



# Identifikasi Penggunaan Chat GPT Pada Esai TOEFL Menggunakan Metode Long Short Term Memory

Karina Natasya Darmawan\*, Silvester Dian Handy Permana, Ketut Bayu Yogha Bintoro

Program Studi Teknik Informatika, Fakultas Sains, Teknik, dan Desain, Universitas Trilogi, Jakarta, Indonesia

Email: <sup>1</sup>karina.natasya29@gmail.com, <sup>2</sup>handy@trilogi.ac.id, <sup>3</sup>ketutbayu@trilogi.ac.id

Email Penulis Korespondensi: karina.natasya29@gmail.com

**Abstrak**—Penggunaan teknologi kecerdasan buatan (AI) semakin meningkat seiring dengan perkembangan teknologi. Teknologi AI yang sering digunakan berupa aplikasi *Chat GPT (Generative Pre-trained Transformers)*. *Chat GPT* merupakan suatu aplikasi yang digunakan oleh masyarakat sebagai sumber informasi, penyusunan suatu kalimat, membuat karya tulis, serta menjawab pertanyaan esai TOEFL. Penggunaan yang berlebihan dapat menyebabkan masyarakat kehilangan kreativitas dan jati diri karena tidak memahami konteks dan terlalu mengandalkan hasil teks AI tersebut sehingga menimbulkan risiko akademis. Pengajar pun mengalami kesulitan untuk membedakan tulisan teks AI dan manusia. Oleh karena itu, penelitian ini digunakan untuk mengidentifikasi apakah esai TOEFL merupakan hasil teks manusia atau GPT. Penelitian ini memiliki tujuan untuk membantu pengajar dalam mengidentifikasi esai TOEFL apakah hasil teks manusia atau GPT. Penelitian menggunakan metode *Long Short Term Memory (LSTM)* untuk mengidentifikasi penggunaan GPT pada esai TOEFL. Terdapat 3 konfigurasi pembagian data pada metode LSTM untuk mendapatkan hasil terbaik. Data penelitian berupa 2 kumpulan data esai TOEFL dengan perintah esai yang sama serta memiliki jumlah sampel sebesar 220 sampel data. Metode LSTM merupakan modifikasi dari algoritma *Recurrent Neural Network (RNN)* dan termasuk ke dalam metode *Deep Learning*. Metode LSTM melibatkan sel memori yang dikontrol oleh tiga gerbang, yaitu *input gate*, *forgot gate*, dan *output gate* serta *hidden state*. Gerbang berfungsi untuk memutuskan dan mengontrol informasi yang ditambahkan, dihapus, dan dikeluarkan dari sel memori. Hasil penelitian merupakan suatu sistem yang dapat membantu pengajar untuk mendeteksi penggunaan GPT pada esai TOEFL. Penelitian berhasil mengidentifikasi penggunaan GPT pada esai TOEFL dalam konfigurasi pembagian data 70:30 dengan skor *loss* sebesar 25,07%, skor akurasi 89,83%, dan skor prediksi 64,32%. Oleh karena itu, diharapkan sistem dari penelitian dapat membantu pengajar dalam identifikasi penggunaan GPT serta memudahkan penilaian esai TOEFL.

**Kata Kunci:** Identifikasi; Chat GPT; LSTM; Manusia; TOEFL

**Abstract**—The use of Artificial Intelligent (AI) technology is increasing along with technological developments. One of the technologies that is often used is *Chat GPT (Generative Pre-trained Transformers)*. *Chat GPT* is an application used for many things such as source of information, write an essay, and answer TOEFL essay questions. Because of its easiness, people will excessively use this that can cause people to lose creativity because they do not understand the material context and rely too much on the AI text result, which poses academic risks. Teachers also have difficulty to distinguish between AI and human text writing. Therefore, this research is to identify whether TOEFL essay are result of human text or GPT. This research used the *Long Short Term Memory (LSTM)* method to identify the use of GPT in TOEFL essay. This research also used 3 different split data configurations to find the best results. This research consists of 2 TOEFL essay datasets with the same prompt and has total of 220 data samples. The LSTM method is a modification of algorithm *Recurrent Neural Network (RNN)* and part of *Deep Learning*. The LSTM method involves memory cell controlled by three gates, such as *input gate*, *forgot fate*, *output gate*, and the *hidden state*. The gates are used to decide and control the information added, deleted, and removed from memory cell. The results of this research is a system that can help teachers detect the use of GPT in TOEFL essay. This research successfully identified the use of GPT in TOEFL essay in a 70:30 data split configuration with a *loss* score of 25,07%, accuracy score of 89,83%, and prediction score of 64,32%. Therefore, it is hoped that this system can help teachers identify the use of GPT and facilitate the assessment of TOEFL essay.

**Keywords:** Identification; Chat GPT; LSTM; Human; TOEFL

## 1. PENDAHULUAN

Teknologi kecerdasan buatan (AI) sudah menjadi salah satu teknologi yang digunakan oleh masyarakat dalam kehidupan sehari-hari, salah satunya adalah aplikasi *Chat GPT (Generative Pre-trained Transformers)* yang merupakan suatu model berbasis jaringan saraf transformator yang berdasarkan pada pengetahuan yang telah terlatih sebelumnya dan bersifat generatif. Aplikasi yang dapat digunakan untuk penerjemahan bahasa, menghasilkan materi pembelajaran, pembuatan karya tulis, dan esai TOEFL. *Chat GPT* merupakan aplikasi yang mudah digunakan dan tidak dapat diberhentikan penggunaannya oleh masyarakat. Aplikasi *Chat GPT* digunakan oleh hampir seluruh masyarakat mulai dari pelajar sampai orang tua. Penggunaan *Chat GPT* pun sering disalahgunakan untuk membantu dalam mengerjakan soal ujian untuk membantu pelajar mencapai nilai yang memuaskan. Penggunaan yang berlebihan tersebut yang dapat menurunkan kemampuan dalam mempelajari keterampilan yang diperlukan untuk menyelesaikan tugas secara efektif [1].

Dari permasalahan tersebut dapat menimbulkan sesuatu kekhawatiran terhadap integritas akademik. Penelitian sebelumnya mengungkapkan beberapa dampak dari penggunaan *Chat GPT* terhadap pengajar, yaitu Penelitian menyatakan bahwa pengajar tidak dapat membedakan secara pasti antara teks AI dan manusia [2] dan kualitas teks AI yang sudah semakin tinggi kualitasnya sehingga meningkatkan kesulitan untuk identifikasi teks AI [3]. Oleh karena itu pentingnya bagi pengajar untuk mengetahui apakah esai TOEFL mengandung teks dari AI atau tidak sehingga pengajar dapat memberikan penilaian yang sesuai.

Beberapa penelitian sebelumnya sudah ada yang membahas mengenai identifikasi teks GPT dan manusia dengan beberapa metode yang berbeda. Penelitian sebelumnya berhasil mengidentifikasi menggunakan analisis sidik jari linguistik dan model *Random Forest* dengan tingkat akurasi sebesar 97.8% serta menyatakan bahwa teknik *Natural*



*Language Processing* (NLP) dapat secara efektif mendeteksi esai yang ditulis oleh *Chat GPT* [4]. Penelitian ini pun mengungkapkan bahwa penelitian dilakukan untuk mencegah penyalahgunaan AI oleh pelajar karena tidak adanya karakteristik dari tulisan GPT yang dihasilkan. Penelitian lainnya pun ada yang membahas mengenai identifikasi GPT menggunakan CNN dan LSTM dan menemukan bahwa model LSTM memiliki nilai presisi yang lebih tinggi dibandingkan dengan model CNN [5]. Oleh karena itu, penelitian yang dirancang menggunakan teknik NLP serta metode LSTM untuk identifikasi penggunaan GPT pada esai TOEFL.

