

IMPLEMENTASI *LONG-SHORT TERM MEMORY* (LSTM) UNTUK GENERASI *FEEDBACK* BERBAHASA INDONESIA PADA SISTEM PENILAIAN ESAI

Delvia Lana Sembu¹

¹Teknik Informatika, Sains dan Teknologi, Universitas Muhammadiyah Sukabumi

delvialanasemba081@ummi.ac.id

Abstract

Essays are used as evaluation tools that require subjective and in-depth responses from students, highlighting high cognitive abilities and good writing skills. The importance of feedback in identifying aspects that need improvement in student responses has been widely recognized. Automated Essay Scoring (AES) is a system that utilizes machine learning technology to automatically evaluate student essays. Despite research on AES in Indonesian language such as SIMPLE, Automated Bahasa Indonesia Essay Evaluation with Latent Semantic Analysis, and E-learning Automated Essay Scoring System Using Text Similarity Matching Text, the implementation of feedback for Indonesian AES is still incomplete. Providing feedback for each essay can be a time-consuming task, especially when there are many essays to be assessed. This can lead to fatigue for teachers and risk reducing the quality of the feedback provided. Long-Short Term Memory (LSTM) has the potential as a feedback system in Indonesian language. As one type of algorithm in the Recurrent Neural Network (RNN) family, LSTM excels in handling complex sequential data, such as essay texts, and accurately identifying patterns and relationships between words in texts. Therefore, this research aims to implement a feedback system in Indonesian AES using the LSTM algorithm. The implementation of LSTM is then carried out for feedback development with outputs consisting of positive and negative feedback sentences, and AES with outputs ranging from one to four scores. The feedback model is then implemented with AES so that the scores generated by AES affect the type of feedback sentences given.

Keywords: Long-Short Term Memory, Feedback, Feedback Sentences, Automated Essay Scoring

Abstrak

Esai digunakan sebagai alat evaluasi yang memerlukan respon subjektif dan mendalam dari siswa, menyoroti kemampuan kognitif yang tinggi dan keterampilan menulis yang baik. Pentingnya umpan balik dalam mengidentifikasi aspek-aspek yang perlu ditingkatkan dalam jawaban siswa telah diakui secara luas. *Automated Essay Scoring* (AES) adalah sistem yang menggunakan teknologi *machine learning* untuk mengevaluasi esai siswa secara otomatis. Meskipun telah ada penelitian tentang AES berbahasa Indonesia seperti SIMPLE, *Automated Bahasa Indonesia Essay Evaluation with Latent Semantic Analysis*, dan *E-learning Automated Essay Scoring System Menggunakan Metode Searching Text Similarity Matching Text*, namun implementasi umpan balik untuk AES berbahasa Indonesia masih belum sepenuhnya dilakukan. Memberikan umpan balik terhadap setiap esai bisa menjadi pekerjaan yang memakan waktu, terutama jika terdapat banyak esai yang perlu dinilai. Ini bisa menyebabkan kelelahan bagi pengajar dan berisiko mengurangi kualitas respon yang diberikan. *Long-Short Term Memory* (LSTM) memiliki potensi sebagai sistem umpan balik dalam bahasa Indonesia. Sebagai salah satu jenis algoritma dalam keluarga *Recurrent Neural Network* (RNN), LSTM unggul dalam mengatasi struktur data sekuensial yang kompleks, seperti teks esai, dan mampu mengidentifikasi pola serta hubungan antar kata-kata dalam teks secara akurat. Oleh karena itu, penelitian ini bertujuan untuk mengimplementasikan sistem umpan balik pada AES berbahasa Indonesia dengan menggunakan algoritma LSTM. Implementasi LSTM kemudian dilakukan untuk pengembangan umpan balik dengan output berupa kalimat umpan balik berjenis positif dan negatif dan AES dengan output berupa skor dengan *range* satu hingga empat. Model umpan balik kemudian diimplementasikan dengan AES sehingga skor yang dikeluarkan oleh AES berpengaruh terhadap jenis kalimat umpan balik yang diberikan.

Kata kunci: *Long-Short Term Memory, Umpan Balik, Kalimat Umpan Balik, Automated Essay Scoring*

©This work is licensed under a Creative Commons Attribution - ShareAlike 4.0 International License

1. Pendahuluan

Esai merupakan cara untuk mengevaluasi siswa yang mengharuskan mereka merespons pertanyaan secara subjektif dan mendalam. Ini menuntut kemampuan kognitif yang tinggi dan keterampilan menulis yang baik. Salah satu bagian penting dari proses pengerjaan esai adalah pemberian umpan balik. Umpan balik dapat memberikan informasi terkait area-area yang perlu ditingkatkan atau diperbaiki dalam jawaban esai yang ditulis [1].

Sebagai hasilnya, esai dianggap sebagai alat evaluasi yang efektif untuk mengukur pemahaman siswa terhadap materi yang dipelajari sebelumnya [2].

Ketika tidak ada umpan balik, siswa mungkin kesulitan menyadari kesalahan atau kelemahan mereka, yang dapat menghambat pemahaman mereka terhadap materi yang dipelajari. Selain itu, umpan balik yang positif juga dapat memotivasi siswa untuk meningkatkan pemahaman mereka [3].

Dalam era kemajuan teknologi yang terus berkembang, telah muncul sistem bernama *Automated Essay Scoring* (AES). Sistem ini menggunakan teknologi *machine learning* yang telah dirancang secara khusus untuk mengevaluasi dan menilai esai-esai yang ditulis oleh siswa. AES telah dikembangkan dalam berbagai bahasa, salah satunya adalah dalam bahasa Indonesia yang mana peran umpan balik menjadi perhatian utama dalam pengembangannya [4].

Dalam penelitian sebelumnya, sistem *Automated Essay Scoring* (AES) dan sistem umpan balik telah dikembangkan dan dievaluasi. Penelitian tersebut berjudul "*A Human-Centric AES and Feedback System for the Development of Ethical Reasoning*", dengan tujuan membuat AES yang lebih fokus pada manusia dan memberikan umpan balik yang baik kepada mahasiswa dalam kursus online dengan banyak peserta. Penelitian tersebut menggunakan dua jenis umpan balik. Pertama, AES menilai esai mahasiswa dan memberikan skor. Kedua, instruktur memilih esai yang perlu diperbaiki berdasarkan skor AES, lalu memberikan umpan balik kepada mahasiswa. Masalahnya, pemberian umpan balik oleh instruktur atau pengajar untuk setiap esai bisa menjadi pekerjaan yang membutuhkan banyak waktu apalagi jika esai yang diperiksa berjumlah banyak. Hal ini dapat membuat pengajar kelelahan dan mempengaruhi kualitas umpan balik yang diberikan [5].

