PENGEMBANGAN OFFICE ADD-IN ASISTEN PENULISAN DOKUMEN BERBASIS *LARGE LANGUAGE MODEL*

SKRIPSI

Program Studi Sarjana Informatika
Jurusan Informatika

Oleh:
REISYA PRATAMA
NIM D1041221032



FAKULTAS TEKNIK
UNIVERSITAS TANJUNGPURA
PONTIANAK
2026

HALAMAN PERNYATAAN

Yang bertanda tangan di bawah ini:

Nama : Reisya Pratama

NIM : D1041221032

menyatakan bahwa dalam SKRIPSI yang berjudul "Pengembangan Office Add-in Asisten Penulisan Dokumen Berbasis *Large Language Model*" tidak terdapat karya yang pernah diajukan untuk memperoleh gelar sarjana di suatu perguruan tinggi manapun. Sepanjang pengetahuan Saya, tidak terdapat karya atau pendapat yang pernah ditulis atau diterbitkan oleh orang lain, kecuali yang secara tertulis diacu dalam naskah ini dan disebutkan dalam Daftar Rujukan.

Demikian pernyataan ini dibuat dengan sebenar-benarnya. Saya sanggup menerima konsekuensi akademis dan hukum di kemudian hari apabila pernyataan yang dibuat ini tidak benar.

Pontianak, 15 Juli 2026



Reisya Pratama
D1041221032

KEMENTERIAN PENDIDIKAN TINGGI, SAINS,
DAN TEKNOLOGI
UNIVERSITAS TANJUNGPURA
FAKULTAS TEKNIK
Jalan Prof. Dr. H. Hadari Nawawi Pontianak 78124
Telepon (0561) 740186, WA: +6282152280907
Email : ft@untan.ac.id Website : http://teknik.untan.ac.id

HALAMAN PENGESAHAN

PENGEMBANGAN OFFICE ADD-IN ASISTEN PENULISAN DOKUMEN BERBASIS *LARGE LANGUAGE MODEL*

Jurusan Informatika
Program Studi Sarjana Informatika

Oleh:

Reisya Pratama

NIM. D1041221032

Telah dipertahankan di depan Penguji Skripsi pada tanggal 15 Juli 2026
dan diterima sebagai salah satu persyaratan untuk memperoleh gelar sarjana

Susunan Penguji Skripsi

Dosen Pembimbing Utama	: Anggi Perwitasari, S.T., M.T. NIP. 198908192019032012
Dosen Pembimbing Kedua	: M. Tsana'uddin Farid, S.Kom., M.Eng. NIP. 199609162024061001
Dosen Penguji Utama	: H. Hengky Anra, S.T., M.Kom. NIP. 197503251999031005
Dosen Penguji Kedua	: Izuardo Zulkarnain, S.Pd., M.T. NIP. 199405172024061001

Pontianak, 15 Juli 2026
Dekan,

Dr.-Ing. Ir. Slamet Widodo, MT, IPM
NIP. 196712231992031002

Pembimbing Utama

Anggi Perwitasari, S.T., M.T.
NIP. 198908192019032012

ii

Skripsi ini didedikasikan kepada kedua orang tuaku Bapak Syaiful Bahri dan Ibu Eny Farida, saudara Risyafa Dwi Rahmatika, serta segenap keluarga besar yang senantiasa memberikan doa dan dukungan**.** Apa yang mereka telah berikan melebihi daripada apa yang pernah aku inginkan.
Terima kasih yang mendalam untuk seluruh Dosen dan Staf Informatika yang telah membagikan ilmu serta memberikan pelayanan terbaik selama masa perkuliahan.

Penghargaan dan rasa syukur yang luar biasa kupersembahkan untuk teman-teman seperjuangan yang telah berproses, berdiskusi, dan saling bahu-membahu menyelesaikan setiap tantangan bersama, khususnya untuk sahabat-sahabat di Plan B dan Pink Flag yang selalu menjadi pendukung, berbagi cerita, dan memberikan bantuan yang tak ternilai harganya. Terima kasih pula kepada seluruh keluarga besar Informatika Angkatan 2022 atas kebersamaan yang luar biasa, serta seluruh rekan responden yang telah meluangkan waktu dalam pengujian sistem.

Dari titisan ilmu-Mu kupersembahkan goresan tanganku bagi ilmu pengetahuan Indonesia

PRAKATA

Puji syukur kehadirat Allah SWT atas segala rahmat, berkah, dan hidayah-Nya, sehingga skripsi dengan judul **"**Pengembangan Office Add-in Asisten Penulisan Dokumen Berbasis *Large Language Model***"** ini dapat diselesaikan dengan baik. Skripsi ini disusun sebagai salah satu persyaratan untuk memperoleh gelar sarjana pada Program Studi Sarjana Informatika, Jurusan Informatika, Fakultas Teknik, Universitas Tanjungpura. Penelitian ini sendiri berfokus pada pengembangan aplikasi Add-in Microsoft Word yang terintegrasi dengan *Large Language Model* (LLM) untuk membantu proses parafrase akademik dengan menerapkan teknik *Masking* Entitas untuk mengamankan format sitasi dan rumus matematika.