Pada penelitian sebelumnya membahas mengenai identifikasi teks AI dan manusia menggunakan metode *DistilBERT* dan LSTM yang mendapatkan nilai akurasi sebesar 99.8% [6]. Penelitian ini menggunakan kumpulan data gabungan antara teks manusia dan AI yang didapatkan dari halaman web *Kaggle*. Kumpulan data yang digunakan tersebut tidak dapat dipastikan apakah teks tersebut merupakan teks asli buatan manusia atau AI. Hal ini memicu peneliti untuk menggunakan kumpulan data yang dapat terbukti bahwa teks tersebut merupakan teks asli manusia dan teks AI. Dengan memvalidasi kumpulan data tersebut, penelitian ini dapat memastikan bahwa hasil penelitian sesuai dengan harapan peneliti, yaitu dapat dengan sesuai mengidentifikasi tulisan buatan manusia dan tulisan hasil AI.

Penelitian sebelumnya juga sudah ada yang membahas mengenai menggunakan suatu perintah untuk membedakan teks manusia dan GPT dan mendapatkan tingkat akurasi sebesar 0.991% [7]. Penelitian ini menggunakan perintah yang berbeda untuk membandingkan hasil esai yang dituliskan manusia dan GPT. Hasil penelitian menemukan bahwa esai buatan manusia memiliki kosakata yang lebih luas dan bervariasi sedangkan esai buatan GPT mempunyai pola bahasa yang konsisten dan berisi lebih banyak paragraf. Oleh karena itu, penelitian yang dirancang menggunakan perintah yang sama untuk esai buatan manusia dan GPT serta menentukan batasan kata yang akan di proses sehingga dapat menerapkan standar kumpulan data yang digunakan.

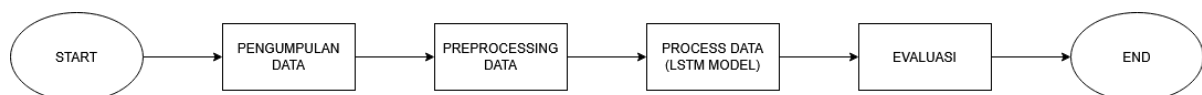
Berdasarkan penelitian – penelitian sebelumnya yang sudah dijelaskan, penelitian yang dirancang memiliki perbedaan yaitu hanya menggunakan metode LSTM yang disertai oleh metode *FastText* sebagai metode *embedding* untuk mengidentifikasi penggunaan *Chat GPT* pada esai TOEFL. Metode *FastText* digunakan karena menggunakan pendekatan *sub-word* dalam pembuatan vektor yang merepresentasikan tiap kata sehingga memberikan fleksibilitas lebih. Dalam metode *FastText*, setiap kata direpresentasikan sebagai rata-rata representasi vektor karakter n-gram beserta kata itu sendiri.

Penelitian ini menetapkan beberapa batasan untuk memastikan penelitian tetap terarah, memiliki standar yang jelas, dan fokus pada permasalahan yang ingin diselesaikan, sehingga tidak menyimpang dari tujuan yang diinginkan. Batasan pertama pada penelitian ini adalah menentukan batasan kata sebesar 500 kata karena perkiraan kata pada esai TOEFL adalah sebesar 300 – 400 kata untuk TOEFL *integrated* dan *independent* [8] sehingga penelitian menambahkan jumlah kata menjadi 500 kata untuk memaksimalkan perkiraan kata tersebut. Batasan kedua adalah bahan penelitian menggunakan esai TOEFL yang menggunakan Bahasa Inggris. Batasan ketiga adalah penelitian hanya dapat mendeteksi berkas yang memiliki ekstensi *.docx*. Batasan terakhir adalah versi *Chat GPT* yang digunakan pada penelitian ini adalah versi GPT-4. Dengan demikian, batasan ini yang menentukan standar bahan penelitian yang digunakan pada penelitian ini.

Berdasarkan uraian di atas, penelitian mempunyai rumusan masalah, yaitu: Bagaimana mengidentifikasi penggunaan *Chat GPT* pada esai TOEFL dengan menggunakan metode *Long Short Term Memory* (LSTM). Penelitian ini dilakukan sesuai dengan rumusan masalah dan batasan tersebut. Oleh karena itu, penelitian ini bertujuan untuk mengidentifikasi penggunaan *Chat GPT* pada esai TOEFL menggunakan metode *Long Short Term Memory* (LSTM). Dengan harapan, penelitian berhasil mengidentifikasi penggunaan GPT dengan nilai akurasi yang tinggi. Hasil penelitian ini diharapkan dapat membantu pengajar dalam identifikasi penggunaan GPT serta dapat memberikan referensi dan kontribusi terkait identifikasi GPT pada esai TOEFL serta terhadap teknologi LSTM.

## 2. METODOLOGI PENELITIAN

Penelitian ini menggunakan metode *Long Short Term Memory* (LSTM) untuk mengidentifikasi penggunaan GPT (*Generative Pre-trained Transformers*) pada esai TOEFL. Langkah – langkah penelitian dapat terlihat pada Gambar 1.



**Gambar 1.** Tahapan Metodologi Penelitian

Gambar 1 menunjukkan tahapan penelitian yang akan dilakukan. Tahapan ini dimulai dengan pengumpulan data esai TOEFL untuk kumpulan data GPT dan manusia yang akan digunakan sebagai bahan penelitian. Tahapan selanjutnya adalah proses *preprocessing* data yang akan membersihkan dan memisahkan data. Tahapan selanjutnya adalah proses data atau implementasi model LSTM yang akan mengonfigurasi dan melatih data. Tahapan akhir adalah proses evaluasi data yang akan menguji performa model serta memprediksi dokumen baru.

### 2.1 Pengumpulan Data

Tahapan ini merupakan tahapan yang dilakukan untuk mengumpulkan data. Pengumpulan data didapatkan berdasarkan pernyataan esai TOEFL dari situs web *Goodreads* pada tahun 2004 [9]. Proses ini memiliki jumlah sampel sebesar 220



data dari 2 kumpulan data, yaitu kumpulan data GPT dengan total sampel sebesar 110 data dan kumpulan data manusia sebesar 110 data. Kumpulan data tersebut didapatkan dengan cara manual, yaitu mengekstrak berdasarkan berkas ataupun dari situs web *chatgpt*. Kumpulan data manusia dapat dipastikan tidak mengandung AI karena menggunakan berkas pada tahun 2004 dan 2018. Sedangkan, GPT merilis demo awal pada tahun 2022. Paragraf dari setiap kumpulan data tersebut akan diberikan label 1 untuk paragraf GPT dan label 0 untuk paragraf manusia.

Di tahapan ini, penelitian pun akan melakukan filter pada kedua kumpulan data tersebut sehingga tidak akan memproses kumpulan data yang memiliki berkas lebih dari 500 kata. Oleh karena itu, penelitian hanya akan memproses kumpulan data yang memiliki jumlah data kurang dari 500 kata. Penelitian memberikan batasan sebesar 500 kata karena perkiraan kata esai TOEFL adalah sebesar 300 – 400 kata [10] sehingga penelitian ini memaksimalkan jumlah kata tersebut menjadi 500 kata. Dapat dilihat pada Tabel 1 mengenai contoh sampel data esai TOEFL yang digunakan pada penelitian ini. Tabel tersebut menunjukkan data perintah yang digunakan pada salah satu contoh sampel beserta kalimat dari kumpulan data manusia dan GPT.

