Improving Large Language Model's Ability to Find the Words Relationship

Sirojul Alam[*], **Jaka Abdul Jabar**, **Fauzi Abdurrachman**, **Bambang Suharjo**
H. A. Danang Rimbawa
Universitas Pertahanan Indonesia, Bogor, Indonesia

Article Info:
Received: 6 June 2024, Revised: 26 October 2024, Accepted: 30 October 2024

Abstract-
Background: It is still possible to enhance the capabilities of popular and widely used large language models (LLMs) such as Generative Pre-trained Transformers (GPT). One method of achieving enhancement is using the Retrieval-Augmented Generation (RAG) architecture, which incorporates outside data into the model to improve LLM capabilities.
Objective: This research aims to prove that the RAG can help LLMs respond more precisely and rationally.
Methods: The method used in this work is utilizing the Huggingface Application Programming Interface (API) for word embedding, storing, and finding the relationship of the words.
Result: The attractively rendered graph clearly shows how well RAG performs. The knowledge obtained is logical and understandable, such as the word Logistic Regression, which is related to accuracy and F1 score and defined as a simple and the best model compared to Naïve Bayes and the Support Vector Machine (SVM) model.
Conclusion: The conclusion is that RAG helps LLMs improve their capability.

Keywords: Ability to Find, Large Language Model's, Words Relationship

Corresponding Author:
Sirojul Alam,
Cyber Defense Engineering Study Program, Universitas Pertahanan Indonesia, Bogor, Indonesia,
Email: sirojul.alam@tp.idu.ac.id

1. INTRODUCTION

Language serves as a crucial tool for communication and expression [1], constantly evolving alongside the progress of time and technology. It enables humans to share information, emotions, and thoughts, fostering socialization and community building as defined as a social bonding tool [2]. Language development is closely linked to cognitive abilities, influencing how individuals express their thoughts and interact with the world. Essentially, computers cannot understand humanity's communication except when equipped with advanced artificial intelligence technology. This poses a challenge for researchers to provide and make computers understand to communicate like human beings [3]. To answer this challenge, researchers have proposed many language models. In the 1990s, researchers proposed the statistical language model (SLM). This model predicts the next word of sentences based on the word frequency [4]. The next language model is the Neural Language Model (NLM). This model can predict and make categorizations of words in a sentence by utilizing a neural network algorithm [5]–[7].

In natural language processing (NLP), the evolution of language models has been marked by significant advancements, particularly with the development of Pre-trained Language Models (PLMs). PLMs are the foundational framework for subsequent models, including large language models (LLMs). PLMs are distinguished by their ability to represent words within a given context effectively [5]. This capability lays the groundwork for scaling up the model's capacity, achieved through iterative improvements in data training methodologies [8], [9].

How to Cite: S. Alam, J. Abdul Jabar, F. Abdurrachman, B. Suharjo, and H. D. Rimbawa, "Improving Large Language Model's Ability to Find the Words Relationship", Jurnal Bumigora Information Technology (BITe), vol. 6, no. 2, pp. 141-148, Dec. 2024. doi: 10.30812/bite.v6i2.4127

The transition of PLMs to LLMs represents a paradigm shift in NLP, characterized by using deep learning techniques to enhance text analysis and generation capabilities [10]. LLMs leverage vast datasets and sophisticated neural network architectures to achieve remarkable proficiency in understanding and generating human-like text. Among the most prominent examples of LLMs is ChatGPT, a conversational artificial intelligence (AI) model that has gained widespread adoption and recognition.

ChatGPT, in its latest iteration, ChatGPT-4, exemplifies the culmination of advancements in LLM technology. ChatGPT has demonstrated enhanced language understanding, context sensitivity, and conversational powers with each successive version. Its robust performance in various natural language understanding and generation tasks has solidified its position as a leading player in conversational AI.

