REPOSITOR, Vol. 7, No. 3, Agustus 2025, Pp. 393-400
ISSN : 2714-7975
E-ISSN : 2716-1382

Penerapan Model Pre-Trained BERT dalam Mendeteksi Teks Buatan ChatGPT

Putri Maharani Isnainiyah*[1], Christian Sri Kusuma Aditya[1], Didih Rizki Chandranegara[1]
Universitas Muhammadiyah Malang
putrimaharani@webmail.umm.ac.id*

Abstrak

Perkembangan AI yang berkembang saat ini tentu saja mempermudah pekerjaan manusia. Tetapi, disamping itu, terdapat beberapa dampak buruk ada AI. Contohnya dampak buruk adanya AI yang dapat mengenerate text seperti adanya potensi saran disinformasi dan resiko pelanggaran terhadap privasi pengguna. Penelitian ini menggunakan BERT untuk mendeteksi teks ChatGPT. Dataset dibagi menjadi kurang dari 200 kata dan lebih dari 200 kata. Optimizer SGD menunjukkan akurasi lebih tinggi pada dataset lebih dari 200 kata (97%) dibandingkan yang kurang (95%). Dengan AdamW, akurasi kedua dataset sama, namun loss pada teks kurang dari 200 kata lebih rendah (0.086) dibanding yang lebih dari 200 kata (0.18).

Kata Kunci: *AI, ChatGPT, BERT*

Abstract

The current development of AI undoubtedly makes human tasks easier. However, there are also several negative impacts associated with AI. For example, the ability of AI to generate text poses risks such as the potential for disinformation and the risk of violating user privacy. This study uses BERT to detect ChatGPT-generated text. The dataset is divided into two categories: less than 200 words and more than 200 words. The SGD optimizer shows higher accuracy for the dataset with more than 200 words (97%) compared to the one with fewer words (95%). With AdamW, both datasets achieve the same accuracy, but the loss for the text with less than 200 words is lower (0.086) compared to the text with more than 200 words (0.18).

Keywords: *AI, ChatGPT, BERT*

1. Pendahuluan

Perkembangan teknologi di zaman ini sudah sangat berkembang terutama di bidang AI atau Artificial Intelligence. Artificial Intelligence adalah bidang multidisiplin yang dapat membuat keputusan tanpa dipengaruhi oleh nilai-nilai pribadi manusia [1]. Kemampuan AI tentu saja dapat menimbulkan dampak baik dan dampak buruk. Salah satunya adalah kemampuan AI untuk mengenerate sebuah text, tentunya dengan kemampuan tersebut, manusia bisa lebih mudah dalam melakukan pekerjaan seperti, merangkum dan membuat sebuah text. Tetapi, tetap saja ada beberapa dampak buruk seperti adanya potensi sebagai sarana disinformasi [2], potensi bias dalam pembuatan konten, dan dapat membawa resiko terhadap privasi pengguna [3].

"Penelitian terkait klasifikasi teks antara AI dan manusia sedang berkembang, salah satunya dilakukan oleh Iyab Katib dan koleganya [4]. Dalam penelitian tersebut, para peneliti menggunakan berbagai metode seperti Decision Tree, SVM Model, XGBoost, CNN Model, ELM Model, serta TSA-LSTMRNN. Metode TSA-LSTMRNN terbukti memberikan akurasi terbaik pada dataset manusia sebesar 93,17%, dan akurasi 93,83% untuk dataset ChatGPT."

Penelitian kedua adalah penelitian berjudul “Deep dive into language traits of AI-generated Abstracts” yang dilakukan Kumar, dkk[5]. Penelitian ini menggunakan lima pendekatan, yakni LDA (Linear Discriminant Analysis), regresi logistik (LR), klasifikasi vektor pendukung (SVC), XGBoost (Extreme Gradient Boosting), serta Extra Trees Classifier (ETC). Akurasi tertinggi didapatkan dengan metode XGBoost dengan SF (Fitur Semantik), LF (fitur linguistik) dan HBH (fitur pragmatic). Akurasi yang didapatkan adalah 93.4%.