**Tabel 1.** Contoh Sampel Data Esai TOEFL

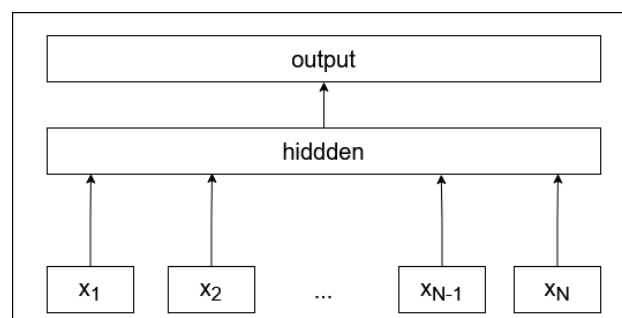
Kumpulan Data	<i>Why do you think people attend college or university? Use specific reasons and examples to support your answer</i>
Chat GPT (Label 1)	<i>There are many reasons why people choose to attend college or university. Some of the most common reasons include the desire to gain new knowledge and skills, the opportunity to pursue a career in a particular field, and the chance to earn a higher salary. ...</i>
Manusia (Label 0)	<i>College is a place that the students can learn more and new knowledge and experience in it. Of course, different people have different reason to study in college. For example, some people want to be to go on a further study after they graduate from the college ...</i>

## 2.2 Preprocessing Data

Tahapan ini merupakan tahapan persiapan data sebelum mengimplementasikan ke dalam model LSTM. Proses ini dilakukan untuk memastikan data ter-standarisasi serta melakukan pembagian data menjadi data latih dan data uji. Tahapan ini melalui beberapa proses, yaitu proses pembersihan data, tokenisasi, *padding*, memuat *FastText embedding*, dan pembagian data. Proses pembersihan data meliputi penghapusan paragraf kosong, *case folding*, perubahan dan penghapusan label HTML dan URL, dan penghapusan tanda baca, serta penghapusan spasi ganda. Proses pembersihan data tersebut dilakukan untuk memastikan bahwa data yang akan diproses sudah bersih.

Proses tokenisasi adalah proses mengubah teks menjadi unit yang lebih kecil disebut token. Hasil dari proses tokenisasi merupakan suatu urutan angka (*sequence*) yang merepresentasikan teks dalam bentuk urutan angka berdasarkan dari kamus kata. Proses *padding* adalah proses menambahkan angka nol ke dalam urutan angka baik di awal ataupun di akhir urutan. Penelitian ini menggunakan penambahan angka nol di awal urutan (*pre-padding*) sehingga semua urutan angka memiliki panjang yang sama berdasarkan panjang maksimum urutan kata dalam data [11].

Proses selanjutnya adalah *word embedding* merupakan suatu teknik pemetaan kata menjadi vektor numerik sehingga dapat merepresentasikan kata dengan hubungan semantik dan informasi kontekstualnya [12]. Penelitian menggunakan metode *FastText* sebagai metode *embedding*. Metode *FastText* adalah metode yang mempresentasikan kata sebagai kombinasi unit bagian kata sehingga dapat menangani kata jarang atau kata yang tidak pernah muncul di kamus [12], [13]. Pada Gambar 2 merupakan arsitektur model *FastText*. Proses *Fasttext embedding* menggunakan vektor kata yang sudah dilatih sebelumnya dengan vektor kata sebesar 2 juta vektor kata yang dilatih dengan informasi pada *Common Crawl* dengan total 600B token serta dimensi sebesar 300 [14]. Dapat terlihat pada Gambar 2 mengenai arsitektur model *FastText*.



**Gambar 2.** Arsitektur Model *FastText*

Proses pembagian data (*split data*) merupakan proses untuk memisahkan kumpulan data menjadi data pelatihan dan data pengujian. Proses pembagian data merupakan proses penting untuk membantu dalam proses melatih data, mengonfigurasi parameter dan menilai performa model. Data pelatihan (*training*) adalah data yang digunakan untuk melatih dan mengembangkan model. Sedangkan data pengujian (*testing*) adalah data yang digunakan setelah pelatihan model untuk melihat kinerja model. Penelitian ini menggunakan metode pengambilan sampel acak berstrata (*stratified random sampling*) yaitu dengan cara memilih sampel data secara acak berdasarkan parameter tertentu. Metode akan



memastikan data terdistribusikan dengan benar saat pembagian data latih dan data uji [15]. Penelitian menggunakan beberapa konfigurasi pembagian data, yaitu 80:20, 70:30, 60:40 untuk mendapatkan hasil yang terbaik. Konfigurasi yang digunakan pada penelitian ini dapat dilihat pada Tabel 2.

**Tabel 2.** Konfigurasi Pembagian Data

No	Data Latih	Data Uji
1	80%	20%
2	70%	30%
3	60%	40%

### 2.3 Proses Data

Tahapan ini merupakan proses implementasi model LSTM yang menggunakan beberapa lapisan. Lapisan yang digunakan pada penelitian ini adalah lapisan *embedding*, LSTM, *dense*, *dropout*, dan *output*. Penelitian menggunakan model *sequential* sehingga setiap lapisan akan ditambahkan secara berurutan dan hanya memiliki satu *input* dan satu *output*. Lapisan *embedding* merupakan lapisan yang mengubah data *input* menjadi vektor numerik.

Metode *Long Short Term Memory* (LSTM) merupakan suatu metode pengembangan dari *Recurrent Neural Network* yang dikembangkan pada tahun 1991 oleh *Sepp Hochreiter* dan *Jurgen Schmidhuber* yang memiliki kemampuan untuk menangani dependensi jangka panjang [16]. Metode LSTM memiliki sel memori (*cell state*) dan 3 jenis gerbang, yaitu gerbang masukan (*input gate*), gerbang lupa (*forgot gate*), gerbang keluaran (*output gate*). Gerbang masukan merupakan gerbang yang berfungsi untuk menambahkan informasi penting ke dalam sel memori. Gerbang lupa merupakan gerbang yang berfungsi untuk menghapus informasi yang tidak lagi diperlukan dari sel memori. Sel memori akan diperbarui dengan cara mempertahankan informasi lama dan menambahkan informasi baru. Gerbang keluaran merupakan gerbang yang berfungsi untuk memberikan keluaran berdasarkan dari sel memori saat ini. Gerbang keluaran adalah gerbang yang akan menentukan *hidden state* yang akan dikirimkan ke sel selanjutnya [16], [17], [18].

Lapisan *dropout* merupakan lapisan yang digunakan untuk mengurangi *overfitting* pada model dengan cara menonaktifkan beberapa masukan berdasarkan dari rasio *dropout* yang diberikan [19]. Lapisan *dense* merupakan lapisan yang terhubung secara keseluruhan (penuh) dan digunakan untuk mempelajari pola dan hubungan yang kompleks dalam data [19]. Lapisan *output* menggunakan lapisan *dense* yang menghasilkan nilai berkisar antara 0 dan 1. Penelitian ini menggunakan fungsi pengaktifan *Sigmoid* untuk klasifikasi biner.

Terdapat beberapa konfigurasi parameter yang digunakan pada penelitian ini, yaitu fungsi aktivasi, pengoptimalan, fungsi *loss*, *epoch*, dan *batch*. Penelitian ini menggunakan metode *early stopping* untuk memberhentikan model jika tidak menemukan adanya peningkatan signifikan pada saat pelatihan model [20].

### 2.4 Evaluasi Data

Tahapan ini merupakan proses evaluasi model LSTM untuk melihat performa kinerja model. Tahapan ini akan menguji menggunakan data uji serta memprediksi pada dokumen baru. Proses evaluasi akan mendapatkan *confusion matrix*, nilai akurasi, presisi, *recall*, F1 dan skor prediksi. Hasil proses ini yang akan menentukan apakah model berhasil atau gagal dalam memprediksi penggunaan GPT pada esai TOEFL.

*Confusion matrix* digunakan untuk menunjukkan visualisasi kinerja model berdasarkan nilai aktual dan nilai prediksi sehingga membantu mengidentifikasi jumlah klasifikasi data. Matriks memiliki 4 kriteria, yaitu TP (*True Positive*) merepresentasikan nilai aktual dan prediksi bernilai positif, TN (*True Negative*) merepresentasikan nilai aktual dan prediksi bernilai negatif, FP (*False Positive*) yang merepresentasikan nilai aktual negatif tetapi nilai prediksi positif, dan FN (*False Negative*) yang merepresentasikan nilai aktual positif tetapi nilai prediksi negatif [21], [22]. Hasil matriks dapat dilihat pada Gambar 3.