Indeed, ChatGPT has seamlessly integrated into our daily lives, serving as a trusted companion and an invaluable resource for various inquiries. This advanced language model has become a go-to platform for seeking information and initiating research endeavors. Its adaptive learning capabilities enable it to continuously refine its knowledge base and enhance its problem-solving skills based on user interactions. As a result, users benefit from an increasingly intelligent and responsive system that effectively addresses their queries and assists them in navigating complex topics. The transformative impact of ChatGPT extends beyond more question-answering functionalities. ChatGPT facilitates seamless communication and collaboration between humans and machines by harnessing the power of NLP and machine learning. Its ability to understand context, interpret nuances, and generate coherent responses enables it to bridge the gap between users and information resources, fostering a more efficient and productive research environment. Researchers caution against placing complete trust in LLMs due to their inherent limitations and potential implications [8]. In fact, the International Red Cross Committee has issued warnings regarding the possibility of technology removing human decision-making from critical life-and-death situations. This underscores the importance of maintaining human oversight and intervention, especially when ethical considerations and moral judgment are paramount.

There is a gap between this research and the previous one [11] namely generalization to work in structured data. Despite the remarkable capabilities of LLMs, there remains room for enhancement. One promising approach to augmenting LLMs is using Retrieval-Augmented Generation (RAG) techniques. RAG integrates external data sources into the LLM framework, enriching its knowledge base and expanding its contextual understanding [9]. This integration empowers LLMs to generate accurate and tailored responses to the inquiry's specific context, enhancing the overall quality of interactions. RAG also serves as a mechanism for mitigating the dissemination of inaccurate information by LLMs during response generation. By leveraging external data sources, RAG enables LLMs to cross-reference and validate the information before generating a response, thereby reducing the likelihood of errors or misleading content. This ensures that LLMs' responses are reliable, trustworthy, and aligned with the user's query. The adoption of RAG represents a significant advancement in the evolution of LLMs. The difference between this research and the previous one is that this research works with unstructured data from texts in an article.

Several studies have explored integrating RAG techniques into LLMs to enhance accuracy and performance. In study [10] a soft memory module was proposed to facilitate the indexing and retrieval of hidden states that humans may not interpret directly. This approach enables the model to access relevant information efficiently, improving its ability to generate contextually appropriate responses. Another research [11] introduced a technique for incorporating RAG into language models, allowing the model to select relevant text snippets even when the input is incomplete. Research [12] introduced SESCore2, a RAG framework that evaluates text generation by integrating retrieval-enhanced synthesis of data and preliminary training on synthetic data. Research [13], [14] proposed the multimodal RAG (MuRAG), which integrates RAG techniques with multimodal data, including text images. MuRAG demonstrates superior performance in tasks related to answering multimodal questions such as WebQA and multimodalQA. This research aims to integrate RAG with LLM so that the LLM can generate informative responses. This research contributes to refining the LLMs to the next level, with rational responses to users.

2. RESEARCH METHOD

This study adopts the RAG architecture in conjunction with LLMs and knowledge graphs to analyze the relationships between words within a document. A knowledge graph is a structured representation of objects, often referred to as semantic networks [15]. It serves as a conduit for organizing and conveying knowledge in the form of a graph, with nodes representing entities and edges denoting the relationships between nodes [15]. Essentially, a knowledge graph encapsulates a body of knowledge regarding the relationships among various entities [14]. This framework facilitates the extraction of insights from data and provides a comprehensive overview of the interconnected entities [16]. The popularity of knowledge graphs has surged, particularly after Google adopted them to augment search results [17].

This research was conducted by utilizing the Huggingface Application Programming Interface (API), a well-known NLP provider. We obtained a token to access the API for free, and then we hit the model that we needed, such as the llama_index model, which is a text-processing model with 7 billion parameters [18] as also utilized by [19], zephyr 7B-beta model, an LLM well-trained model [20], this model is a scaled-up model from the mistral-AI [21]. Our research material is an article about performance measurement of the Naïve Bayes (NB) model, Support Vector Machine (SVM) model, and logistic regression model in a sentiment analysis [22]. The article contains 7,947 words, providing a comprehensive analysis of the performance of NB, SVM, and Logistic Regression (LR) in Indonesian immigration sentiment analysis. This word count indicates that the article is in-depth and offers extensive data, methodology, and discussion.