Keberhasilan penyusunan skripsi ini tidak lepas dari bimbingan, dukungan, dan bantuan dari berbagai pihak. Oleh karena itu, penulis menyampaikan terima kasih yang sebesar-besarnya kepada Ibu Anggi Perwitasari, S.T., M.T., selaku Dosen Pembimbing Utama, dan Bapak M. Tsana'uddin Farid, S.Kom., M.Eng., selaku Dosen Pembimbing Kedua, yang telah memberikan banyak bimbingan, arahan, masukan, serta diskusi yang intensif. Ucapan terima kasih juga penulis tujukan kepada Bapak H. Hengky Anra, S.T., M.Kom., selaku Dosen Penguji Utama, dan Bapak Izuardo Zulkarnain, S.Pd., M.T., selaku Dosen Penguji Kedua, atas segala koreksi, saran, dan masukan konstruktifnya, serta kepada Bapak Mariyadi, S.Pd., M.Pd selaku validator dalam tahapan *Human Validation* yang telah membantu menyempurnakan penelitian ini.

Penulis menyadari bahwa skripsi ini masih jauh dari kesempurnaan, oleh karena itu penulis mengharapkan masukan konstruktif agar dapat menyempurnakannya di masa mendatang. Akhir kata, penulis berharap penelitian ini dapat memberikan manfaat nyata bagi perkembangan ilmu pengetahuan di Indonesia.

Pontianak, 15 Juli 2026



Reisya Pratama

ABSTRAK

Peningkatan penggunaan alat bantu parafrase berbasis *Large Language Model* (LLM) di kalangan mahasiswa untuk penyusunan naskah akademik masih menghadapi kendala fragmentasi alur kerja akibat ketergantungan pada platform web eksternal yang terpisah dari perangkat lunak standar Microsoft Word. Dari sisi teknis, sifat model generatif yang bekerja secara probabilistik rawan menimbulkan halusinasi yang merusak elemen dokumen bersifat deterministik atau mutlak, seperti format sitasi otomatis dan notasi rumus matematika, serta berisiko mengalami pergeseran makna yang dapat mengubah atau menghilangkan esensi dari argumen ilmiah aslinya. Penelitian ini bertujuan untuk mengembangkan sebuah *Office* Add-in Microsoft Word berbasis model Gemini 3 Flash untuk mempermudah proses parafrase akademik secara langsung di dalam jendela penulisan. Pengembangan sistem menggunakan model *Prototyping* dengan menerapkan teknik *Masking* Entitas pada level struktur *Office Open XML* (OOXML) untuk melindungi keutuhan format sitasi (Mendeley dan Zotero) serta rumus matematika menggunakan token penanda khusus (*placeholder*). Untuk menjamin kualitas hasil penulisan, diimplementasikan arsitektur *Multi-agent Cyclic* berupa siklus umpan balik antara agen generator dan agen *evaluator* yang mengintegrasikan metrik ParaPLUIE sebagai penilai internal secara *real-time* serta metrik BLEU untuk mengukur tingkat variasi kata (*lexical diversity*). Hasil pengujian integritas data menunjukkan tingkat keberhasilan 100% dalam melindungi entitas akademik pada sampel berentitas, meskipun ditemukan celah teknis berupa 1 kasus (0,71%) halusinasi *placeholder* pada teks tanpa entitas. Pengujian efektivitas orkestrasi agen mencatat bahwa 98,02% dari 202 sesi berhasil memenuhi standar metrik pada iterasi pertama, dengan kemampuan pemulihan kasus pergeseran makna pada iterasi lanjutan. Evaluasi *Human Validation* terhadap hasil parafrase melalui metode triangulasi penilai (pakar kebahasaan dan mahasiswa) menghasilkan skor rata-rata gabungan sebesar 4,3812 dari skala 5,00 (87,62%). Terakhir, pengujian *usability* menggunakan *System Usability Scale* (SUS) terhadap 42 responden memperoleh skor rata-rata 72,02 (Grade B / kategori "Baik"), membuktikan bahwa aplikasi layak dan dapat diterima (*acceptable*) oleh pengguna untuk menunjang aktivitas penulisan karya ilmiah mahasiswa.