		Nilai Aktual	
		Positif	Negatif
Nilai Prediksi	Positif	TP	FP
	Negatif	FN	TN

**Gambar 3.** Confusion Matrix



$$Accuracy = \frac{TP+TN}{TP+TN+FP+FN} \quad (1)$$

$$Precision = \frac{TP}{TP+FP} \quad (2)$$

$$Recall = \frac{TP}{TP+FN} \quad (3)$$

$$F1_{score} = 2 \times \frac{precision \times recall}{precision + recall} \quad (4)$$

Pada persamaan 1 dapat terlihat rumus untuk mendapatkan nilai akurasi. Nilai akurasi adalah nilai yang merepresentasikan seberapa sering keakuratan model dalam memprediksi. Pada persamaan 2 merupakan rumus untuk mendapatkan nilai presisi. Nilai presisi adalah nilai yang merepresentasikan seberapa akurat model dalam memprediksi kelas positif. Pada persamaan 3 merupakan rumus untuk mendapatkan nilai *recall*, yang menunjukkan seberapa baik model dalam menemukan semua data positif yang sebenarnya. Pada persamaan 4 merupakan nilai *F1*, yang mencerminkan keseimbangan antara nilai presisi dan *recall* [22], [23].

Penelitian ini juga melakukan proses prediksi model menggunakan dokumen baru. Proses ini akan memberikan skor prediksi berkisar 0 dan 1. Skor prediksi tersebut yang akan menentukan label yang digunakan apakah paragraf dokumen tersebut merupakan hasil dari GPT atau manusia. Label 0 menandakan bahwa paragraf tersebut teridentifikasi buatan manusia. Sedangkan, label 1 menandakan bahwa paragraf tersebut buatan dari GPT.

### 3. HASIL DAN PEMBAHASAN

#### 3.1 Kumpulan Data

Penelitian ini mengumpulkan 2 jenis kumpulan data, yaitu kumpulan data manusia dan kumpulan data *Generative Pre-trained Transformers* (GPT) yang merupakan sebuah data esai TOEFL dengan perintah esai. Penelitian menggunakan buku *Answers to All TOEFL Essay Questions* dari situs web *Goodreads* yang dipublikasi pada tahun 2004 [9] dan *An Essay Collection for TOEFL* pada tahun 2018 [24] untuk mendapatkan kumpulan data manusia dengan cara menyalin teks tersebut ke dalam dokumen *word*. Sedangkan, untuk kumpulan data GPT didapatkan dengan cara memasukkan perintah esai TOEFL yang sama ke dalam situs web *ChatGPT* lalu disalin ke dalam dokumen *word*. Penelitian menggunakan rumus *cochran* untuk menentukan ukuran sampel minimum yang dibutuhkan penelitian [25]. Berdasarkan perhitungan rumus *cochran* maka jumlah sampel yang dibutuhkan adalah 96 data. Oleh karena itu, penelitian menggunakan jumlah sampel 220 untuk kedua kumpulan data tersebut. Penelitian sengaja meningkatkan jumlah sampel per kumpulan data menjadi 110 data untuk mengurangi risiko kekurangan data pada saat proses pembersihan data. Penelitian akan melakukan proses filter data pada kumpulan data sehingga tidak akan menyertakan dokumen dengan jumlah kata yang lebih dari 500 kata. Penelitian mendapatkan 575 paragraf dari kumpulan data GPT dan 470 paragraf dari kumpulan data manusia. Paragraf yang ditemukan pada kumpulan data tersebut masih memiliki paragraf kosong yang akan diproses pada proses pembersihan data. Penelitian pun menambahkan label 1 pada paragraf GPT dan label 0 pada paragraf manusia. Penelitian akan menggabungkan kedua jenis paragraf tersebut ke dalam satu kumpulan data untuk memudahkan pada saat proses selanjutnya. Pada Tabel 3 merupakan suatu deskripsi mengenai paragraf yang digunakan, seperti jenis data, label dan teks paragraf yang digunakan pada penelitian. Dari tabel tersebut dapat terlihat, bahwa penelitian ini menemukan suatu paragraf kosong yang akan di proses pada tahapan selanjutnya.

**Tabel 3.** Label Paragraf Teks

Kumpulan Data	Label	Teks Paragraf
GPT	1	<i>There are many reasons why people choose to attend college or ...</i>
GPT	1	<i>Another problem is the issue of overload. With so much information available online, it can be overwhelming ...</i>
GPT	1	
Manusia	0	<i>Nowadays, a lot of teens work in jobs while they are studying. Some of them really need the money they earn to ...</i>
Manusia	0	<i>The information technology is developing so rapidly that nearly all families in Hong Kong have at least one computer. Not only can ...</i>

#### 3.2 Preprocessing Data

##### a. Pembersihan Data

Terdapat beberapa proses yang dilakukan pada tahapan ini, yaitu penghapusan paragraf kosong, *case folding*, perubahan dan penghapusan label HTML dan URL, penghapusan tanda baca, dan penghapusan spasi ganda. Penghapusan paragraf kosong untuk memastikan penelitian hanya memproses paragraf yang memiliki isi teks paragraf saja. Proses lainnya pada tahapan ini dilakukan sehingga semua paragraf memiliki standar format penulisan yang sama. Hasil dari setiap proses pembersihan data dapat dilihat pada Tabel 4.



**Tabel 4.** Proses Pembersihan data

Proses	Teks sebelumnya	Hasil akhir
Case folding	<i>In my opinion, higher education should be available to &lt;strong&gt;all&lt;/strong&gt; students. &lt;br&gt; Why do I think so? It's because ...</i>	<i>in my opinion, higher education should be available to &lt;strong&gt;all&lt;/strong&gt; students. &lt;br&gt; why do i think so? it's because ...</i>
Penghapusan dan Perubahan label HTML dan URL	<i>in my opinion, higher education should be available to &lt;strong&gt;all&lt;/strong&gt; students. &lt;br&gt; why do i think so? it's because ...</i>	<i>in my opinion, higher education should be available to all students. why do i think so? it's because ...</i>
Penghapusan tanda baca	<i>in my opinion, higher education should be available to all students. why do i think so? it's because ...</i>	<i>in my opinion, higher education should be available to all students. why do i think so it's because ...</i>
Penghapusan spasi ganda	<i>in my opinion, higher education should be available to all students. why do i think so it's because ...</i>	<i>in my opinion, higher education should be available to all students. why do i think so it's because ...</i>

b. Tokenisasi & Padding

Proses tokenisasi adalah proses untuk mengubah teks paragraf menjadi suatu urutan indeks integer. Proses akan mempelajari keseluruhan teks paragraf sehingga dapat memberikan indeks pada kata tersebut, dengan indeks terkecil menandakan frekuensi kata yang sering ditemukan. Proses *padding* adalah proses penambahan angka 0 pada indeks integer, sehingga setiap urutan memiliki panjang maksimum urutan yang sama. Proses ini dilakukan untuk menjaga ukuran sehingga konsisten serta mempermudah dalam pemrosesan, seperti proses *batch*. Contoh proses tokenisasi dapat dilihat pada Tabel 5.

