

RANCANG BANGUN SISTEM PENILAIAN DAN UMPAN BALIK OTOMATIS PADA JAWABAN TEKS DENGAN *SENTENCE TRANSFORMERS* DAN LLM *GEMMA* BERBASIS WEB

Muhamad Fazril Maulana

¹Jurusan, Teknik Informatika dan Komputer, Politeknik Negeri Jakarta

Depok, Jawa Barat

muhamad.fazril.maulana.tik21@mhswn.pnj.ac.id

Abstract - The manual essay assessment process is often time-consuming, labor-intensive, and potentially subjective, hindering teachers from providing effective feedback. This research aims to design and build a web-based automatic assessment and feedback system as a solution. The system was developed as a Learning Management System (LMS) using the Laravel framework, integrated with deep learning models. The research method used is the AI Project Life Cycle. The assessment model was built using pre-trained Sentence Transformers and Cosine Similarity, reinforced with Part-of-Speech (POS) Tagging to improve accuracy in understanding sentence meaning. For feedback, the system utilizes the Large Language Model (LLM) Gemma to generate constructive responses. The assessment model testing achieved an accuracy of 95%. Functionality testing through Black Box Testing showed that all features perform as expected. Usability evaluation using User Acceptance Testing (UAT) and the System Usability Scale (SUS), involving 25 respondents (teachers and students), indicated an excellent level of acceptance with an average SUS score of 89.3.

Keywords: Natural Language Processing, Part of Speech Tagging, Sentence Transformers, LLM Gemma, Automatic Feedback.

Abstrak-- Proses penilaian esai secara manual seringkali memakan waktu, tenaga, dan berpotensi subjektif, sehingga menghambat guru dalam memberikan umpan balik yang efektif. Penelitian ini bertujuan untuk merancang dan membangun sistem penilaian dan umpan balik otomatis berbasis web sebagai solusi. Sistem ini dikembangkan sebagai Learning Management System (LMS) menggunakan framework Laravel yang diintegrasikan dengan model deep learning. Metode penelitian yang digunakan adalah AI Project Life Cycle. Model penilaian dibangun menggunakan pre-trained Sentence Transformers dan Cosine Similarity, serta diperkuat dengan Part-of-Speech (POS) Tagging untuk meningkatkan akurasi dalam memahami makna kalimat. Untuk umpan balik, sistem memanfaatkan Large Language Model (LLM) Gemma guna menghasilkan respons yang konstruktif. Hasil pengujian model penilaian mencapai akurasi 95%. Pengujian fungsionalitas melalui Black Box Testing menunjukkan semua fitur berjalan sesuai harapan. Evaluasi kegunaan dengan metode User Acceptance Testing (UAT) dan System Usability Scale (SUS) yang melibatkan 25 responden (guru dan siswa) menunjukkan tingkat penerimaan sangat baik dengan skor SUS rata-rata 89,3.

Kata kunci: Natural Language Processing, Part of Speech Tagging, Sentence Transformers, LLM Gemma, Umpan Balik Otomatis.

I. PENDAHULUAN

Proses penilaian pada soal esai sering kali menjadi tantangan bagi tenaga pendidik, terutama dalam hal konsistensi. Kadangkala, berpotensi terjadi penilaian yang bias dimana hasil yang didapatkan dapat berbeda antar jawaban yang padahal memiliki makna yang sama [1]. Selain itu, metode manual cenderung memakan waktu yang

lama, sekitar 7-10 menit untuk menilai satu siswa, dan memerlukan konsentrasi tinggi yang sulit dipertahankan saat jumlah siswa banyak [2], [3]. Dari hasil wawancara dengan guru di SMK Grafika Yayasan Lektur, ditemukan bahwa metode penilaian yang masih dilakukan secara manual memakan waktu cukup lama, terutama jika jumlah siswa banyak dan waktu terbatas. Hal ini menyulitkan guru memberikan umpan balik secara efektif. Aktivitas

mengoreksi soal esai dirasa melelahkan, dan pemberian umpan balik menambah beban kerja. Kondisi ini dikhawatirkan berdampak pada kesehatan guru, yang dapat memengaruhi konsistensi penilaian dan kualitas umpan balik siswa.