Dari pemaparan di atas, penelitian ini akan melakukan mengenai deteksi teks buatan ChatGPT menggunakan model pretrained BERT. BERT adalah salah satu metode NLP (Natural Language Representation from Transformer yang dilatih dengan langkah pre-training dan fine tuning. Dikarenakan arsitekturnya terdiri dari multi-layer bidirectional Transformer, BERT dapat

menangkap hubungan kata dari inputan (self-attention). Selain itu, BERT dapat membaca inputan dari dua arah secara bersamaan. Dengan kemampuan tersebut, BERT dapat membuat representasi kata dengan lebih tepat[6].

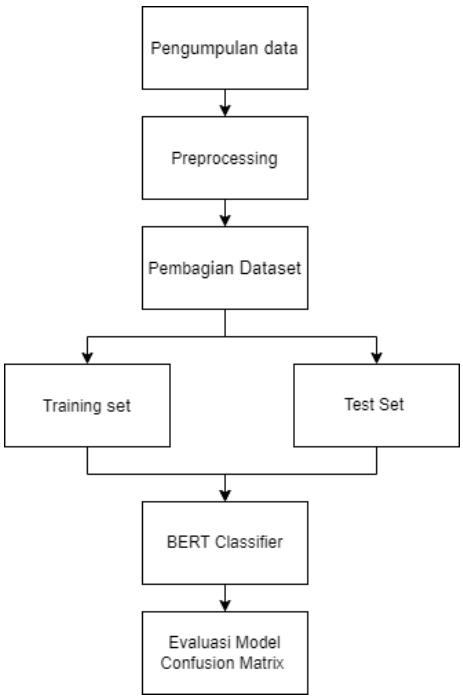
Pada penelitian ini akan ada beberapa tahap. Tahap tersebut akan diawali dari pengumpulan data, setelah itu dilakukan preprocessing, pembagian dataset, proses training dan proses evaluasi. Dengan adanya penelitian ini, diharapkan dapat tercipta model deteksi teks buatan ChatGPT yang lebih akurat dan efisien dengan penggunaan BERT sebagai model. Dengan menggunakan BERT, diharapkan model dapat menangkap makna kontekstual setiap kata secara lebih baik dibandingkan metode NLP lainnya.

2. Metode Penelitian

Penelitian ini memakai BERT sebagai metode untuk mengenali teks yang dihasilkan oleh ChatGPT. Pada penelitian ini akan ada beberapa tahap. Tahap tersebut akan diawali dari pengumpulan data, setelah itu dilakukan preprocessing, pembagian dataset, proses training dan proses evaluasi.

2.1 Alur Penelitian

Pada penelitian ini, data yang sudah terkumpul akan dilakukan preprocessing berupa data cleaning dan juga tokenization menggunakan BERT. Setelah itu, dataset akan dipisahkan rasio 75%:25% (train set : test set). Selanjutnya dilakukan tahap training. Pada tahap ini, model akan dilatih menggunakan data training yang telah dibagi. Setelah proses pelatihan selesai, model akan dievaluasi menggunakan data testing untuk mengukur performanya. Pada Gambar 1 menunjukkan alur dari penelitian ini.



Gambar 1. Alur Penelitian

2.2 Analisis Masalah

Penelitian ini secara garis besar bertujuan untuk membuat sebuah model yang dapat mendeteksi teks buatan chatgpt dengan tingkat akurasi yang tinggi. Model ini akan menggunakan algoritma BERT yang memliki kemampuan untuk memahami konteks bahasa dan relasai antar kata.

Dalam penelitian ini, terdapat dua aspek utama yang menjadi fokus utama analisis, yaitu seberapa tinggi akurasi yang didapatkan oleh model BERT untuk mendeteksi teks buatan chatgpt. Selain iitu, penelitian ini juga akan menganalisis apakah panjang teks akan mempengaruhi akurasi model.

