Conversational Agent for Website Engagement using LLaMA 2

Aishwarya S
Department of Information Technology
Sri Venkateswara College of Engineering
Sriperumbudur, India
2020it0113@svce.ac.in

Bollineni Manogna
Department of Information Technology
Sri Venkateswara College of Engineering
Sriperumbudur, India
2020it0563@svce.ac.in

P. Leela Rani
Department of Information Technology
Sri Venkateswara College of Engineering
Sriperumbudur, India
leela@svce.ac.in

***Abstract*— In an era marked by the ubiquitous deluge of information, the project titled "Conversational Agent for Website Engagement using LLaMA 2" emerges as a beacon of innovation, offering a transformative solution to heighten user engagement on websites. The project strategically employs the cutting-edge capabilities of LLaMA 2, an advanced artificial intelligence developed by Meta AI. At its core, this initiative endeavors to craft a conversational agent that transcends the limitations of traditional chatbots. Powered by the nuanced understanding and generation of human-like text, this agent aspires to create an environment conducive to natural and context-aware conversations on websites.**

The overarching objective is to simplify the user experience, allowing individuals to seamlessly connect their queries with the most pertinent and meaningful content. Distinguishing itself from conventional approaches, the conversational agent integrates an inventive technique known as Graph-Based Prioritization. By incorporating this novel approach, the project aims to elevate the conversational interface to new heights of effectiveness and user satisfaction. In the intricate domain of information retrieval, the project strategically leverages the capabilities of PageRank, a venerable algorithm recognized for its efficacy.

Keywords—conversational agent, chat bot, natural language processing, sentiment analysis.

I. INTRODUCTION

Natural Language Processing (NLP) is a vibrant and revolutionary domain within Artificial Intelligence (AI), dedicated to empowering computers to comprehend, interpret and produce human language. The interdisciplinary field merges principles from linguistics, computer science and Machine Learning (ML). The applications of NLP span a broad spectrum, ranging from the development of chatbots for real-time customer assistance to the creation of automated translation services that facilitate communication across diverse languages. NLP acts as the translator, deciphering user messages and generating natural language responses for chatbots. This empowers chatbots to hold meaningful conversations, enhancing user experience and efficiency.

Chatbots serve as valuable tools for lead generation and customer engagement. By proactively engaging visitors, answering their questions and guiding them through the website, chatbots can capture valuable leads and nurture them towards conversion. Additionally, chatbots can gather valuable feedback from users, helping businesses gain insights into customer preferences and areas for improvement. By leveraging chatbot technology, businesses can create more meaningful experiences for their online audience, driving higher engagement, customer satisfaction and ultimately, business success.

Chatbots have become ubiquitous in various real-world scenarios, offering seamless interactions and assistance across diverse industries. From customer service platforms to e-commerce websites, chatbots are deployed to provide immediate support and information to users. In healthcare, they aid in scheduling appointments and answering basic medical queries. Educational institutions utilise chatbots for student support and course enrolment guidance. Their versatility makes them indispensable tools in modern-day interactions between businesses and consumers. Incorporating chatbots into websites guarantees constant availability, rapid responses, personalised interactions and enhanced engagement. As advancements in AI continue, chatbots are expected to play an even greater role in shaping the future of customer service.

II. PROBLEM STATEMENT

In the era of information overload on websites, users often encounter the challenge of navigating through overwhelming content to find relevant information. Outdated and obsolete website-specific information from AI models necessitates the need for a web-based conversational agent that delivers precise and accurate responses to user queries, providing an efficient solution for customer engagement and informed decision-making. This project distinguishes itself from traditional chatbot implementations by fostering natural, context-aware conversations within the website environment. By seamlessly bridging the gap between user queries and website content, the conversational agent aims to enhance the overall user experience.

III. RELATED WORK

Recent research has underscored the significance of conversational agents in augmenting user engagement across diverse online platforms. Chen et al. (2019) delved into the realm of e-commerce by implementing conversational agents adept at providing personalized product recommendations and assistance, leveraging sentiment analysis to tailor responses to user preferences. This approach not only elevated user satisfaction but also bolstered conversion rates, showcasing the potential of conversational agents as effective tools for customer interaction. Concurrently, Smith et al. (2020) navigated the educational landscape, devising a context-aware chatbot that dynamically adapted its responses to cater to the

evolving learning objectives and progress of users. This adaptability not only deepened engagement but also yielded enhanced learning outcomes, indicating the versatility of conversational agents in diverse domains.

