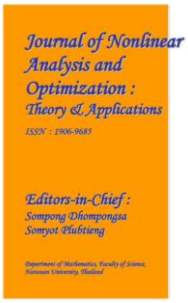
Journal of Nonlinear Analysis and Optimization
Vol. 15, Issue. 1, No.15 : 2024
ISSN : **1906-9685**



Journal of Nonlinear Analysis and Optimization : Theory & Applications

ISSN : 1906-9685

Editors-in-Chief :
Sompong Dhompongsa
Somyot Plubtieng

Department of Mathematics, Faculty of Science, Naresuan University, Thailand

Paper ID: ICRTEM24_143   **ICRTEM-2024 Conference Paper**

ENHANCING LARGE LANGUAGE MODEL EFFICIENCY FOR ENGINEERING STUDENTS: A WEB-BASED INTEGRATION WITH OPTIMIZED PROMPTS

[#1]D. SHRAVANI, ***UG Student,***

[#2]K. NAGENDRA, ***UG Student,***

[#3]D. S. V. BHASKARA VARMA, ***UG Student,***

[#4]Dr. S. KIRUBAKARAN, ***Professor,***

Department of CSE,

CMR COLLEGE OF ENGINEERING & TECHNOLOGY, HYDERABAD,

ABSTRACT - As the integration of artificial intelligence (AI) becomes increasingly relevant in various industries, engineering students seek efficient and innovative ways to leverage AI technologies for their academic and practical needs. This project aims to empower engineering students to use Large Language Models effectively by developing a web-based integration that optimizes the use of prompts. The proposed solution addresses the challenges engineering students may encounter while interacting with Large Language Models like ChatGPT, Google Bard etc., which includes generating precise responses, extracting relevant technical information, and integrating the AI model seamlessly into engineering workflows. To overcome these challenges, the project will be focused on enhancing the user experience through intuitive prompt engineering. Ultimately, this project aims to equip engineering students with a powerful and user-friendly tool that enhances their AI-assisted learning experience. By efficiently using Large Language Models through optimized prompts and seamless integration into their academic workflows, engineering students can access valuable insights, expedite their problem-solving capabilities, and foster a deeper understanding of complex engineering concepts.

Keywords- *AI Integration, Engineering Education Tools, Web-based Assistance, Prompt Optimization, Technical Information Retrieval, Enhanced Learning, Problem-Solving Acceleration, LLM Applications.*

I. INTRODUCTION

In today's AI-driven landscape, engineering students are actively seeking innovative ways to leverage OpenAI's ChatGPT. Our project aims to empower these students by creating a web-based integration that optimizes prompts for ChatGPT. We're addressing challenges like precise responses, extracting technical information, and seamless integration into engineering workflows. Our goal is to enhance the user experience with refined prompt engineering, providing a user-friendly tool for improved AI-assisted learning. By mastering ChatGPT with tailored prompts, we aim to enable students to gain valuable insights, enhance problem-solving, and deepen their understanding of engineering concepts. This project bridges the gap between AI and engineering education, transforming the future of learning in this field.

II. RELATED WORK

In the quest for innovation and efficiency, modern projects frequently rely on existing solutions as fundamental building blocks for development. This approach not only recognizes the expertise and advancements of those who came before us but also nurtures a collaborative ecosystem where ideas can evolve and confront new challenges. In our project, we wholeheartedly embrace this ethos, conscientiously integrating elements from existing solutions to enrich our endeavor. These existing solutions serve as guiding lights, offering insights and frameworks that shape the direction of our project.

A. **Training language models to follow instructions with human feedback (InstructGPT).** Recent large LMs prioritize webpage token prediction over user instructions (Radford et al., 2019; Brown et al., 2020; Fedus et al., 2021; Rae et al., 2021; Thoppilan et al., 2022), causing misalignment. Aligning LMs involves training for helpfulness, honesty, and harmlessness (Leike et al., 2018; Askell et al., 2021).Using RLHF (Christiano et al., 2017; Stiennon et al., 2020), we fine-tune GPT-3 to create InstructGPT. This entails hiring contractors, collecting human-written demonstrations, training a reward model, and fine-tuning with PPO (Schulman et al., 2017).Evaluation includes human ratings and automatic assessments on various NLP datasets. Models of different sizes (1.3B, 6B, and 175B parameters), all GPT-3 based, are trained.