2.3 Dataset

Data yang digunakan adalah gabungan dari data “ChatGPT Classification Dataset” dan “GPT vs. Human: A Corpus of Research Abstracts”. Kedua dataset tersebut didapatkan dari kaggle.com. Pada penelitian ini sendiri akan terdapat dua macam dataset, yaitu Dataset kurang dari 200 kata dan Dataset lebih dari 200 kata. Kedua dataset akan memiliki dua kelas, yaitu kelas human generated dan GPT generated yang masing-masing berjumlah 226.

Dalam pengumpulan dataset ini, terdapat beberapa tahap. Tahap pertama, kedua dataset awal akan dibagi menjadi dua kategori, yaitu dataset dengan kata kurang dari 200 dan dataset dengan kata lebih dari 200. Selanjutnya, kategori dari kedua dataset akan digabungkan. Setelah itu, dilakukan analisis terhadap persebaran kelas pada masing-masing kategori. Agar masing-masing kategori memliki jumlah yang sama, dilakukan pemotongan data berdasarkan jumlah kelas terendah.

2.4 Preprocessing

Preprocessing dilakukan agar data dapat diproses untuk dilakukan training. Terdapat dua tahapan preprocessing pada penelitian ini. Tahap pertama adalah Data cleaning. Data cleaning adalah penghapusan elemen yang tidak relevan, seperti tanda baca atau tanda hubung. Tahap kedua merupakan proses case folding, di mana data akan dikonversi menjadi huruf kecil. Ini bertujuan agar teks menjadi lebih konsisten dalam penggunaan huruf sehingga analisis menjadi lebih mudah.

2.5 Pembagian Dataset

Setelah data diproses, akan dilakukan pembagian menjadi training set dan test set. Selanjutnya, training set akan dibagi ulang menjadi training set dan validation set. Training set digunakan saat melakukan pelatihan model. Validation set digunakan untuk proses validasi model. Untuk Testing set digunakan untuk melakukan evaluasi pada model yang telah terbentuk. Pada penelitian ini, pembagian dataset akan memiliki rasio 75% data training dan 15% data test. Untuk data validation akan mendapat 10% dari data training.

2.6 Tokenizing

Tokenizing adalah tahap teks akan dipisahkan menjadi kata-kata yang akan disebut menjadi token. Dengan adanya token, model akan lebih mudah untuk memahami teks [7]. Tokenisasi pada penelitian in akan menggunakan token BERT base. BERT base menyediakan 30.552 dalam vocabularynya [8]. Inputan pada tokenisasi akan mengalami tiga tahapan. Yang pertama, akan mendapat token khusus, yaitu token [CSL] yang mempresentasikan agrerat dari rangkaian token dan token [SEP] yang akan memisahlan inputan yang memiliki lebih dari 1 kalimat. Selanjutnya, token-token tersebut akan dimapping berdasarkan vocabulary BERT. Dan tahap terakhir, yaitu pembuatan token attention task. Token attention mask adalah proses di mana token akan dibagi menjadi dua yaitu token untuk mempresentasikan token [PAD] dan token 1 akan mempresentasikan token kata asli.

2.7 BERT classifier

Penelitian ini menggunakan modle BERT yang memiliki arsitektur multi-layer transformer. BERT memiliki keunggulan untuk menangkap hubungan dari inputan (self-attention) dan juga BERT dorancang untuk melatih secara dua arah yang menggabungkan konteks kiri dan kanan di semua layer [9].

Setelah dilakukan tokenizing, langkah selanjutnya adalah proses training. Untuk proses training, tahap pertama yaitu melakukan inisialisasi model BERT. Pada inisialisasi model, ditetapkan parameter num_labels untuk menentukan jumlah kelas dari klasifikasi. Setelah itu, dilakuakn pemuatan data loaders yang akan membantu data menjadi lebih terstruktur dan sistematis [10]. Selanjutnya, ditetapkan beberapa parameter, seperti optimizer dan learning rate. Terdapat dua jenis optimizer untuk pengujian penelitian ini.