Perkembangan teknologi pemrosesan bahasa alami atau *Natural Language Processing* (NLP) telah mendorong lahirnya berbagai sistem penilaian esai otomatis atau *Automated Essay Scoring* (AES). Beberapa penelitian terdahulu telah menerapkan metode seperti pencocokan kata kunci berbasis frasa nomina [4], penggunaan *Term Frequency-Inverse Document Frequency* (TF-IDF) dengan *Cosine Similarity* [5], serta algoritma *machine learning* seperti *Support Vector Machine* (SVM) dan *Latent Semantic Analysis* (LSA) [3]. Meskipun demikian, metode-metode tersebut masih memiliki keterbatasan mendasar, yaitu kesulitan dalam memahami makna semantik dan konteks kalimat secara mendalam, serta belum mampu menilai jawaban esai dengan variasi kalimat kompleks namun bermakna sama secara akurat.

Penelitian ini menawarkan inovasi dengan merancang sistem penilaian dan umpan balik otomatis yang terintegrasi menggunakan arsitektur *deep learning modern*, yaitu *Sentence Transformers*. *Sentence Transformers* unggul dalam menghasilkan representasi vektor kalimat yang sensitif terhadap konteks semantik dan cocok untuk tugas-tugas di tingkat kalimat seperti perbandingan similaritas [6], dan diperkuat dengan teknik *Part-of-Speech* (POS) *Tagging* untuk mengidentifikasi potensi konflik makna dalam kalimat. Penilaian kesamaan makna antara jawaban siswa dan jawaban referensi dilakukan dengan *cosine similarity*, yang efektif mengukur kedekatan semantik antar vektor. Selain memberikan skor, sistem ini juga menggunakan *Large Language Model* (LLM) *Gemma* untuk menghasilkan umpan balik konstruktif bagi siswa.

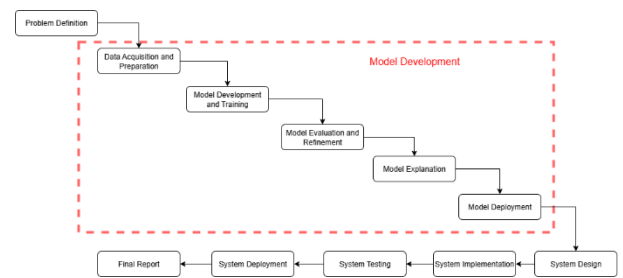
II. METODOLOGI PENELITIAN

A. Rancangan Penelitian

Penelitian ini bertujuan mengembangkan aplikasi penilaian otomatis untuk jawaban teks menggunakan model *pre-trained Sentence Transformers* dengan teknik *semantic embedding*. Algoritma *Cosine Similarity* digunakan untuk mengukur kemiripan antara jawaban siswa dan jawaban referensi. Selain itu, model LLM *Gemma* diintegrasikan untuk memberikan umpan balik informatif yang konstruktif kepada siswa.

B. Tahapan Penelitian

Penelitian ini menggunakan metode AI Project Life Cycle sebagai pendekatan pengembangan. Metode ini menawarkan langkah-langkah sistematis yang mudah diterapkan dalam proyek berbasis kecerdasan buatan [7]. Tahapan-tahapan dalam metode penelitian ditunjukkan pada Gambar 1.



Gambar 1 Alur AI Project Life Cycle

1. Pendefinisian Masalah (*Problem Definition*)

Penelitian ini bertujuan mengembangkan solusi teknologi untuk mengatasi masalah subjektivitas penilaian, lamanya waktu penilaian, dan kurangnya umpan balik pada jawaban esai siswa di SMK Grafika Yayasan Lektor.

