Streamlining Text Generation with AI Powered Prompt Simplification Strategies

Maddula Ratna Mohitha[1], Panduranga Vital Terlapu*[2], B. Anusrilekha[3], R. Narendra[4], P. Uday Shekar[5] ,P. Krishna Chaitanya[6] , T. Rohith Kumar[7]

[1,2,3,4,5,6,7]Department of Computer Science & Engineering, Aditya Institute of Technology and Management, Tekkali-532201, India.

{r.mohithamaddula, *vital2927, baratamanusrilekha7, narendrarokkam145, udayperuri9, potnurukrishnachaitanya, thyadirohith}@gmail.com

Abstract. In recent times, generating understandable prompts for AI has been a significant problem, which in turn leads to inaccurate results. In essence, the problem is about finding ways to make AI understand and respond accurately to the prompts given to it, which is crucial for improving its overall performance and usefulness in various applications. This paper proposes a novel approach to enhance AI comprehension by generating tailored understanding prompts through prompt engineering techniques in Natural Language Processing, along with advanced Transformer-based deep learning models. Our project integrates these techniques to transform a base prompt into a set of diverse and comprehensive understanding prompts. To ensure data security, we have also implemented data encryption standard and Blowfish encryption algorithms to protect sensitive information during the transformation process. The resulting prompts will be used to train AI models, enabling them to grasp nuanced details and context when responding to user queries. The paper's significance lies in its potential to improve the quality of AI-generated responses across a range of applications, including natural language understanding, question answering, and content generation. Crucially, the developed web application, constructed using the MERN Stack, promises more reliable and insightful interactions with AI systems, effectively bridging the gap between human comprehension and AI-generated content.

Keywords: Artificial Intelligence, Prompt Engineering, Natural Language Processing, MERN Stack

1 Introduction

Prompt engineering is a methodology employed within the domain of natural language processing (NLP) in artificial intelligence (AI).[1] It involves crafting textual descriptions that delineate the specific tasks expected of the AI system. These prompts serve as detailed instructions for the desired output that the model should generate. Guided by this input, the AI produces an output, which can take various forms, with the aim of using human-understandable text conversationally to interact with models.[2] By embedding the task description directly in the input, the model gains flexibility and expands its range of capabilities.

K. R. Madhavi et al. (eds.), *Proceedings of the International Conference on Computational Innovations and Emerging Trends (ICCIET 2024)*, Advances in Computer Science Research 112,
https://doi.org/10.2991/978-94-6463-471-6_16

models.[2] By embedding the task description directly in the input, the model gains flexibility and expands its range of capabilities.

1.1 Problem Statement

In the realm of Artificial Intelligence (AI), the ability to understand and respond accurately to prompts has emerged as a pivotal challenge in recent times. This difficulty often leads to inaccurate results, hindering AI's overall performance and applicability across various domains. Addressing this issue requires innovative solutions that empower AI systems to comprehend prompts effectively, thereby enhancing their utility in real-world applications.

1.2 Proposed System

This paper proposes a novel approach to tackle the problem of AI comprehension by leveraging prompt engineering techniques in Natural Language Processing (NLP) alongside advanced Transformer-based deep learning models.[3] By integrating these methodologies, the project aims to refine base prompts into a diverse set of tailored understanding prompts. Moreover, to safeguard sensitive data during the transformation process, robust encryption algorithms such as the Data Encryption Standard and Blowfish are implemented, ensuring data security throughout.

The resulting set of prompts will serve as training data for AI models, enabling them to grasp nuanced details and contextual cues when responding to user queries.[4] The significance of this initiative lies in its potential to elevate the quality of AI-generated responses across a spectrum of applications, encompassing natural language understanding, question answering, and content generation. It enhances individuals' comprehension of AI-generated content, hence increasing the overall reliability and depth of the process.

2 Literature Survey

The literature survey highlights recent advancements in prompt-based techniques, including consistency-based self-adaptive prompting, automatic prompting, and verbalizer generation, resulting in improved few-shot learning and LLM accuracy.