Penelitian lain menggabungkan *deep learning* ke dalam AES untuk menciptakan sistem penilaian ujian yang efisien. Penelitian ini berjudul "*Integrating Deep Learning into An Automated Feedback Generation System for Automated Essay Scoring*". Umpan balik dalam penelitian ini dibuat berdasarkan skor esai yang diberikan oleh AES setelah melalui dua tahap pengembangan [6]. Kedua penelitian tersebut telah menerapkan umpan balik pada AES yang mereka kembangkan.

Terdapat penelitian tentang sistem *Automated Essay Scoring* (AES) dalam bahasa Indonesia. Salah satunya adalah SIMPLE: Sistem Penilai Esei Otomatis Untuk Menilai Ujian Dalam Bahasa Indonesia, yang bertujuan untuk membuat sistem evaluasi jawaban ujian dalam Bahasa Indonesia. Selain itu, ada penelitian lain yang bernama *Automated Bahasa Indonesia Essay Evaluation with Latent Semantic Analysis* yang mengintegrasikan sistem penilaian otomatis Bahasa Indonesia ke dalam platform e-learning menggunakan metode analisis semantik laten. Penelitian lainnya, yang berjudul *E-learning Automated Essay Scoring System Menggunakan Metode Searching Text Similarity Matching Text*, mengembangkan platform e-learning dengan sistem *Automated Essay Scoring* dalam Bahasa Indonesia. Namun, dalam sistem penilaian otomatis Bahasa Indonesia tersebut, belum dilakukan pengimplementasian sistem umpan balik.

Natural Language Processing (NLP) merupakan salah satu cabang dari *Artificial Intelligence* (AI) yang bertujuan untuk mengajarkan komputer dalam memahami teks. NLP memungkinkan komputer untuk membaca teks kemudian memahaminya dan menghasilkan teks baru [7].

Dalam NLP, salah satu data yang digunakan merupakan data teks. Setiap kata dalam sebuah data teks atau kalimat mengandung banyak makna yang penting, untuk itu, waktu dan urutan kata berperan penting dalam pengembangan kalimat. Salah satu algoritma NLP yang sering digunakan untuk tujuan ini adalah *Recurrent Neural Network* (RNN) [8]. RNN menggunakan *output* dari *layer* sebelumnya untuk dijadikan *input* pada *layer* selanjutnya [9].

Salah satu dari algoritma RNN adalah *Long-Short Term Memory* (LSTM). LSTM merupakan jenis jaringan saraf yang sangat baik dalam memproses data berurutan seperti teks. Dalam sebuah penelitian yang berjudul "*Integrating Deep Learning into An Automated Feedback Generation System for Automated Essay Scoring*", LSTM telah terbukti lebih baik daripada jenis jaringan saraf lainnya dalam memprediksi skor esai. Keunggulan LSTM juga terletak pada kemampuannya untuk memahami struktur teks esai dan meningkatkan prediksi skor atribut-esai. Ini sangat penting dalam *Automated Essay Scoring* (AES) untuk memberikan umpan balik yang baik [10].

Sayangnya, pada sistem penilaian otomatis bahasa Indonesia yang telah dikembangkan, pengimplementasian sistem umpan balik belum dilakukan. Maka, tujuan dari penelitian ini adalah menerapkan sistem umpan balik pada *Automated Essay Scoring* (AES) dalam bahasa Indonesia menggunakan algoritma *Long-Short Term Memory* (LSTM).

2. Metode Penelitian

Untuk mencapai tujuan penelitian, diperlukan penggunaan suatu metode. Metode ini membantu merancang langkah-langkah penelitian yang lebih terstruktur dan sistematis, sehingga penelitian menjadi lebih mudah dikelola. Metode juga memastikan peneliti tetap fokus pada jalur yang jelas dan mendapatkan gambaran besar tentang penelitian yang akan dilakukan. Dalam penelitian ini, digunakan metode *Sample, Explore, Modify, Model, and Assess* (SEMMA), dengan langkah-langkah berikut:

2.1 Pengumpulan Data (*Sample*)

Penelitian ini menggunakan dua jenis data, yaitu data kumpulan esai dari [Kaggle](#) berbentuk *Comma Seperated Value* (CSV) untuk tahap *Automated Scoring System* (AES) dan kumpulan kalimat umpan balik yang digenerasi oleh AI Chat-GPT dan Bard untuk tahap pengembangan umpan balik.

2.2 Deskripsi Data (*Explore*)

Penelitian ini menggunakan dua jenis dataset. Pertama, untuk mengembangkan *Automated Essay Scoring* (AES), digunakan dataset kumpulan esai yang sudah memiliki skor terkait untuk setiap esai. Kedua, untuk pengembangan umpan balik, dataset dikumpulkan dan disimpan dalam format file CSV berdasarkan fitur-fitur yang telah ditentukan sebelumnya. Terdapat delapan fitur yang ditetapkan untuk dataset pengembangan umpan balik, yang disesuaikan dengan *output* yang dihasilkan dari AES.

2.3 Modifikasi Data (*Modify*)

Proses ini menekankan persiapan data melalui pembersihan, transformasi, dan modifikasi agar sesuai untuk pemodelan. Data yang diterapkan pada *Automated Essay Scoring* (AES) dan pengembangan umpan balik belum optimal, sehingga memerlukan transformasi data untuk digunakan dalam pengembangan model.

2.3.1 Data Preprocessing

Untuk pengembangan AES, data terjemahan awalnya ke Bahasa Indonesia. Setiap kolom *dataset* kemudian diubah menjadi *list*, dilakukan *stopwords removing*, *lowercasing*, tokenisasi, *stripping*, *splitting*, dan penghapusan karakter *non-alphabetic*.

Sedangkan untuk pengembangan umpan balik, data diubah menjadi *list*, kemudian di-tokenisasi. Kalimat umpan balik yang telah di-tokenisasi kemudian diubah menjadi sekuens bilangan bulat. Terakhir, dilakukan *padding* untuk memastikan setiap sekuens memiliki panjang yang sama.

2.3.2 Feature Engineering

Feature creation adalah proses menambah, menghapus, atau membuat fitur baru pada dataset. *Feature creation* diterapkan pada tahap pengembangan AES [11]. Dalam tahap ini, sebuah rubrik penilaian dibuat untuk menentukan *output* dari AES. Berdasarkan rubrik tersebut, setelah menerjemahkan *dataset*, hanya fitur esai yang digunakan. Dari fitur esai tersebut, dibuat empat *dataset* baru dengan fitur yang sesuai dengan rubrik masing-masing.