2.7.1 SGD

SGD (Stochastic gradient descent) adalah optimizer yang akan memperbarui bobot tanpa menunggu seluruh epoch selesai. Selain itu, akan ada parameter momentum yang akan membantu percepatan dan stabilitas konvergensi algoritma optimasi sehingga algoritma dapat mengurangi fluktuasi yang terjadi selama proses optimasi.

2.7.2 AdamW

AdamW adalah optimizer dari variasi Adam yang merupakan gabungan dari optimasi Momentum dan RMSprop. Untuk memperbarui bobot, Adam akan menggunakan estimasi momen pertama (mean) dan kedua (uncentered variance). Pada adamW, terdapat parameter weight decay yang dapat membantu regulasi model.

Setelah dilakukan inisialisasi optimizer dan learning rate, akan dilakukan pengaturan dropout layer, nilai epoch, dan batch size.

2.8 Evaluasi Model

Setelah dilakukan pelatihan model dengan training set, akan dilakukan evaluasi dengan test set menggunakan confusion matrix. Confusion matrix memiliki empat nilai yang terdiri dari nilai sebenarnya dan nilai prediksi. Terdapat empat kelas dalam confusion matrix, yaitu True Positive (TP) yang berarti model berhasil memprediksi sebagai kelas positif dengan benar, False Positive (FP) yang berarti model salah memprediksi sebagai kelas positif, True Negative (TN) yang berarti model dengan benar memprediksi sebagai kelas negatif dan False Negative (FN) berarti model dengan salah memprediksi sebagai kelas negatif.

3. Hasil Penelitian dan Pembahasan

Penelitian ini melakukan uji dengan membagi dataset menjadi dua dataset, yaitu dataset dengan kurang dari 200 kata dan dataset dengan lebih dari 200 kata. Kedua dataset memliki jumlah data sebesar 452, yang masing-masing kelas memliki jumlah 226. terdapat dua kelas pada dataset ini, yaitu kelas 0 (human generated) dan kelas 1 (GPT generated).

3.1 Proses Training

Setelah melewati tahap preprocessing, data melalui tahap pembagian dataset. Selanjutnya akan dilakukan proses tokenisasi. Pada proses tokenisasi akan ditetapkan panjang maksimum untuk token. Dikarenakan pada penelitian ini terdapat dua dataset yang memilki panjang yang berbeda, maka setiap dataset juga akan memiliki skenario panjang maksimum yang berbeda juga. Pada dataset kata kurang dari 200 akan ditetapkan panjang maksimum sebesar 300 dan untuk dataset kata lebih dari 200 akan ditetapkan panjang maksimum sebesar 512.

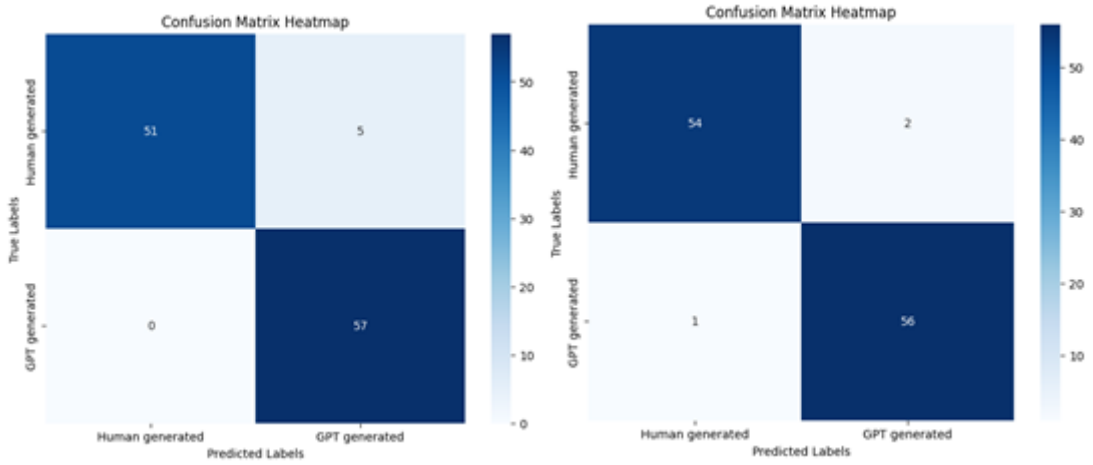
Setelah dilakukan tokenisasi, data akan digabung menggunakan data loaders dengan fungsi dari Dataset.from_tensor_slices(). Selanjutnya, dilakukan pemuatan model BERT dengan jumlah kelas 2. Untuk nilai learning rate yang ditetapkan adalah sebesar 0.00003. Terdapat dua skenario dalam penelitian ini, yaitu yang pertama akan menggunakan fungsi optimasi SGD dan yang kedua menggunakan fungsi optimasi AdamW.