Table 1. Researchers Evolutions on different models Descriptions

Author	Year	Description	Results
Tianyu Gao et al. [5]	2021	Prompt-based fine-tuning and demonstrations.	LM-BFF achieves up to 30% absolute improvement in few-shot learning.
Xingchen Wan et al. [6]	2023	Consistency-based Self-adaptive Prompting	COSP boosts LLM performance by up to 15% in zero-shot.
Yuhang Zhou et al. [7]	2023	Automatic prompt and verbalizer generation.	Best average accuracy: 73.2%
Qingyan Guo et al. [8]	2023	EVOPROMPT: Evolutionary Prompt Optimization	It enhances LLM performance by up to 25%.
Yuxian Gu et al. [9]	2021	Pre-trained Prompt Tuning (PPT) framework.	PPT outperforms full-model fine-tuning in both full-data and few-shot scenarios.
Simran Arora et al. [10]	2022	Ask Me Anything Prompting (AMA)	AMA prompting method boosts LLM performance by 10.2% on average.
Albert Lu et al. [11]	2023	Prompt-centric analysis technique.	Analysis of prompt constraints on GPT-3, BLOOM, and OPT.

Ning Ding et al. [12]	2021	Prompt-learning with cloze-style prompts.	Improved entity typing in supervised, few-shot, and zero-shot scenarios.
Laria Reynolds et al. [13]	2021	Few-shot prompt analysis.	0-shot prompts outperform few-shot prompts in probing GPT-3's capabilities.

3 Methodology

The study focuses on creating a user-friendly web interface using the MERN Stack, integrating APIs and AI models for prompt generation and response accuracy, and offering customizable prompt creation based on user preferences and topic relevance. Fig. 1 shows the Streamlining Text Generation with AI-Powered Prompt Proposal Working Model in detailed.

Web Interface Development Using MERN Stack

Begin by designing and developing a user-friendly web interface with the MERN Stack (MongoDB, Express.js, React.js, Node.js). [14] The interface is easy to use and responsive so that users can interact seamlessly. We Implement user authentication, registration, and profile management features to provide users with secure access and personalized experiences.

API Integration with Prompt Generation

Integrate APIs into our project to enable prompt generation functionality. so we included APIs or services that provide prompt generation or natural language processing (NLP) functionality that meets your project's needs. These APIs are seamlessly integrated into our web interface so that users can easily access prompt generation features[15][16].

Integration of AI for Response Generation

We have connected the created web application with the ChatGPT 3.5 Turbo API to accurately respond to prompts. This involves integrating a pre-trained AI model to generate responses. Whenever a user enters a query, our application fetches the data from the pre-trained model. Based on the user's input, it provides the required response to the users.

Basic Prompt Generation

The focus is on adding functionality that generates base prompts based on user-selected topics. Using the MERN Stack's capabilities to retrieve and display base prompts related to the user's selected topic. Create code to dynamically generate base prompts based on topic relevance, diversity, and accuracy.

Custom Prompt Creation

Provide users with the ability to create custom prompts. Create an intuitive interface that allows users to enter their prompt ideas or modify existing prompts based on the system's suggestions. Add functionality to securely store these custom prompts in MongoDB and associate them with the appropriate user accounts for future access.