We utilize the obtained document as research material, then upload and read it. This process utilizes a function named SimpleDirectoryReader that is available in the llama library [19]. The Zephyr 7B-beta library reads the document material, and the result is stored in a repository. If the document has not been successfully read, the reading process will continue to be repeated. In the next stage, our model is used to identify relevant context from the text contained in the research material. The model then creates an index to generate graphs based on the identified context. This is done to offer input to LLM to generate responses efficiently. In the last stage, we visualize the result with a graph. These stages are depicted in Figure 1.

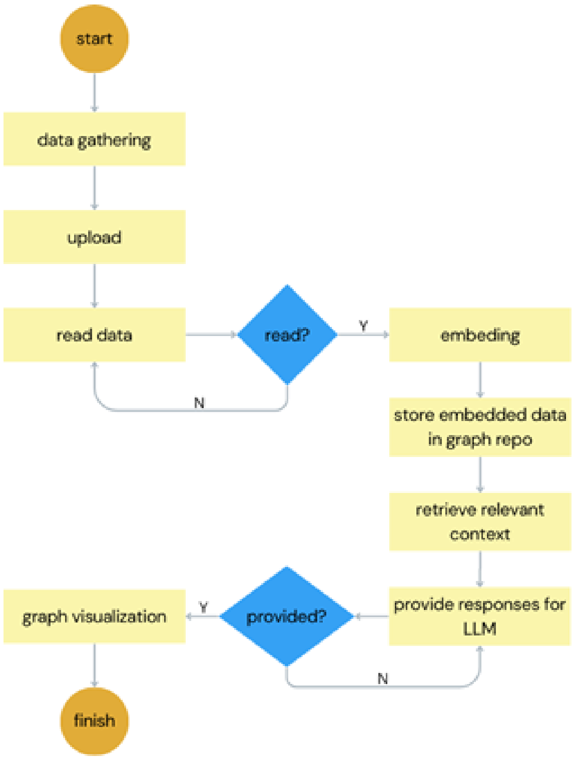

Figure 1. Research Stages

In detail, as depicted in Figure 1, the process begins with data gathering, where relevant information is collected from various sources, and we use [22], to serve as the foundation for the RAG workflow. Once the data is gathered, it is uploaded into the system, which prepares it for the subsequent steps. After the upload, the system proceeds to read the data, evaluating its completeness and ensuring it meets the necessary standards for further processing. Upon successful reading, the data undergoes embedding, transforming the text or information into vector representations. These vectors enable the model to identify and retrieve contextual similarities between data points. The embedded data is stored in a graph repository, a structured storage system that organizes the vectors, making it easier for the model to access and retrieve relevant information as needed. With the data stored as embeddings, the system retrieves the relevant context in response to the user's queries or needs. This retrieval process ensures that only the most pertinent information is selected, allowing the system to provide a focused and accurate response.

Following context retrieval, the selected information is prepared as a response for the LLM. This response is formulated based on the relevant context identified within the repository. Another decision point checks if the LLM has successfully provided the required context. If it has, the workflow proceeds to graph visualization; otherwise, the system loops back to retrieve additional context. In the final step, the relationships and structure of the data are visualized as a graph. This visual representation offers an intuitive and comprehensive view of the data and its connections, supporting further analysis and understanding. With the graph visualization complete, the RAG process concludes, providing a clear and organized output based on the original data collection and embedding stages.

We use several libraries and dependencies such as llama_index, pyvis, Ipython, langchain, and pypdf. We also use diagnostic logging to monitor the program. We then call the essential libraries of this research. All the libraries are currently operating efficiently and exhibiting excellent performance. The ServiceContext module is specifically designed to acquire contextual data, which is pivotal for the overall functionality of our system. The KnowledgeGraphIndex is tasked with the generation and manipulation of knowledge graphs. This module is critical in structuring and analyzing the interconnected data, enabling a deeper understanding of the underlying patterns and relationships. The simpleGraphStore is responsible for storing data graphs, ensuring that all data is systematically organized and readily accessible for processing and analysis. At the end of the research stages, we visualize the obtained data.