**Tabel 5.** Contoh Proses Tokenisasi

Proses Tokenisasi	Hasil Tokenisasi
Contoh kalimat	<i>This is sentence</i> <i>Another sentence</i>
Tokenisasi	[ <i>this, is, sentence</i> ] [ <i>another, sentence</i> ]
Membuat kamus kata (word index)	{ <i>sentence: 1, this: 2, is: 3, another: 4</i> }
Konversi ke urutan angka	<i>This is sentence</i> → [2, 3, 1] <i>Another sentence</i> → [4, 1]
Pre-padding	[2, 3, 1] → [2, 3, 1] [4, 1] → [0, 4, 1]

c. Memuat *FastText Embedding*

Proses ini adalah proses untuk memuat atau mengakses berkas vektor kata yang sudah diunduh pada situs web *fasttext.cc*. Berkas yang digunakan memiliki vektor kata sebesar 2 juta vektor kata yang telah dilatih dengan informasi bagian kata pada *Common Crawl* dengan total 600B token serta dimensi sebesar 300 [14]. Proses ini akan membuat suatu matriks yang menyimpan vektor *embedding* dengan semua kosakata dengan baris sesuai dengan indeks dan kolom sesuai dengan dimensi *embedding*. Dalam proses di penelitian ini akan memastikan bahwa setiap kata dalam kamus kata memiliki matriks *embedding* baik itu dari vektor *Fasttext* ataupun vektor acak. Metode *FastText* adalah proses tokenisasi bukan hanya berdasarkan kata tetapi juga berdasarkan karakter n-gram kata. Contoh proses *fasttext* dapat dilihat pada Tabel 6 dengan contoh kata *eating*.

**Tabel 6.** Contoh Proses *FastText*

Proses <i>FastText</i>	Hasil <i>FastText</i>
Contoh kata	<i>eating</i>
Pembentukan <i>subword</i> (n-grams)	N = 3 [ <i>“ea”, “ear”, “ati”, “tin”, “ing”, “ng”</i> ]
Representasi vector kata	[ <i>-9.700e-03 -8.990e-02 ... 4.110e-02 2.440e-02</i> ]
Representasi vector <i>subword</i>	<i>Subword: creating, Vector: [ 9.000e-03 -6.970e-02 ... -5.220e-02]</i> <i>Subword: eating, Vector: [-9.700e-03 -8.990e-02 ... 2.440e-02]</i>
Kemiripan kata	<i>eat: 0.7600497603416443</i> <i>munching: 0.7471524477005005</i>

d. Pembagian Data

Proses ini adalah proses pembagian data menjadi data latih dan data uji menggunakan metode pengambilan sampel acak berstrata (*stratified random sampling*). Metode ini digunakan untuk memastikan bahwa data terdistribusi dengan



benar serta mewakili setiap kelas secara proporsional. Penelitian ini menggunakan 3 konfigurasi pembagian data dari total 1015 total paragraf. Konfigurasi pertama adalah 80:20 dengan data latih sebesar 812 dan data uji sebesar 203. Konfigurasi kedua adalah 70:30 dengan data latih sebesar 710 dan data uji sebesar 305. Konfigurasi ketiga adalah 60:40 dengan data latih sebesar 609 dan data uji sebesar 406.

Contoh sampel data latih yang didapatkan pada pembagian data tersebut adalah [ 0 0 0 0 ... 1174 2486 1139]. Sedangkan, untuk contoh salah satu label data latih yang digunakan adalah 1, yang menandakan bahwa data latih tersebut berasal dari data GPT. Hal ini menandakan bahwa salah satu sampel yang ada pada data latih merupakan sampel dari data GPT yang memiliki label 1.

### 3.3 Model LSTM

Tahapan ini adalah tahapan implementasi model LSTM dengan tujuan untuk mendefinisikan model, membuat kompilasi model, serta melatih model. Proses model LSTM melalui beberapa lapisan, yaitu lapisan *input*, *embedding*, *LSTM*, *dense*, *dropout*, dan *output*. Proses model LSTM dilakukan menggunakan model *sequential* yang merupakan suatu tumpukan di mana setiap lapisan memiliki tepat satu *input* dan satu *output* [26]. Lapisan pertama adalah lapisan *input* yang digunakan untuk menerima data dan memetakan bentuk data yang diterima serta merupakan lapisan yang tidak melakukan perhitungan [27]. Parameter yang digunakan pada lapisan ini adalah parameter *shape* untuk menentukan dimensi data *input*. Penelitian ini menggunakan nilai dari panjang maksimum pada urutan yang terdapat dalam *sequence* sebagai nilai parameter *shape*. Lapisan kedua adalah lapisan *embedding* untuk mengubah data menjadi suatu vektor numerik. Parameter yang digunakan pada lapisan ini menggunakan konfigurasi dari *Fasttext embedding* seperti ukuran kosakata unik, ukuran vektor *embedding*, dan matriks *embedding*. Pada penelitian ini, parameter *trainable* harus memiliki nilai *false* untuk mempertahankan vektor dan tidak perlu diperbarui. Penelitian ini mendapatkan jumlah *embedding* parameter sebesar 1.489.800 parameter yang tidak dapat dilatih.

Lapisan ketiga adalah lapisan LSTM yang merupakan lapisan inti untuk memproses ukuran data dan menangkap ketergantungan jangka panjang dalam model. Lapisan melalui 4 gerbang, yaitu gerbang *input*, *forget*, *cell*, dan *output* sehingga akan memproses data secara berurutan. Penelitian menggunakan 64 unit pada lapisan LSTM sehingga mendapatkan jumlah parameter sebesar 93.440 parameter yang dapat dilatih.

Lapisan keempat adalah lapisan *dense* yang merupakan lapisan yang terhubung secara keseluruhan. Lapisan di mana setiap neuron di lapisan terhubung ke setiap neuron dalam lapisan sebelumnya. Lapisan ini digunakan untuk mempelajari pola dan hubungan kompleks dalam data. Penelitian ini menggunakan parameter *Rectified Linear Unit Activation* (ReLU) sehingga dapat mengurangi masalah gradien yang hilang serta mampu menghasilkan nilai 0 yang sebenarnya dan mudah dioptimalkan [28]. Penelitian menggunakan 64 unit pada lapisan ini sehingga mendapatkan jumlah parameter sebesar 4.160 parameter yang dapat dilatih.

Lapisan kelima adalah lapisan *dropout* untuk mengurangi *overfitting* yang dapat terjadi pada penelitian ini. Proses ini dilakukan dengan cara menghilangkan neuron sementara dalam jaringan neural sehingga dapat menciptakan jaringan baru dari jaringan induknya. Neuron dihilangkan berdasarkan dari probabilitas yang diberikan. Lapisan ini digunakan untuk mencegah unit untuk beradaptasi sehingga tidak bergantung pada salah satu jenis data serta dapat meningkatkan generalisasi model. Penelitian ini menggunakan nilai probabilitas sebesar 0.2 dan tidak menghasilkan parameter sehingga jumlah parameter sebesar 0.

Lapisan terakhir adalah lapisan *output* yang menggunakan lapisan *dense* dengan fungsi pengaktifan *sigmoid* sehingga menghasilkan nilai berkisar antara 0 dan 1. Hasil keluaran tersebut sesuai dengan probabilitas standar sehingga dapat dikonversi menjadi nilai biner [29]. Penelitian dapat menentukan hasil berdasarkan klasifikasi biner, di mana jika hasil kurang dari sama dengan 0.5 dianggap sebagai 0 (data manusia) sedangkan jika hasil lebih dari sama dengan 0.5 dianggap sebagai 1 (data GPT). Penelitian ini menggunakan unit 1 pada lapisan ini sehingga memiliki parameter sebesar 65 parameter yang dapat dilatih.

