IMPLEMENTASI FEW-SHOT LEARNING UNTUK PREDIKSI KALIMAT SOLUSI DARI MASALAH PADA ARTIKEL ILMIAH MENGGUNAKAN MODEL *LARGE LANGUAGE MODELS* (LLM)

***Eka Nurul Azizah, *Sam Farisa Chaerul Haviana**
*Teknik Informatika, Fakultas Teknologi Industri, Universitas Islam Sultan Agung

Correspondence Author: ekaazizah@std.unissula.ac.id

Abstract
Pesatnya pertumbuhan jumlah artikel ilmiah menghadirkan tantangan baru dalam mengekstraksi informasi yang relevan, khususnya dalam mengidentifikasi hubungan antara permasalahan dan solusinya. Penelitian ini bertujuan untuk mengembangkan model prediksi solusi menggunakan Large Language Models (LLM) dengan pendekatan Few-Shot Learning. Model yang diterapkan adalah Llama 3.2, yang telah disesuaikan dengan dataset hasil ekstraksi dari 100 artikel ilmiah, diklasifikasikan ke dalam empat kategori utama: Problem-Solution, Tantangan-Jawaban, Peluang-Jawaban, dan Kelemahan-Peningkatan. Proses pengolahan data mencakup tahapan pre-processing, seperti case folding, tokenizing, filtering, dan stemming, guna meningkatkan kualitas data sebelum model dilatih. Evaluasi model dilakukan menggunakan metrik ROUGE untuk menilai akurasi prediksi solusi yang dihasilkan. Hasil penelitian menunjukkan bahwa pendekatan Few-Shot Learning mampu mengenali pola hubungan masalah-solusi dengan lebih efektif dibandingkan metode konvensional. Selain itu, sistem berbasis website juga dikembangkan untuk mempermudah akses dan pemanfaatan model oleh mahasiswa dalam menyelesaikan tugas akhir. Walaupun model menunjukkan kinerja yang baik, tantangan dalam menangani pertanyaan yang berbeda jauh dari contoh yang diberikan masih menjadi kendala yang perlu disempurnakan dalam penelitian mendatang.

Keyword: *Few-Shot Learning, Large Language Model, Llama 3.2*, Prediksi Solusi, Artikel Ilmiah

1. PENDAHULUAN

Artikel ilmiah berperan penting dalam kemajuan ilmu pengetahuan dan teknologi. Setiap tahunnya, jumlah publikasi ilmiah mengalami peningkatan yang signifikan. Berdasarkan penelitian [1], publikasi ilmiah global tumbuh sekitar 4% per tahun, dengan lebih dari 4 juta artikel diterbitkan pada 2021. Pertumbuhan yang pesat ini menimbulkan tantangan baru dalam mengelola serta mengekstrak informasi yang relevan dari artikel-artikel tersebut.

Perkembangan teknologi kecerdasan buatan, khususnya *Large Language Models (LLM),* telah membuka peluang baru dalam pengolahan dan analisis teks. Penelitian yang dilakukan oleh menunjukkan bahwa LLM memiliki kemampuan yang menjanjikan dalam memahami konteks dan struktur artikel ilmiah dengan akurasi mencapai 89%. Namun, tantangan utama dalam penggunaan LLM konvensional adalah kebutuhan akan data training yang sangat besar dan resource komputasi yang intensif.

Few-shot Learning muncul sebagai solusi potensial untuk mengatasi keterbatasan tersebut [2]. dalam publikasinya mendemonstrasikan bahwa pendekatan *Few-shot Learning* dapat menghasilkan performa yang kompetitif dengan hanya menggunakan 5-10 contoh training, dibandingkan dengan metode tradisional yang membutuhkan ribuan contoh. Keunggulan ini sangat relevan dalam konteks analisis artikel ilmiah, dimana setiap bidang memiliki karakteristik dan terminologi yang unik.

Keunggulan utama *Few-shot Learning* dibandingkan *fine-tuning* terletak pada efisiensi dan fleksibilitasnya. *Fine-tuning* membutuhkan waktu training yang lama dan komputasi intensif, sementara *Few-shot Learnig* dapat beradaptasi secara *real-time* dengan contoh minimal [3] Dalam konteks prediksi solusi dari masalah artikel ilmiah, *Few-shot Learning* memungkinkan model untuk memahami pola hubungan masalah-solusi dengan lebih efisien dan dapat beradaptasi dengan berbagai domain keilmuan.