2. Pengumpulan dan Persiapan Data (*Data Acquisition and Preparation*)

Tahap ini meliputi pengumpulan dan persiapan data melalui studi literatur dan wawancara dengan guru di SMK Grafika Yayasan Lektor untuk memahami kebutuhan dan tantangan. Dataset berupa pasangan soal, kunci jawaban, dan jawaban siswa juga dikumpulkan untuk pengembangan dan kalibrasi model AI. Data kemudian dilakukan pemrosesan dengan langkah seperti *lowercasing*, penghapusan tag HTML, normalisasi karakter non-alfanumerik, dan normalisasi spasi agar bersih dan konsisten untuk pengukuran kemiripan semantik.

3. Pengembangan dan Pelatihan Model (*Model Development and Training*)

Tahap ini berfokus pada pemilihan dan penerapan model untuk mengukur kemiripan semantik dan memberikan umpan balik. Dua model utama digunakan:

- Model Penilaian, Menggunakan *Sentence Transformers* (*paraphrase-multilingual-MiniLM-L12-v2*) untuk mengukur kemiripan makna antara jawaban siswa dan kunci jawaban. Model ini mendukung berbagai bahasa, termasuk Indonesia, dan diperkuat dengan POS tagging untuk analisis lebih mendalam.

- b. Model Umpan Balik, Menggunakan LLM *Gemma* (*open-source*) untuk menghasilkan kalimat umpan balik berdasarkan skor dan jawaban siswa dengan teknik *prompt engineering* untuk meningkatkan kualitas respons sesuai kebutuhan aplikasi.

4. Evaluasi Model (*Model Evaluation and Refinement*)

Evaluasi Model dilakukan terhadap dua model utama:

- a. Model Penilaian dievaluasi menggunakan dataset dan penyesuaian manual nilai ambang batas (*threshold*) untuk menentukan apakah jawaban benar atau salah.
- b. Model Umpan Balik (LLM *Gemma*) dievaluasi dengan menggunakan dataset *benchmark* untuk mendeteksi potensi halusinasi, dengan membandingkan output model terhadap jawaban benar, salah, dan terbaik yang bertujuan untuk mengurangi halusinasi.

5. Penjelasan Model (*Model Explanation*)

Tahap ini menggunakan pendekatan *Explainable AI* (XAI) untuk menjelaskan proses pengambilan keputusan oleh model *pre-trained Sentence Transformers*, guna meningkatkan transparansi dan kepercayaan. Dua metode XAI yang digunakan adalah SHAP dan LIME, yang bertujuan untuk memvisualisasikan kata-kata yang memengaruhi skor yang dihasilkan model.

6. Penerapan Model (*Model Deployment*)

Tahap ini melibatkan *deployment* dua model menggunakan *Flask API* dengan *endpoint* terpisah. Model penilaian menerima input JSON berisi soal, kunci jawaban, jawaban siswa, dan jenis soal, lalu menghasilkan skor kemiripan. Model umpan balik menerima data serupa ditambah skor dan bobot soal, yang digunakan untuk menghasilkan umpan balik kontekstual melalui LLM *Gemma*. Kedua *endpoint* mengembalikan output dalam format JSON.

7. Desain Sistem (*System Design*)

Tahap ini bertujuan mengidentifikasi kebutuhan sistem secara mendalam melalui wawancara dengan tenaga pendidik di SMK Grafika Yayasan Lektor, diperoleh informasi untuk merancang sistem. Hasilnya divisualisasikan dalam bentuk Use Case diagram untuk menggambarkan interaksi pengguna terhadap sistem.

8. Implementasi Sistem (*System Implementation*)

Tahap ini mencakup pengembangan aplikasi berbasis *learning management system* (LMS). Aplikasi dibangun menggunakan Laravel 11 dan Tailwindcss, serta terhubung dengan database MySQL. Sistem ini juga diintegrasikan dengan model *machine learning* yang telah dideploy secara terpisah melalui *Flask API*.

9. Pengujian Sistem (*System Testing*)

Tahap ini menguji sistem menggunakan *Black Box*, SUS, dan UAT. *Black Box* menguji fungsionalitas fitur, sementara SUS dan UAT melibatkan guru dan siswa untuk menilai kelayakan dan kesesuaian sistem dengan kebutuhan pengguna.