In tandem with advancements in conversational agent technology, researchers have explored innovative methodologies to refine content prioritization and information retrieval. Li et al. (2018) introduced a graph-based approach for news article recommendation, leveraging semantic relationships to enhance the relevance and diversity of recommendations. Such techniques hold promise for optimizing conversational agents tasked with navigating vast repositories of information, ensuring that users receive tailored and pertinent responses to their queries. Moreover, the foundational PageRank algorithm, pioneered by Brin and Page (1998) for web page ranking, continues to inform strategies for information retrieval. By incorporating these methodologies, the proposed project aims to harness the power of graph-based prioritization and PageRank algorithms to streamline the conversational agent's ability to sift through the deluge of website content, delivering curated responses that align closely with user intent.

In addition to content prioritization, sentiment analysis emerges as a pivotal component in refining the conversational agent's interactions with users. Liu et al. (2021) demonstrated the efficacy of sentiment-aware chatbots in social media platforms, enabling nuanced responses that resonate with user emotions. Integrating sentiment analysis into the conversational agent equips it with the capability to discern user sentiment and tailor responses accordingly, fostering empathetic and contextually relevant interactions. By amalgamating insights from these diverse research strands, the proposed project aims to sculpt a sophisticated conversational agent poised to navigate the complexities of website engagement, thereby enriching the user experience in an era dominated by information abundance.

IV. PROPOSED SYSTEM

A. *Overview*

In a realm characterised by the overwhelming abundance of information, the initiative titled "Conversational Agent for Website Engagement using LLaMA 2" emerges as a beacon of innovation, offering a transformative solution to enhance user engagement on websites. Strategically harnessing the advanced capabilities of LLaMA 2, this project aims to develop a conversational agent that surpasses the limitations of traditional chatbots. With its foundation rooted in the nuanced comprehension and generation of human-like text, this agent endeavours to cultivate an environment conducive to natural and contextually aware conversations on websites.

The primary goal is to streamline the user experience, enabling individuals to seamlessly connect their inquiries with the most relevant and meaningful content. Setting itself apart from conventional approaches, the conversational agent incorporates an innovative technique called Graph-Based Prioritization. By integrating this unique approach, the project seeks to elevate the conversational interface to new levels of effectiveness and user satisfaction.

B. *Architecture of conversational agent for website engagement using LLaMa2.*

The architecture diagram of the proposed model is depicted below in figure 1.

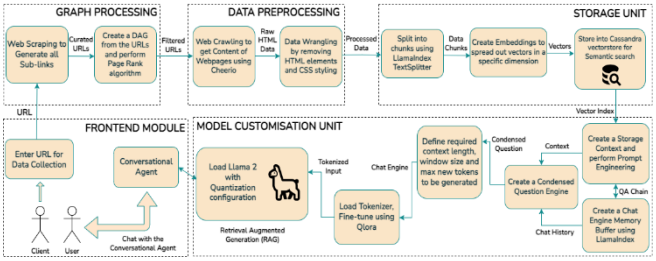

Figure 1 Proposed Architecture Diagram

C. *Model Description*

LLaMA 2, developed by Meta AI and launched in July 2023, represents an advancement over the original LLaMA LLM. Trained on an expanded dataset with 40% more data and featuring double the context length, it stands out for its extensive training on a vast corpus of text and code. Notably, LLaMA 2 boasts an impressive size, housing 70 billion parameters, thus positioning it among the largest publicly available LLMs. Renowned for its minimal violation rates in both single and multi-turn prompts, the model's creativity tends to increase alongside the growth in parameters. However, for tasks like Question-Answering (QA)[31], precision rather than creativity is paramount. Hence, there are limitations on utilising a 7 billion parameter model, prioritising precise responses to prompts. LLaMA 2 demonstrates a remarkable capability to comprehend the intent behind a prompt and generate responses consistent with that intent, even when faced with complex or challenging prompts.

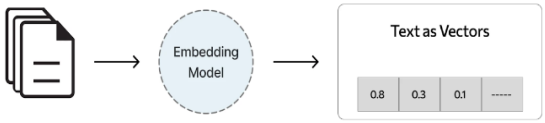

Figure 2 Embedding Model structure

An embedding refers to the conversion of words or phrases into numerical values represented in continuous vector space as shown in Figure 3.2. Such representations reflect the semantic connections among words by positioning related ones nearby. Words with shared meanings tend to cluster closely in this vector space during training using approaches like GloVe[42] and Transformer[43], applied to sizable text collections. Consequently, finding meaningful correlations among words becomes feasible, facilitating improved performance in natural language processing tasks.

D. *Web Scraping*

Web scraping is the automated process of extracting data from websites. It involves using software tools or programming scripts to access and retrieve information from web pages. Web scraping allows users to gather data from various websites in a structured format, which can then be

analysed, processed or stored for various purposes such as market research, price monitoring, content aggregation or data analysis.