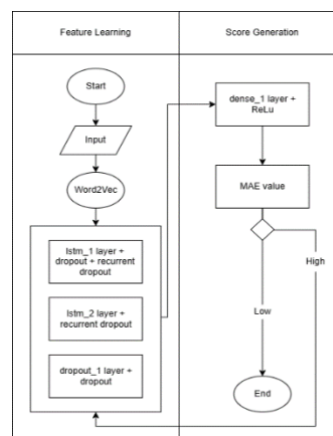
Selanjutnya adalah *feature extraction*. *Feature extraction* adalah proses mengekstraksi fitur dari *dataset* yang digunakan tanpa melakukan perubahan pada *dataset* aslinya [11]. Dalam pengembangan AES, langkah ekstraksi fitur menggunakan *Word2vec* juga dilakukan untuk melatih model.

2.4 Pemodelan Data (*Model*)

Pada tahap pemodelan, secara umum proses AES dan pengembangan umpan balik terbagi menjadi dua tahap, yaitu tahap pembelajaran fitur dan tahap pengembangan, dengan rincian sebagai berikut:

2.4.1 Automated Essay Scoring (AES)

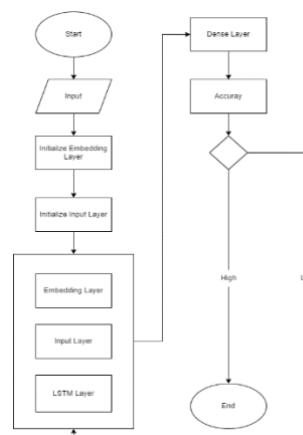
Pada tahap pembelajaran fitur, metode *word embedding*, *word2vec*, digunakan pada data *input*. Kemudian, model dengan dua lapisan LSTM diterapkan untuk memproses data input dan sekuensial. Regularisasi *dropout* juga diterapkan untuk mencegah *overfitting*, diikuti oleh penggunaan lapisan *Dense* untuk menghasilkan skor akhir dengan aktivasi *ReLU*. Tahap-tahap tersebut digambarkan menggunakan *flowchart* pada Gambar 2.1 sebagai berikut:



Gambar 2.1 Flow Chart Model Automated Essay Scoring (AES)

2.4.2 Pengembangan Umpan Balik

Untuk pengembangan model umpan balik, digunakan lapisan *Embedding* dan LSTM untuk memproses serta mengekstraksi informasi dari kalimat input. *Input Layer* digunakan untuk menerima data *input*, dan *Dense Layer* digunakan untuk menghasilkan kalimat *feedback* yang sesuai dengan konteks kalimat input dengan aktivasi *Softmax*. Seperti digambarkan pada Gambar 2.2 seperti sebagai berikut:



Gambar 2.2 Flow Chart Model Pengembangan Umpan Balik

2.5 Evaluasi Data (Assess)

Setelah model dibangun, evaluasi dilakukan menggunakan metrik BLEU untuk kalimat umpan balik. Evaluasi dilakukan dengan menghitung kesamaan persentase *n-gram* antara kalimat umpan balik yang dihasilkan oleh model dengan kalimat umpan balik dari *dataset* yang digunakan.

3. Hasil dan Pembahasan

3.1 Pengumpulan Data (Sample)

Data yang digunakan dalam penelitian ini terbagi menjadi dua bagian, yaitu *dataset* untuk *Automated Scoring Essay* (AES) dan *dataset* untuk pengembangan umpan balik. Untuk AES, data yang digunakan berasal dari *The William and Flora Hewlett Foundation* (Hewlett), yang merupakan *dataset* yang mendukung *Automated Student Assessment Prize* (ASAP).

Untuk pengembangan umpan balik, data yang dimanfaatkan adalah sekumpulan kalimat umpan balik yang diperoleh dari berbagai sumber di internet.

3.2 Deskripsi Data (Explore)

3.2.1 Automated Essay Scoring (AES)

Data yang dipakai dalam proses pengembangan AES terdiri dari sekumpulan jawaban esai yang telah dievaluasi oleh pakar, dengan rata-rata panjang antara 150 hingga 550 kata. Setiap respons esai dikelompokkan ke dalam empat jenis esai, seperti dideskripsikan pada Tabel 3.1 berikut:

Tabel 3.1 Tipe Esai

Tipe Esai	Deskripsi
<i>Persuasive</i> (Persuasif)	Tipe esai persuasive atau persuasif merupakan jenis esai yang ditulis untuk meyakinkan pembaca terhadap pandangan atau pendapat penulis.
<i>Narrative</i> (Naratif)	Tipe esai ini merupakan jenis esai yang ditulis untuk menceritakan sesuatu, seperti pengalaman penulis.
<i>Ekspository</i> (Ekspositori)	Tipe ekspositori merupakan jenis esai yang ditulis guna menjelaskan, menguraikan atau memberikan informasi tentang suatu topik.
<i>Source Dependent Response</i>	Tipe ini merupakan jenis esai yang ditulis sebagai respon terhadap teks sumber yang diberikan.

Dataset ini memiliki sepuluh kolom atau atribut dengan penjelasan seperti pada Tabel 3.2 berikut:

Tabel 3.2 Deskripsi *Dataset*

Fitur	Deskripsi	Fitur
essay_id	Indentifikasi unik untuk setiap dataset	essay_id
essay_set	Nama set esai	essay_set

essay	Esai	essay
rater1_domain1	Penilaian oleh human-rater pertama	rater1_domain1
rater2_domain1		rater2_domain1
rater3_domain1		rater3_domain1
rater2_domain2	Penilaian oleh human-rater kedua	rater2_domain2
rater2_domain2		rater2_domain2
domain1_score	Skor akhir esai	domain1_score
domain2_score		domain2_score

Hanya satu atribut dari kesepuluh atribut *dataset* yang dipertimbangkan, yaitu atribut esai. Atribut ini kemudian digunakan untuk membentuk *dataset* baru sesuai dengan standar penilaian yang telah ditentukan. Standar penilaian tersebut dideskripsikan pada Tabel 3.3 untuk analisis konten, Tabel 3.4 untuk struktur dan organisasi, Tabel 3.5 untuk kesalahan ejaan serta Tabel 3.6 untuk kesalahan ejaan.