10. Penerapan Sistem (*System Deployment*)

Tahap ini meliputi implementasi sistem ke lingkungan produksi dengan mendeploy aplikasi ke *server hosting* dan memastikan koneksi database berjalan lancar

11. Laporan Akhir (*Final Report*)

Tahap ini menyusun laporan akhir yang mendokumentasikan seluruh proses pengembangan aplikasi.

C. Objek Penelitian

Objek penelitian dari penelitian ini adalah sistem penilaian dan umpan balik otomatis berbasis web yang terintegrasi dalam Learning Management System (LMS), yang akan digunakan di SMK Grafika Yayasan Lektor sebagai studi kasus.

II. HASIL DAN PEMBAHASAN

A. Pendefinisian Masalah (*Problem Definition*)

Hasil observasi dan wawancara di SMK Grafika Yayasan Lektor mengungkap beberapa permasalahan dalam penilaian, yaitu: subjektivitas dalam penilaian, beban kerja guru yang tinggi, waktu koreksi yang lama, serta tidak adanya umpan balik bagi siswa. Hal ini menunjukkan perlunya sistem penilaian otomatis yang konsisten dan mampu memberikan umpan balik yang bersifat membangun.

B. Pengumpulan dan Persiapan Data (*Data Acquisition and Preparation*)

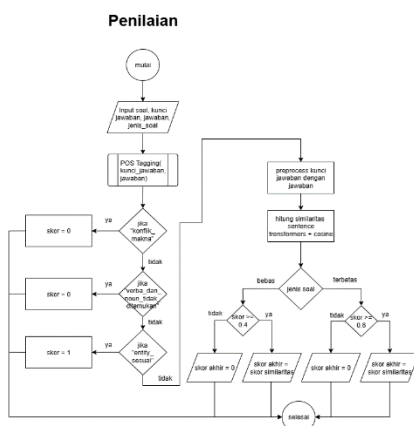
Data diperoleh dari tujuh mata pelajaran di SMK Grafika Yayasan Lektor, terdiri dari 100 entri berisi soal, jawaban siswa, kunci jawaban, jenis soal, dan label (benar/salah). Sebanyak 80% digunakan untuk menentukan nilai *threshold* guna mengoptimalkan *hyperparameter*, sedangkan 20%

sisanya digunakan untuk validasi sistem terhadap data yang belum pernah dilatih sebelumnya.

C. Pengembangan dan Pelatihan Model (*Model Development and Training*)

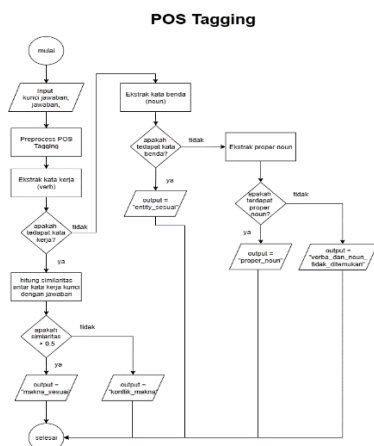
1. Model Penilaian

Model penilaian dikembangkan menggunakan *Sentence Transformers* dan *Part of Speech (POS) Tagging* untuk memahami makna atau tujuan kalimat. Terdapat dua jenis soal yang diuji, yaitu soal terbatas (seperti definisi atau proses) dan soal bebas (seperti opini atau penjabaran manfaat). Pengujian *threshold* memperoleh nilai ambang batas optimal sebesar 0,6 untuk soal terbatas dan 0,4 untuk soal bebas.