E. Web Crawling

Web crawling is the automated process of systematically browsing the internet to index and collect information from web pages. It involves using web crawlers or bots, also known as spiders, to visit web pages, follow links and retrieve content. Web crawling is commonly used by search engines to index web pages for search results. This process is fundamental for maintaining up-to-date search engine databases and for various other applications such as data mining, content aggregation and website monitoring.

F. Data Preparation

Effective training and fine-tuning of LLMs rely heavily on thorough data preparation. The quantity and quality of the data utilised during training significantly influence the performance of the LLM. Several key considerations come into play during data preparation:

- Data Size: LLMs necessitate large volumes of data to achieve optimal training outcomes. More extensive datasets generally lead to better LLM performance.
- Data Diversity: It's crucial to expose LLMs to a diverse array of text data, reflective of the contexts in which they will operate. This ensures the model learns a broad spectrum of language patterns and minimises the risk of biased or repetitive text generation.
- Data Quality: Clean and well-formatted data is essential for effective LLM training. This involves eliminating errors, inconsistencies and irrelevant information from the dataset.

The Alpaca format offers a structured approach to data preparation for LLM fine-tuning. Each data point in this JSON format comprises three key fields:

- instruction: A concise task description guiding the model's action.
- input: Optional contextual information aiding the model's understanding.
- output: A required field providing the correct answer or output corresponding to the instruction.

The instruction field should be clear and informative, furnishing all necessary details for the model to perform the task accurately. While the input field is optional, it can enhance the model's comprehension by providing additional context. The output field is mandatory and must supply the correct response to the instruction. Overall, the Alpaca format offers versatility, making it suitable for fine-tuning LLMs across various tasks.

G. Tokenization of Data

The LLaMA tokenizer[38] is a byte-level Byte-Pair-Encoding (BPE)[37] tokenizer that is used to encode text for the LLaMA LM. It is based on the SentencePiece tokenizer, but it has been modified to better handle the specific needs of the LLaMA 2 model. The LLaMA tokenizer works by splitting text into bytes and then grouping the bytes into pairs. The pairs are then merged into larger and larger units until a vocabulary of a fixed size is created. The vocabulary is then used to encode the text into a sequence of integers. The LLaMA tokenizer has a number of advantages over other tokenizers, including:

- It is able to handle a wide range of languages and character sets.
- It is able to encode rare and infrequent words.
- It is able to encode subword units, which can help to improve the performance of LLMs.

H. Storage Module

The storage module encompasses both the database and the array of procedures employed to efficiently store and manage data within the database. A database[39] refers to an organised compilation of data usually stored and administered electronically through a DataBase Management System (DBMS)[39]. With a DBMS, users can perform various operations such as creating, deleting, etc. within the database. The content extracted using web crawling techniques is stored in the database for training and use by the model.

I. Ranking

In the process of engaging with embeddings, the practice of ranking consists of evaluating the degree of closeness amongst a specified collection of vectors, consequently establishing their relative positions within the embedding space. Diverse measurements, such as cosine similarity, Euclidean distance or Manhattan distance, can serve to quantify proximity, subsequently yielding dissimilar rankings.

The utilisation of cosine similarity to discover the term most comparable to a particular animal name in an embedding space encompassing numerous animal names. Initiate by computing the cosine similarity between the particular name and every remaining vector, followed by arranging the findings in decreasing order, ultimately designating the outcome featuring the peak score signifies the optimal match corresponding to semantic affinity.

J. Reranking

Reranking is the procedure of reshuffling the initial ordering produced after performing a primary ranking operation over a series of objects or elements in response to certain criteria, frequently grounded in enhanced knowledge, refined constraints or user feedback. This technique aims at improving the relevance and utility of the original list, thus offering superior matches tailored to preferences or objectives of the users.

Reranking incorporates additional factors beyond the original ranking scope, enhancing precision and accuracy. Through sophisticated calculations, reranked lists showcase heightened fidelity towards end-user expectations, boosting satisfaction levels.

The essence of reranking lies in iterative optimization cycles, wherein fresh parameters or signals get introduced, guiding successive reassessment phases. Subsequent rounds adjust the existing rankings dynamically, accounting for revised conditions and augmented intelligence, culminating in more comprehensive and nuanced arrangements. Overall, reranking amplifies the impact of decision-making processes in diverse areas, such as information retrieval, recommendations, summarization and conversational AI agents.

