = 7, FN = 2, dan nilai TN = 228-(219+7+2) = 0, adalah sebagai berikut:

$$Accuracy = \frac{219+0}{219+0+7+2} = 0.9605 \text{ atau } 96.05\%$$



Gambar 17. Visualisasi atau Grafik *Training* dan *Validation Model*

Gambar 18. Hasil Akurasi Model Pada Data Uji (*Validation*)

3.2. Evaluasi *Confusion Matrix*

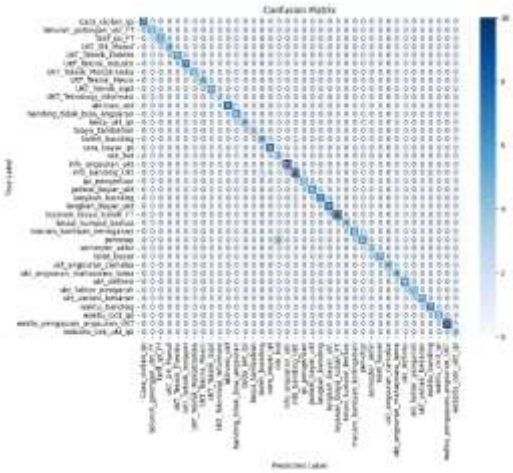
Evaluasi *confusion matrix* pada penelitian ini digunakan untuk menghitung kinerja model, di mana metode ini akan mencari nilai dari persentase diantaranya yaitu *accuracy*, *precision*, *recall*, dan *f1-score* [24]. **Gambar 19**. adalah hasil evaluasi *confusion matrix* dari program *chatbot* yang menerapkan *Natural Language Processing* (NLP) (*preprocessing* dan *feature extraction*) dan *Bidirectional Long Short-Term Memory* (BiLSTM). **Gambar 20**. Menunjukkan hasil *classification report* antara *true label* dengan *predicted label* dari data uji percakapan *chatbot* berjumlah 38 *tag*/*label*. Sebelumnya telah dibahas untuk hasil akurasi, selanjutnya pada **persamaan 12**. sampai **persamaan 14**. merupakan perhitungan untuk *precision*, *recall*, dan *f1-score*.