Gambar 2 Flowchart Model Penilaian

Dilakukan input data berupa soal, kunci jawaban, dan jawaban siswa untuk kemudian diproses. Tahap selanjutnya adalah menjalankan subproses POS tagging seperti ditunjukkan pada gambar berikut:



Gambar 3 Flowchart POS Tagging

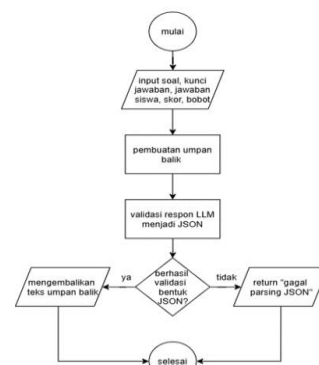
Dalam tahap pemrosesan POS tagging, dilakukan pemrosesan selanjutnya, kunci

jawaban dan jawaban siswa akan diproses untuk dipecah menjadi kata-kata, kemudian diidentifikasi kata kerja, kata benda, dan *proper noun* (kata benda khusus). Hasil dari proses ini adalah pengambilan keputusan apakah kedua teks memiliki makna yang sama, mengandung konflik makna, entitas yang sesuai, *proper noun* yang cocok, atau kata kerja dan kata benda tidak ditemukan. Hasil dari proses ini kemudian dievaluasi berdasarkan ketentuan berikut:

- Jika output dari subproses adalah “makna_sesuai”, yang berarti memiliki maksud yang sama, maka proses penilaian akan dilanjutkan ke tahap berikutnya.
- Jika output yang dihasilkan adalah “konflik_makna”, artinya makna pada jawaban bertentangan dengan kunci jawaban. Dalam kasus ini, model secara otomatis akan memberikan skor kemiripan akhir sebesar 0.
- Jika output adalah “entity_sesuai”, yang berarti terdapat kesamaan kata benda antara kunci jawaban dan jawaban siswa, maka model akan memberikan skor kemiripan sebesar 1.
- Jika output adalah “proper_noun”, yang berarti tidak ditemukan kata benda umum, namun ditemukan *proper noun*, maka proses penilaian akan tetap dilanjutkan ke tahap berikutnya.
- Jika output yang dihasilkan adalah “verba_dan_noun_tidak_ditemukan”, artinya tidak ditemukan kata kerja maupun kata benda pada kedua teks. Dalam kondisi ini, model menyimpulkan bahwa jawaban tidak relevan dan memberikan skor sebesar 0.

2. Model Umpan Balik

Umpan Balik



Gambar 4 Flowchart Model Umpan Balik

Dalam proses pembuatan umpan balik, sistem menerima input berupa data soal, kunci jawaban, jawaban siswa, bobot soal, dan skor yang diperoleh. Pembuatan umpan balik dilakukan dengan bantuan model LLM *Gemma 3 27B* yang

diakses melalui layanan OpenRouter. Model ini menghasilkan teks umpan balik berdasarkan konteks soal, kunci jawaban, jawaban siswa, skor penilaian, dan bobot soal.

Digunakan teknik *prompt engineering*, yaitu metode menyusun perintah (*prompt*) secara spesifik, jelas serta instruksi eksplisit agar model LLM memberikan respons yang sesuai. Output yang dihasilkan kemudian divalidasi dengan menggunakan library *Pydantic*, untuk memastikan output dari model LLM memiliki struktur sesuai dengan format yang diharapkan.

D. Evaluasi Model (*Model Evaluation and Refinement*)

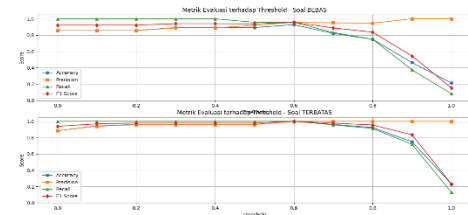
Tahap ini bertujuan mengevaluasi kinerja model yang didasarkan pada hasil pengujian kedua model selama proses pengembangan.