Tabel 3.3 Rubrik Analisis Konten

Jumlah Kata Benda (Noun)	Skor	Jumlah Kata Benda (Noun)
>= 0	1	>= 0
> 68	2	> 68
> 115	3	> 115

Tabel 3.4 Rubrik Struktur dan Organisasi

Jumlah Kata (Word)	Skor	Jumlah Kata (Word)
>= 0	1	>= 0
>= 83	2	>= 83
>= 140	3	>= 140

Tabel 3.5 Rubrik Kesalahan Ejaan

Jumlah Spelling Error	Skor	Jumlah Spelling Error
>= 0	4	>= 0
>= 7	3	>= 7
>= 13	2	>= 13

Tabel 3.6 Rubrik Relevansi dengan Konten

Skor Content Relevance	Skor Akhir	Skor Content Relevance
<= 3	1	<= 3
<= 6	2	<= 6
<= 9	3	<= 9

Batasan angka untuk setiap rentang skor ditentukan dengan menghitung nilai minimum dan maksimum dari setiap objek yang dinilai. Sebagai contoh, untuk rubrik analisis konten, dilakukan perhitungan nilai minimum dan maksimum untuk *noun_count*.

```
Noun Limit
max_noun_count = df['noun_count'].max()
min_noun_count = df['noun_count'].min()
```

Berdasarkan nilai minimum dan maksimum tersebut, batasan jumlah kata benda atau noun untuk setiap skor ditentukan secara manual.

```
Noun Limit
max_noun_count = df['noun_count'].max()
min_noun_count = df['noun_count'].min()
```

Hasil pengelompokan secara manual tersebut kemudian diimplementasikan dan diperiksa.

```
Score Assigner
def score_assigner(count, criteria):
    if count > criteria[3]:
        return 4
    elif count > criteria[2]:
        return 3
    elif count > criteria[1]:
        return 2
    elif count >= criteria[0]:
        return 1
    else:
        return 0
```

Hal ini dilakukan untuk memastikan dataset yang dibagi untuk setiap range skor seimbang. Berikut contoh jumlah pembagian dataset untuk range skor rubrik analisis konten dengan objek yang dinilai adalah kata kerja. Hasil akhir dataset tersebut dapat dilihat pada Tabel 3.7 untuk analisis konten, Tabel 3.8 untuk struktur dan organisasi, Tabel 3.9 untuk kesalahan ejaan dan Tabel 3.10 untuk fitur relevansi dengan konten.

Tabel 3.7 Fitur Analisis Konten

Fitur	Deskripsi	Fitur
id_esai	Identifikasi untuk setiap esai	id_esai
esai	Esai	esai
clean_essay	Esai yang telah dibersihkan	clean_essay

Tabel 3.8 Fitur Struktur dan Organisasi

Fitur	Deskripsi	Fitur
id_esai	Identifikasi untuk setiap esai	id_esai
esai	Esai	esai
clean_essay	Esai yang telah dibersihkan	clean_essay

Tabel 3.9 Fitur Kesalahan Ejaan

Fitur	Deskripsi	Fitur
id_esai	Identifikasi untuk setiap esai	id_esai
esai	Esai	esai

```
clean_essay Esai yang telah dibersihkan clean_essay
```

Tabel 3.10 Fitur Relevansi dengan Konten

Fitur	Deskripsi	Fitur
id_esai	Identifikasi untuk setiap esai	id_esai
esai	Esai	esai
clean_essay	Esai yang telah dibersihkan	clean_essay

3.2.2 Pengembangan Umpan Balik

Kalimat umpan balik yang terkumpul sesuai dengan output dari model *Automated Essay Scoring* (AES), mencakup penilaian terhadap rubrik analisis konten, struktur dan organisasi, kesalahan ejaan, dan relevansi konten. Pada Tabel 3.11 berikut adalah atribut-atribut yang digunakan dalam pengembangan model umpan balik.

Tabel 3.11 Fitur *Dataset* Pengembangan Umpan Balik

Fitur	Deskripsi	Jumlah
sao_pos	Kalimat umpan positif untuk rubrik Struktur dan Organisasi	120
sao_neg	Kalimat umpan negatif untuk rubrik Struktur dan Organisasi	120
ca_pos	Kalimat umpan positif untuk rubrik Analisis Konten	120
ca_neg	Kalimat umpan negatif untuk rubrik Analisis Konten	120
gas_pos	Kalimat umpan positif untuk rubrik Kesalahan Ejaan	120
gas_neg	Kalimat umpan negatif untuk rubrik Kesalahan Ejaan	120
cor_pos	Kalimat umpan positif untuk rubrik Relevansi dengan Konten	120
cor_neg	Kalimat umpan negatif untuk rubrik Relevansi dengan Konten	120

Total jumlah kalimat umpan balik dalam *dataset* yang dipakai untuk pengembangan model ini mencapai 960 kalimat.

3.3 Modifikasi Data (*Modify*)

3.3.1 *Automated Essay Scoring* (AES)

Dataset yang diperoleh berisi esai dalam Bahasa Inggris, sehingga memerlukan terjemahan ke dalam Bahasa Indonesia. Proses terjemahan dilakukan secara otomatis menggunakan kode yang mengimplementasikan library *EasyGoogleTranslate*.

```
EasyGoogleTranslate
from easygoogletranslate import
EasyGoogleTranslate
```

Terjemahan hasilnya kemudian disimpan dalam format *Comma Separated Value* (CSV). Dari *dataset* yang

telah diterjemahkan, dibuat empat dataset baru sesuai dengan prosedur sebelumnya. Adapun contoh hasil terjemahan tersebut terlihat pada Tabel 3.12 berikut:

Tabel 3.12 Terjemahan Dataset

Bahasa Inggris	Bahasa Indonesia
Dear local newspaper, I think effects computers have on people are great learning skills/affects because they give us time to chat with friends/new people, helps us learn about the	Surat kabar lokal yang terhormat, menurut saya pengaruh komputer terhadap manusia adalah keterampilan/pengaruh belajar yang hebat karena komputer memberi kita waktu untuk ngobrol dengan teman/orang baru, membantu kita belajar tentang

Tahap selanjutnya merupakan persiapan data. Pada tahap persiapan data, dilakukan penambahan *id* dan pembersihan data. Proses pembersihan data mencakup *lowercasing*, penghapusan tanda baca atau *punctuation*, dan tokenisasi.

Menggunakan *dataset* yang telah diterjemahkan dan diproses, kemudian dibuat empat *dataset* baru yakni *dataset* untuk analisis konten, struktur dan organisasi, kesalahan ejaan, dan relevansi dengan konten.

Dataset untuk analisis konten mengalami perulangan pada setiap kata menggunakan *library nltk*. Dalam setiap perulangan, kata-kata diperiksa untuk menemukan kata benda, dan jumlahnya kemudian dikelompokkan berdasarkan rubrik yang telah ditetapkan sebelumnya untuk menentukan skornya.

Content Analysis

```
def count_pos_tags(essay, pos_tag):
    words = [word for sentence in essay for word in sentence]
    pos_tags = nltk.pos_tag(words)
    return sum(1 for pos in pos_tags if pos.startswith(pos_tag))
df['noun_count'] = df['clean_essay'].apply(lambda essay: count_pos_tags(essay.split(), 'N'))
```

Rubrik struktur dan organisasi mengukur total kata dalam setiap esai berdasarkan tokenisasi dari fitur *clean_essay*. Skor ditetapkan berdasarkan total kata tersebut sesuai dengan rubrik yang telah ditetapkan sebelumnya.