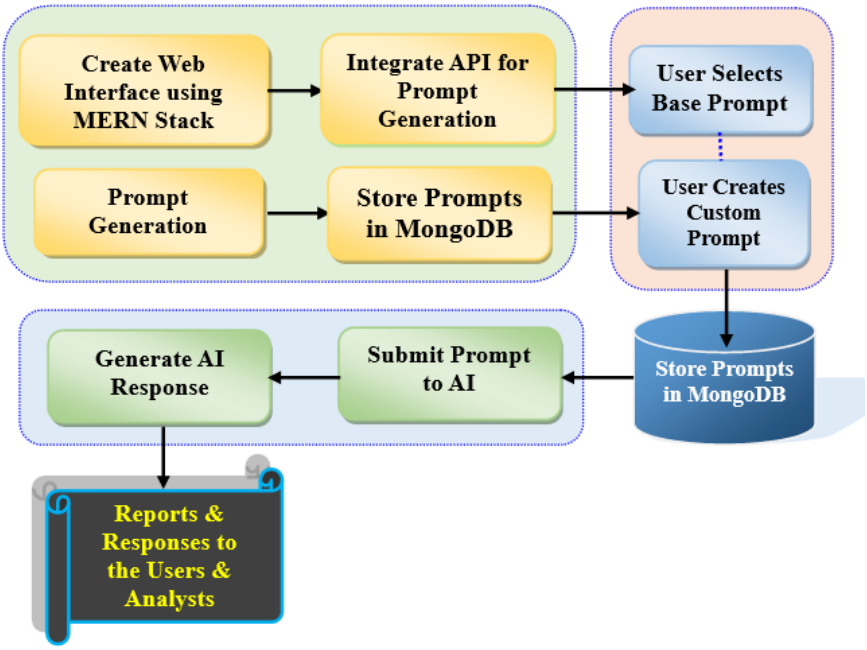

Fig. 1. Proposal Working Model

4 Data Security Measures

In our application, ensuring the security of user data and generated responses is paramount. To achieve this, we employ robust encryption algorithms such as DES (Data Encryption Standard) and Blowfish. These algorithms play a crucial role in safeguarding sensitive information from unauthorized access or tampering. In this, we discuss encryption algorithms like DES and Blowfish, which transform plaintext data into ciphertext for secure communication. These algorithms are essential for safeguarding sensitive information, ensuring user privacy, and fostering trust in security measures.

4.1 Data Encryption Standard (DES)

DES is a symmetric-key encryption algorithm that enhances data security by encrypting user data before transmission over the network, preventing unauthorized access. It also protects stored data in databases, ensuring data remains unreadable even with unauthorized access. DES is used to generate secure authentication tokens, providing an extra layer of security to verify user identity, reducing the risk of unauthorized access or identity theft.

Secure Transmission: When users interact with our application, their data, including login credentials, prompts, and responses, are encrypted using DES before transmission over the network. This prevents eavesdropping and interception by malicious entities.

Stored Data Protection: User data stored in our databases is encrypted using DES. Even if unauthorized access occurs, the encrypted data remains unreadable without the decryption key, thus maintaining confidentiality.

3. Authentication Tokens: DES encryption is utilized to generate secure authentication tokens. These tokens verify the identity of users during their session, ensuring that only authorized individuals can access the application's functionalities.

4.2 Blowfish Encryption Algorithm

Blowfish is a symmetric-key block cipher known for its robustness and flexibility. Its usage enhances the security of our application in the following ways:
Enhanced Encryption Strength: Blowfish employs variable key lengths, making it adaptable to evolving security requirements. By utilizing Blowfish encryption, we ensure that user data and responses are protected with high-level security measures.
Protection Against Brute Force Attacks: Blowfish's key expansion mechanism and complex encryption process significantly increase resistance against brute force attacks. This deters unauthorized parties from decrypting sensitive information through exhaustive trial and error methods.
Data Integrity Assurance: In addition to encryption, Blowfish also provides integrity checks through cryptographic hashing. This ensures that the data remains intact and unaltered during transmission and storage, maintaining its reliability and trustworthiness.
By incorporating DES and Blowfish encryption algorithms into our application's security framework, we prioritize the confidentiality, integrity, and authenticity of user data and generated responses. This proactive approach to data security fosters trust among our users, assuring them that their information is safeguarded against potential threats.

5 Implementation

5.1 User Registration and Authentication

User Sign-up: Implements a user registration page where users can sign up by providing necessary information like username, email, and password.
Admin Verification: Admin verifies the user by generating an OTP and sending it to the user's registered email or phone number.
User Authentication: Users enter the OTP to authenticate their account. Only verified users can proceed.
Login: Verified users can log in using their credentials. Implement login functionality where users enter their username/email and password to access the application.
Forgot Password: Provide a "Forgot Password" option where users can reset their password by verifying their identity through email or phone number.