K. Low Rank Adaptation

LoRA fine-tuning presents a method for refining LLMs that is more efficient and impactful compared to traditional fine-tuning approaches. Traditional methods entail retraining all parameters of the LLM, which can be computationally burdensome and time-intensive, particularly for exceedingly large LLMs. Conversely, LoRA fine-tuning targets only a small subset of the parameters of the LLM, rendering it notably faster and more resource-efficient. This technique operates by utilising two low-rank matrices to approximate the weight matrix of the LLM. These matrices are then fine-tuned on a task-specific dataset. Subsequently, after fine-tuning the low-rank matrices, they are integrated into the weight matrix of the LLM to tailor the LLM for the new task. LoRA fine-tuning has demonstrated effectiveness across various NLP tasks, encompassing text classification, question answering and summarization. LoRA implements a strategy where it freezes the weights of the pre-trained model and introduces trainable rank decomposition matrices into each layer of the Transformer architecture. Additionally, LoRA adopts float16 precision for its parameters, while employing int8 precision for the parameters of the LLaMA-2 model.

L. Quantized Low Rank Adaptation

QLoRA involves propagating gradients through a frozen, 4-bit quantized pretrained LM into LoRA. After confirming that 4-bit QLoRA matches 16-bit performance consistently across various scales, tasks, and datasets, a comprehensive investigation of instruction fine-tuning is conducted on the largest open-source LMs available for research. The performance of instruction fine-tuning on these models is assessed.

M. Retrieval-Augmented Generation

RAG is a technique that enhances the accuracy and reliability of generative AI models by incorporating facts fetched from external sources. This method is designed to address the limitations of LLMs by integrating neural information retrieval with neural text generation. The retrieval component of RAG acts as a high-powered reading module, rapidly indexing relevant data from vast knowledge stores, while the generation component serves as a creative writing module, synthesising key information from retrieved passages into coherent narratives. The architecture allows RAG systems to overcome the limitations of LLMs, such as Generative Pre-trained Transformer (GPT-4)[55], which cannot explicitly access external datasets within their foundation models. By offloading content ingestion to specialised standalone retrieval modules, generators can focus on mastering linguistic dexterity. The approach significantly expands the reachable information compared to what can be stored internally only, enabling RAG to provide more accurate and nuanced responses. Additionally, RAG mitigates harmful model biases by exposing generators to more diverse perspectives through broader content retrieval and it enables genuine fact-checking against original sources. The method moves closer to how humans leverage knowledge acquisition to unlock reasoning and communication superpowers far beyond our biological programming.

N. Amount of training data

Table 1 shows the impact of large amounts of training data. LLaMA 2 represents an upgraded version of LLaMA 1, trained on a fresh blend of publicly available data. The pre-training corpus was expanded by 40% and the context length of the model was doubled. It employs grouped-query attention. Variants of LLaMA 2 with 7B, 13B and 70B parameters have been released. Additionally, there were 34B variants which will not be released. LLaMA 2-Chat, an optimised version of LLaMA 2 for dialogue scenarios, has been developed. Variants with 7B, 13B and 70B parameters have been released. The initial version of LLaMA 2-Chat underwent supervised fine-tuning, with training conducted on 2 trillion tokens of data, offering a favourable performance-cost trade-off. LLaMA 2 exhibits a strong command of English, as evidenced in Table 2.1, thanks to its training on an extensive dataset containing a significant amount of English text and code.

Language	Percentage	Language	Percentage	Language	Percentage
en	89.70%	ru	0.13%	uk	0.07%
unknown	8.38%	nl	0.12%	ko	0.06%
de	0.17%	it	0.11%	ca	0.04%
fr	0.16%	ja	0.10%	sr	0.04%
sv	0.15%	pl	0.09%	id	0.03%
zh	0.13%	pt	0.09%	cs	0.03%
es	0.13%	vi	0.08%	fi	0.03%
da	0.02%	ro	0.03%	hu	0.03%
sl	0.01%	bg	0.02%	no	0.03%

Table 1. Language Distribution in Pre-training Data LLaMA 2

O. Markdown And Streaming Response

Markdown in LLMs is to structure and format text, enhancing the readability and presentation of generated content. This use of Markdown allows for the creation of documents with headings, lists, links, and other formatting elements that can be easily interpreted and displayed in various platforms. The integration of Markdown with LLMs, such as GPT-4 and Llama 2, is particularly beneficial for tasks requiring the generation of structured content. By incorporating Markdown syntax into the text generation process, LLMs can produce outputs that are not only coherent and contextually relevant but also well-organised and visually appealing. This integration facilitates the creation of documents, guides, and other forms of content that are easier to navigate and understand, thereby improving the overall

user experience. Moreover, the use of Markdown in conjunction with LLMs enables the generation of content that can be seamlessly integrated into existing documentation systems, websites, and other digital platforms. This integration simplifies the process of content creation and editing, making it more accessible to a wider range of users.