Structure and Organization

```
def count_words(tokens):
    return len(tokens)
df['word_count'] = df['clean_essay'].apply(count_words)
```

Penyusunan *dataset* untuk mendeteksi Kesalahan Ejaan memanfaatkan *library symspell*. Proses dimulai dengan inisialisasi dan penerapan *library symspell* untuk setiap kata dalam fitur *clean_essay*. Selanjutnya, kesalahan ejaan dihitung dengan membandingkan kata dalam *clean_essay* dengan kamus yang disediakan sebagai

referensi. Jumlah kesalahan ejaan kemudian dinilai berdasarkan rubrik yang telah ditetapkan sebelumnya.

Spelling Error

```
def create_symspell(words):
    symspell = SymSpell()
    for word in words:
        symspell.create_dictionary_entry(word, 1)
    return symspell
def count_spell_errors(symspell, tokens):
    error_count = 0
    for word in tokens:
        suggestions = symspell.lookup(word, Verbosity.CLOSEST, max_edit_distance=2, include_unknown=True)
        if not suggestions or suggestions[0].term != word:
            error_count += 1
    return error_count
```

Dalam kriteria penilaian terakhir, skor akhir dihitung menggunakan rumus berikut:

$$\frac{x + y + z}{3} \qquad 4.1$$

Keterangan:

- x* = *structure_and_organization_final_score*
- y* = *content_analysis_final_score*
- z* = *grammar_and_language_final_score*

3.3.2 Pengembangan Umpan Balik

Setiap atribut dalam dataset pengembangan umpan balik diubah menjadi daftar string.

String List

```
corpus_content_analysis_pos = df['ca_pos'].astype(str).tolist()
corpus_content_analysis_neg = df['ca_neg'].astype(str).tolist()
```

Proses selanjutnya tokenisasi diterapkan pada dataset yang telah diubah menjadi *list*. Ini dimulai dengan menginisialisasi tokenizer dan kemudian melakukan tokenisasi pada setiap kata dalam dataset.

Tokenisasi

```
def tokenize_feedback(corpus):
    tokenizer = Tokenizer()
    tokenizer.fit_on_texts(corpus)
    total_words = len(tokenizer.word_index) + 1
    return tokenizer, total_words
```

Fitur yang telah ditokenisasi kemudian diubah menjadi urutan angka atau vektor untuk memungkinkan pemrosesan oleh model.

Training Sequences

```
def training_sequences(corpus, tokenized_corpus, input_sequences):
    for line in corpus:
        token_list = tokenized_corpus.texts_to_sequences([line])[0]
        for i in range(1, len(token_list)):
            n_gram_sequence = token_list[i:i+1]
```

```
input_sequences.append(n_gram_sequence)
```

Selanjutnya, urutan input akan diproses untuk digunakan sebagai masukan model.

Process Sequence Data

```
def process_sequence_data(input_sequences,
total_words):
    max_sequence_len = max([len(x) for x in
input_sequences])
    input_sequences =
np.array(pad_sequences(input_sequences,
maxlen=max_sequence_len, padding='pre'))

    xs = input_sequences[:, :-1]
    labels = input_sequences[:, -1]
    ys =
tf.keras.utils.to_categorical(labels,
num_classes=total_words)

    return xs, ys, max_sequence_len,
input_sequences
```

Pada Tabel 3.13 berikut adalah hasil output dari fungsi tersebut.

Tabel 3.13 Output Process Sequence Data

Output	Deskripsi	Fungsi
xs	Input sequence tanpa elemen terakhir	Model dikembangkan untuk memprediksi setiap kata selanjutnya, sehingga input sequences tidak memiliki elemen terakhir agar model dapat memprediksinya
ys	One-hot encoded label	Digunakan sebagai label output untuk hasil latihan.
max_sequence_len	Panjang maksimum input sequences	Menentukan input sequences yang dihasilkan saat pelatihan.

Pertama, dilakukan proses *reshaping* dengan menentukan ukuran minimum dari semua input xs. Ukuran minimum tersebut adalah sebesar 1318.

Reshape

```
min_size = min(xs_ca_neg.shape[0],
xs_ca_pos.shape[0],
xs_sao_pos.shape[0],
xs_ca_neg.shape[0],
xs_cor_neg.shape[0])
```

Setelah itu, dilakukan sampling acak dengan panjang sampel yang sama dengan *min_size* yang telah ditetapkan sebelumnya. Tujuannya adalah untuk menjaga keseimbangan antara fitur positif dan negatif.

Reshape

```
indices_to_keep_ca_pos =
np.random.choice(xs_ca_pos.shape[0],
size=min_size, replace=False)
indices_to_keep_ca_neg =
np.random.choice(xs_ca_neg.shape[0],
size=min_size, replace=False)
```

3.4 Pemodelan (Model)

3.4.1 Automated Essay Scoring (AES)

Pemodelan untuk AES dilakukan dengan menggunakan fungsi *sequential* dari library Keras.

Automated Essay Scoring

```
def get_model():
    model = Sequential()
    model.add(LSTM(300, dropout=0.4,
recurrent_dropout=0.4, input_shape=(1, 300),
return_sequences=True))
    model.add(LSTM(64,
recurrent_dropout=0.4))
    model.add(Dropout(0.5))
    model.add(Dense(4, activation='relu'))

    model.compile(loss='mean_squared_error',
optimizer='rmsprop', metrics=['mae'])
    model.summary()
    return model
```

Adapun fungsi dari setiap layer dan parameternya dituliskan pada Tabel 3.14 berikut:

Tabel 3.14 AES Model

Layer	Parameter	Deskripsi
LSTM 1	unit	Menangkap pola dalam data. Unit yang digunakan berjumlah 300.
	dropout	Untuk menghindari overfitting. Nilai dropout yang digunakan adalah 0.4
	recurrent_dropout	Untuk menghindari pola pada hubungan temporal antar unit yang terlalu kompleks. Nilai recurrent dropout yang digunakan adalah 0.4
	input_shape	Menentukan ukuran masukan. Input shape yang digunakan adalah (1, 300)
	return_sequences	Mengembalikan setiap output sequences. Return sequences yang digunakan adalah True. Apabila return sequences yang digunakan adalah False,

LSTM 2	unit	maka layer tidak akan mengembalikan output sequences. Menangkap pola dalam data. Unit yang digunakan berjumlah 64.
	recurrent_dropout	Untuk menghindari pola pada hubungan temporal antar unit yang terlalu kompleks. Nilai recurrent dropout yang digunakan adalah 0.4
Dropout	-	Untuk menghindari overfitting. Nilai dropout yang digunakan adalah 0.5
Dense	neuron	Menghasilkan output. Jumlah neuron yang digunakan adalah 4
	activation	Mengaktivasi neuron. Aktivasi yang digunakan merupakan ReLu. ReLu digunakan karena aktivasi relu menghasilkan output yang linear.
Compile	loss	Mengevaluasi model. Loss yang digunakan merupakan Mean Squared Error (MSE). Semakin kecil nilai MSE, maka semakin baik model.
	optimizer	Menyesuaikan bobot dan bias. Optimizer yang digunakan merupakan RMSprop. RMSprop digunakan karena berdasarkan percobaan, RMSprop menghasilkan nilai yang baik.
	metrics	Mengevaluasi model. Loss yang digunakan merupakan Mean Absolute Error (MAE). Semakin kecil nilai MAE, maka semakin baik model.