Pada skenario pertama atau SGD, nilai yang ditetapkan untuk nilai momentum adalah 0.9. Pada skenario kedua atau adamw, nilai yang ditetapkan untuk nilai weight decay adalah sebesar 0.02. Dan selanjutnya dilakukan pengaturan nilai dropout sebesar 0.2 untuk mengurangi overfitting.

3.2 Evaluasi Model

3.2.1 Skenario 1 (SGD)

Akurasi test yang didapatkan dari model dengan dataset kurang dari 200 kata adalah 96% dengan nilai loss sebsar 17%. Sedangkan akurasi test set pada dataset lebih dari 200 kata adalah 97% dengan nilai loss 7.73%. Pada gambar 2, menunjukkan confusion matrix dari dataset kurang dari 200 kata dan dataset lebih dari 200 katadengan fungsi optimizer adamW.



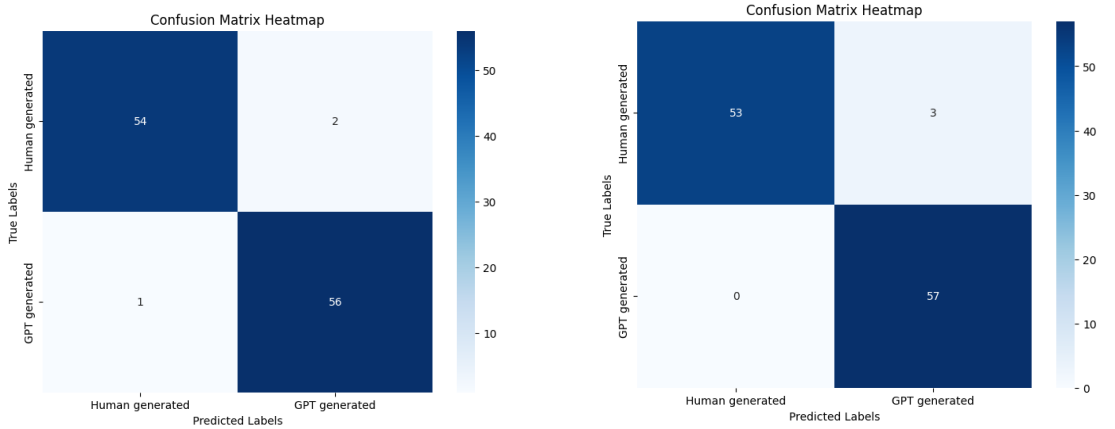
Gambar 2. Confusion Matrix pada Dataset <200 Kata dan Dataset >200 Kata Skenario 1

Dari confusion matrix pada Gambar 2. dataset kurang dari 200 kata, dapat dilihat bahwa dari data sebanyak 133, model dapat memprediksi kelas human generated dengan benar (True negative) sejumlah 51. Lalu, untuk data yang sebenarnya adalah kelas human generated, tetapi salah diprediksi sebagai kelas GPT generated (False positif) berjumlah 5. Selanjutnya, untuk data yang berhasil diprediksi sebagai kelas GPT generated (True positif) berjumlah 57. Dan untuk kelas yang sebenarnya GPT generated, tetapi diprediksi sebagai human generated (False negative) berjumlah 0.