B. **LaMDA: Language Models for Dialog Applications.** Language model pre-training in NLP [1-12] is enhanced by combining unlabeled text with scaling model and dataset sizes [13]. GPT-3 [12], a 175B parameter model trained on unlabeled text, excels in few-shot learning. Dialog models [14-18] leverage Transformers' ability to represent long-term dependencies in text. LaMDA, a family of Transformer-based neural language models designed for dialog, ranges from 2B to 137B parameters and is pre-trained on a dataset of 1.56T words. LaMDA performs diverse tasks: generating, filtering for safety, grounding on knowledge, and re-ranking. Scaling improves quality, but scaling with fine-tuning enhances LaMDA overall. Quality, safety, and groundedness are key metrics. LaMDA adapts to roles, with fine-tuning models being more helpful.

C. **BlenderBot 3.** Pre-training large language models has significantly advanced open-domain dialogue agents (Adiwardana et al., 2020; Zhang et al., 2020; Roller et al., 2021). Recent studies indicate that fine-tuning language models (Roller et al., 2021; Thoppilan et al., 2022; Ouyang et al., 2022; Bai et al., 2022) yields significant improvements. Concerns persist regarding scalability and alignment with user interests when using paid crowdworkers. Advocacy for public deployment of these agents aims to stimulate innovation (Roller et al., 2020; Shuster et al., 2021b). This report introduces BlenderBot 3 (BB3) to facilitate accessible and reproducible research (Sonnenburg et al., 2007; Pineau et al., 2021). In 2022, researchers introduced BB3, a transformer model derived from OPT175B (Zhang et al., 2022) and fine-tuned for modular tasks (Shuster et al., 2022). They explored human feedback training and robust learning algorithms (Xu et al., 2022b; Ju et al., 2022), reporting BB3's superior performance compared to existing chatbots and releasing model resources.

III. METHODS AND EXPERIMENTAL DETAILS

A. *High-level methodology*

The methodology of our project functions akin to a machine learning algorithm that learns from past mistakes and avoids repeating them. Our project's architecture revolves around presenting text prompts to users rather than requiring them to formulate prompts themselves. This design simplifies the user experience, allowing them to focus on obtaining the information they need without unnecessary delays caused by prompting. However, utilizing Large Language Models (LLMs) for prompting requires a certain level of familiarity with the subject matter, which is typically facilitated by the website or system.

Central to our project is the implementation of a feedback model, which enables administrators to provide feedback on prompts and content. Administrators play a pivotal role in manipulating the prompts and content within the project, as they are responsible for curating notes and ensuring accuracy. While this feedback process may require time, it ultimately streamlines the prompt generation process for all users.

The website or system is designed to present prompts in a simple and precise manner, aligning with the expectations of students and organizations. Upon receiving user input regarding the desired subject, the system maps it to relevant subtopics or sections, thereby facilitating efficient information retrieval. It's important to note that specifying the subject is mandatory for accurate prompt mapping and subsequent output generation.

The efficiency of our system's output is attributed to the use of the Palm API, developed and utilized by Google. This API boasts a vast database and robust capabilities, enabling accurate and reliable output generation. By leveraging the Palm API, repetitive actions can be automated, ensuring consistent performance and desired outcomes.

In summary, our project's methodology revolves around efficient prompt generation and output generation facilitated by user feedback and the utilization of advanced APIs like Palm API. Through streamlined processes and accurate output generation, we aim to enhance the user experience and facilitate seamless access to information.