1. Evaluasi Model Penilaian
 - a. Evaluasi *Part of Speech (POS) Tagging*

POS tagging digunakan untuk meningkatkan akurasi penilaian otomatis dengan membantu model memahami makna kalimat. *Sentence Transformers*, efektif dalam menghitung kemiripan kata dan struktur, namun kesulitan membedakan kalimat yang mirip secara struktural namun bertentangan maknanya. Misalnya, "Fotosintesis menghasilkan oksigen" dan "Fotosintesis tidak menghasilkan oksigen" memiliki struktur dan kata yang mirip, namun maknanya bertentangan. Tanpa *POS tagging*, *Sentence Transformers* dapat memberikan skor kemiripan yang tinggi. Dengan menggabungkan *Sentence Transformers* dan *POS tagging*, sistem dapat mendeteksi perbedaan makna seperti negasi atau kontradiksi, sehingga dapat menghasilkan skor kemiripan yang lebih akurat.

- b. Evaluasi Nilai Ambang Batas (*Threshold*)

Nilai ambang batas (*threshold*) digunakan untuk menentukan batas minimum kemiripan antara jawaban siswa dan kunci jawaban yang dianggap benar. Dalam penelitian ini, *threshold* ditentukan secara empiris melalui percobaan pada 80% data uji dari dataset yang berjumlah 100. Pengujian dilakukan dengan mencoba berbagai nilai *threshold* dari 0.10 hingga 1.00, untuk mengevaluasi dampaknya terhadap penilaian model.



Gambar 5 Hasil Metrik Evaluasi Nilai Threshold

Hasil evaluasi menunjukkan untuk soal jenis bebas, *threshold* 0.40 memberikan nilai optimal, dengan *recall* sempurna (1.00) dan *F1-score* 0.94, memungkinkan model mengenali variasi jawaban siswa. Sementara untuk soal jenis terbatas, *threshold* 0.60 menghasilkan performa terbaik dengan nilai akurasi, *presisi*, *recall*, dan *F1-score* sebesar 1.00, cocok untuk menilai jawaban yang lebih tepat.

c. Evaluasi Akurasi Model

Model diuji menggunakan 20% data sisanya sebagai data validasi. evaluasi dilakukan dengan menerapkan *threshold* yang telah diuji sebelumnya: *threshold* 0.60 untuk soal terbatas, karena memberikan hasil evaluasi terbaik, dan *threshold* 0.40 untuk soal bebas.



Gambar 6 Confusion Matrix evaluasi akurasi model

Hasil pengujian menunjukkan bahwa untuk soal bebas, *threshold* 0.40 memberikan *F1-score* yang lebih tinggi dan *recall* sempurna, yang menunjukkan kemampuan model dalam mengenali variasi jawaban siswa. Untuk soal terbatas, *threshold* 0.60 dipertahankan untuk memastikan penilaian yang tepat.

2. Evaluasi Model Umpan Balik
 - a. Pengujian Halusinasi Model LLM

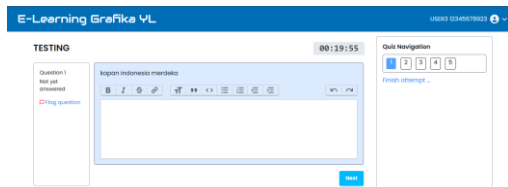
Evaluasi model *Large Language Model* (LLM) dalam memberikan umpan balik bebas dari halusinasi, yaitu kondisi di mana model memberikan informasi yang tampak benar namun tidak sesuai fakta. Pengujian dilakukan menggunakan dataset *benchmarking* dari Github yang berisi 818 data.

Berdasarkan *Use Case*, sistem melibatkan tiga aktor utama: admin, guru, dan siswa, masing-masing dengan peran dan akses berbeda. Semua aktivitas diawali dengan proses login. Admin mengelola data master seperti user, jurusan, sekolah, tahun akademik, kelas, dan mata pelajaran. Guru bertugas mengelola materi, membuat quiz, dan

menilai. Siswa mengikuti ujian yang dibuat guru dan melihat hasilnya.

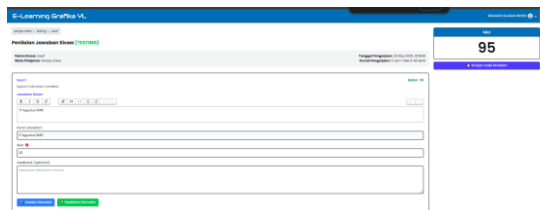
H. Implementasi Sistem (*System Implementation*)

Berikut merupakan implementasi sistem yang menunjukkan alur kerja aplikasi secara keseluruhan, seperti ditampilkan pada gambar di bawah ini.