Adapun hasil dari model AES adalah sebagai berikut:

Hasil dari model AES tersebut berupa skor dengan *range* skor satu hingga empat. Input untuk model AES berupa teks esai seperti pada Tabel 3.15 berikut:

Tabel 3.15 Sample Input Essay

Input Essay
Surat kabar lokal yang terhormat, Saya memahami bahwa orang-orang sedang mencoba untuk memutuskan apakah menggunakan komputer adalah hal yang baik atau buruk, dan saya pikir ini akan membantu mereka mengambil keputusan. (...)

Kemudian *output* dari model AES tersebut mengeluarkan empat *output*, masing-masing satu untuk setiap rubrik penilaian dengan range skor satu sampai empat. Berikut merupakan *output* dari model AES menggunakan *sample input* esai yang diberikan sebelumnya yang dituliskan pada Tabel 3.16 berikut.

Tabel 3.16 Sample Output AES

Fitur		Deskripsi	Fitur
Dimension Score	1	Struktur dan Organisasi	1.6722069
Dimension Score	2	Analisis Konten	1.6536046
Dimension Score	3	Kesalahan Ejaan	3.8370764
Dimension Score	4	Relevansi dengan Konten	2.6249323

3.4.2 Pengembangan Umpan Balik

Lapisan pertama dalam model pengembangan umpan balik adalah lapisan *embedding*. Argumennya mencakup jumlah kata maksimum dari setiap *dataset*, yang ditentukan dengan menggunakan fungsi *max*. Ini menentukan ukuran matriks lapisan *embedding*. Jumlah dimensi untuk merepresentasikan setiap kata adalah 64. Argumen terakhir menentukan panjang data input, yang setara dengan maksimum dari panjang sekuens dataset analisis konten dikurangi satu.

```
Embedding Layer
embedding_layer =
Embedding(max(total_words_content_analysis_neg
9,
total_words_content_analysis_pos,
total_words_structure_and_organization_neg,
total_words_structure_and_organization_pos,
total_words_grammar_and_language_pos,
total_words_grammar_and_language_neg,
total_words_content_relevance_neg,
total_words_content_relevance_pos),
64,
input_length=max_sequence_len_ca_neg-1)
```


Lapisan kedua adalah *Bidirectional Layer* dengan 20 unit. Unit digunakan untuk mempelajari pola dalam proses pelatihan.

Bidirectional LSTM Layer

```
lstm_layer = Bidirectional(LSTM(20))
```

Layer berikutnya adalah Input Layer untuk menentukan bentuk masukan dari setiap dataset yang akan digunakan oleh model, dengan bentuk masukan sepanjang maksimum dari setiap sekuens dataset.

Input Layer

```
input_ca_neg = Input(shape=(max_sequence_len_ca_neg-1,))
input_ca_pos = Input(shape=(max_sequence_len_ca_pos-1,))
input_sao_neg = Input(shape=(max_sequence_len_sao_neg-1,))
input_sao_pos = Input(shape=(max_sequence_len_sao_pos-1,))
input_gas_neg = Input(shape=(max_sequence_len_gas_neg-1,))
input_gas_pos = Input(shape=(max_sequence_len_gas_pos-1,))
input_cor_neg = Input(shape=(max_sequence_len_cor_neg-1,))
input_cor_pos = Input(shape=(max_sequence_len_cor_pos-1,))
```

Setelah penginisialisasian lapisan *embedding*, lapisan *embedding* tersebut digunakan dengan menggunakan bentuk masukan yang telah ditentukan sebelumnya sebagai parameter. Implementasi lapisan *embedding* kemudian disimpan dalam variabel masing-masing.

Embedding Layer Implemented

```
embedding_ca_neg = embedding_layer(input_ca_neg)
embedding_ca_pos = embedding_layer(input_ca_pos)
embedding_sao_neg = embedding_layer(input_sao_neg)
embedding_sao_pos = embedding_layer(input_sao_pos)
embedding_gas_neg = embedding_layer(input_gas_neg)
embedding_gas_pos = embedding_layer(input_gas_pos)
```

Layer Bidirectional diterapkan pada lapisan *embedding* dari setiap dataset.

Bidirectional LSTM Layer Implemented

```
lstm_output_ca_neg = lstm_layer(embedding_ca_neg)
lstm_output_ca_pos = lstm_layer(embedding_ca_pos)
lstm_output_sao_neg = lstm_layer(embedding_sao_neg)
lstm_output_sao_pos = lstm_layer(embedding_sao_pos)
lstm_output_gas_neg = lstm_layer(embedding_gas_neg)
lstm_output_gas_pos = lstm_layer(embedding_gas_pos)
lstm_output_cor_neg = lstm_layer(embedding_cor_neg)
lstm_output_cor_pos = lstm_layer(embedding_cor_pos)
```

Setelah itu dilakukan penerapan *Dense layer* sebagai lapisan *output* dengan jumlah unit yang sama dengan total kata dalam setiap *dataset*. Fungsi aktivasi yang digunakan adalah *softmax*. Layer ini terhubung dengan *output* dari *Bidirectional LSTM Layer* sebagai input.

Dense Layer

```
dense_ca_neg = Dense(total_words_content_analysis_neg,
activation='softmax',
name='output_neg')(lstm_output_ca_neg)
dense_ca_pos = Dense(total_words_content_analysis_pos,
activation='softmax',
name='output_pos')(lstm_output_ca_pos)
lstm_output_cor_pos = lstm_layer(embedding_cor_pos)
dense_sao_neg = Dense(total_words_structure_and_organization_neg,
activation='softmax',
name='output_sao_neg')(lstm_output_sao_neg)
dense_sao_pos = Dense(total_words_structure_and_organization_pos,
activation='softmax',
name='output_sao_pos')(lstm_output_sao_pos)
dense_gas_neg = Dense(total_words_grammar_and_language_neg,
activation='softmax',
name='output_gas_neg')(lstm_output_gas_neg)
dense_gas_pos = Dense(total_words_grammar_and_language_pos,
activation='softmax',
name='output_gas_pos')(lstm_output_gas_pos)
```

```
dense_cor_neg = Dense(total_words_content_relevance_neg,
activation='softmax',
name='output_cor_neg')(lstm_output_cor_neg)
```

```
dense_cor_pos = Dense(total_words_content_relevance_pos,
activation='softmax',
name='output_cor_pos')(lstm_output_cor_pos)
```