$$Precision = \frac{TP}{TP+FP} \tag{12}$$

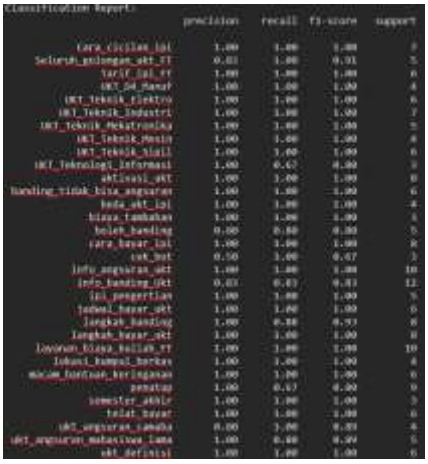
$$Recall = \frac{TP}{TP+FN} \tag{13}$$

$$F1 - Score = \frac{2*Precision*Recall}{Precision+Recall} \tag{14}$$

ISSN: 1829-667X / E-ISSN: 2460-9021
DOI:10.31315/telematika.v22i2.14637



Gambar 19. Hasil *Confusion Matrix*



	precision	recall	f1-score	support
cara_cicilan_ipi	1.00	1.00	1.00	8
Seberapa_puasnya_ukt_IT	0.83	1.00	0.91	5
tarif_ipi_fr	1.00	1.00	1.00	6
UKT_D4_Manaf	1.00	1.00	1.00	4
UKT_Teknik_Elektro	1.00	1.00	1.00	6
UKT_Teknik_Industri	1.00	1.00	1.00	7
UKT_Teknik_Mekatronika	1.00	1.00	1.00	5
UKT_Teknik_Mesin	1.00	1.00	1.00	6
UKT_Teknik_Sipil	1.00	1.00	1.00	6
UKT_Teknologi_Informasi	1.00	0.67	0.80	3
aktivasi_ukt	1.00	1.00	1.00	8
banding_tidak_bisa_angsuran	1.00	1.00	1.00	6
beda_ukt_ipi	1.00	1.00	1.00	4
biaya_tambahan	1.00	1.00	1.00	4
boleh_banding	0.00	0.00	0.00	5
cara_bayar_ipi	1.00	1.00	1.00	8
cek_ipi	0.50	1.00	0.67	3
info_angsuran_ukt	1.00	1.00	1.00	10
info_banding_UKt	0.83	0.83	0.83	12
ipi_pengertian	1.00	1.00	1.00	5
jadwal_bayar_ukt	1.00	1.00	1.00	6
langkah_banding	1.00	0.88	0.93	8
langkah_bayar_ukt	1.00	1.00	1.00	8
layanan_biaya_kuliah_IT	1.00	1.00	1.00	10
lokasi_kampus_berikan	1.00	1.00	1.00	4
macam_bantuan_keringanan	1.00	1.00	1.00	6
penutup	1.00	0.67	0.80	9
semester_akhir	1.00	1.00	1.00	3
telat_bayar	1.00	1.00	1.00	6
ukt_angsuran_cawaba	0.80	1.00	0.89	4
ukt_angsuran_mahasiswa_lama	1.00	0.80	0.89	5
ukt_definisi	1.00	1.00	1.00	6

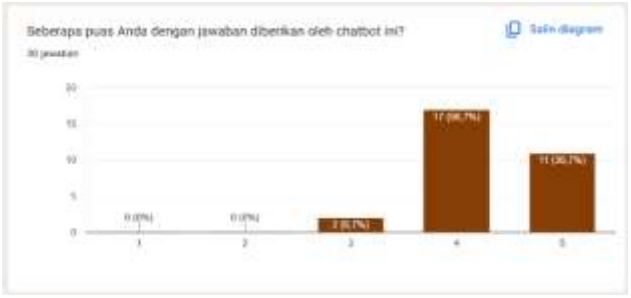
Gambar 20. Hasil *Classification Report*

3.3. Tingkat Kepuasan *Chatbot* dengan Metode CSAT

Pengujian terhadap *user*, di mana beberapa *user* akan mencoba menggunakan atau berinteraksi dengan *bot Telegram*. Setelah selesai melakukan interaksi dengan *bot*, *user* akan diarahkan untuk mengisi kuesioner yang telah disediakan. Pengujian *user* menggunakan responden yang berasal dari mahasiswa Fakultas Teknik Universitas Tidar sebanyak 30 responden. Kuesioner ini menerapkan metode *Customer Satisfaction Score* (CSAT), di mana fokus dari metode ini adalah untuk mengukur "Tingkat Kepuasan Pengguna" setelah berinteraksi dengan *bot* [25]. Terdapat 3 (tiga) kategori pertanyaan yang merujuk ke dalam metode CSAT, pertanyaan tersebut yaitu "Seberapa puas Anda dengan jawaban yang diberikan oleh *chatbot* ini?", "Apakah *chatbot* ini cukup baik dalam memahami pertanyaan dan memberikan jawaban yang mudah dipahami?", dan "Apakah kinerja dari *chatbot* ini cukup memenuhi harapan Anda?". Hasil dari pengisian kuesioner tersebut ditunjukkan pada **Gambar 21**. sampai **Gambar 23**. dengan hasil tertinggi berada di *range score* 4 (Puas) dan diikuti oleh 5 (Sangat Puas). **Persamaan 15**. merupakan rumus perhitungan metode CSAT.