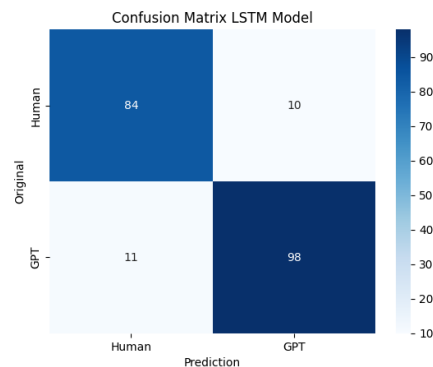
### 3.4 Evaluasi LSTM

Tahapan ini adalah tahapan evaluasi pada pelatihan model LSTM yang sudah dibuat pada tahapan sebelumnya sehingga dapat mengukur seberapa baik model dalam melakukan prediksi. Penelitian ini melakukan proses evaluasi pada data uji dan mendapatkan beberapa nilai seperti nilai *loss*, akurasi, presisi, *recall*, dan *F1* serta mendapatkan *confusion matrix*. Nilai *loss* digunakan untuk melihat seberapa baik model digeneralisasi ke data yang belum pernah dilihat. Nilai akurasi digunakan untuk memeriksa seberapa banyak prediksi model yang benar dibandingkan dengan total prediksi yang dibuat. Nilai presisi digunakan untuk memeriksa seberapa baik model dapat memprediksi nilai positif yang benar dibandingkan dengan semua prediksi yang dianggap positif. Nilai *recall* digunakan untuk memeriksa seberapa baik model dalam memprediksi nilai positif benar dibandingkan dengan semua nilai data positif. Sedangkan, nilai *F1* digunakan untuk melihat nilai keseimbangan antara nilai presisi dan *recall*. *Confusion matrix* akan mengidentifikasi jumlah data yang diprediksi salah ataupun benar serta menunjukkan visualisasi 4 kriteria, yaitu Positif Benar (TP), Negatif Benar (TN), Positif Salah (FP), dan Negatif Salah (FN).

Pada konfigurasi model 80:20 memiliki nilai *loss* sebesar 0.25148603320121765, nilai akurasi sebesar 0.896551724137931, nilai presisi sebesar 0.9074074074074074, nilai *recall* sebesar 0.8990825688073395, dan skor *F1* sebesar 0.9032258064516129. Sedangkan untuk *confusion matrix* pada konfigurasi model ini dapat dilihat pada Gambar 4 yang menunjukkan 4 kriteria. Penelitian ini mendapatkan nilai TP sebesar 98 karena label asli dan prediksi merupakan teks GPT, nilai TN sebesar 84 karena label asli dan prediksi merupakan teks manusia, nilai FP sebesar 10 karena label

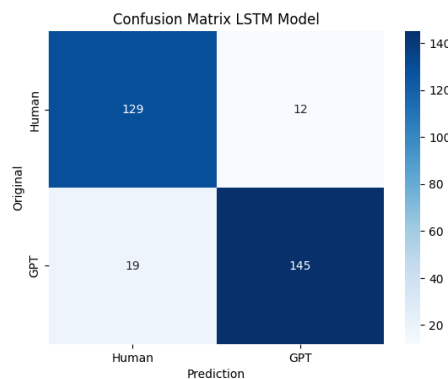


asli merupakan teks manusia tetapi di prediksi menjadi teks GPT, dan nilai FN sebesar 11 karena label asli merupakan teks GPT tetapi di prediksi menjadi teks manusia. Gambar 4 merupakan hasil visualisasi mengenai hasil *confusion matrix* pada konfigurasi model 80:20.



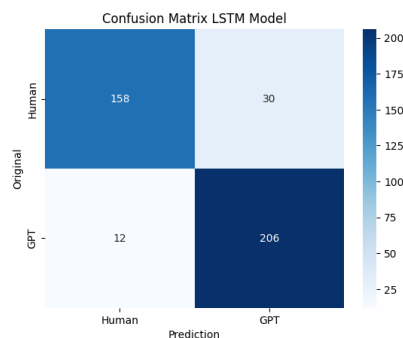
**Gambar 4.** *Confusion Matrix* Model LSTM 80:20

Pada konfigurasi model 70:30 memiliki nilai *loss* sebesar 0.2507959306240082, nilai akurasi sebesar 0.898360655737705, nilai presisi sebesar 0.9235668789808917, nilai *recall* sebesar 0.8841463414634146, dan skor *F1* sebesar 0.8841463414634146. Sedangkan untuk *confusion matrix* pada konfigurasi model ini dapat dilihat pada Gambar 5 yang menunjukkan 4 kriteria. Penelitian ini mendapatkan nilai TP sebesar 145 karena label asli dan prediksi merupakan teks GPT, nilai TN sebesar 129 karena label asli dan prediksi merupakan teks manusia, nilai FP sebesar 12 karena label asli merupakan teks manusia tetapi di prediksi menjadi teks GPT, dan nilai FN sebesar 19 karena label asli merupakan teks GPT tetapi di prediksi menjadi teks manusia. Gambar 5 merupakan hasil visualisasi mengenai hasil *confusion matrix* pada konfigurasi model 70:30.



**Gambar 5.** *Confusion Matrix* Model LSTM 70:30

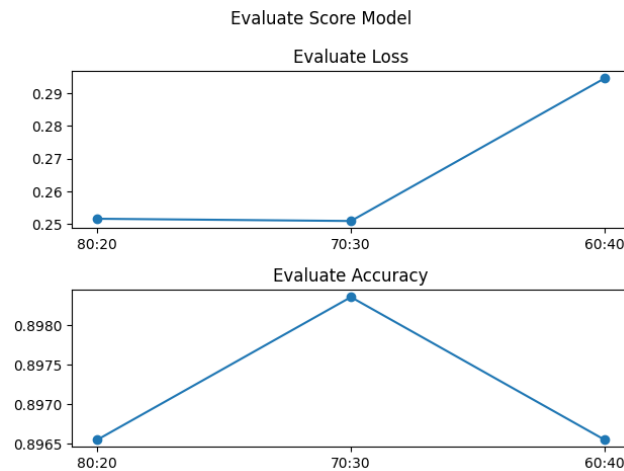
Pada konfigurasi model 60:40 memiliki nilai *loss* sebesar 0.29463595151901245, nilai akurasi sebesar 0.896551724137931, nilai presisi sebesar 0.8728813559322034, nilai *recall* sebesar 0.944954128440367, dan skor *F1* sebesar 0.9074889867841409. Sedangkan untuk *confusion matrix* pada konfigurasi model ini dapat dilihat pada Gambar 6 yang menunjukkan 4 kriteria. Penelitian ini mendapatkan nilai TP sebesar 206 karena label asli dan prediksi merupakan teks GPT, nilai TN sebesar 158 karena label asli dan prediksi merupakan teks manusia, nilai FP sebesar 30 karena label asli merupakan teks manusia tetapi di prediksi menjadi teks GPT, dan nilai FN sebesar 12 karena label asli merupakan teks GPT tetapi di prediksi menjadi teks manusia. Gambar 6 merupakan hasil visualisasi mengenai hasil *confusion matrix* pada konfigurasi model 60:40.



**Gambar 6.** *Confusion Matrix* Model LSTM 60:40

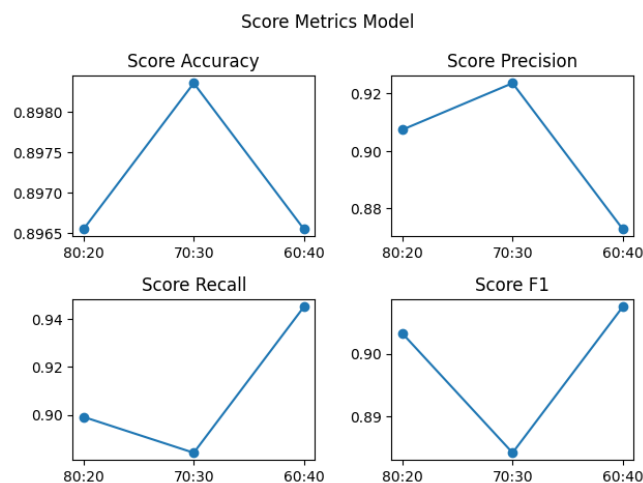


Untuk perbandingan hasil evaluasi ketiga model LSTM tersebut dapat dilihat pada Gambar 7 yang menunjukkan nilai evaluasi *loss* dan akurasi dari ketiga model tersebut. Pada gambar tersebut dapat terlihat bahwa skor evaluasi *loss* di konfigurasi 70:30 lebih rendah dibandingkan nilai *loss* pada konfigurasi lainnya. Serta skor evaluasi *accuracy* di konfigurasi 70:30 lebih tinggi dibandingkan skor evaluasi akurasi pada konfigurasi lainnya.