Kemudian, *Model layer* disiapkan untuk menghubungkan *input* dan *output* dari model.

```
Model
model = Model(inputs=[input_ca_neg,
input_ca_pos, input_sao_neg, input_sao_pos,
input_gas_neg, input_gas_pos, input_cor_neg,
input_cor_pos], outputs=[dense_ca_neg,
dense_ca_pos, dense_sao_neg, dense_sao_pos,
dense_gas_neg, dense_gas_pos, dense_cor_neg,
dense_cor_pos])
```

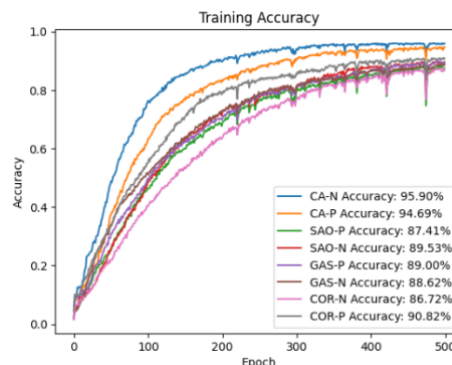
Setelah pengembangan model, langkah selanjutnya adalah melakukan kompilasi model. Ini melibatkan penggunaan *categorical_crossentropy* sebagai *loss function*, *adam* sebagai *optimizer*, dan *accuracy* sebagai *metrics*.

```
Compile
model.compile(loss='categorical_crossentropy',
optimizer='adam', metrics=['accuracy'])
```

Proses terakhir adalah pelatihan model, yang dilakukan dengan menggunakan fitur dan label dari setiap dataset sebagai argumen, dengan jumlah epoch sebanyak 500 dan verbose yang diatur ke 1.

```
Fit
history = model.fit([xs_ca_neg, xs_ca_pos,
xs_sao_neg, xs_sao_pos, xs_gas_neg,
xs_gas_pos, xs_cor_neg, xs_cor_pos],
[ys_ca_neg, ys_ca_pos, ys_sao_neg, ys_sao_pos,
ys_gas_neg, ys_gas_pos, ys_cor_neg,
ys_cor_pos], epochs=500, verbose=1)
```

Hasil akurasi dari pemodelan sistem umpan balik adalah sebagai berikut.



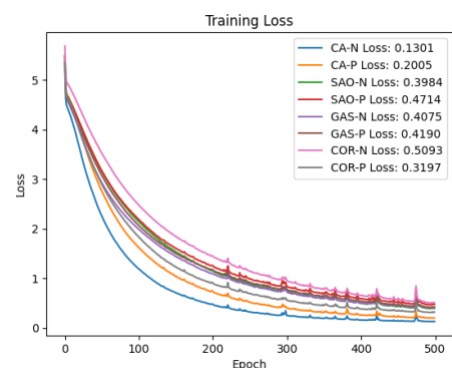
Gambar 3.1 Accuracy Model Umpan Balik

Hasil akurasi lebih jelasnya dapat dilihat pada Tabel 3.17 berikut.

Tabel 3.17 Accuracy Model Umpan Balik

Feedback	Accuracy
sao_pos	87.41%
sao_neg	89.53%
ca_pos	94.69%
ca_neg	95.90%
gas_pos	89.00%
gas_neg	88.62%
cor_pos	90.82%
cor_neg	86.72%

Adapun *loss* untuk pemodelan sistem umpan balik adalah sebagai berikut.



Gambar 3.2 Loss Model Umpan Balik

Output dari model pengembangan umpan balik merupakan sebuah kalimat umpan balik positif atau negatif untuk setiap rubrik. Pada Tabel 3.18 berikut merupakan contoh umpan balik yang dikembangkan oleh model untuk rubrik konten analisis.

Tabel 3.18 Sample Kalimat Umpan Balik Analisis Konten

Rubrik	Kalimat Umpan Balik
Konten Analisis Positif	Esaimu memiliki kesimpulan yang relevan dan menarik serta memancing pembaca untuk berpikir kritis dan dan

Konten Analisis Negatif	kejelasan esaimu sumber mempertimbangkan sudut pandang	2.6249323	COR Positif	Esaimu memberikan perspektif yang unik menjadikannya relevan dan menarik bagi pembaca dan meyakinkan dijelaskan lebih kritis relevan isu isu topik topik
	Esaimu perlu lebih didukung oleh bukti bukti yang lebih memadai dan relevan untuk mendukung klaim klaim yang diajukan klaim klaim sebelumnya			

3.4.3 Implementasi

Sistem umpan balik diintegrasikan ke dalam *Automated Essay Scoring* (AES) yang sudah kembangkan. AES akan memberikan hasil berupa penilaian nilai untuk setiap kriteria, dan kemudian menggunakan pernyataan kondisional dengan keterangan seperti pada Tabel 3.19 berikut.

Tabel 3.19 Keterangan Umpan Balik dan Skor Akhir

Rubrik	Skor >= 2	Skor < 2
Konten Analisis	Positif	Negatif
Struktur dan Organisasi	Positif	Negatif
Kesalahan Ejaan	Negatif	Positif
Relevansi dengan Konten	Positif	Negatif

Hasil dari penerapan model AES dengan model umpan balik menggunakan input esai yang sama seperti yang tercantum dalam Tabel 3.15. Dalam konteks ini, skor dan umpan balik yang dihasilkan berdasarkan input esai tersebut dapat dilihat pada Tabel 3.20.

Tabel 3.20 Sample Implementasi AES dan Sistem Umpan Balik

Skor	Jenis Umpan Balik	Kalimat Umpan Balik
1.6722069	SAO Negatif	Esaimu perlu memiliki alur yang lebih jelas dan logis sehingga pembaca dapat mengikuti argumen atau analisis inti lebih baik memerlukan dan Esaimu perlu lebih berbobot dan memberikan kontribusi yang signifikan terhadap bidang ilmu yang dibahas dengan menyajikan analisis yang lebih mendalam dan
1.6536046	CA Negatif	Esaimu perlu menggunakan kata kata kata yang repetitif memengaruhi pemahaman yang mengganggu sepenuhnya sepenuhnya terstruktur digunakan ambigu lebih mudah beberapa
3.8370764	GAS Negatif	

3.5 Evaluasi (Assess)

Evaluasi untuk kalimat umpan balik dilakukan menggunakan metrik BLEU, yang dijalankan dengan kode sebagai berikut.

```
BLEU
for x in range(len(references)):

    results = bleu.compute(
        predictions=predictions,
        references=lists[x],
        max_order=1
    )

    bleu_score = results['bleu']

    total_bleu_score += bleu_score

    precision_score = results['precisions'][0]

    total_precision_score += precision_score

    bp_score = results['brevity_penalty']

    total_bp_score += bp_score
```

Hasil evaluasi menunjukkan hasil seperti pada Tabel 3.21 berikut:

Tabel 3.21 Evaluasi BLEU Score

Rubrik	Precision	Brevity Penalty	BLEU Score
Konten Analisis Positif	0.14	0.5	0.06
Konten Analisis Negatif	0.21	0.56	0.11
Struktur dan Organisasi Positif	0.22	0.61	0.13
Struktur dan Organisasi Negatif	0.13	0.69	0.07
Kesalahan Ejaan Positif	0.22	0.62	0.13

Kesalahan Ejaan Negatif	0.21	0.59	0.11
Relevansi dengan Konten Positif	0.20	0.72	0.14
Relevansi dengan Konten Negatif	0.15	0.62	0.09

Penilaian evaluasi untuk skor BLEU dianggap cukup baik apabila skor BLEU yang dihasilkan lebih dari atau sama dengan 0.3. Namun, hasil penelitian menunjukkan bahwa skor BLEU dari kalimat umpan balik yang dihasilkan oleh sistem masih di bawah 0.3. Oleh karena itu, dapat disimpulkan bahwa kalimat umpan balik yang dihasilkan oleh model belum cukup baik dan belum sebaik umpan balik yang diberikan oleh manusia.