Studi komparatif menunjukkan bahwa *Few-shot Learning* menghasilkan performa yang lebih stabil dibandingkan *zero-shot learning*, terutama dalam tugas yang membutuhkan pemahaman kontekstual yang dalam mendemonstrasikan bahwa *Few-shot Learning Learning* learning mencapai F1-score 15-20% lebih tinggi dibandingkan zero-shot learning dalam tugas analisis teks ilmiah.

Implementasi *Few-shot Learning* dalam konteks prediksi kalimat solusi dari artikel ilmiah menghadapi beberapa tantangan spesifik. Penelitian [4] mengidentifikasi bahwa variasi gaya penulisan dan struktur artikel antar disiplin ilmu dapat mempengaruhi akurasi prediksi. Selain itu, kompleksitas bahasa ilmiah dan penggunaan terminologi teknis menambah tingkat kesulitan dalam proses ekstraksi informasi.

Studi komparatif menunjukkan bahwa *Few-shot Learning* menghasilkan performa yang lebih stabil dibandingkan *zero-shot learning*, terutama dalam tugas yang membutuhkan pemahaman kontekstual yang dalam. Integrasi LLM dengan *Few-shot Learning* membuka perspektif baru dalam automasi analisis artikel ilmiah [5]. melaporkan peningkatan efisiensi sebesar 75% dalam proses identifikasi solusi ketika menggunakan pendekatan terintegrasi ini. Kemampuan sistem untuk belajar dari contoh terbatas sambil mempertahankan pemahaman kontekstual yang kuat dari LLM menciptakan sinergi yang menjanjikan.

Berdasarkan tantangan dan peluang tersebut, penelitian ini mengusulkan implementasi *Few-shot Learning* dengan *model llama 3.2* untuk prediksi kalimat solusi dari masalah pada artikel ilmiah. Pendekatan ini diharapkan dapat mengatasi keterbatasan dataset berlabel sambil mempertahankan akurasi yang tinggi dalam ekstraksi solusi. Mendemonstrasikan keberhasilan penerapan *Few-shot Learning* dengan model bahasa multilingual, yang dapat menjadi dasar untuk pengembangan sistem yang lebih komprehensifberkontribusi secara signifikan terhadap produktivitas dan kualitas penelitian akademik secara keseluruhan [6]

Implementasi sistem ini diharapkan dapat memberikan kontribusi signifikan dalam meningkatkan efisiensi penelitian ilmiah, mengoptimalkan pemanfaatan pengetahuan yang ada, dan mempercepat inovasi lintas domain. [7] telah meletakkan fondasi yang kuat untuk pengembangan arsitektur transformer dan optimasi *model llama 3.2, yang* menjadi basis untuk penelitian ini

1.1. N*atural Language Processing (NLP)*

Natural Language Processing (NLP) merupakan kemampuan suatu komputer untuk memahami dan mengolah bahasa manusia, baik dalam bentuk lisan maupun tulisan, sebagaimana digunakan dalam komunikasi sehari-hari. *NLP* secara sederhana bertujuan untuk *Natural Language Processing (NLP)* adalah area integral dari ilmu komputer antara pembelajaran mesin dan linguistik yang mana komputasi digunakan secara luas, yang ditujukan untuk membuat komputer mengerti pernyataan yang ditulis menggunakan bahasa manusia. Pengolahan bahasa alami timbul untuk meringankan pekerjaan user dan untuk memenuhi keinginan terhubung dengan komputer dalam bahasa murni. Karena semua pengguna mungkin tidak fasih dalam bahasa khusus mesin*, NLP* melayani pengguna yang tidak memiliki cukup waktu untuk mempelajari bahasa baru atau mendapatkan kesempurnaan di dalamnya *Natural Language Processing* berdasarkan basisnya bisa dikelompokkan dalam dua komponen ialah *Natural Language Generation (NLG) dan Natural Language Understanding (NLU)* yang mengembangkan tugas guna memahami dan menghasilkan teks*. NLP* bertindak sebagai pilar dasar untuk pengenalan bahasa, yang digunakan oleh Siri (Apple) dan Google. Hal ini memungkinkan teknologi untuk mengenali teks bahasa alami manusia dan perintah berbasis ucapan dan mencakup dua komponen utama *Natural Language Generation (NLG) dan Natural Language Understanding (NLU). NLP* lebih menyulitkan daripada NLG, karena bahasa alami memiliki struktur dan bentuk yang sangat kompleks, memetakan inputan yang diberikan pengguna dan menganalisis beberapa fitur Bahasa [8].