5.2 User Login and Home Page

User Login: Once authenticated, users are redirected to the home page.
Home Page: Design a user-friendly home page that introduces the application's features and options.
Application Features: Provide an overview of the application's main features, such as base prompt page and custom prompt page.

5.3 Base Prompt Page

Prompt Input: Users can input a question or topic of interest.
Prompt Suggestions: Based on the input, provide various prompt suggestions relevant to the topic.
Prompt Selection: Users can select and use the prompt suggestions that best fit their needs.
Response Generation: Implement functionality to provide accurate responses based on the selected prompt suggestions.

5.4 Custom Prompt Page

Custom Prompt Submission allows users to create personalized prompts based on their specific needs and preferences. The system uses algorithms to generate responses that align with the user's prompt specifications, ensuring personalized and relevant interactions. This enhances the user experience and engagement by analysing the prompt's topic, role, objective, tone, audience, and format. Fig. 2 shows the detailed analysis of custom prompt page set up.

Input Fields: Provide input fields for users to specify details such as topic, objective, tone, audience, and format.
User Choices: Users can make selections or input information for each category.
Custom Prompt Submission: We will enable users to submit their custom prompt specifications. These specifications will include:
Topic: The subject matter or theme around which the prompt revolves.
Act as: The role or perspective the user wants the prompt to assume.
Objective: The desired outcome or purpose of the prompt.
Tone: The emotional or linguistic quality desired in the prompt.
Audience: The intended recipients or readers of the prompt.
Format: The structure or presentation style of the prompt.

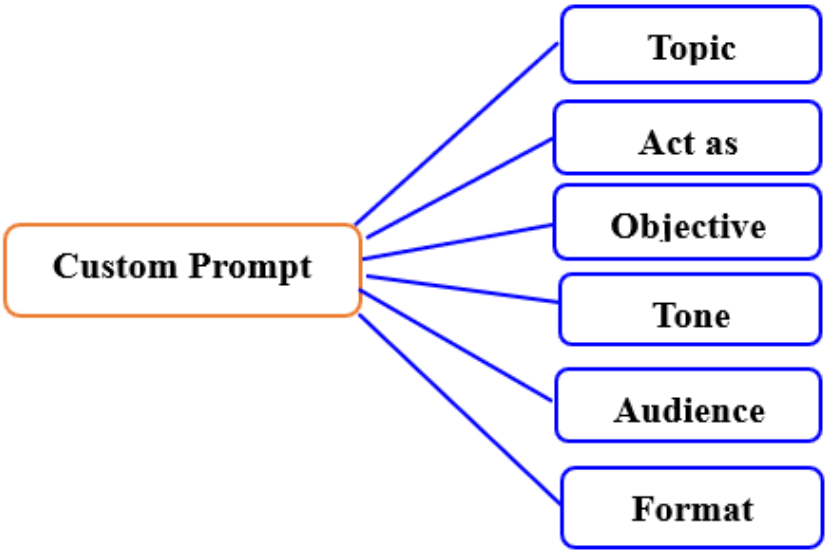

Fig. 2. Custom Prompt Options

Response Generation: Develop algorithms to generate responses tailored to the user's custom prompt specifications. Fig. 3 shows the flow process of response generation. These steps outline the basic functionality and flow of the application. Each step involves both frontend (user interface) and backend (server-side logic) development to ensure a seamless user experience. Additionally, consider implementing security measures such as encryption for sensitive user data and authentication tokens to protect user sessions.