Sedangkan, dari confusion matrix dataset lebih dari 200 kata dapat dilihat bahwa dari data sebanyak 133, model dapat memprediksi kelas human generated dengan benar (True negative) sejumlah 54. Lalu, untuk data yang sebenarnya adalah kelas human generated, tetapi salah diprediksi sebagai kelas GPT generated (False positif) berjumlah 2. Selanjutnya, untuk data yang berhasil diprediksi sebagai kelas GPT generated (True positif) berjumlah 56. Dan untuk kelas yang sebenarnya GPT generated, tetapi diprediksi sebagai human generated (False negative) berjumlah 1.

3.2.2 Skenario 2 (AdamW)

Akurasi test yang didapatkan dari model dengan dataset kurang dari 200 kata adalah 97% dengan nilai loss sebsar 8.62%. Sedangkan akurasi test set pada dataset lebih dari 200 kata adalah 97% dengan nilai loss 18.05%. Pada Gambar 2, menunjukkan confusion matrix dari dataset kurang dari 200 kata dan dataset lebih dari 200 katadengan fungsi optimizer adamw.



Gambar 3. Confusion Matrix pada Dataset <200 Kata dan Dataset >200 Kata Skenario 2

Dari confusion matrix pada Gambar 3, dapat dilihat bahwa dari data sebanyak 133, model dapat memprediksi kelas human generated dengan benar (True negative) sejumlah 54. Lalu, untuk data yang sebenarnya adalah kelas human generated, tetapi salah diprediksi sebagai kelas

GPT generated (False positif) berjumlah 2. Selanjutnya, untuk data yang berhasil diprediksi sebagai kelas GPT generated (True positif) berjumlah 56. Dan untuk kelas yang sebenarnya GPT generated, tetapi diprediksi sebagai human generated (False negative) berjumlah 1.

Dari confusion matrix di atas, dapat dilihat bahwa dari data sebanyak 133, model dapat memprediksi kelas human generated dengan benar (True negative) sejumlah 53. Lalu, untuk data yang sebenarnya adalah kelas human generated, tetapi salah diprediksi sebagai kelas GPT generated (False positif) berjumlah 3. Selanjutnya, untuk data yang berhasil diprediksi sebagai kelas GPT generated (True positif) berjumlah 57. Dan untuk kelas yang sebenarnya GPT generated, tetapi diprediksi sebagai human generated (False negative) berjumlah 0.

3.3 Hasil Keseluruhan

Pada Tabel 1 menunjukkan perbandingan hasil akurasi dan nilai loss dalam kedua skenario terhadap data testing.

Tabel 1. Tabel Perbandingan Akurasi

	Optimizer SGD		Optimizer AdamW	
	Acc	Loss	Acc	Loss
Dataset kurang dari 200 kata	0.96	0.17	0.97	0.086
Dataset lebih dari 200 kata	0.97	0.073	0.97	0.18

Dilihat dari hasil pengujian yang telah dilakukan, dataset dengan kurang dari 200 kata lebih optimal saat menggunakan optimizer adamW (0.97) dibandingkan menggunakan optimizer SGD (0.96). Sedangkan, untuk dataset dengan kata lebih dari 200, pada masing-masing optimizer memiliki nilai akurasi test yang sama, hanya saja pada optimizer SGD nilai loss test (0.073) yang didapatkan lebih kecil dibandingkan saat menggunakan optimizer AdamW (0.18) sehingga dapat disimpulkan dataset dengan kata lebih dari 200 lebih optimal jika memakai optimizer SGD.

Secara keseluruhan, panjang data mempengaruhi akurasi model. Dataset dengan kata lebih dari 200 lebih akurat menggunakan SGD, sedangkan dengan AdamW, panjang data tidak terlalu berpengaruh pada akurasi, tetapi dataset kurang dari 200 kata memiliki nilai loss yang lebih rendah. AdamW memiliki keunggulan dalam kecepatan pelatihan karena menggunakan laju pembelajaran adaptif untuk memperbarui bobot jaringa, berbeda dengan SGD yang menggunakan fixed learning rate sehingga kurang fleksibel.