Kata kunci: *Large Language Model*, *Masking* Entitas, *Office* Add-in, Parafrase Akademik, ParaPLUIE.

ABSTRACT

The increasing use of *Large Language Model* (LLM)-based paraphrasing tools among university students for academic writing continues to face workflow fragmentation due to reliance on external web platforms that are disconnected from the standard Microsoft Word environment. From a technical perspective, the probabilistic nature of generative models makes them prone to hallucinations that can compromise deterministic document elements, such as automated citation formatting and mathematical notation, while also introducing semantic shifts that may alter or omit the original scientific arguments. This study aims to develop a Microsoft Word Office Add-in powered by the Gemini 3 Flash model to facilitate academic paraphrasing directly within the writing interface. The system was developed using the Prototyping model by implementing an Entity *Masking* technique at the Office Open XML (OOXML) structural level to preserve the integrity of citation formats (Mendeley and Zotero) and mathematical equations through the use of special *placeholder* tokens. To ensure paraphrase quality, a *Multi-agent Cyclic* architecture was implemented, consisting of a feedback loop between a generator agent and an *evaluator* agent that integrates the ParaPLUIE metric for *real-time* internal quality assessment and the BLEU metric to measure lexical diversity. Data integrity testing achieved a 100% success rate in protecting academic entities in samples containing such entities, although one technical issue (**0.71%**) involving *placeholder* hallucination was observed in text without academic entities. Evaluation of the agent orchestration demonstrated that 98.02% of 202 sessions satisfied the predefined quality metrics on the first iteration, with subsequent iterations successfully recovering cases involving semantic drift. Human validation of the paraphrased outputs, conducted using a triangulation approach involving a language expert and university students, produced a combined average score of 4.3812 out of 5.00 (87.62%)**.** Finally, usability evaluation using the System Usability Scale (SUS) with 42 respondents yielded an average score of 72.02 (Grade B, "Good" category), indicating that the application is both usable and acceptable for supporting students in academic writing.

Keywords: *Academic Paraphrasing*, *Large Language Model*, *Office* Add-in, *Masking* Entitas, *ParaPLUIE.*

DAFTAR ISI

HALAMAN PERNYATAAN i
HALAMAN PENGESAHAN.....**Error! Bookmark not defined.**
PRAKATA iv
ABSTRAK v
ABSTRACT vi
DAFTAR ISI vii
DAFTAR GAMBAR xi
DAFTAR TABEL xii
DAFTAR KODE PROGRAM xiii
DAFTAR LAMPIRAN xiv
GLOSARIUM xv
BAB I PENDAHULUAN 1
1.1 Latar Belakang 1
1.2 Perumusan Masalah 3
1.3 Tujuan dan Sasaran Penelitian 3
1.4 Pembatasan Masalah 3
1.5 Sistematika Penulisan 4
BAB II TINJAUAN PUSTAKA 6
2.1 Penelitian Terkait 6
2.2 Parafrase 7
2.3 Office Add-In 7
2.3.1 Office.js 8
2.3.2 OOXML Markdown 8
2.4 Node.js 9
2.5 *Large Language Model* 9
2.5.1 Gemini 3 Flash 10
2.5.2 Gemini 2.5 Flash Lite 11
2.5.3 Vertex AI 12
2.6 LogProb 12
2.7 *Prompt Engineering* 13
2.7.1 Prompt LM-CPPF 14

2.7.2 Gemini Best Practice Prompt 14
2.7.3 Prompt EPA (Easy Prompt Augmentation) 15
2.8 Format Pertukaran Data (JSON) 17
2.9 Basis Data 18
2.9.1 PostgreSQL 18
2.9.2 Supabase 19
2.10 Arsitektur *Pipeline* 19
2.11 *Multi-Agent Cyclic* 20
2.12 *Regular Expression* 20
2.13 *Masking* Entitas & *Placeholder* 21
2.14 Metrik Evaluasi Teks 22
2.14.1 Metrik BLEU 22
2.14.2 Metrik ParaPLUIE 24
2.15 *Flowchart* Diagram 26
2.16 Model *Prototype* 27
2.17 Pengujian Sistem 28
2.17.1 Black Box Testing 28
2.17.2 Human Validation 29
2.17.3 System Usability Scale 31
2.18 Korelasi Spearman 32
BAB III METODOLOGI PENELITIAN 34
3.1 Metode Pengembangan Sistem 34
3.2 Alur Penelitian 35
3.3 Alat & Bahan Penelitian 36
3.3.1 Perangkat Keras 36
3.3.2 Perangkat Lunak 37
3.4 Analisis Kebutuhan 37
3.4.1 Analisis Pengguna 38
3.4.2 Kebutuhan Fungsional 38
3.4.3 Kebutuhan Non-Fungsional 39
3.5 Perancangan Prototype 39
3.5.1 Arsitektur Sistem 39