1.2. *Large Language Processing*

Large Language *Model(LLM*) adalah salah satu jenis model Bahasa kecerdasan buatan yang termasuk ke dalam *Natural Language Processing (NLP). LLM* dapat membuat teks dengan kemampuan seperti layaknya manusia. LLM ditandai dengan banyak parameter dan dilatih pada kumpulan teks yang besar, serta memiliki kemampuan untuk menghasilkan output yang relevan kontekstual dan gramatikal. Sedangkan *langchain* merupakan kerangka kerja yang digunakan dalam pengembangan aplikasi dengan LLM. *Langchain* memungkinkan pengembang untuk membangun aplikasi menggunakan LLM untuk meningkatkan penyesuaian, akurat, dan relevansi dari model. Penelitian yang pernah dilakukan oleh Oughuzan Topsakal dan T.Cetin Akinci (2023) tentang *Creating Large Language Model Applications Utilizing Langchain: A Primer on Developing LLM Apps Fa*st. Penelitian ini memiliki peran besar dalamdunia pengembangan aplikasi dengan *Large Language Model (LLM).* Penelitian ini juga menggunakan *Langchain* sebagai kerangka kerja yang memungkinkan pengembang untuk memanfaatkan LLM dengan baik, efektif dan efisien. *Langchain* juga dapat meningkatkan kualitas dan relevansi outputyang dihasilkan. Melalui penelitian ini Topsakal dan Akinci menunjukkan bahwa LLM tidak hanya sebuah alat yang kuat untuk memproses teks, namun juga dapat

diterapkan secara luas dalam berbagai konteks, termasuk penulisan kode dan tugas-tugas lainnya dalam bidang kecerdasan buatan [9].

1.3. *Few-Shot Learning*

Indikator evaluasi dalam *Few-shot Learning* sangat penting untuk menilai kinerja model yang dilatih dengan data beranotasi terbatas, yang ditugaskan untuk mengklasifikasikan kategori baru secara akurat hanya dengan beberapa sampel berlabel. Biasanya, evaluasi ini dilakukan dalam pengaturan N-way K-shot, di mana setiap tugas terdiri dari support set dan verification set yang mencakup N kategori (way) dan K sampel dalam support set untuk setiap kategori. Akurasi algoritma ditentukan dengan mengujinya pada verification set, mengulangi evaluasi beberapa kali, dan menghitung rata-rata akurasi. Secara lebih spesifik, akurasi klasifikasi pada metode *Few-shot Learning* dihitung berdasarkan proporsi instance yang diklasifikasikan dengan benar dibandingkan dengan ukuran total dataset.

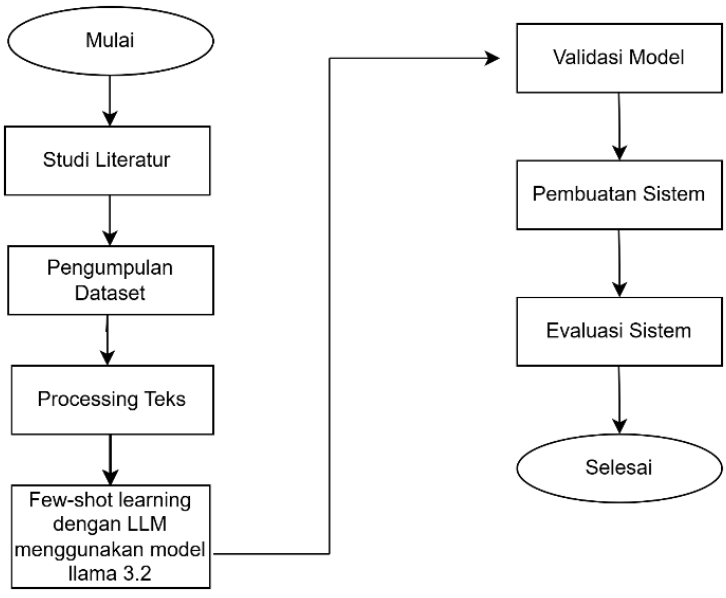
Selain akurasi klasifikasi dan efisiensi meta-learning, penting juga untuk mempertimbangkan metrik kinerja lain seperti precision, recall, dan F1-score. Metrik ini memberikan pemahaman yang lebih mendalam tentang kinerja model dengan mempertimbangkan false positive, false negative, serta keseimbangan antara precision dan recall. Dengan mengevaluasi model menggunakan kombinasi akurasi, efisiensi meta-learning, dan metrik relevan lainnya, peneliti dapat menilai efektivitas model secara komprehensif dalam skenario *Few-shot Learning*. [10]