**Gambar 7.** Skor Evaluasi Model

Untuk perbandingan hasil evaluasi ketiga model LSTM tersebut dapat dilihat pada Gambar 8 yang menunjukkan skor metrik dari ketiga konfigurasi model. Skor metrik tersebut mencakup skor akurasi, presisi, *recall*, dan *F1*. Dari gambar tersebut, dapat terlihat bahwa skor akurasi dan presisi di konfigurasi 70:30 lebih tinggi dibandingkan skor pada konfigurasi lainnya. Pada konfigurasi 70:30 untuk skor *recall* dan *F1* memiliki skor yang lebih rendah dibandingkan skor lainnya. Untuk skor *recall* dan *F1* tertinggi terdapat pada konfigurasi model 60:40.



**Gambar 8.** Skor Evaluasi Matriks Model

Dari hasil evaluasi tersebut, dapat terlihat bahwa konfigurasi model 70:30 memiliki nilai hasil evaluasi terbaik karena memiliki skor akurasi tertinggi serta nilai *loss* terendah dibandingkan dengan skor evaluasi pada konfigurasi model lainnya.

### 3.5 Prediksi LSTM

Tahapan terakhir pada penelitian ini adalah tahapan prediksi model LSTM untuk melihat hasil prediksi model terhadap dokumen baru sehingga dapat melihat performa kinerja model terhadap data yang belum pernah dilihat sebelumnya. Penelitian ini melakukan prediksi berdasarkan 2 dokumen esai TOEFL baru, yaitu dokumen dengan esai TOEFL manusia dan dokumen dengan esai TOEFL GPT. Dokumen prediksi manusia memiliki 4 paragraf dan dokumen prediksi GPT memiliki 5 paragraf. Tahapan ini akan memberikan hasil probabilitas yang akan menentukan setiap paragraf merupakan paragraf dari manusia atau GPT. Jika hasil probabilitas kurang dari 0.5 akan dianggap sebagai data manusia dengan label 0. Sedangkan, jika hasil probabilitas lebih dari 0.5 akan dianggap sebagai data GPT dengan label 1.

Terdapat beberapa proses yang perlu dilakukan sebelum penelitian melakukan prediksi pada dokumen tersebut. Proses pertama adalah pengecekan jumlah kata dokumen sehingga penelitian hanya akan memproses dokumen dengan jumlah kata yang kurang dari 500 kata (batas maksimal kata). Proses kedua adalah pembersihan data paragraf sehingga setiap data sesuai dengan standar yang sudah ditetapkan pada penelitian ini. Proses terakhir adalah tokenisasi dan *padding*,



yaitu pengubahan data yang sudah di standarisasi menjadi urutan integer lalu menambahkan *pre-padding* sehingga urutan memiliki panjang yang sama. Setelah melakukan proses tersebut, penelitian sudah dapat melakukan prediksi terhadap data tersebut.

Pada prediksi di konfigurasi 80:20 untuk dokumen prediksi manusia memiliki skor prediksi sebesar 0.00904789 sehingga model memprediksi bahwa paragraf tersebut merupakan paragraf manusia dengan label 0. Skor prediksi tersebut berlaku terhadap setiap paragraf dari dokumen prediksi manusia yang berjumlah 4 paragraf. Sedangkan, untuk prediksi pada dokumen GPT memiliki skor prediksi sebesar 0.27768672 sehingga model memprediksi bahwa paragraf tersebut adalah paragraf manusia dengan label 0. Skor prediksi tersebut berlaku terhadap setiap paragraf dari dokumen prediksi GPT yang berjumlah 5 paragraf. Oleh karena itu, konfigurasi 80:20 terbukti gagal dalam memprediksi penggunaan GPT pada esai TOEFL.

Pada prediksi di konfigurasi 70:30 untuk dokumen manusia memiliki skor prediksi sebesar 0.0048279 sehingga model memprediksi bahwa paragraf tersebut merupakan paragraf manusia dengan label 0. Setiap paragraf pada dokumen prediksi manusia memiliki skor prediksi yang sama di keempat paragraf tersebut. Sedangkan, untuk prediksi pada dokumen GPT memiliki skor prediksi sebesar 0.6432308 sehingga model memprediksi bahwa paragraf tersebut adalah paragraf GPT dengan label 1. Skor prediksi tersebut berlaku terhadap seluruh paragraf yang terdapat pada dokumen prediksi GPT. Oleh karena itu, konfigurasi 70:30 terbukti berhasil dalam memprediksi penggunaan GPT pada esai TOEFL.

Pada prediksi di konfigurasi 60:40 untuk dokumen manusia memiliki skor prediksi sebesar 0.00596699 sehingga model memprediksi bahwa paragraf tersebut merupakan paragraf manusia dengan label 0. Skor prediksi tersebut berlaku pada keempat paragraf dari dokumen prediksi manusia. Sedangkan, untuk prediksi pada dokumen GPT memiliki skor prediksi sebesar 0.85626435 sehingga model memprediksi bahwa paragraf tersebut adalah paragraf GPT dengan label 1. Skor prediksi tersebut berlaku terhadap seluruh paragraf yang terdapat pada dokumen prediksi GPT. Oleh karena itu, konfigurasi 60:40 terbukti berhasil dalam memprediksi penggunaan GPT pada esai TOEFL.

#### 4. KESIMPULAN

Data yang digunakan pada penelitian ini merupakan 2 kumpulan data esai TOEFL manusia dan GPT dengan total sampel sebesar 220 data. Penelitian menggunakan 3 konfigurasi data sebagai proses pembagian data, yaitu 80:20, 70:30, dan 60:40. Hasil penelitian terbaik didapatkan pada model LSTM 70:30 dengan hasil skor evaluasi *loss* terendah sebesar 0.2507959306240082, skor akurasi tertinggi sebesar 0.898360655737705 dan skor presisi tertinggi sebesar 0.9235668789808917. Model tersebut pun dapat memprediksi penggunaan GPT pada dokumen baru dengan skor prediksi sebesar 0.6432308. Model LSTM 60:40 pun dapat memprediksi penggunaan GPT pada dokumen baru dengan skor prediksi sebesar 0.85626435 tetapi memiliki skor *loss* dan presisi terendah, yaitu skor *loss* sebesar 0.29463595151901245 dan skor presisi sebesar 0.8728813559322034. Sedangkan, model LSTM 80:20 gagal untuk memprediksi penggunaan GPT pada dokumen baru dengan skor prediksi sebesar 0.27768672. Saran untuk pengembangan selanjutnya dari penelitian ini adalah menggunakan *OpenAI API* untuk mendapatkan kumpulan data GPT tersebut tanpa melalui proses ekstraksi manual. Meningkatkan dan perbanyak keragaman dari kumpulan data yang digunakan seperti dari aplikasi *Perplexity*, *Gemini*, dan *Grok* sehingga dapat mempelajari dan mengetahui format penulisan manusia dan AI lainnya. Dapat mengidentifikasi penggunaan GPT untuk per kalimat dalam setiap dokumen sehingga hasil identifikasi penggunaan GPT lebih detail karena penelitian ini hanya dapat mengidentifikasi penggunaan secara per paragraf.