3.5.2 Arsitektur *Pipeline* 42
3.5.3 Skema Basis Data 45
3.5.4 Alur Kerja Sistem 48
3.5.5 Algoritma *Masking* 52
3.5.6 Algoritma Generasi dan Evaluasi (*Multi-Agent Cyclic*) 55
3.5.7 Perancangan *Prompt* 59
3.5.8 Algoritma *Unmasking* 61
3.5.9 Perancangan Antarmuka 65
3.6 Implementasi 67
3.6.1 Pengembangan *Frontend* (*Office* Add-in) 67
3.6.2 Pengembangan *Backend* dan Integrasi API 67
3.7 Pengujian 68
3.7.1 Pengujian Fungsional (*Black Box Testing*) 68
3.7.2 Pengujian Human Validation 69
3.7.3 Pengujian Usability (*System Usability Scale*) 70
3.8 Analisis 70
3.8.1 Analisis Kinerja Algoritma *Masking* Entitas 71
3.8.2 Analisis Kinerja dan Mekanisme Arsitektur *Multi-agent Cyclic* 71
3.8.3 Analisis *Human Validation* dan Korelasi Metrik 72
BAB IV HASIL DAN PEMBAHASAN 73
4.1 Implementasi Sistem 73
4.1.1 Pengembangan Prototipe Awal 73
4.1.2 Pengembangan Prototipe Lanjutan 83
4.2 Hasil Implementasi 92
4.2.1 Antarmuka Utama dan *mode* Pemrosesan 92
4.2.2 Mekanisme Pemrosesan dan Visualisasi Hasil Parafrase 93
4.2.3 Fitur Evaluasi dan Umpan Balik 95
4.2.4 Aksi Finalisasi (*Insert & Copy*) 97
4.3 Hasil Pengujian 98
4.3.1 BlackBox Testing 98
4.3.2 Human Validation 100
4.3.3 System Usability Scale 104

4.4 Analisis dan Pembahasan 105
4.4.1 Analisis Kinerja Algoritma *Masking* Entitas 105
4.4.2 Analisis Kinerja dan Mekanisme Arsitektur *Multi-agent Cyclic*..... 107
4.4.3 Korelasi Metrik Internal dengan Human Validation 112
BAB V PENUTUP..... 116
5.1 Kesimpulan..... 116
5.2 Saran..... 117
DAFTAR RUJUKAN 119

DAFTAR GAMBAR

Gambar 3.1 *Flowchart* Alur Penelitian 35
Gambar 3.2 Arsitektur Sistem 40
Gambar 3.3 Arsitektur *Pipeline* Sistem 42
Gambar 3.4 Skema Basis Data 45
Gambar 3.5 *Flowchart* Alur Kerja Sistem 49
Gambar 3.6 *Flowchart* Algortima *Masking* Entitas 52
Gambar 3.7 *Flowchart* Algoritma Generasi dan Evaluasi 56
Gambar 3.8 *Flowchart* Perancangan *Prompt* 59
Gambar 3.9 *Flowchart Unmasking* Entitas 61
Gambar 3.10 Rancangan Antarmuka (*Wireframe*) 65
Gambar 4.1 Panel Utama dan Tampilan Add-Ins pada Microsoft Word 92
Gambar 4.2 Tampilan Saat Teks Terseleksi 93
Gambar 4.3 Tampilan Saat Proses Parafrase Berjalan 93
Gambar 4.4 Tampilan Hasil Saat *Diff View* Aktif 94
Gambar 4.5 Tampilan Saat *Highlight* Kalimat 94
Gambar 4.6 Tampilan Hasil Saat *Diff View* Non-Aktif 95
Gambar 4.7 Tampilan Peringatan Saat Metrik Tidak Memenuhi Standar 96
Gambar 4.8 Tampilan Kotak Evaluasi dan Umpan Balik 96
Gambar 4.9 Tampilan Saat Teks Berhasil Disisipkan 97
Gambar 4.10 Tampilan Saat Teks Berhasil Disalin 97
Gambar 4.11 Proporsi Sampel Berentitas dan Entitas Terdeteksi 106
Gambar 4.12 Distribusi Skor ParaPLUIE & BLEU pada Seluruh Parafrase 109