1.4. *Ollama*

Ollama merupakan inovasi terbaru dalam dunia teknologi kecerdasan buatan (AI) yang memungkinkan pengguna menjalankan model bahasa besar (*Large Language Models* atau LLM) secara lokal di perangkat mereka. Teknologi ini dirancang untuk memenuhi kebutuhan akan akses LLM yang lebih aman, cepat, dan efisien tanpa harus bergantung pada koneksi *internet. Ollama* menyediakan antarmuka berbasis command-line yang sederhana dan mudah digunakan, sehingga pengguna dapat dengan cepat mengunduh, mengelola, dan mengoperasikan berbagai model AI [11]. Salah satu model AI yang saya gunakan melalui *Ollama* adalah *Llama 3.2,* versi terbaru yang menawarkan peningkatan signifikan dalam performa, kecepatan pemrosesan, dan efisiensi memori. *Llama 3.2* dirancang untuk memberikan hasil yang lebih akurat dan responsif, menjadikannya pilihan unggul untuk implementasi lokal AI, termasuk pemrosesan bahasa alami, pengembangan kode, dan aplikasi berbasis AI lainnya. Dengan *Ollama dan Llama 3.2,* pengguna kini memiliki solusi AI canggih yang dapat diakses kapan saja tanpa perlu koneksi internet

2. METODE PENELITIAN

Penelitian ini menggunakan pendekatan *Few-shot Learning* menggunakan LLM. Dalam penelitian ini, LLM akan di training untuk memprediksi solusi dari kalimat masalah yang terdapat dalam artikel ilmiah. Langkah-langkah yang harus dilakukan dalam penelitian ini adalah sebagai berikut :



Gambar 1 Alur Penelitian

2.1. Studi Literatur

Peneliti akan meninjau berbagai sumber literatur, termasuk *e-book,* artikel, jurnal, penelitian sebelumnya seperti tesis dan skripsi, serta informasi dari berbagai situs web yang relevan dengan topik penelitian. Studi literatur ini berfokus pada pemahaman teori mengenai *Few-shot Learning,* (LLM), metode

prediksi berbasis teks, serta penerapan model *machine learning* untuk menganalisis solusi, tantangan, peluang, dan kelemahan yang diidentifikasi dalam artikel ilmiah.

2.2. Pengumpulan Dataset

Pengumpulan dataset dalam penelitian ini dilakukan secara manual karena belum tersedia dataset yang sesuai dengan kebutuhan penelitian. Dataset ini disusun dengan mereview 100 artikel. Artikel yang digunakan diambil dari Jurnal Teknologi Informasi dan Ilmu Komputer (JTIIK).

2.3. Data Pre Processing

Dataset yang telah terkumpul akan melalui tahap *pre-processing data* untuk meningkatkan kualitas data. Pada tahap ini, data mentah akan diproses melalui serangkaian langkah pembersihan dan standardisasi agar siap digunakan dalam analisis selanjutnya. Berikut adalah tahapan *pre-processing data yang* akan dilakukan:

1. *Case folding* atau normalisasi huruf merupakan salah satu tahapan dalam text preprocessing yang bertujuan untuk menyamakan format huruf dengan mengubah seluruh teks menjadi huruf kecil atau huruf besar. Proses ini dilakukan agar teks lebih mudah dibandingkan dan dianalisis tanpa terpengaruh oleh perbedaan penggunaan huruf besar. Case folding yang dilakukan yaitu text.lower untuk mengubah teks menjadi huruf kecil semua. Mengubah semua karakter menjadi huruf kecil membantu meningkatkan konsistensi data, membuat analisis dan pemrosesan lebih lanjut menjadi lebih mudah [12].
2. *Tokenizing,* adalah proses membagi atau memotong teks menjadi kata-kata yang membentuknya. Tujuan dari *tokenizing* adalah untuk mempermudah proses selanjutnya seperti perhitungan kata, pembobotan kata, dan transformasi data menjadi vektor dengan dimensi tinggi. Dengan *tokenizing, data* teks dapat diolah dengan lebih mudah dan akurat sebelum dilakukan analisis lebih lanjut [13].
3. *Filtering* merupakan proses dalam text preprocessing setelah tokenisasi*, filtering* dilakukan untuk mengambil kata penting hasil tokenisasi*. Proses filtering* dalam membuang kata-kata yang tidak digunakan atau stopwordterdapat dalam bag of words [14].
4. *Stemming* merupakan proses mengubah kata menjadi bentuk dasarnya*. Stemming* dilakukan untuk menyeragamkan bentuk kata. Tujuan dari proses *stemmingadalah* menghilangkan imbuhan-imbuhan baik itu berupa prefiks, sufiks, maupun konfiks yang ada pada setiap *kata Stemming* dalam penelitian ini dilakukan berdasarkanaturan morfologi bahasa Indonesia.

2.4. Pemilihan dan Konfigurasi Model

Setelah data tersedia, langkah berikutnya adalah memilih dan mengatur *model Llama 3.2* untuk proses *Few-shot Learning.* Model ini dipilih karena kemampuannya dalam memahami serta menghasilkan teks sesuai dengan instruksi yang diberikan. Pada tahap konfigurasi tetap diperhatikan agar model dapat beradaptasi dengan baik tanpa mengalami overfitting. Selain itu, format input-output juga disesuaikan dengan merancang prompt tertentu agar model lebih memahami tugasnya dalam memberikan solusi berdasarkan pertanyaan yang diberikan. Format prompt yang digunakan adalah: Dengan pendekatan ini, model memahami bahwa input berupa pertanyaan atau permasalahan yang memerlukan jawaban dalam bentuk solusi, dengan batasan tidak lebih dari dua kalimat.

2.5. Few-Shot Model

Few-shot merupakan proses penyesuaian model yang telah melalui tahap pelatihan sebelumnya (pre-trained model) agar lebih efektif dalam menyelesaikan tugas tertentu atau menangani data tertentu. Model pra-terlatih umumnya dilatih dengan kumpulan data yang luas dan bervariasi, sehingga mampu memahami berbagai jenis data secara umum.

Few-shot model *LLama* 3.2 pada penelitian ini dilakukan untuk menyesuaikan model agar lebih optimal dalam memprediksi solusi berdasarkan kalimat permasalahan pada artikel ilmiah. Proses ini menggunakan pendekatan pembelajaran *sequence to sequence*, dimana model dilatih untuk memahami hubungan antara input berupa kalimat masalah dan output berupa solusi yang sesuai dengan konteks.

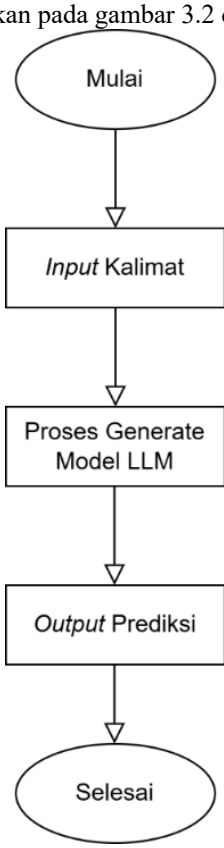
2.6. Evaluasi dan Pengujian Model

Setelah model selesai dilatih, dilakukan validasi menggunakan metrik evaluasi seperti *ROUGE (Recall-Oriented Understudy for Gisting Evaluation*). Metrik ini mengukur kesamaan antara solusi yang diprediksi oleh model dengan solusi yang ada dalam dataset. Validasi bertujuan untuk mengukur relevansi, akurasi, dan kualitas prediksi model.

ROUGE (Recall-Oriented Understudy for Gisting Evaluation) merupakan metode evaluasi yang banyak digunakan dalam *Natural Language Processing*, khusunya untuk menilai kualitas ringkasan otomatis dan terjemahan mesin. Evaluasi ini dilakukan dengan membandingkan hasil prediksi solusi yang dihasilkan oleh sistem dengan data acuan, yang disebut *ground truth summarization*. *ROUGE* bekerja menggunakan pendekatan intrinsic, yakni menganalisis tingkat kesamaan antara hasil prediksi dan referensi yang diharapkan.