We utilize the Pyvis library to represent the knowledge graph visually. The notebook function from Pyvis is particularly advantageous for our purposes as it seamlessly integrates with Google Colab notebooks, the platform we employ for all our work processes. To ensure compatibility and accessibility, we configure the cdn_resources parameter accordingly, aligning the availability of resources with our setup. We then set the directed function to true, indicating that the graph represents directional relationships between nodes, and we save the results of our analysis to facilitate easy retrieval should further examination be required.

3. RESULT AND DISCUSSION

The graph visualization, as depicted in Figure 2, Figure 3, and Figure 4, provides insightful details and critical information. It highlights the accuracy of the NB classifier algorithm and identifies the model that delivers the highest performance in sentiment analysis. This highlights its effectiveness in classifying textual data based on sentiment, providing a quantitative measure that aids in assessing the model's reliability and precision. The knowledge graph is designed to be straightforward and easily interpretable, allowing viewers to comprehend the relationships and data points represented quickly. This simplicity in design facilitates a better understanding of the underlying analytical result, enhancing the usability of the visualized data for both technical and non-technical audiences.

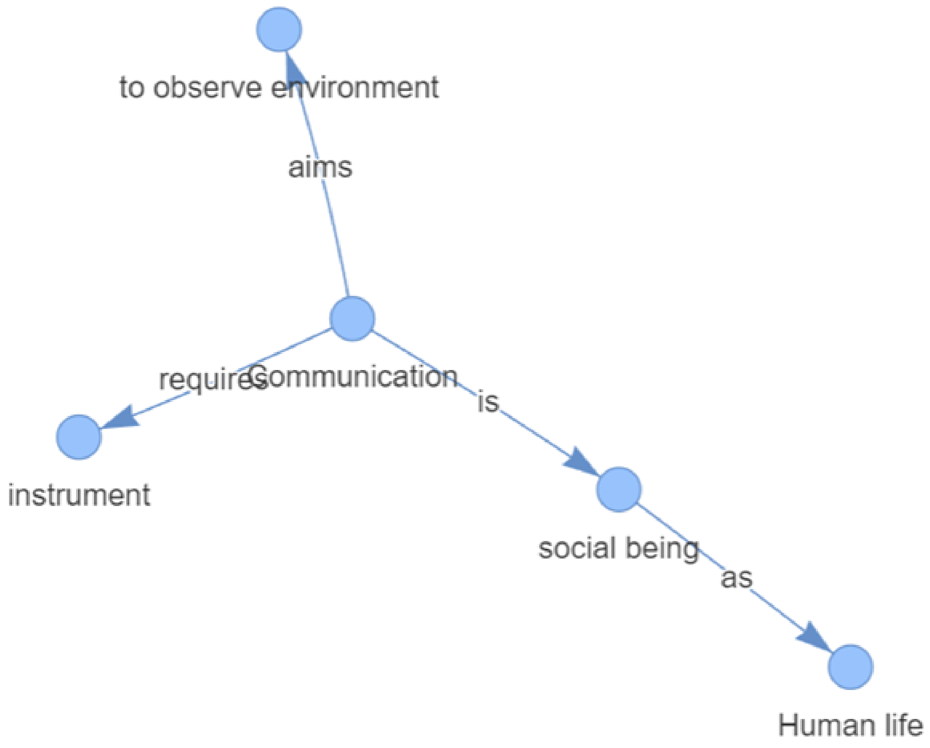

Figure 2. Knowledge Graph Result

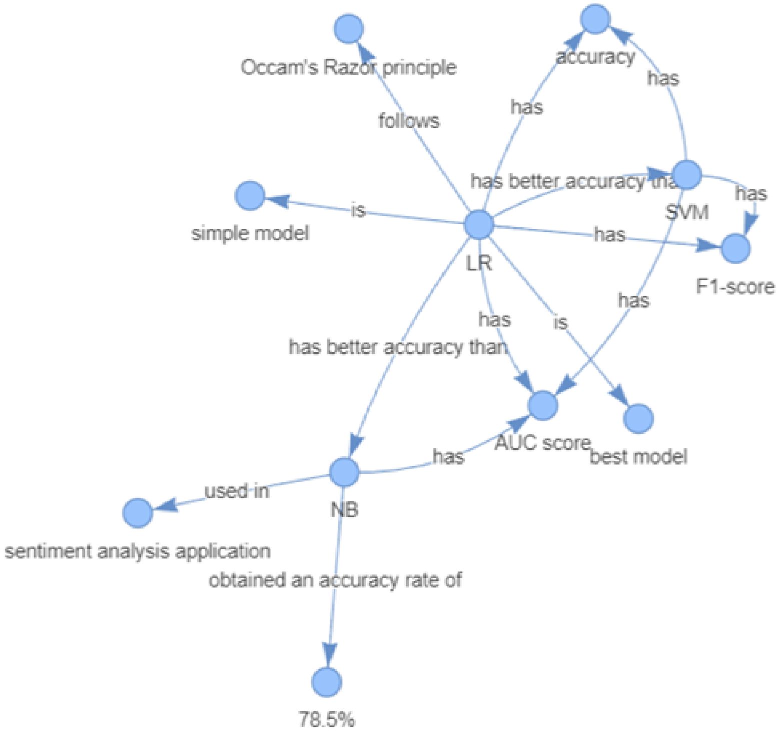

Figure 3. Knowledge Graph Result

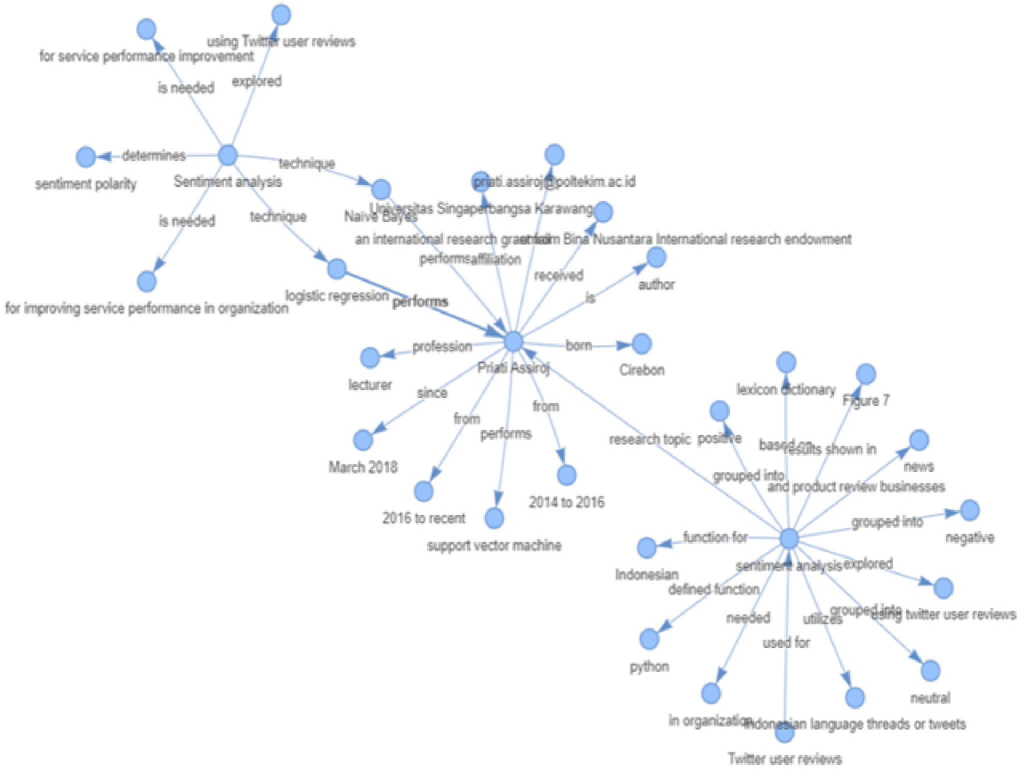

Figure 4. Knowledge Graph Result

The finding of this research, as depicted in Figure 3, shows that the knowledge graph shows that the LR model outperforms both NB and SVM models in terms of effectiveness for sentiment analysis of Indonesian immigration data. This graph indicates that logistic regression delivers the highest accuracy and ranks as the simplest model compared to the other two. The simplicity of logistic regression makes it particularly appealing, as it suggests a lower barrier to implementation and interpretation, which can be crucial for practical applications where complexity