DAFTAR TABEL

Tabel 2.1 Penelitian Terkait 6
Tabel 2.2 Penjelasan Simbol-Simbol *Flowchart*..... 26
Tabel 2.3 Skala Penilaian *Human Validation*..... 30
Tabel 2.4 Komponen Pertanyaan Kuesioner SUS..... 31
Tabel 2.5 Klasifikasi Skor *System Usability Scale* (SUS)..... 32
Tabel 3.1 Skenario Pengujian Fungsional (*Black Box Testing*)..... 69
Tabel 4.1 Hasil Pengujian Fungsional (*Black Box Testing*) Tahap Awal 82
Tabel 4.2 Ringkasan Iterasi Pengembangan Prototipe Lanjutan..... 83
Tabel 4.3 Hasil Pengujian Fungsional (*Black Box Testing*) Final..... 98
Tabel 4.4 Hasil Pengujian *Human Validation*..... 101
Tabel 4.5 Evaluasi Human Validation per *mode* Pemrosesan..... 102
Tabel 4.6 Hasil Pengolahan Data System Usability Scale 104
Tabel 4.7 Perbandingan Kinerja *mode* Formal dan *mode* Akademik..... 111
Tabel 4.8 Distribusi Kategori Latensi Pemrosesan pada Seluruh Percobaan..... 111
Tabel 4.9 Hasil Perhitungan Korelasi Spearman..... 112

DAFTAR KODE PROGRAM

Kode Program 4.1 Identifikasi Sitasi Mendeley *Reference* 74
Kode Program 4.2 *Regular Expression* untuk Objek Matematika (OMath).....75
Kode Program 4.3 Substitusi OMath dengan Token [[MATH_n]] 76
Kode Program 4.4 Substitusi Sitasi dengan Token [[CIT_n]] 76
Kode Program 4.5 Penghapusan Pembungkus oMathPara saat Un*Masking* 77
Kode Program 4.6 Ekstraksi Isi dari Paket OOXML *Content Control*.....78
Kode Program 4.7 *Regular Expression* untuk Sitasi Mendeley & Zotero.....87

DAFTAR LAMPIRAN

Lampiran A Data Seluruh Log Parafrase A-1

GLOSARIUM

Application Programming Interface (API)	Jembatan komunikasi digital yang menghubungkan sistem *backend* aplikasi dengan layanan AI Gemini melalui platform Vertex AI.
Arsitektur *Pipeline*	Metode penyusunan proses data secara berurutan, di mana hasil keluaran dari satu tahapan langsung menjadi masukan untuk tahap berikutnya.
Black Box Testing	Metode pengujian sistem yang fokus pada fungsinya (melihat kesesuaian *input* dan *output*) tanpa perlu memeriksa struktur internal atau detail kode programnya.
BLEU (Bilingual Evaluation Understudy)	Metrik atau skor untuk mengukur seberapa mirip kata-kata dalam teks hasil parafrase dengan teks aslinya.
Ceiling Effect	Fenomena dalam statistik di mana alat ukur tidak mampu lagi membedakan variasi kualitas sampel karena hampir semua sampel sudah mencapai nilai maksimal yang ditetapkan.
Context-switching	Gangguan fokus atau beban mental yang dialami pengguna saat harus terus-menerus berpindah antar-aplikasi saat sedang bekerja.
Easy Prompt Augmentation (EPA)	Metode otomatis untuk memperbanyak contoh instruksi berkualitas bagi AI dengan cara saling memparafrase contoh yang sudah ada, untuk

	meningkatkan performa model tanpa perlu mengetik banyak contoh manual.
Masking Entitas	Teknik menyembunyikan bagian penting teks (seperti sitasi atau rumus) dengan label sementara agar bagian tersebut aman dan tidak ikut rusak/berubah saat teks diparafrase oleh AI.
Few-shot Prompts	Teknik memberikan beberapa contoh acuan langsung di dalam instruksi agar AI memahami format atau hasil akhir yang diinginkan pengguna.
Diagram Flowchart	Diagram visual yang menggunakan simbol-simbol standar untuk menggambarkan langkah-langkah logis dari alur kerja suatu sistem atau algoritma.
Halusinasi (AI Hallucination)	Kondisi ketika AI membuat-buat informasi sendiri yang tidak keliru.
Human Validation	Proses pengujian langsung oleh sebagai standar utama untuk memastikan hasil parafrase akurat secara makna dan memenuhi standar penulisan ilmiah.
JSON (JavaScript Object Notation)	Format teks ringan dan standar yang digunakan untuk mengirim dan menukar data antarkomponen sistem agar mudah dibaca manusia maupun diproses komputer.