Gambar 10 Halaman ujian siswa

Gambar 10 merupakan tampilan halaman quiz dari sisi siswa yang sedang melaksanakan *quiz*. siswa dapat menjawab soal yang tersedia serta melakukan navigasi antar soal dengan mudah.



Gambar 11 Halaman koreksi ujian

Gambar 11 adalah tampilan dalam proses koreksi, guru dapat memanfaatkan fitur utama aplikasi, yaitu tombol “koreksi otomatis” dan tombol “feedback otomatis”. Selain itu, guru juga diberikan kebebasan untuk melakukan penilaian dan umpan balik secara manual.

I. Pengujian Sistem (*System Testing*)

Pengujian sistem dilakukan untuk memastikan aplikasi berjalan sesuai kebutuhan pengguna, dengan tiga pendekatan utama: *Black Box Testing*, *User Acceptance Testing* (UAT), dan *System Usability Scale* (SUS).

a. *Black Box Testing*

Hasil pengujian Black Box menunjukkan semua skenario berjalan baik dan aplikasi siap digunakan, namun masih ada kemungkinan munculnya kasus tak terduga di penggunaan nyata karena pengujian hanya mencakup skenario tertentu.

b. *User Acceptance Testing* (UAT)

User Acceptance Testing (UAT) adalah pengujian yang ditujukan untuk pengguna akhir [9]. Dalam penelitian ini, 17 siswa dan 8 guru menguji langsung aplikasi web serta menilai kuesioner yang mencakup tiga aspek: pengalaman pengguna, fungsionalitas aplikasi, dan penerapan model machine learning.

Berdasarkan hasil pengujian siswa, rata-rata tingkat kepuasan siswa berkisar antara 84,7% hingga 95,3%. Sementara itu, pada pengujian guru, rata-rata tingkat kepuasan guru berkisar antara 85% hingga 95%.

Secara keseluruhan, aplikasi diterima dengan baik oleh guru dan siswa, baik dari segi kemudahan penggunaan, fungsionalitas, maupun kualitas fitur penilaian dan umpan balik otomatis.

c. *System Usability Scale* (SUS)

Pengujian SUS juga dilakukan oleh guru dan siswa di SMK Grafika Yayasan Lektur, yang terfokus pada tingkat kegunaan aplikasi. Pengujian ini menggunakan sepuluh pertanyaan SUS dengan format jawaban skala *Likert* 1 hingga 5, yang mencakup kalimat negatif dan positif [10].

Berdasarkan hasil perhitungan menggunakan metode SUS, diperoleh nilai rata-rata sebesar 89,3. Nilai ini tergolong cukup tinggi dan termasuk dalam kategori *grade* “A” menurut standar SUS, yang menunjukkan bahwa aplikasi yang dikembangkan berfungsi dengan baik dan mendapatkan respons positif dari pengguna.

III. KESIMPULAN DAN SARAN

Penelitian ini berhasil mengembangkan aplikasi penilaian dan umpan balik otomatis berbasis *Learning Management System* dengan mengimplementasikan model *pre-trained Sentence Transformers* dan *part of speech tagging* untuk mengonversi teks menjadi representasi vektor. Kemudian, tingkat kemiripan dihitung menggunakan *cosine similarity* dengan menerapkan ambang batas (*threshold*) optimal, yaitu 0,4 untuk soal bebas dan 0,6 untuk soal terbatas, yang ditentukan melalui pengujian secara mandiri dan menghasilkan akurasi sebesar 95% pada data uji. Selain itu, model LLM *Gemma* dimanfaatkan untuk menghasilkan umpan balik yang efektif dengan teknik *prompt engineering*, yang memberikan respons terstruktur dalam format JSON mencakup empat komponen utama yaitu: penjelasan kesalahan, kunci jawaban yang benar, saran perbaikan, dan motivasi.