2.7. Analisis system

Pada penelitian ini, penulis akan membuat sistem prediksi solusi berbasis website yang bertujuan untuk memberikan layanan informasi tugas akhir kepada mahasiswa di program studi Teknik Informatika UNISSULA. Untuk alurnya akan digambarkan pada gambar 3.2 dengan flowchart



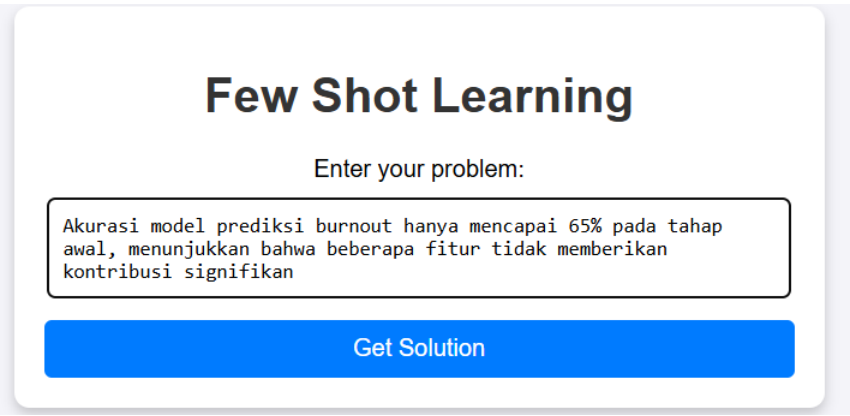
Gambar 2. Alur kerja system

Berikut adalah tahapan alur kerja sistem :

a. Mulai : Pertama *user* memasuki halaman tampilan awal yang menggambarkan proses dari *website* aplikasi sistem prediksi solusi
b. Input Kalimat : Kemudian *user* memasukkan kalimat permasalahan yang tgerdapat dalam artikel, setelah itu tekan tombol "*Get Solution"* untuk meunuggu hasil nya keluar.
c. Permasalahan diproses :Selanjutnya sistem akan mulai memproses permasalahan yang diberikan oleh *user* dengan mencocokkan data jawaban yang sudah dilatih.
d. Output Prediksi : Setelah permasalahan diproses, model akan memberikan *output* prediksi, yaitu kalimat solusi yang relevan dengan masalah yang diberikan. Solusi ini dihasilkan berdasarkan pola-pola yang telah dipelajari oleh model selama proses *few-shot*.