Large Language Model (LLM)	Model kecerdasan buatan berbasis teks berskala besar yang dilatih menggunakan data teks masif untuk memahami dan menghasilkan bahasa manusia yang alami.
Logprobs (Logarithmic Probabilities)	Nilai angka yang menunjukkan tingkat kepastian AI saat memilih kata demi kata dalam pembuatan teks.
Mendeley contentControls	Struktur data bawaan Microsoft Word yang digunakan oleh plugin Mendeley untuk menyimpan data referensi atau sitasi karya ilmiah agar tidak rusak saat teks diubah.
Metrik ParaPLUIE	Metode penilaian otomatis berbasis AI untuk mendeteksi kesamaan makna antara teks asli dan hasil parafrase. Skor positif berarti maknanya terjaga, sedangkan skor negatif berarti maknanya melenceng.
Model Prototype	Metode pengembangan sistem yang dilakukan secara bertahap dan berulang, dimulai dari membuat rancangan awal (prototipe) lalu disempurnakan terus-menerus berdasarkan evaluasi pengguna.
Multi-agent Cyclic	Pola desain sistem yang melibatkan beberapa peran AI yang bekerja secara berputar dalam sebuah siklus untuk memperbaiki kualitas hasil dari LLM secara otomatis.
Node.js	Lingkungan untuk menjalankan kode JavaScript di sisi *server* (*backend*) yang

	berfungsi mengatur komunikasi antara *Frontend, database*, dan layanan AI.
Office Add-in	Aplikasi tambahan berbasis web yang dipasang langsung di dalam Microsoft Word untuk memperluas fitur bawaan Word (seperti menambahkan fitur parafrase).
Office Open XML (OOXML)	Format dasar standar dokumen Microsoft Word (berupa paket file XML) yang dibongkar dalam penelitian ini untuk mendeteksi dan mengamankan letak sitasi serta rumus matematika.
OMath (Office Math)	Format khusus dari Microsoft Word yang digunakan untuk menyimpan susunan rumus dan notasi matematika secara terstruktur di dalam dokumen.
Parafrase Akademik	Proses menulis ulang ide atau kutipan ilmiah menggunakan kalimat sendiri tanpa mengubah makna aslinya, serta tetap mencantumkan sumber asli sebagai bentuk kejujuran ilmiah.
Patchwriting	Kesalahan dalam memparafrase di mana penulis hanya mengganti beberapa kata dengan sinonimnya tanpa merombak struktur kalimat secara signifikan. Teks hasilnya dinilai gagal karena masih terlalu mirip dengan sumber asli.
Placeholder	Label penanda sementara (seperti [[CIT_n]] atau [[MATH_n]]) yang digunakan untuk mengamankan posisi sitasi atau rumus

	matematika selama proses parafrase berlangsung.
Plagiarisme Parafrase	Tindakan mengambil ide orang lain lalu mengubahnya secara minimal, sehingga struktur kalimat atau pilihan katanya masih sangat mirip dan dianggap menjiplak.
Prompt Engineering	Teknik menyusun instruksi atau perintah yang jelas dan terstruktur agar model AI dapat menghasilkan jawaban yang tepat dan sesuai dengan batasan yang diinginkan.
Regular Expression (RegEx)	Pola kode khusus yang digunakan untuk mencari, mencocokkan, atau menyaring karakter teks tertentu (dalam hal ini digunakan untuk melacak kode sitasi Mendeley).
Self-correction	Kemampuan sistem untuk memeriksa dan memperbaiki kesalahan pada hasilnya sendiri secara otomatis melalui mekanisme umpan balik sebelum ditampilkan ke pengguna.
Semantic Drift	Kondisi di mana hasil parafrase mengalami perubahan arti sehingga maknanya melenceng dari maksud teks asli.
System Usability Scale (SUS)	Kuesioner standar berisi 10 pertanyaan untuk mengukur tingkat kemudahan, efisiensi, dan kenyamanan sebuah aplikasi saat digunakan oleh pengguna. Skor di atas 68 menandakan sistem layak digunakan.

Task Pane	Panel menu samping di dalam Microsoft Word yang menjadi wadah utama bagi pengguna untuk berinteraksi dengan fitur parafrase tanpa perlu menutup dokumen.
Vertex AI	Platform kecerdasan buatan terpadu milik Google Cloud yang digunakan dalam penelitian ini untuk mengakses model Gemini 3 Flash, mengelola parameter generasi, dan mengaktifkan fitur logprobs untuk perhitungan metrik ParaPLUIE.

BAB I
PENDAHULUAN

1.1 Latar Belakang

Perkembangan teknologi Kecerdasan Buatan (AI), khususnya *Large Language Model* (LLM), telah membawa perubahan besar pada alat bantu penulisan ilmiah. Perangkat parafrase yang dulunya hanya mengandalkan penggantian sinonim kata demi kata, kini telah berkembang menjadi sistem generatif yang mampu memahami konteks kalimat secara mendalam [1]. Kemampuan ini membuat penggunaan alat bantu parafrase berbasis AI meningkat pesat di kalangan mahasiswa untuk membantu meningkatkan efisiensi dan kualitas penyusunan naskah akademik, seperti laporan tugas akhir dan skripsi [2]. Meskipun teknologi ini sangat membantu dalam merombak struktur kalimat secara bervariasi, penerapannya dalam aktivitas penulisan ilmiah saat ini masih menghadapi kendala, baik dari sisi integrasi alur kerja maupun keandalan teknis dari model generatif tersebut.