might hinder usability. The result depicted in Figure 4 enriches our understanding by detailing the thematic focus and authorship linked to the research material. It explicitly reveals that the scope of the study encompasses sentiment analysis using logistic regression, naïve Bayes, and support vector machine models, applying sentiment polarity techniques to scrutinize data collected from Twitter. This analysis categorizes sentiment into positive, negative, and neutral, utilizing a lexicon dictionary to assist in the classification. Such detailed mapping of topics and methodologies not only clarifies the scope of the research but also highlights the comprehensive approach taken to explore sentiment analysis, providing a clear framework for understanding the depth and application of the study within the context of social media data.

The knowledge graph's structure, as depicted in these figures, is a valuable tool for quickly identifying the key elements and relationships within the research. By providing a clear visual breakdown of the models used and their comparative performances, the graph allows researchers and practitioners alike to gauge the effectiveness of different analytical techniques at a glance. The integration of topic-specific details, such as the source of the data and the method of analysis (sentiment analysis based on a lexicon dictionary), offers a concise yet comprehensive overview of the research's operational framework. This method of presentation not only facilitates easier comprehension but also aids in disseminating research findings to a broader audience, ensuring that the implications and utility of the study are readily accessible to those interested in the field of sentiment analysis. The methodology also offers a new approach to enhancing semantic understanding through LLM-powered relationship mapping. This research is supported by previous research [23], which shows that RAG improves LLMs performance.

4. CONCLUSION