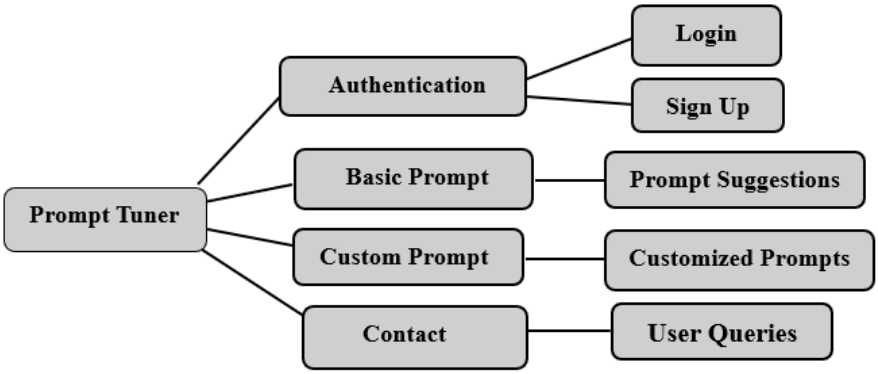

Fig. 3. Flow of process

6 Results and Discussions

The paper introduces cutting-edge technologies, employing prompt engineering in NLP and the MERN Stack. This innovative approach refines base prompts, enhancing AI comprehension significantly. Advanced Transformer-based models form the backbone of an automated pipeline for prompt generation, ensuring accurate and context-aware responses. Robust encryption using DES and Blowfish algorithms protects sensitive data. Integration of the MERN Stack optimizes data storage, server-side development, dynamic UI creation, and backend support, prioritizing security, and user experience. The paper underscores a commitment to advancing AI while safeguarding data integrity and confidentiality.

The results demonstrate the effectiveness of a proposed approach to enhance AI comprehension through tailored understanding prompts. Utilizing Natural Language Processing (NLP) and transformer-based deep learning models, a diverse set of prompts was generated, enhancing AI models' ability to understand nuanced details. The automated pipeline integrated these techniques, streamlining the process and ensuring data security through Diffing and Blowfish encryption algorithms. Our study aims to improve AI-generated responses in natural language understanding, question answering, and content generation. By providing tailored prompts, it enhances user experience and utility. A web application using MERN Stack facilitates reliable interactions between users and AI models. Fig. 4 shows the Accuracies of Learners. The Deep Learning NLP outperformed all other NLP methods in accuracy, with 79% accuracy. Rule-based NLP had the lowest accuracy, while Prompt Engineering demonstrated the highest accuracy (88%), indicating its potential for effective natural language processing.

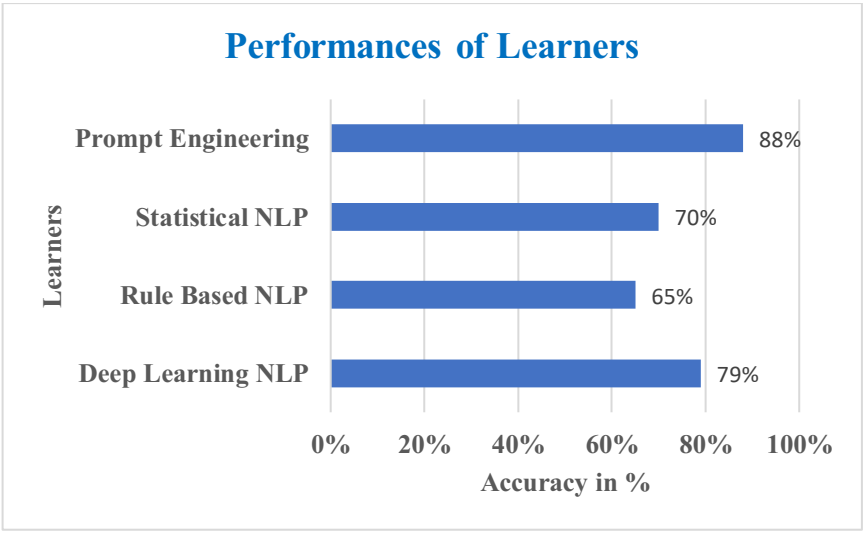

Fig.4. Performances for the NLP and Prompt Engineering Learners

On the Base Prompt page, when a user enters a topic, the prompt tuner attempts to grasp it and provides the user with default prompt suggestions from the trained data related to the topic. The user can then choose the desired prompts to generate the required response. On the Custom Prompt page, users could customize prompts using options such as topic, role, objective, tone, audience, and format. This allows users to tailor their prompts and generate responses accordingly. Fig. 5 shows the results of "Base Prompt page, users are presented with intuitive prompt suggestions, streamlining their input process, and facilitating smoother interactions with the system". Fig. 6 shows the results of "Users are empowered with selective prompt options on the custom prompt page, facilitating tailored outputs effectively".