Saat ini, Microsoft Word menjadi salah satu perangkat lunak standar yang banyak digunakan oleh mahasiswa dalam menyusun dokumen ilmiah [3]. Namun, sebagian besar alat bantu parafrase berbasis AI yang tersedia saat ini masih berbentuk platform web eksternal yang terpisah dari lingkungan penulisan utama. Kondisi ini menyebabkan terjadinya fragmentasi alur kerja, di mana mahasiswa harus terus-menerus menyalin teks, berpindah ke peramban web, dan menempelkannya kembali ke Microsoft Word. Perpindahan ini berpotensi menurunkan efisiensi kerja dan serta menambah durasi penyelesaian tugas [4]. Penelitian terkait menunjukkan bahwa mengintegrasikan alat pendukung langsung ke dalam aplikasi dokumen pengolah kata dapat mempermudah pengguna karena semua proses dilakukan dalam aplikasi [5]. Hal ini juga terbukti menjadi solusi yang lebih diminati, seperti yang terlihat pada kesuksesan plugin manajemen referensi Mendeley yang banyak diadopsi karena kemampuannya membantu penulisan secara langsung tanpa meninggalkan jendela kerja Word [6].

Dari sisi teknis, penerapan langsung LLM untuk kebutuhan akademik menghadapi tantangan terkait karakteristik data. Model bahasa generatif bekerja secara probabilistik, yaitu memprediksi kata berikutnya berdasarkan probabilitas [7]. Di sisi lain, naskah akademik memuat elemen-elemen penting yang bersifat

deterministik atau mutlak, seperti format sitasi otomatis dan notasi rumus matematika. Sifat probabilistik LLM ini sering kali menimbulkan halusinasi yang merusak struktur sitasi otomatis dan mengubah variabel rumus. Penelitian terkait membuktikan bahwa integrasi token penanda khusus atau *placeholder* efektif dalam memandu LLM untuk melokalisasi dan menandai batas-batas pada teks [8]. Prinsip penandaan posisi ini dapat diadaptasi dalam proses *Masking* untuk melindungi dan menyembunyikan elemen-elemen spesifik. Dengan menggunakan pendekatan ini, elemen-elemen tersebut dapat terjaga tanpa mengubah konten dan format aslinya.

Selain risiko kerusakan format, proses generasi teks pada LLM standar juga rawan mengalami pergeseran makna [9]. Pergeseran ini terjadi ketika kalimat hasil parafrase terlihat mengalir dengan alami, tetapi esensi atau maksud dari argumen ilmiah aslinya justru berubah atau hilang. Untuk menjamin kualitas hasil penulisan, sistem tidak hanya mengandalkan satu kali proses pembuatan teks, melainkan membutuhkan mekanisme evaluasi mandiri yang bekerja secara berulang [10]. Pendekatan ini dapat diwujudkan melalui arsitektur *Multi-agent Cyclic*, di mana agen pembuat teks (generator) dan agen pengoreksi (*evaluator*) bekerja dalam sebuah siklus umpan balik [11]. Di dalam siklus tersebut, metrik ParaPLUIE yang berupa LLM *judge* dapat diintegrasikan pada agen *evaluator* untuk bertindak sebagai penilai internal yang bertugas mendeteksi perubahan semantik sebelum hasil akhir diberikan kepada pengguna [12].

Berdasarkan berbagai permasalahan tersebut, penelitian ini bertujuan untuk mengembangkan sebuah aplikasi Add-in Microsoft Word untuk parafrase akademik berbasis LLM menggunakan arsitektur *Multi-agent Cyclic*. Aplikasi ini dibangun memanfaatkan *Office JavaScript API*. Di dalamnya, teknik *Masking* Entitas diimplementasikan untuk melindungi keutuhan format sitasi dan rumus matematika, sementara siklus *Multi-agent* dengan kendali metrik ParaPLUIE digunakan untuk menjaga makna teks. Sebagai tambahan, untuk hasil yang lebih optimal, performa sistem juga akan divalidasi secara internal menggunakan metrik BLEU untuk mengukur tingkat variasi kata (*lexical diversity*).