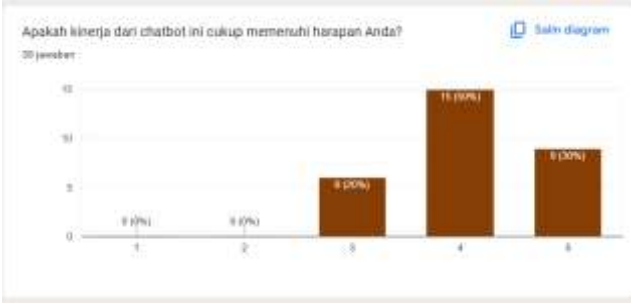
$$CSAT\ Score = \frac{jumlah\ respon\ positif\ (Puas\ \&\ Sangat\ Puas)}{jumlah\ atau\ total\ responden} * 100\% \quad (15)$$

ISSN: 1829-667X / E-ISSN: 2460-9021
DOI:10.31315/telematika.v22i2.14637



Gambar 21. Diagram 'Kepuasan jawaban yang diberikan'

Gambar 22. Diagram 'Memahami pertanyaan dan jawaban mudah dipahami'



Gambar 23. Diagram 'Kinerja sesuai harapan'

Berdasarkan dari **persamaan 15**. tersebut perhitungan dari kuesioner dengan metode *Customer Satisfaction Score* (CSAT) pada **Gambar 21**. sampai **Gambar 23**. adalah sebagai berikut:

1. Perhitungan dari point 'Kepuasan jawaban yang diberikan'

   $CSAT\ Score = \frac{17+11}{30} * 100\% = 93.3\%$

2. Perhitungan dari point 'Memahami pertanyaan dan jawaban mudah dipahami'

   $CSAT\ Score = \frac{16+12}{30} * 100\% = 93.3\%$

3. Perhitungan dari point 'Kinerja sesuai harapan'

   $CSAT\ Score = \frac{15+9}{30} * 100\% = 80\%$

4. Jika dari ketiga point tersebut dihitung nilai rata-rata, maka dihasilkan:

ISSN: 1829-667X / E-ISSN: 2460-9021
DOI:10.31315/telematika.v22i2.14637

$$Rata - rata = \frac{93.3+93.3+80}{3} * 100\% = 88.86\%$$

4. Kesimpulan dan Saran

Berdasarkan penelitian yang telah dilakukan terkait klasifikasi teks pada *chatbot* dengan tema UKT maupun biaya kuliah yang terdapat di Fakultas Teknik Universitas Tidar. Sistem *chatbot* yang di rancang mampu mengklasifikasikan teks pertanyaan yang dimasukan oleh *user* dengan benar dan tepat terkait permasalahan UKT maupun biaya kuliah di Fakultas Teknik Universitas Tidar. Penelitian menggunakan pendekatan *Natural Language Processing* (NLP) dan *Deep Learning Bidirectional Long Short-Term Memory* (BiLSTM) menghasilkan akurasi sebesar 96.05%. Di mana program *chatbot*, *preprocessing* yang digunakan yaitu *lower casing*, *punctuation removal*, tokenisasi, *padding*, *stop word removal* dan *stemming* dengan Sastrawi. *Feature extraction* berupa *word embedding FastText*. Hasil pengujian *user* dengan menerapkan metode *Customer Satisfaction Score* (CSAT) memberikan hasil jawaban responden dengan rata-rata 88.86% pada *range* 4 (Puas) dan 5 (Sangat Puas).

Saran yang bisa dikembangkan agar penelitian menjadi lebih baik seperti, menambahkan *dataset* yang lebih beragam dengan berbagai variasi tema selain yang digunakan dalam penelitian ini, dengan tujuan agar pertanyaan yang diajukan *user* memiliki cakupan yang lebih luas terkait informasi kuliah. Keterbatasan jumlah *sample* responden penelitian yang dilakukan menyebabkan hasil tingkat kepuasan pengguna tidak dapat mewakili secara menyeluruh, sehingga disarankan untuk menambahkan *sample* yang lebih banyak dan beragam. Menerapkan atau deploy *chatbot* ke dalam hosting agar bisa diakses secara 24 jam. Melakukan penelitian dengan model pembelajaran yang lebih luas terkait AI *chatbot* untuk menangani masalah bahasa gaul atau *slang* dalam konteks Bahasa Indonesia, sehingga mampu meningkatkan kinerja dari *chatbot* khususnya dalam menangani *slang*, maupun *typo*.