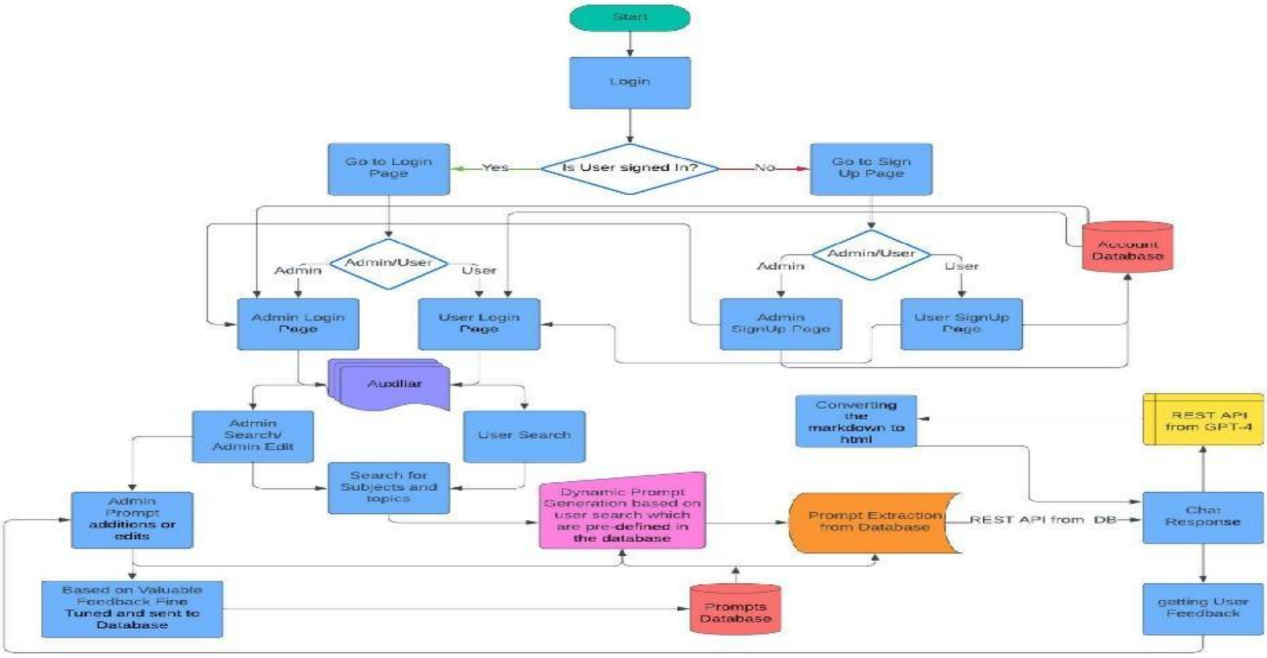


Fig.Architecture of the Mod

B. **Dataset**

Our dataset mainly includes text prompts sent to the Palm API, particularly those generated using an older version of the InstructGPT models. These prompts were created on an interactive platform through supervised learning on a portion of our demonstration data. Our team majorly focused on Engineering aspects. Our dataset where it is a Single-Model Architecture where it allows the model to handle a wide array of tasks, including generating responses, filtering out unsafe or inappropriate responses, grounding responses in known sources, and re-ranking responses based on quality.

To train the very first InstructGPT models, we asked admins to write prompts themselves. This is because we needed an initial source of instruction-like prompts to bootstrap the process, and these kinds of prompts weren’t often submitted to the regular Large Language Model on the API.

IV. RESULTS AND DISCUSSIONS

The exploration of existing solutions sheds light on the diverse approaches and methodologies available to enhance the capabilities of language models in engineering contexts. Each solution offers unique benefits and insights, contributing to the overarching goal of improving language model performance for engineering tasks.

Training Language Models with Human Feedback:

Approach: This solution emphasizes the importance of training language models with human feedback, leveraging reinforcement learning to enhance their ability to understand and follow instructions accurately.
Applicability to Engineering: Precision and accuracy are paramount in engineering tasks, making this approach highly relevant. Training models to interpret engineering instructions can significantly improve their efficiency in handling technical queries.
Benefits: The resulting models possess a deeper understanding of engineering terminology and context, enabling ChatGPT to provide more accurate and actionable responses in engineering domains.

LaMDA: Language Models for Dialog Applications:

Approach: LaMDA focuses on enhancing conversational abilities in language models, aiming to make interactions more natural and engaging.
Applicability to Engineering: While not tailored specifically for engineering tasks, LaMDA'sconversational capabilities can enhance user engagement in engineering-related queries.
Benefits: LaMDA facilitates more natural and context-rich conversations, making it easier for engineers to interact with language models and discuss complex engineering problems.

BlenderBot 3: A Deployed Conversational Agent:

Approach: BlenderBot 3 prioritizes responsible and continuous learning, with a focus on safety mechanisms and collecting user feedback to enhance conversational abilities.
Applicability to Engineering: In engineering contexts, where accuracy and safety are crucial, BlenderBot 3's features are highly desirable for providing accurate and safe information.
Benefits: The safety mechanisms and continuous learning of BlenderBot 3 ensure accurate responses in engineering domains, while user feedback fine-tunes the model's understanding of engineering concepts.