Oleh karena itu, untuk pengembangan selanjutnya, disarankan untuk terus meningkatkan kualitas model, termasuk menyempurnakan proses analisis semantik dan meningkatkan performa POS *tagging*. Selain itu, perlu memperluas skala dan variasi dataset guna meningkatkan kemampuan model agar lebih adaptif dan akurat dalam berbagai konteks. Perbaikan juga perlu dilakukan pada sisi infrastruktur server dan mekanisme *deployment* model, sehingga aplikasi dapat berjalan lebih ringan, cepat, dan efisien.

IV. REFERENSI

- [1] E. L. Amalia¹, A. J. Jumadi, I. A. Mashudi³, W. Wibowo⁴, P. N. Malang, and P. Korespondensi, "Analisis Metode Cosine Similarity Pada Aplikasi Ujian Online Esai Otomatis (Studi Kasus Jti Polinema) Cosine Similarity Method Analysis on Automatic Esai Online Test Application," *J. Teknol. Inf. dan Ilmu Komput.*, vol. 8, no. 2, pp. 343–348, 2021, doi: 10.25126/jtiik.202184356.
- [2] U. D. Mariyani, W. Setyaningsih, and R. Agustina, "Pengembangan Sistem Koreksi Jawaban Esai Otomatis Menggunakan Naive Bayes Dan Pengujian Menggunakan User Acceptance Test (UAT)," *Rainstek J. Terap. Sains dan Teknol.*, vol. 4, no. 1, pp. 61–73, 2022, doi: 10.21067/jtst.v4i1.6857.
- [3] H. K. Bharata and H. Sulistyowati, "Optimasi Sistem Penilaian Ujian Essay Online Menggunakan Support Vector Machine (SVM) Dan Latent Semantic Analysis (LSA) Dengan Bahasa R," *Sci. J. Inf. Syst. Technol. Appl. Comput. Eng.*, vol. 10, no. 2, pp. 63–70, 2020.
- [4] N. Chamidah, M. M. Santoni, H. N. Irmanda, and R. Astriratma, "Penilaian Esai Pendek Otomatis dengan Pencocokan Kata Kunci Frasa Nomina," *Techno.Com*, vol. 20, no. 4, pp. 489–498, 2021, doi: 10.33633/tc.v20i4.5043.
- [5] A. R. Lahitani, "Automated Essay Scoring menggunakan Cosine Similarity pada Penilaian Esai Multi Soal," vol. 22, no. 2, pp. 107–118, 2022.
- [6] G. S. Ascione and V. Sterzi, "A comparative analysis of embedding models for patent similarity," 2022.
- [7] F. Azimah and K. Rizky Nova Wardani, "Sistem Pendeteksi Gejala Awal Covid-19 dengan Penggunaan Metode Al Project Cycle," *J. Locus Penelit. dan Pengabd.*, vol. 1, no. 6, pp. 405–418, 2022, doi: 10.36418/locus.v1i6.135.
- [8] M. Renze, "The Effect of Sampling Temperature on Problem Solving in Large Language Models," pp. 1–12, 2024.
- [9] M. A. Chamida, A. Susanto, and A. Latubessy, "Analisa User Acceptance Testing Terhadap Sistem Informasi Pengelolaan Bedah Rumah Di Dinas Perumahan Rakyat Dan Kawasan Permukiman Kabupaten Jepara," *Indones. J. Technol. Informatics Sci.*, vol. 3, no. 1, pp. 36–41, 2021, doi: 10.24176/ijtis.v3i1.7531.
- [10] I. K. Wardani, P. Utomo, A. Budiman, and D. N. Amadi, "Pemanfaatan Metode Design Thinking dan Pengujian SUS untuk UI/UX Aplikasi Home Care Madiun Berbasis Android," *J. Comput. Inf. Syst. Ampera*, vol. 4, no. 2, pp. 106–125, 2023, [Online]. Available: <https://journal-computing.org/index.php/journal-cisa/index>