#### REFERENCES

- [1] I. Belcic and C. Stryker, "What is GPT (generative pre-trained transformer)? | IBM." Accessed: May 31, 2025. [Online]. Available: <https://www.ibm.com/think/topics/gpt>
- [2] J. Fleckenstein, J. Meyer, T. Jansen, S. D. Keller, O. Köller, and J. Möller, "Do teachers spot AI? Evaluating the detectability of AI-generated texts among student essays," *Computers and Education: Artificial Intelligence*, vol. 6, Jun. 2024, doi: 10.1016/j.caeai.2024.100209.
- [3] A. Fiedler and J. Döpke, "Do humans identify AI-generated text better than machines? Evidence based on excerpts from German theses," *International Review of Economics Education*, vol. 49, p. 100321, Jun. 2025, doi: 10.1016/J.IREE.2025.100321.
- [4] A. Mizumoto, S. Yasuda, and Y. Tamura, "Identifying ChatGPT-generated texts in EFL students' writing: Through comparative analysis of linguistic fingerprints," *Applied Corpus Linguistics*, vol. 4, no. 3, Dec. 2024, doi: 10.1016/j.acorp.2024.100106.
- [5] Y. Luan and S. Lin, "Research on Text Classification Based on CNN and LSTM," *2019 IEEE International Conference on Artificial Intelligence and Computer Applications (ICAICA)*, pp. 352–355, Mar. 2019, doi: 10.1109/ICAICA.2019.8873454.
- [6] T. Rahman *et al.*, "Human vs AI: Evaluating the Effectiveness of Deep Learning in AI Text Detection," in *2024 27th International Conference on Computer and Information Technology, ICCIT 2024 - Proceedings*, Institute of Electrical and Electronics Engineers Inc., 2024, pp. 2736–2741. doi: 10.1109/ICCIT64611.2024.11021749.
- [7] R. An, Y. Yang, F. Yang, and S. Wang, "Use prompt to differentiate text generated by ChatGPT and humans," *Machine Learning with Applications*, vol. 14, p. 100497, Dec. 2023, doi: 10.1016/j.mlwa.2023.100497.
- [8] GlobalExam, "TOEFL Writing - Online Training." Accessed: Jun. 08, 2025. [Online]. Available: <https://global-exam.com/blog/en/toefl-writing-online-training/>
- [9] Toeflessays.Com and G. Gao, "Answers to All TOEFL Essay Questions." Accessed: Jun. 28, 2025. [Online]. Available: <https://www.goodreads.com/book/show/41726611-answers-to-all-toefl-essay-questions>



- [10] M. Goodine, "Master the TOEFL Writing Section in 2025 | Test Resources." Accessed: Jun. 08, 2025. [Online]. Available: <https://www.toeflresources.com/writing-section/>
- [11] S. Zhu and F. Chollet, "Understanding masking & padding | TensorFlow Core." Accessed: Jan. 05, 2025. [Online]. Available: [https://www.tensorflow.org/guide/keras/understanding\\_masking\\_and\\_padding#padding\\_sequence\\_data](https://www.tensorflow.org/guide/keras/understanding_masking_and_padding#padding_sequence_data)
- [12] Krithika, "Introduction to FastText Embeddings and its Implication -." Accessed: Jan. 05, 2025. [Online]. Available: <https://www.analyticsvidhya.com/blog/2023/01/introduction-to-fasttext-embeddings-and-its-implication/>
- [13] J. Barnard, "What Are Word Embeddings? | IBM." Accessed: Jan. 05, 2025. [Online]. Available: <https://www.ibm.com/think/topics/word-embeddings>
- [14] T. Mikolov, E. Grave, P. Bojanowski, C. Puhresch, and A. Joulin, "Advances in Pre-Training Distributed Word Representations," *LREC 2018 - 11th International Conference on Language Resources and Evaluation*, pp. 52–55, Dec. 2017, Accessed: Feb. 04, 2025. [Online]. Available: <https://arxiv.org/abs/1712.09405v1>
- [15] A. S. Gillis, "What is data splitting and why is it important?" Accessed: Jan. 09, 2025. [Online]. Available: <https://www.techtarget.com/searchenterpriseai/definition/data-splitting>
- [16] M. Bansal, A. Goyal, and A. Choudhary, "A comparative analysis of K-Nearest Neighbor, Genetic, Support Vector Machine, Decision Tree, and Long Short Term Memory algorithms in machine learning," *Decision Analytics Journal*, vol. 3, p. 100071, Jun. 2022, doi: 10.1016/J.DAJOUR.2022.100071.
- [17] V. Gallan, "LSTM (Long Short Term Memory). Sebelum mempelajari tentang LSTM, kita... | by Varelino Gallan | Bina Nusantara IT Division | Medium." Accessed: Feb. 09, 2025. [Online]. Available: <https://medium.com/bina-nusantara-it-division/lstm-long-short-term-memory-d29779e2ebf8>
- [18] A. Satyo Bayangkari Karno, "Prediksi Data Time Series Saham Bank BRI Dengan Mesin Belajar LSTM (Long ShortTerm Memory)," *Journal of Information and Information Security (JIFORTY)*, vol. 1, no. 1, pp. 1–8, 2020, [Online]. Available: <http://ejurnal.uharajaya.ac.id/index.php/jiforty>
- [19] M. Umer *et al.*, "Impact of convolutional neural network and FastText embedding on text classification," *Multimed Tools Appl*, vol. 82, no. 4, pp. 5569–5585, Feb. 2023, doi: 10.1007/S11042-022-13459-X/FIGURES/3.
- [20] K. Chowdhury, "10 Hyperparameters to keep an eye on for your LSTM model — and other tips | by Kuldeep Chowdhury | Geek Culture | Medium," Geek Culture. Accessed: Feb. 10, 2025. [Online]. Available: <https://medium.com/geekculture/10-hyperparameters-to-keep-an-eye-on-for-your-lstm-model-and-other-tips-f0ff5b63fcd4>
- [21] M. A. H. Wadud, M. M. Kabir, M. F. Mridha, M. A. Ali, M. A. Hamid, and M. M. Monowar, "How can we manage Offensive Text in Social Media - A Text Classification Approach using LSTM-BOOST," *International Journal of Information Management Data Insights*, vol. 2, no. 2, p. 100095, Nov. 2022, doi: 10.1016/J.IJIMEI.2022.100095.
- [22] N. K. Hemalatha, R. N. Brunda, G. S. Prakruthi, B. V. B. Prabhu, A. Shukla, and O. S. J. Narasipura, "Sugarcane leaf disease detection through deep learning," *Deep Learning for Sustainable Agriculture*, pp. 297–323, Jan. 2022, doi: 10.1016/B978-0-323-85214-2.00003-3.
- [23] P. K. Yechuri and S. Ramadass, "Classification of Image and Text Data Using Deep Learning-Based LSTM Model," *Traitement du Signal*, vol. 38, no. 6, pp. 1809–1817, Dec. 2021, doi: 10.18280/ts.380625.
- [24] englishclubmshk, "An Essay Collection For TOEFL," 2018. Accessed: Jun. 28, 2025. [Online]. Available: <https://englishclubmshk.wordpress.com/wp-content/uploads/2018/09/toefl-essays.pdf>
- [25] S. K. Ahmed, "How to choose a sampling technique and determine sample size for research: A simplified guide for researchers," *Oral Oncology Reports*, vol. 12, p. 100662, Dec. 2024, doi: 10.1016/J.OOR.2024.100662.
- [26] fchollet, "The Sequential model." Accessed: Jul. 01, 2025. [Online]. Available: [https://keras.io/guides/sequential\\_model/](https://keras.io/guides/sequential_model/)
- [27] "Keras Input Layer - GeeksforGeeks." Accessed: Jul. 01, 2025. [Online]. Available: <https://www.geeksforgeeks.org/deep-learning/keras-input-layer/>
- [28] B. Krishnamurthy, "ReLU Activation Function Explained | Built In." Accessed: Jul. 03, 2025. [Online]. Available: <https://builtin.com/machine-learning/relu-activation-function>
- [29] B. Janeczko and G. Srivastava, "The use of deep learning in image analysis for the study of oncology," *Internet of Multimedia Things (IoMT): Techniques and Applications*, pp. 133–150, Jan. 2022, doi: 10.1016/B978-0-32-385845-8.00011-3.