Comparison:

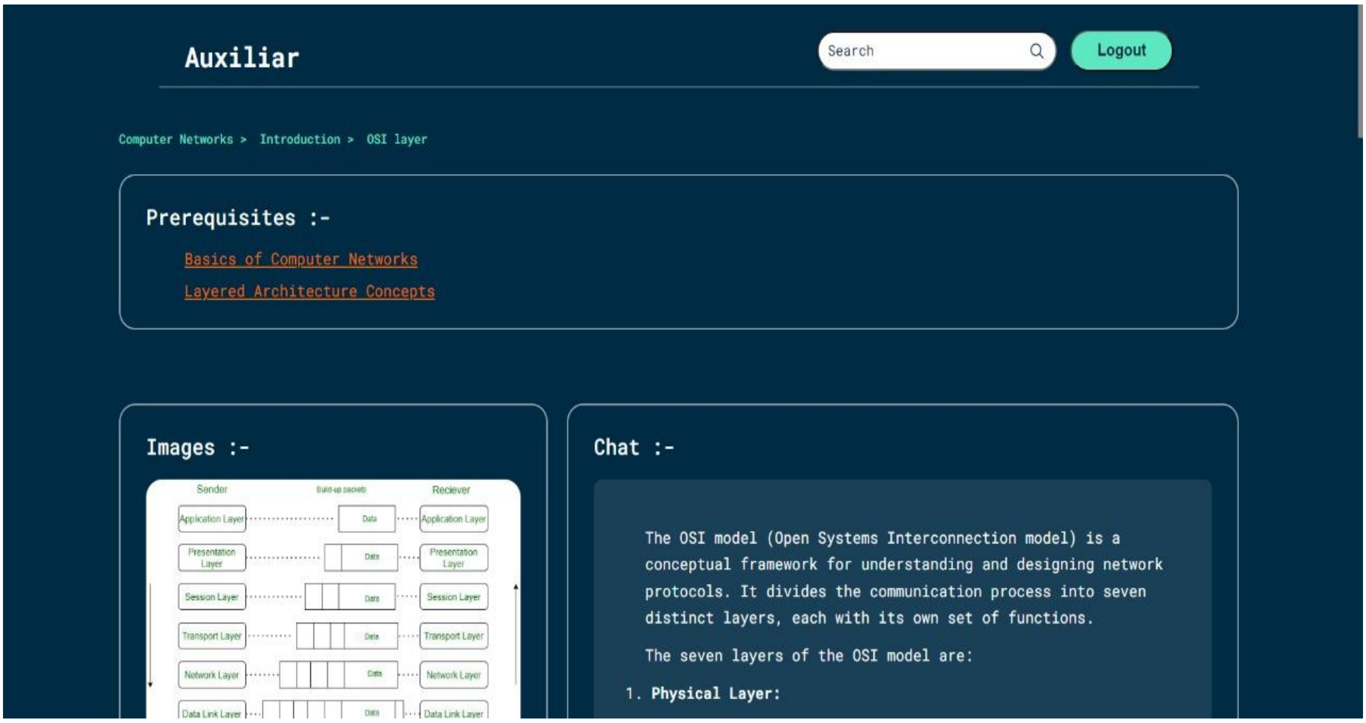
Each solution brings its own merits to the table. While the first solution focuses on precision and task-specific understanding, LaMDA enhances conversational aspects, and BlenderBot 3 prioritizes responsible engagement and continuous learning. Combining aspects of all three solutions presents an ideal approach, leveraging engineering-specific data for technical foundation, enhancing conversational aspects, and ensuring responsible and continuous learning.

Integration:

By integrating features from these solutions, a language model like ChatGPT can efficiently handle a wide range of engineering tasks while maintaining safety and user-friendliness in its responses.

The exploration of existing solutions underscores the importance of continual improvement and innovation in language model development. By incorporating elements from diverse methodologies, our project aims to enhance ChatGPT's capabilities for engineering students, providing efficient and accurate responses while prioritizing safety and user engagement. Through the integration of these insights, our project strives to optimize language model performance in engineering contexts, ultimately facilitating seamless interactions and knowledge dissemination.

Below, you'll find a series of images showcasing our innovative web project in action. These visuals provide a glimpse into the user interface, features, and functionality of our web application. Take a moment to explore and discover how our project can revolutionize your online experience. From streamlined interactions to intuitive design, witness firsthand the power and potential of our creation. Dive in and envision the possibilities with our web project.