The RAG architecture is a pivotal approach aimed at bolstering the powers of LLMs. Our research endeavors to harness this architecture's potential through meticulous experimentation by delving into its application within a specific document context. Our study meticulously scrutinizes the intricate interplay between words within the documents, seeking to unearth nuanced relationships that underpin its semantic fabric. Our experiments unveil a dynamic landscape of interconnected concepts, shedding light on underlying semantic associations that imbue the document with meaning. By seamlessly integrating experimental findings with a knowledge graph, we transcend mere word-level analysis, unveiling a comprehensive depiction of the intricate web of semantic connections that permit the document's discourse. This study underscores the transformative potential of the RAG architecture in unlocking deeper semantic insights and fostering a more nuanced understanding of textual data.

ACKNOWLEDGEMENT

We want to express our deepest gratitude and highest appreciation to the Universitas Pertahanan Indonesia for supporting the authors from the initial stages of the research design to the publication of this study. The university's assistance and resources have been invaluable in enabling the successful completion and dissemination of our research findings.

REFERENCES

[1] Z. Dai and J. Liu, "Communotion and the Evolution of Human Language," *Journal of Arts and Humanities*, vol. 8, no. 9, pp. 100–110, Oct. 2019. DOI: 10.18533/journal.v8i9.1737.

[2] L. Damjanovic, S. G. B. Roberts, and A. I. Roberts, "Language as a tool for social bonding: Evidence from wild chimpanzee gestural, vocal and bimodal signals," *Philosophical Transactions of the Royal Society B: Biological Sciences*, vol. 377, no. 1860, p. 20 210 311, Sep. 2022. DOI: 10.1098/rstb.2021.0311.

[3] A. M. Turing, "Computer Machinary and Intelligence," *Mind, New Series*, vol. 59, no. 236, pp. 433–460, 1950.

[4] J. S. Nixon and F. Tomaschek, "Introduction to the special issue emergence of speech and language from prediction error: Error-driven language models," *Language, Cognition and Neuroscience*, vol. 38, no. 4, pp. 411–418, Apr. 2023. DOI: 10.1080/23273798.2023.2197650.

[5] M. Peters *et al.*, "Deep Contextualized Word Representations," *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*, pp. 2227–2237, 2018. DOI: 10.18653/v1/N18-1202.

[6] M. Shanahan, "Talking about Large Language Models," *Commun. ACM*, vol. 67, no. 2, pp. 68–79, Jan. 2024. DOI: 10.1145/3624724.