Daftar Pustaka

[1] B. Raharjo, "Deep Learning dengan Python," *Penerbit Yayasan Prima Agus Teknik*, pp. 1–131, 2022.

[2] C. Prianto, R. Andarsyah, and N. H. Harani, "Rancang Bangun Kamus Digital Berbasis Chatbot Menggunakan Pendekatan Pattern Matching," *JURNAL MEDIA INFORMATIKA BUDIDARMA*, vol. 6, no. 4, pp. 2327–2334, 2022.

[3] E. S. Eriana and A. Zein, "Artificial Intelligence (AI)," 2023.

[4] M. Mustaqim, A. Gunawan, Y. B. Pratama, and I. Zaliman, "Pengembangan Chatbot Layanan Publik Menggunakan Machine Learning Dan Natural Languange Processing," *Journal of Information Technology and society*, vol. 1, no. 1, pp. 1–4, 2023.

[5] K. A. Nugraha and D. Sebastian, "Chatbot Layanan Akademik Menggunakan K-Nearest Neighbor," *Jurnal Sains dan Informatika*, vol. 7, no. 1, pp. 11–19, 2021.

[6] M. F. Fadli, G. A. Buntoro, and F. Masykur, "Penerapan Algoritma Neural Network Pada Chatbot Pmb Universitas Muhammadiyah Ponorogo Berbasis Web," *JuSiTik: Jurnal Sistem dan Teknologi Informasi Komunikasi*, vol. 6, no. 1, pp. 13–22, 2022.

ISSN: 1829-667X / E-ISSN: 2460-9021
DOI:10.31315/telematika.v22i2.14637

[7] N. A. Purwitasari and M. Soleh, "Implementasi Algoritma Artificial Neural Network Dalam Pembuatan Chatbot Menggunakan Pendekatan Natural Language Parocessing," *JURNAL ILMU PENGETAHUAN DAN TEKNOLOGI (IPTEK)*, vol. 6, no. 1, 2022.

[8] K. Widianto, S. A. Wibowo, and U. Sunarya, "KLASIFIKASI TEKS BERBASIS LONG SHORT-TERM MEMORY UNTUK CHATBOT KONSELING GANGGUAN KECEMASAN SOSIAL," *Telkatika: Jurnal Telekomunikasi Elektro Komputasi & Informatika*, vol. 2, no. 2, 2023.

[9] R. P. Putra, A. H. Pratomo, and R. I. Perwira, "Text Message Classification using Multiclass Support Vector Machine on Information Service Chatbot in the Informatics Department UPN 'Veteran' Yogyakarta," *Telematika: Jurnal Informatika dan Teknologi Informasi*, vol. 19, no. 3, pp. 295–310, 2022.

[10] I. R. Hendrawan and E. Utami, *Natural Language Processing: Eksplorasi Sentimen Masyarakat dalam Evaluasi Produk Lokal Indonesia menggunakan Algoritma Bag of Words, TF-IDF, Word2Vec, dan Doc2Vec*. Penerbit Andi, 2023. [Online]. Available: https://books.google.co.id/books?id=caPgEAAAQBAJ

[11] A. Hikmah, F. Azmi, and R. A. Nugrahaeni, "Implementasi Natural Language Processing Pada Chatbot Untuk Layanan Akademik," *eProceedings of Engineering*, vol. 10, no. 1, 2023.

[12] 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, 2023.

[13] R. Siringoringo, J. Jamaluddin, R. Perangin-angin, E. J. G. Harianja, G. Lumbantoruan, and E. N. Purba, "Model Bidirectional Lstm Untuk Pemrosesan Sekuensial Data Teks Spam," *METHOMIKA: Jurnal Manajemen Informatika & Komputerisasi Akuntansi*, vol. 7, no. 2, pp. 265–271, 2023.