Computer Networks > Introduction > OSI layer

Prerequisites :-

Basics of Computer Networks

Layered Architecture Concepts

Images :-

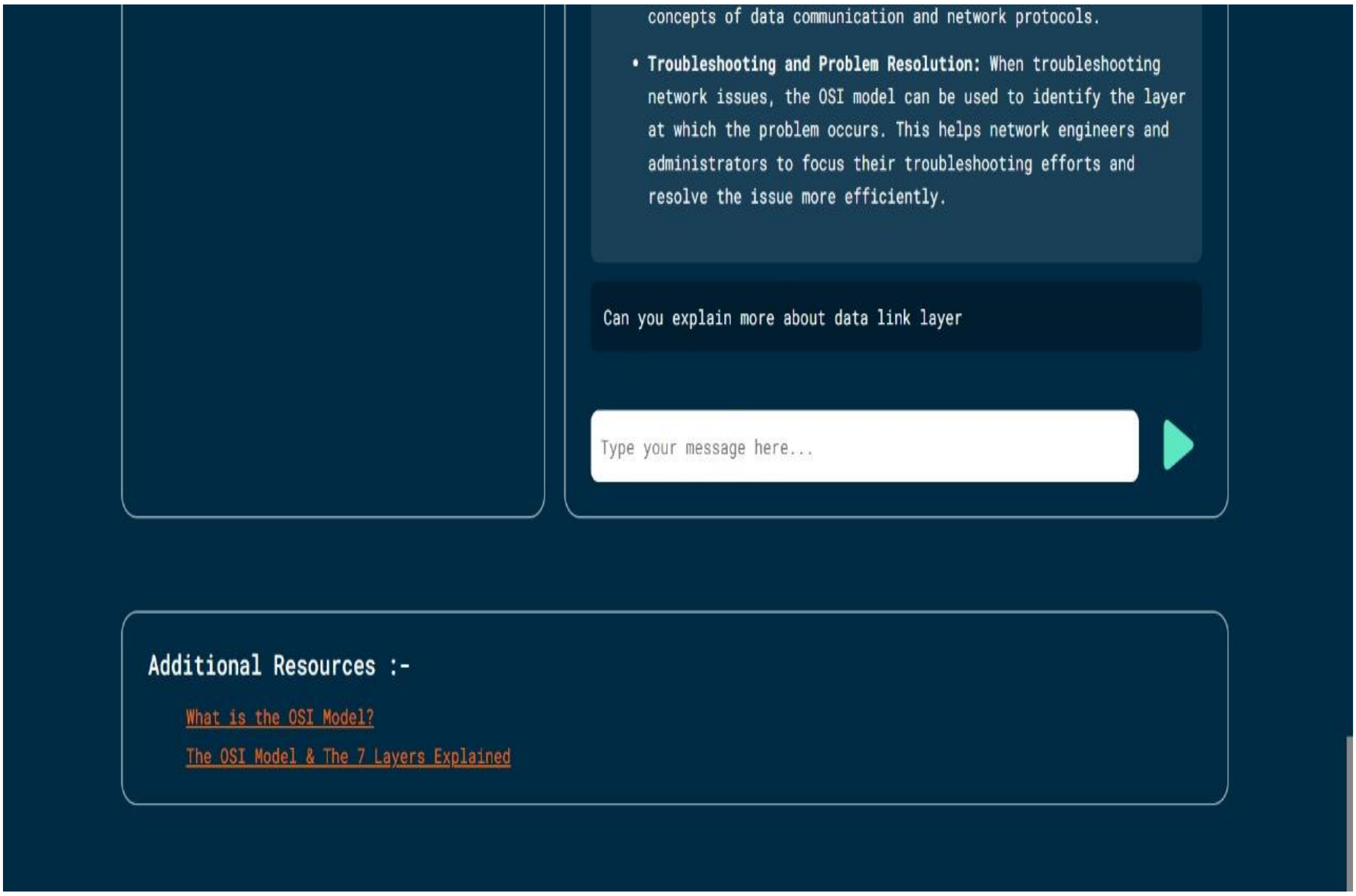
Chat :-

The OSI model (Open Systems Interconnection model) is a conceptual framework for understanding and designing network protocols. It divides the communication process into seven distinct layers, each with its own set of functions.

The seven layers of the OSI model are:

1. **Physical Layer:**

Fig. User Interface - 1



concepts of data communication and network protocols.