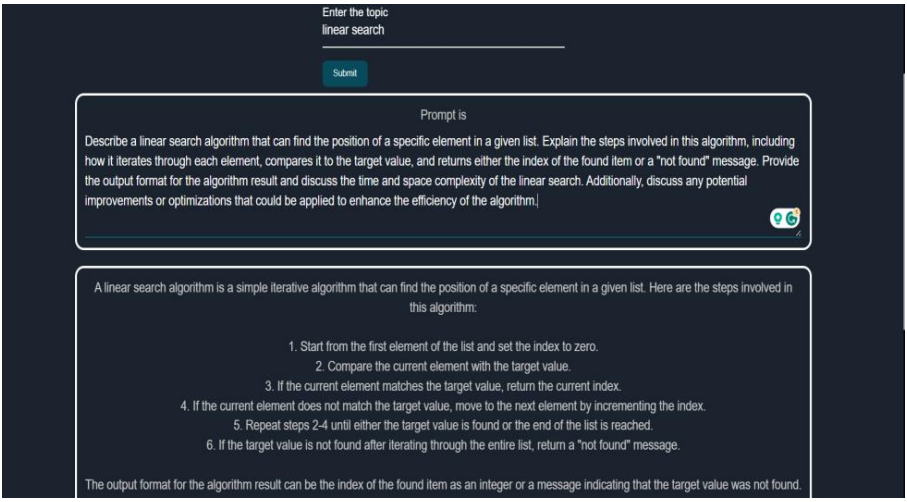

Fig.5. In the Base Prompt page, users are presented with intuitive prompt suggestions, streamlining their input process, and facilitating smoother interactions with the system.

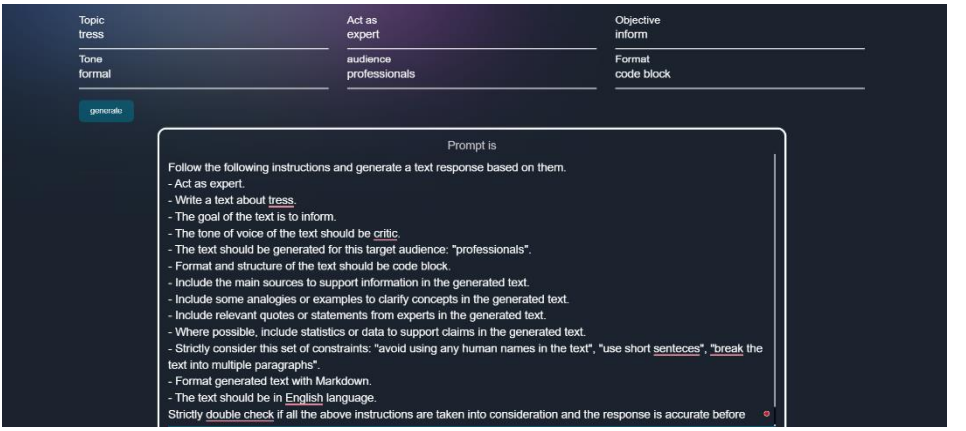

Fig. 6. Users are empowered with selective prompt options on the custom prompt page, facilitating tailored outputs effectively.

7 Conclusion

The paper “Streamlining Text Generation with AI Powered Prompt Simplification Strategies” demonstrates the significant strides made in leveraging cutting-edge technologies such as NLP and the MERN Stack to enhance AI comprehension through tailored prompts. By employing prompt engineering techniques, we have refined base prompts to create a diverse and comprehensive set, thereby greatly augmenting AI models' nuanced understanding capabilities. The integration of advanced Transformer-based deep learning models further solidifies the effectiveness of our automated prompt generation pipeline. Moving forward, the insights gained from this project pave the way for continued advancements in AI-driven technologies, promising even greater precision and context sensitivity in AI-generated outputs.