[14] F. Y. Fiddin, A. Komarudin, and M. Melina, "Chatbot Informasi Penerimaan Mahasiswa Baru Menggunakan Metode FastText dan LSTM," *Journal of Applied Computer Science and Technology*, vol. 5, no. 1, pp. 33–39, 2024.

[15] Y. S. H. Langgeng, E. I. Setiawan, S. Imron, and J. Santoso, "Long short-term memory-based chatbot for vocational registration information services," *Journal of Applied Data Sciences*, vol. 4, no. 4, pp. 414–430, 2023.

[16] S. J. Angelina, A. B. P. Negara, and H. Muhardi, "Analisis Pengaruh Penerapan Stopword Removal Pada Performa Klasifikasi Sentimen Tweet Bahasa Indonesia," *JUARA (Jurnal Aplikasi dan Riset Informatika)*, vol. 2, no. 1, pp. 165–173, 2023.

[17] D. O. Sihombing, "Implementasi Natural Language Processing (NLP) dan Algoritma Cosine Similarity dalam Penilaian Ujian Esai Otomatis," *Jurnal Sistem Komputer dan Informatika (JSON)*, vol. 4, no. 2, pp. 396–406, 2022.

[18] R. N. Aziza, T. S. Ardanti, E. Yosrita, and R. F. Ningrum, "Pembangunan Aplikasi dan Klasifikasi Pertanyaan Chatbot Informasi Akademik Menggunakan Metode Cosine Similarity dan Naïve Bayes," *KILAT*, vol. 12, no. 2, pp. 169–179, 2023.

ISSN: 1829-667X / E-ISSN: 2460-9021
DOI:10.31315/telematika.v22i2.14637

[19] E. Hashmi, S. Y. Yayilgan, M. M. Yamin, S. Ali, and M. Abomhara, "Advancing fake news detection: hybrid deep learning with fasttext and explainable AI," *IEEE Access*, 2024.

[20] S. Ghosal and A. Jain, "Depression and suicide risk detection on social media using fasttext embedding and xgboost classifier," *Procedia Comput Sci*, vol. 218, pp. 1631–1639, 2023.

[21] R. A. Sekarwati, A. Sururi, M. A. Rakhmat, and A. Wibowo, "Survei Metode Pengujian Chatbot pada Media Sosial untuk Mengukur Tingkat Akurasi Survey of Chatbot Testing Methods on Social Media to Measure Accuracy," *vol*, vol. 11, pp. 172–182, 2021.

[22] R. Andarsyah, C. Y. Pratama, and H. D. Kishendrian, "Implementasi Code Coverage Pada Chatbot Telegram Sebagai Media Alternatif Sistem Informasi," *Jurnal Teknik Informatika*, vol. 14, no. 2, pp. 112–117, 2022.

[23] R. P. Yuwan, R. Soelistijadi, and E. Zuliarso, "Implementasi Chatbot Telegram untuk Meningkatkan Kualitas Layanan Jaringan Internet Pada Layanan ICONNET Menggunakan Penerapan Metode Action Research (AR)," *Jurnal JTIK (Jurnal Teknologi Informasi dan Komunikasi)*, vol. 8, no. 1, pp. 40–48, 2024.

[24] A. Ainurrohmah, "Akurasi Algoritma Klasifikasi pada Software Rapidminer dan Weka," in *PRISMA, Prosiding Seminar Nasional Matematika*, 2021, pp. 493–499.

[25] K. S. Y. Putri, I. M. A. D. Suarjaya, and W. O. Vihikan, "Sistem Rekomendasi Skincare Menggunakan Metode Content Based Filtering dan Collaborative Filtering," *Decode: Jurnal Pendidikan Teknologi Informasi*, vol. 4, no. 3, pp. 764–774, 2024.