- **Troubleshooting and Problem Resolution:** When troubleshooting network issues, the OSI model can be used to identify the layer at which the problem occurs. This helps network engineers and administrators to focus their troubleshooting efforts and resolve the issue more efficiently.

Can you explain more about data link layer

Additional Resources :-

What is the OSI Model?

The OSI Model & The 7 Layers Explained

Fig. User Interface - 2

V. CONCLUSION

In conclusion, the background work and results we have explored offer valuable insights into the potential methodologies for achieving our project's final output. Through the adoption of these methods, we can pave the way for a language model that excels in engineering applications while prioritizing precision, engagement, and safety.

Training Language Models with Human Feedback: This approach stands as a cornerstone for enhancing task-specific precision in engineering applications. By integrating human feedback mechanisms, we can refine the model's understanding and ensure responsible and user-friendly interactions.

LaMDA: Although not tailored specifically for engineering, LaMDA's conversational enhancements can greatly benefit interactions with technical professionals. Its ability to improve the naturalness and engagement of conversations adds depth to the model's capabilities.

BlenderBot 3: Particularly well-suited for engineering applications, BlenderBot 3's emphasis on responsible AI and continuous learning ensures the provision of accurate and reliable information in engineering domains. Its safety mechanisms and user feedback loops contribute to a trustworthy and efficient model.

A comprehensive approach entails integrating these methods harmoniously to create a robust language model. By training models for precision, improving conversational abilities, and prioritizing responsible and safe interactions, we can realize the full potential of our project in engineering applications.

With the completion of this project, we envision a language model that not only meets but exceeds the expectations of users in engineering contexts. By leveraging the methodologies discussed and implementing them effectively, we are poised to deliver a solution that empowers users, enhances productivity, and fosters innovation in engineering domains.

References

[1] Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L. Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, Amanda Askell, Peter Welinder Paul Christiano, Jan Leike, Ryan Lowe, OpenAI, “Training language models to follow instructions with human feedback”, arXiv:2203.02155v1 [cs.CL] 4 Mar 2022.

[2] Alec Radford, Karthik Narasimhan, Tim Salimans, Ilya Sutskever, OpenAI, “Improving Language Understanding by GenerativePre-Training”,https://s3-us-west-2.amazonaws.com/openai-assets/research-covers/language-unsupervised/language_understanding_paper.pdf

[3] Romal Thoppilan, Daniel De Freitas, Jamie Hall, Noam Shazeer, Apoorv Kulshreshtha, Heng-Tze Cheng, Alicia Jin, Taylor Bos, Leslie Baker, Yu Du YaGuang Li, Hongrae Lee Huaixiu Steven Zheng Amin Ghafouri Marcelo Matthew Lamm Viktoriya Kuzmina Joe Fenton Aaron Cohen Rachel Bernstein Ray Kurzweil Blaise Aguera-Arcas Claire Cui Marian Croak Ed Chi Quoc Le, Google, “LaMDA: Language Models for Dialog Applications”, arXiv:2201.08239v3 [cs.CL] 10 Feb 2022

[4] Kurt Shuster† , Jing Xu† , Mojtaba Komeili† , Da Ju† , Eric Michael Smith, Stephen Roller, Megan Ung, Moya Chen, Kushal Arora+, Arthur Szlam, Y-Lan Boureau, Melanie Kambadur, Jason Weston Meta AI + Mila / McGill University, “BlenderBot 3: a deployed conversational agent that continually∗ learns to responsibly engage”, arXiv:2208.03188v3 [cs.CL] 10 Aug 2022.

[5] Amelia Glaese* , Nat McAleese* , Maja Trebacz* , John Aslanides* , et al *Equal contributions, all affiliations DeepMind, “Improving alignment of dialogue agents via targeted human judgements”, arXiv:2209.14375v1 [cs.LG] 28 Sep 2022.

[6] Ekin, Sabit (2023): “Prompt Engineering For ChatGPT: A Quick Guide To Techniques, Tips, And Best Practices. TechRxiv. Preprint.” https://doi.org/10.36227/techrxiv.22683919.v2

[7] https://www.deeplearning.ai/

[8] https://learnprompting.org/docs/basics/instructions

[9] Prompt engineering: https://www.youtube.com/@engineerprompt

[10] https://chat.openai.com/

[11] https://bard.google.com/chat

[12] https://ai.meta.com/llama/

[13] Google LaMDA | Discover AI use cases (gpt3demo.com)