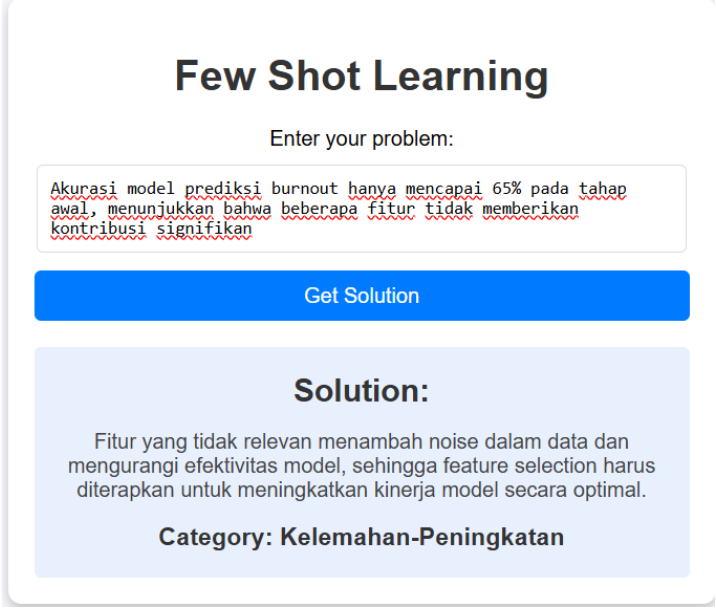
HASIL DAN ANALISA

3.1. Tampilan *Input kalimat Problem*

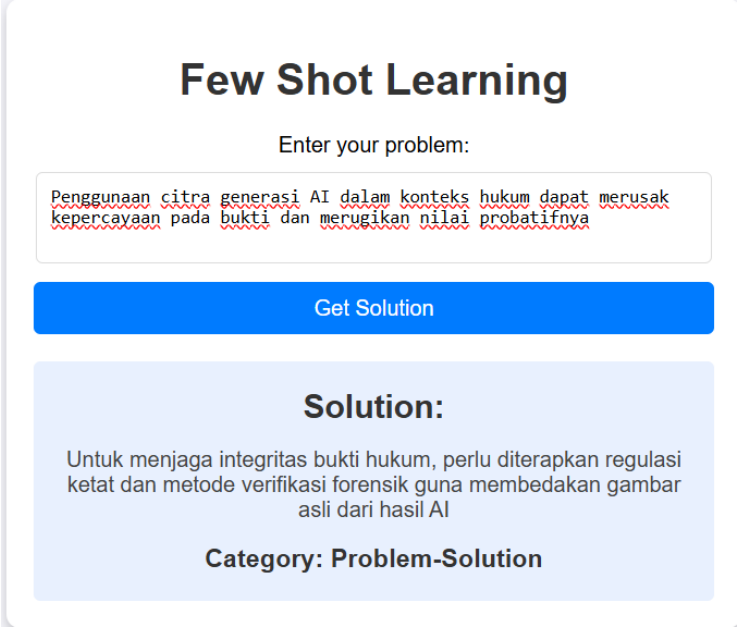


Gambar *3.* Tampilan saat *input* kalimat *problem*

Pada Gambar 3 tampilan pada saat pengguna memasukkan kalimat masalah untuk mendapatkan prediski.



Gambar 4. Tampilan *Outpu* Prediksi



Gambar 5. Tampilan *Output* Prediksi

Dapat diamati pada gambar 3, 4 dan 5, Sistem menghasilkan prediksi solusi berdasarkan permasalahan yang dimasukkan oleh pengguna. Prediksi yang diberikan dinilai relevan dengan permasalahan yang berkaitan dengan teknologi. Dari dua kali percobaan *demo* aplikasi, sistem terbukti mampu memproses input dan memberikan solusi yang sesuai dengan permasalahan yang diajukan oleh pengguna.

3.2. Hasil Evaluasi *Rouge*

Tabel 1. Hasil Evaluasi *Rouge*

ROUGE-1	ROUGE-2	ROUGE-L
0.1352	0.0171	0.1081

Model few-shot menunjukkan kesesuaian yang cukup baik dengan referensi berdasarkan evaluasi *ROUGE. Skor ROUGE-1* sebesar 0.1352 menunjukkan 13,52% unigram sesuai, sementara *ROUGE-2* hanya

1,71% karena kesesuaian bigram lebih sulit dicapai. *ROUGE-L* mencatat 10,81%, mengindikasikan urutan kata yang cukup relevan.

Meskipun hasilnya cukup baik, model ini masih dapat ditingkatkan dengan memastikan kualitas contoh *few-shot* yang representatif serta menerapkan augmentasi data seperti paraphrasing atau *back translation* untuk meningkatkan akurasi tanpa perlu *fine-tuning* penuh.

3. KESIMPULAN

Model *LLaMA* 3.2 dalam skenario *few-shot learning* menunjukkan kesesuaian yang cukup baik dengan referensi, meskipun akurasi prediksi masih terbatas. *Skor ROUGE-1* sebesar 0.1352 menunjukkan model dapat menangkap kata yang sesuai, namun skor *ROUGE-2 yang* lebih rendah (0.0171) mengindikasikan kesulitan dalam mengenali hubungan antar *kata. ROUGE-L* sebesar 0.1081 menunjukkan kesesuaian urutan kata yang cukup baik.

Untuk meningkatkan akurasi, diperlukan optimasi dengan memilih contoh yang lebih representatif, menerapkan augmentasi data seperti paraphrasing atau back translation, serta mengeksplorasi jumlah contoh yang optimal. Selain itu, penggunaan data yang lebih relevan dan metrik evaluasi tambahan seperti *BLEU* atau *BERTScore* dapat memberikan penilaian yang lebih komprehensif.

DAFTAR PUSTAKA

[1] L. Bornmann dan R. Mutz, "Growth rates of modern science: A bibliometric analysis based on the number of publications and cited references," *J. Assoc. Inf. Sci. Technol.*, vol. 66, no. 11, hal. 2215–2222, 2015, doi: 10.1002/asi.23329.