4. Kesimpulan

Dari semua pengujian yang telah dilakukan, dapat dilihat dengan optimizer SGD, dataset kata lebih dari 200 memilki akurasi yang lebih tinggi (97%) daripada dataset kata kurang dari 200 (95%). Untuk pengujian dengan optimizer AdamW, kedua dataset memiliki hasil akurasi test yang sama, yaitu 0.97. Tetapi, nilai loss test dari dataset kurang dari 200 kata lebih rendah (0.086) daripada dataset dengan kata lebih dari 200 (0.18).

Secara keseluruhan, panjang data mempengaruhi akurasi model. Dataset dengan kata lebih dari 200 lebih akurat menggunakan SGD, sedangkan dengan AdamW, panjang data tidak terlalu berpengaruh pada akurasi, tetapi daatset kurang dari 200 kata memiiliki nilai loss yang lebih rendah.

Referensi

[1] Wahyudi, T. (2023, Juni). Studi Kasus Pengembangan dan Penggunaan Artificial Intelligence (AI) Sebagai Penunjang Kegiatan Masyarakat Indonesia. *Indonesian Journal on Software Engineering (IJSE)*, *9*(1), 28-32. https://doi.org/10.31294/ijse.v9i1.15631

[2] Barman, D., & Guo, Z. (2024, June). The Dark Side of Language Models: Exploring the Potential of LLMs in Multimedia Disinformation Generation and Dissemination. *Machine Learning with Applications*, *16*. https://doi.org/10.1016/j.mlwa.2024.100545

[3] Wach, K., Duong, C. D., Ejdys, J., Kazlauskaite, R., Korzynski, P., Mazurek, G., Paliszkiewicz, J., & Ziemba, E. (2023, June). The dark side of generative artificial intelligence: A critical analysis of controversies and risks of ChatGPT. *Entrepreneurial*

Business and Economics Review, *11*(2), 7-30. DOI:10.15678/EBER.2023.110201

[4] Katib, I., Assiri, F. Y., Abdushkour, H. A., Hamed, D., & Ragab, M. (2023). Differentiating Chat Generative Pretrained Transformerfrom Humans: Detecting ChatGPT-Generated Text and HumanText Using Machine Learning. *Mathematics*, *Vol 11*(15), 3400. https://doi.org/10.3390/math11153400

[5] Kumar, V., Bharti, A., Verma, D., & Bhatnagar, V. (2023). Textual Analysis and Detection of AI Generated Academic Texts. arXiv:2312.10617

[6] Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019, Maret 24). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. https://doi.org/10.48550/arXiv.1810.04805

[7] Toraman, C., Yilmaz, E. H., Sahinuc, F., & Ozcelik, O. (2023, Maret 25). Impact of Tokenization on Language Models: An Analysis for Turkish. ACM Transactions on Asian and Low-Resource Language Information Processing, 22(4), 1-21. https://doi.org/10.1145/3578707

[8] Khadhraoui, M., Bellaaj, H., Ammar, M. B., Hamam, H., & Jmaiel, M. (2022, Maret 11). Survey of BERT-Base Models for Scientific Text Classification: COVID-19 Case Study. *Applied Sciences*, *12*, 2891. https://doi.org/10.3390/app12062891

[9] Devlin, J., Chang, M.-W., Lee, K., & Toutanova, K. (2019, Maret 24). BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. https://doi.org/10.48550/arXiv.1810.04805

[10] Atmaja, R. M. R. W. P. K., & Yustanti, W. (2021, Juli 12). Analisis Sentimen Customer Review Aplikasi Ruang Guru Dengan Metode BERT (Bidirectional Encoder Representations from Transformers). *Journal of Emerging Information System and Business Intelligence (JEISBI)*, *Vol. 2*(No. 3), 55-62.