References

1. Mesko, B. Prompt engineering as an important emerging skill for medical professionals: tutorial. Journal of Medical Internet Research, 25, e50638. (2023)
2. White, J., Fu, Q., Hays, S., Sandborn, M., Olea, C., Gilbert, H., ... & Schmidt. A prompt pattern catalog to enhance prompt engineering with chatgpt. arXiv preprint arXiv:2302.11382. (2023)
3. Shin, T., Razeghi, Y., Logan IV, R. L., Wallace, E., & Singh, S. Autoprompt: Eliciting knowledge from language models with automatically generated prompts. arXiv preprint arXiv:2010.15980. (2020)
4. Zamfirescu-Pereira, J. D., Wei, H., Xiao, A., Gu, K., Jung, G., Lee, M. G., ... & Yang, Q. Herding AI cats: Lessons from designing a chatbot by prompting GPT-3. In Proceedings of the 2023 ACM Designing Interactive Systems Conference (pp. 2206-2220). (2023)
5. Madhavi, K. Reddy, Padmavathi Kora, L. Venkateswara Reddy, Janagaraj Avanija, K. L. S. Soujanya, and Prabhakar Telagarapu. "Cardiac arrhythmia detection using dual-tree wavelet transform and convolutional neural network." Soft Computing 26, no. 7 (2022): 3561-3571.
6. Madhavi, K.R., Suri, V.K., Mahalakshmi, V., Reddy, R.O., Reddy, C.S.k. (2023). Federated Madhavi, K. Reddy, S. Viswanadha Raju, and J. Avanija. "Data Labeling and Concept Drift Detection using Rough Entropy For Clustering Categorical Attributes." HELIX 7, no. 5 (2017): 2077-2085.

7. Guo, Q., Wang, R., Guo, J., Li, B., Song, K., Tan, X., ... & Yang, Y. Connecting large language models with evolutionary algorithms yields powerful prompt optimizers. arXiv preprint arXiv:2309.08532. (2023)
8. Gu, Y., Han, X., Liu, Z., & Huang, M. Ppt: Pre-trained prompt tuning for few-shot learning. arXiv preprint arXiv:2109.04332. (2021)
9. Avanija, J., G. Sunitha, and K. Reddy Madhavi. "Semantic Similarity based Web Document Clustering Using Hybrid Swarm Intelligence and FuzzyC-Means." Helix 7, no. 5 (2017): 2007-2012.
10. Arora, S., Narayan, A., Chen, M. F., Orr, L., Guha, N., Bhatia, K., ... & Re, C. Ask me anything: A simple strategy for prompting language models. In The Eleventh International Conference on Learning Representations. (2022)
11. Lu, A., Zhang, H., Zhang, Y., Wang, X., & Yang, D. Bounding the capabilities of large language models in open text generation with prompt constraints. arXiv preprint arXiv:2302.09185. (2023).
12. Ding, N., Chen, Y., Han, X., Xu, G., Xie, P., Zheng, H. T., ... & Kim, H. G. Prompt-learning for fine-grained entity typing. arXiv preprint arXiv:2108.10604. (2021).
13. Reynolds, L., & McDonell, K. Prompt programming for large language models: Beyond the few-shot paradigm. In Extended Abstracts of the 2021 CHI Conference on Human Factors in Computing Systems (pp. 1-7). (2021)
14. Porter, P., Yang, S., & Xi, X. The Design and Implementation of a RESTful IoT Service Using the MERN Stack. In 2019 IEEE 16th International Conference on Mobile Ad Hoc and Sensor Systems Workshops (MASSW) (pp. 140-145). IEEE. (2019)
15. Terlapu, P. V., Yugandhar, M., Ramesh, B., Kumar, B. V., & Pemula, R. (2022, January). Student cognitive learning capability (sclc) prediction system using pca-ann based model. In 2022 International Conference on Computing, Communication and Power Technology (IC3P) (pp. 11-18). IEEE.
16. Vital, T. P., Sangeeta, K., & Kumar, K. K. (2021). Student classification based on cognitive abilities and predicting learning performances using machine learning models. International Journal of Computing and Digital Systems, 10(1), 63-75.