[7] J. Hoffmann *et al.*, *Training Compute-Optimal Large Language Models*, Mar. 2022. DOI: 10.48550/arXiv.2203.15556.

[8] M. Lamparth and J. Schneider, "Why the Military Can't Trust AI," *Foreign Affairs*, pp. 1–8, Apr. 2024.

[9] J. Ge *et al.*, "Development of a liver disease–specific large language model chat interface using retrieval-augmented generation," *Hepatology*, vol. 80, no. 5, pp. 1158–1168, Nov. 2024. DOI: 10.1097/HEP.0000000000000834.

[10] U. Khandelwal *et al.*, *Generalization through Memorization: Nearest Neighbor Language Models*, Feb. 2020. DOI: 10.48550/arXiv.1911.00172.

[11] K. Guu *et al.*, "REALM: Retrieval-Augmented Language Model Pre-Training," in *Proceedings of the 37th International Conference on Machine Learning*, PMLR, Nov. 2020, pp. 3929–3938.

[12] W. Xu *et al.*, "SESCORE2: Learning Text Generation Evaluation via Synthesizing Realistic Mistakes," in *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, Toronto, Canada: Association for Computational Linguistics, 2023, pp. 5166–5183. DOI: 10.18653/v1/2023.acl-long.283.

[13] C. Gutierrez and J. F. Sequeda, "Knowledge graphs," *Communications of the ACM*, vol. 64, no. 3, pp. 96–104, Mar. 2021. DOI: 10.1145/3418294.

[14] M. Boudin *et al.*, "The OREGANO knowledge graph for computational drug repurposing," *Scientific Data*, vol. 10, no. 1, p. 871, Dec. 2023. DOI: 10.1038/s41597-023-02757-0.

[15] V. K. Chaudhri *et al.*, "Knowledge graphs: Introduction, history, and perspectives," *AI Magazine*, vol. 43, no. 1, pp. 17–29, Mar. 2022. DOI: 10.1002/aaai.12033.

[16] N. Torabian *et al.*, "Enhancing Knowledge graph with Selectional Preferences," *Research Square Platform LLC*, pp. 1–23, Nov. 2023. DOI: 10.21203/rs.3.rs-3620069/v1.

[17] R. Ludolph, A. Allam, and P. J. Schulz, "Manipulating Google's Knowledge Graph Box to Counter Biased Information Processing During an Online Search on Vaccination: Application of a Technological Debiasing Strategy," *Journal of Medical Internet Research*, vol. 18, no. 6, e137, Jun. 2016. DOI: 10.2196/jmir.5430.

[18] G. Michelet and F. Breitinger, "ChatGPT, Llama, can you write my report? An experiment on assisted digital forensics reports written using (local) large language models," *Forensic Science International: Digital Investigation*, vol. 48, p. 301 683, Mar. 2024. DOI: 10.1016/j.fsidi.2023.301683.

[19] K. I. Roumeliotis, N. D. Tselikas, and D. K. Nasiopoulos, "LLMs in e-commerce: A comparative analysis of GPT and LLaMA models in product review evaluation," *Natural Language Processing Journal*, vol. 6, p. 100 056, Mar. 2024. DOI: 10.1016/j.nlp.2024.100056.

[20] L. Tunstall *et al.*, *Zephyr: Direct Distillation of LM Alignment*, Oct. 2023. DOI: 10.48550/arXiv.2310.16944.

[21] A. Q. Jiang *et al.*, *Mistral 7B*, Oct. 2023. DOI: 10.48550/arXiv.2310.06825.

[22] P. Assiroj, A. Kurnia, and S. Alam, "The performance of Naïve Bayes, support vector machine, and logistic regression on Indonesia immigration sentiment analysis," *Bulletin of Electrical Engineering and Informatics*, vol. 12, no. 6, pp. 3843–3852, Dec. 2023. DOI: 10.11591/eei.v12i6.5688.

[23] W. Chen *et al.*, *MuRAG: Multimodal Retrieval-Augmented Generator for Open Question Answering over Images and Text*, Oct. 2022. DOI: 10.48550/arXiv.2210.02928.