4. Kesimpulan

Berdasarkan penelitian yang dilakukan, algoritmya *Long-Short Term Memory* (LSTM) telah berhasil diimplementasikan sebagai sistem umpan balik untuk *Automated Essay Scoring* (AES) berbahasa Indonesia.

Implementasi algoritma LSTM untuk AES berbahasa Indonesia menggunakan 960 *dataset*. Pemodelan dilakukan dengan menggunakan *layer* LSTM, *categorical crossentropy* sebagai *loss function*, *adam* sebagai *optimizer*, *metrics accuracy* dan 500 *epochs*. *Output* dari model sistem umpan balik berupa kalimat umpan balik.

Untuk model AES, pemodelan dilakukan dengan menggunakan 12934 *dataset*, *layer* LSTM dengan masing-masing *layer* menggunakan 300 dan 64 *neuron*, *mean squared error* sebagai *loss function*, *rmsprop* sebagai *optimizer*, dan *metrics mae*. *Output* dari model AES merupakan skor dengan rentang nilai satu hingga empat.

Model AES kemudian diimplementasikan dengan model pengembangan umpan balik. Sehingga, nilai yang dihasilkan model AES menentukan umpan balik yang diberikan. Dengan keterangan seperti pada Tabel 4.1 berikut:

Tabel 4.1 Keterangan Umpan Balik dan Skor Akhir

Rubrik	Skor ≥ 2	Skor < 2
Konten Analisis	Positif	Negatif
Struktur dan Organisasi	Positif	Negatif
Kesalahan Ejaan	Negatif	Positif
Relevansi dengan Konten	Positif	Negatif

Terakhir, hasil evaluasi untuk setiap *sample* kalimat umpan balik yang dikembangkan model memperlihatkan hasil seperti pada Tabel 4.2 berikut:

Tabel 4.2 Evaluasi BLEU Score

Rubrik	Precision	Brevity Penalty	BLEU Score
Konten Analisis Positif	0.14	0.5	0.06
Konten Analisis Negatif	0.21	0.56	0.11
Struktur dan Organisasi Positif	0.22	0.61	0.13
Struktur dan Organisasi Negatif	0.13	0.69	0.07
Kesalahan Ejaan Positif	0.22	0.62	0.13
Kesalahan Ejaan Negatif	0.21	0.59	0.11
Relevansi dengan Konten Positif	0.20	0.72	0.14
Relevansi dengan Konten Negatif	0.15	0.62	0.09

Berdasarkan hasil evaluasi, skor BLEU untuk setiap kalimat umpan balik bernilai kurang dari 0.3. BLEU skor dianggap cukup baik jika nilai yang diperoleh adalah sama dengan atau lebih dari 0.3.

Oleh karena itu dapat disimpulkan bahwa sistem umpan balik telah berhasil diimplementasikan ke AES berbahasa Indonesia menggunakan algoritma *Long-Short Term Memory* (LSTM) namun, kalimat umpan balik yang dihasilkan belum cukup baik.

Daftar Rujukan

- [1] A. Amalia, D. Gunawan, Y. Fithri, and I. Aulia, "Automated Bahasa Indonesia essay evaluation with latent semantic analysis," in *Journal of Physics: Conference Series*, Institute of Physics Publishing, Jul. 2019. doi: 10.1088/1742-6596/1235/1/012100.
- [2] F. Rahutomo *et al.*, "Open Problems in Indonesian Automatic Essay Scoring System," *International Journal of Engineering & Technology*, pp. 156–160, 2018, doi: 10.17632/6gp8m72s9p.1.
- [3] V. V. Ramalingam, A. Pandian, P. Chetry, and H. Nigam, "Automated Essay Grading using Machine Learning Algorithm," in *Journal of Physics: Conference Series*, Institute of Physics Publishing, Apr. 2018. doi: 10.1088/1742-6596/1000/1/012030.
- [4] R. A. Rajagede, "Improving Automatic Essay Scoring for Indonesian Language using Simpler Model and Richer

-
- Feature,” *Kinetik: Game Technology, Information System, Computer Network, Computing, Electronics, and Control*, pp. 11–18, Feb. 2021, doi: 10.22219/kinetik.v6i1.1196.
- [5] A. V. Y. Lee, A. C. Luco, and S. C. Tan, “A Human-Centric Automated Essay Scoring and Feedback System for the Development of Ethical Reasoning,” *Educational Technology and Society*, vol. 26, no. 1, pp. 147–159, 2023, doi: 10.30191/ETS.202301_26(1).0011.
- [6] C. Lu and M. Cutumisu, “Integrating Deep Learning into An Automated Feedback Generation System for Automated Essay Scoring,” 2021. [Online]. Available: <https://educationaldatamining.org/edm2021/>
- [7] P. Philip and S. Minhas, “A Brief Survey on Natural Language Processing Based Text Generation and Evaluation Techniques,” *VFAST Transactions on Software Engineering*, vol. 10, 2022.
- [8] S. Santhanam, “Context based Text-generation using LSTM networks,” Apr. 2020, [Online]. Available: <http://arxiv.org/abs/2005.00048>
- [9] A. Biswal, “Power of Recurrent Neural Networks (RNN): Revolutionizing AI.” Accessed: Apr. 12, 2024. [Online]. Available: <https://www.simplilearn.com/tutorials/deep-learning-tutorial/rnn>
- [10] M. A. Hussein, H. A. Hassan, and M. Nassef, “A Trait-based Deep Learning Automated Essay Scoring System with Adaptive Feedback,” 2020. [Online]. Available: www.ijacsa.thesai.org
- [11] H. Patel, “What is Feature Engineering — Importance, Tools and Techniques for Machine Learning.” Accessed: Apr. 14, 2024. [Online]. Available: <https://towardsdatascience.com/what-is-feature-engineering-importance-tools-and-techniques-for-machine-learning-2080b0269f10>
-