1.2 Perumusan Masalah

Dari latar belakang yang telah dijabarkan, berikut rumusan masalah dari penelitian ini:

1. Bagaimana merancang dan membangun Add-in Microsoft Word yang terintegrasi dengan *Large Language Model?*
2. Bagaimana menerapkan teknik *Masking* Entitas untuk menjaga keutuhan struktur dan format elemen sitasi dan rumus matematika pada hasil parafrase akademik?
3. Bagaimana mengimplementasikan arsitektur *Multi-agent Cyclic* dengan metrik *ParaPLUIE* sebagai mekanisme evaluasi mandiri untuk meminimalisir risiko pergeseran makna pada hasil parafrase?

1.3 Tujuan dan Sasaran Penelitian

Berdasarkan rumusan masalah yang telah ditetapkan, maka tujuan dari penelitian ini adalah:

1. Merancang dan membangun Add-in Microsoft Word yang terintegrasi dengan *Large Language Model.*
2. Mengimplementasikan teknik *Masking* Entitas pada elemen sitasi dan rumus matematika untuk menjaga keutuhan struktur dan format nya pada hasil parafrase.
3. Mengimplementasikan arsitektur *Multi-agent Cyclic* dengan metrik ParaPLUIE sebagai mekanisme evaluasi untuk meminimalisir risiko semantic drift pada hasil parafrase.

1.4 Pembatasan Masalah

Agar penelitian ini tetap fokus dan terarah, maka ditetapkan batasan masalah sebagai berikut:

1. Model AI: Gemini 3 Flash bertindak sebagai agen generator teks, dan Gemini 2.5 Flash Lite bertindak sebagai agen *evaluator* metrik.
2. Objek Proteksi (*Masking* Entitas)
    - Rumus Matematika: Terbatas pada format *Office Math* (OMath) lewat deteksi tag menggunakan *Regular Expression* (RegEx).

- Sitasi Otomatis: Terbatas pada gaya sitasi aktif dari Mendeley Desktop & Zotero, serta Mendeley Reference Manager baru (struktur *Content Controls*).
- Format Data: Proses *Masking* hanya berlaku pada elemen sitasi dan rumus yang tersimpan dalam struktur terstandarisasi Office Open XML (OOXML), seperti tag *omath* untuk rumus serta blok *field* atau *Content Controls* untuk sitasi; elemen dengan format lain (seperti rumus atau sitasi berupa gambar hasil *scan*, atau sitasi manual berupa teks biasa) berada di luar cakupan *Masking* sistem ini.

3. Karakteristik Input: Input berupa teks berbahasa Indonesia tanpa format lain, dengan batasan ukuran 1 hingga 750 kata per sesi parafrase.
4. Metrik Evaluasi: Penilaian kualitas dibatasi pada metrik ParaPLUIE dan metrik BLEU, serta tidak mengintegrasikan alat cek plagiarisme pihak ketiga (seperti Turnitin) secara langsung.

1.5 Sistematika Penulisan

Untuk memberikan gambaran yang jelas mengenai isi laporan penelitian ini, sistematika penulisan disusun ke dalam lima bab dengan uraian sebagai berikut:

BAB I PENDAHULUAN menguraikan latar belakang permasalahan yang mendasari penelitian, perumusan masalah, tujuan dan sasaran penelitian, pembatasan masalah agar penelitian tetap fokus dan terarah, serta sistematika penulisan laporan.

BAB II TINJAUAN PUSTAKA membahas penelitian-penelitian terdahulu yang relevan dengan topik penelitian, serta menguraikan landasan teori yang menjadi dasar dalam pengembangan sistem, meliputi konsep-konsep dan teknologi yang digunakan dalam penelitian ini.

BAB III METODOLOGI PENELITIAN menjelaskan metode pengembangan sistem yang digunakan, alur penelitian, alat dan bahan penelitian (perangkat keras dan perangkat lunak), analisis kebutuhan (pengguna, fungsional, dan non-fungsional), perancangan prototipe sistem yang meliputi arsitektur sistem, arsitektur *Pipeline*, skema basis data, alur kerja sistem, algoritma masking dan *Unmasking*, perancangan *prompt*, serta perancangan antarmuka. Selain itu, bab ini

juga menjelaskan tahap implementasi, pengujian, dan analisis yang dilakukan terhadap sistem.

BAB IV HASIL DAN PEMBAHASAN memaparkan hasil implementasi sistem yang telah dikembangkan, hasil pengujian fungsional, hasil pengujian *human validation*, hasil pengujian *usability* (*System Usability Scale*), serta analisis dan pembahasan terhadap hasil-hasil yang diperoleh, termasuk korelasi antara metrik evaluasi internal dengan hasil *human validation*.

BAB V PENUTUP berisi kesimpulan yang diperoleh dari keseluruhan penelitian yang telah dilakukan, serta saran-saran untuk pengembangan dan penyempurnaan sistem pada penelitian selanjutnya.