[2] D. Samuel, J. Barnes, R. Kurtz, S. Oepen, L. Øvrelid, dan E. Velldal, "Direct parsing to sentiment graphs," *Proc. Annu. Meet. Assoc. Comput. Linguist.*, vol. 2, hal. 470–478, 2022, doi: 10.18653/v1/2022.acl-short.51.

[3] Y. Chen, R. Zhong, S. Zha, G. Karypis, dan H. He, "Meta-learning via Language Model In-context Tuning," *Proc. Annu. Meet. Assoc. Comput. Linguist.*, vol. 1, hal. 719–730, 2022, doi: 10.18653/v1/2022.acl-long.53.

[4] *P. Liu, W. Yuan, J. Fu, Z. Jiang, H. Hayashi, dan G. Neubig, "Pre-train, Prompt, and Predict: A Systematic Survey of Prompting Methods in Natural Language Processing," ACM Comput. Surv.*, vol. 55, no. 9, hal. 1–46, 2023, doi: 10.1145/3560815.

[5] P. Keicher, "Machine Learning in Top Physics in the ATLAS and CMS Collaborations," hal. 4–9, 2023, [Daring]. Tersedia pada: http://arxiv.org/abs/2301.09534

[6] X. V. Lin *dkk., "Few-shot Learning with Multilingual Generative Language Models," Proc. 2022 Conf. Empir. Methods Nat. Lang. Process. EMNLP 2022, hal. 9019–9052, 2022, doi: 10.18653/v1/2022.emnlp-main.616.*

[7] Y. Liu *dkk.*, "RoBERTa: A Robustly Optimized BERT Pretraining Approach," no. 1, 2019, [Daring]. Tersedia pada: http://arxiv.org/abs/1907.11692

[8] *R. Dale, B. L. Iverson, P. B. Dervan, dan M. R. F. Hadi, "Natural Language Processing menggunakan Metode Fuzzy String Matching pada Chatbot berbasis Web pada Aplikasi Whatsapp," Handb. Nat. Lang. Process. Second Ed.*, hal. 7823–7830, 2010, [Daring]. Tersedia pada: http://digilib.uinsby.ac.id/id/eprint/51730%0Ahttp://digilib.uinsby.ac.id/51730/2/Muhammad Rifqy Fakhrul Hadi_H06217015.pdf

[9] *F. Rizki, A. Sutiyo, N. S. Harahap, S. Agustian, dan R. M. Candra, "KLIK: Kajian Ilmiah Informatika dan Komputer Implementasi Question Answering Berbasis Chatbot Telegram Pada Tafsir Al-Jalalain Menggunakan Langchain dan LLM," Media Online*, vol. 4, no. 5, hal. 2464–2472, 2024, doi: 10.30865/klik.v4i5.1784.

[10] *Q. Qi, A. Ahmad, dan W. Ke, "Image classification based on few-shot learning algorithms: a review," Indones. J. Electr. Eng. Comput. Sci.*, vol. 35, no. 2, hal. 933–943, 2024, doi: 10.11591/ijeecs.v35.i2.pp933-943.

[11] Y. Wan *dkk.*, "Deep Learning for Code Intelligence: Survey, Benchmark and Toolkit," *ACM Comput. Surv.*, vol. 1, no. 1, 2024, doi: 10.1145/3664597.

[12] R. R. Salam, M. F. Jamil, Y. Ibrahim, R. Rahmaddeni, S. Soni, dan H. Herianto, "Analisis Sentimen Terhadap Bantuan Langsung Tunai (BLT) Bahan Bakar Minyak (BBM) Menggunakan Support Vector Machine," *MALCOM Indones. J. Mach. Learn. Comput. Sci.*, vol. 3, no. 1, hal. 27–35, 2023, doi: 10.57152/malcom.v3i1.590.

[13] A. Ariansyah dan U. Indahyanti, "Fitur Ekstraksi pada Pemodelan Topik Menggunakan Metode Latent Dirichlet Allocation pada Peristiwa Kebocoran Data," no. 2, hal. 1–24, 2024.

[14] E. Yuniar, D. S. Utsalinah, dan D. Wahyuningsih, "Implementasi Scrapping Data Untuk Sentiment Analysis Pengguna Dompet Digital dengan Menggunakan Algoritma Machine Learning," *J. Janitra Inform. dan Sist. Inf.*, vol. 2, no. 1, hal. 35–42, 2022, doi: 10.25008/janitra.v2i1.145.