Streaming response[58] in LLMs is implemented to enhance user experience by delivering content in real-time, reducing the wait time for users. This approach is particularly beneficial for applications that generate text, where waiting for the entire generation process to complete can lead to user drop-off. The streaming response mechanism allows for the immediate display of results to users, making the application feel more responsive and interactive. There are multiple methods to implement streaming responses, including polling, Server-Sent Events (SSE)[58] and WebSockets. Polling is considered the simplest approach for applications that do not require real-time bi-directional communication, such as text or code generation applications. SSE is another method that involves sending server-sent events from the server to the client, allowing for the streaming of generated tokens to the UI. WebSockets, on the other hand, allow for a persistent bi-directional communication channel between the client and server, making them suitable for applications that require real-time transfer of packets from both ends. However, for applications using LLMs that primarily generate text or code, WebSockets might be considered overkill due to the lack of need for real-time bi-directional communication. The implementation of streaming responses in LLM applications involves receiving the streaming response from LLMs, delivering the streaming response to the client-side, and receiving the streaming response on the client side. The choice of method depends on the specific requirements of the application, such as the need for real-time updates and the complexity trade-off between real-time functionality and implementation complexity.

P. Model Customization Module

1. Zero-shot Prompting

Zero-shot prompting refers to a technique in NLP where a LM generates responses to prompts without being explicitly trained on those prompts. In other words, the model is asked to generate responses to tasks or questions it has never seen before, without any fine-tuning or specific training on those tasks. Instead, zero-shot prompting relies on the understanding of language by the model and its ability to generalise from its training data to infer appropriate responses to novel prompts. Here are some of the potential benefits of zero-shot prompting:

- Versatility: Zero-shot prompting allows a model to handle a wide range of tasks without the need for task-specific training data or fine-tuning.
- Scalability: Zero-shot prompting models can be applied to a diverse array of tasks without the need to retrain or fine-tune the model.
- Generalisation: By leveraging its understanding of language and patterns learned during training, a zero-shot prompting model can generalise to unseen tasks or prompts, making it adaptable to new challenges or scenarios.

2. Fine-Tuning LLMs

Fine-tuning LLMs involves adjusting their parameters to optimise performance for specific tasks. This is achieved by training the model on a relevant dataset tailored to the task at hand. The extent of fine-tuning necessary varies based on task complexity and dataset size. LLMs often yield diverse outputs for the same instruction, necessitating fine-tuning to refine their understanding of desired output formats and input analysis. Due to their extensive knowledge base, LLMs typically require minimal data for fine-tuning. This contrasts with traditional LLM training methods, which involve training on large text and code datasets. Fine-tuning offers a cost-effective and time-efficient alternative, as it enables training on smaller datasets. Supervised learning entails training the model to predict correct outputs for each input example in the dataset.

3. Fine-Tuning

Full fine-tuning is a highly effective technique involving training the entire LLM on a task-relevant dataset[50]. Despite being computationally demanding, it often results in superior performance. When computational resources are limited, repurposing may be considered as an alternative. However, for optimal results, full fine-tuning is recommended. In this process, all parameters of a pre-trained model are updated for the specific task, in contrast to partial fine-tuning where only select parameters are adjusted. This method involves training the model with task-specific data, refining its hyperparameters, and evaluating its performance on test data. In LLaMA 2, core model layers[51] such as the decoder, attention and MLP layers are typically fine-tuned during the full fine-tuning process.

4. Database Management

The Database Management Module leverages a vector database, it is a specific database category designed for storing and organising vectors, which represent data objects mathematically and are also referred to as vector embeddings. These databases are finely tuned to streamline the storage, retrieval, and querying processes of vectors, focusing on their similarity or distance from other vectors. In contrast to conventional database querying methods that rely on exact matches or predetermined criteria, vector databases excel in identifying the most similar or pertinent data based on semantic or contextual significance.

V. CONCLUSION

Sentiment analysis is a powerful tool that can be used to understand and respond to public opinion. As machine learning and artificial intelligence continue to develop, sentiment analysis is likely to be used in even more ways in the real world. ABSA is a type of sentiment analysis that goes beyond simply identifying the overall sentiment of a piece of text. Instead, ABSA identifies the specific aspects of a product, service, or experience that people are talking about, and the sentiment they express towards those aspects. ABSA has a wide range of real-world applications. ABSA is

a powerful tool that can be used to improve the customer experience and increase sales in E-commerce.

LLaMA 2 is a state-of-the-art natural language processing library that can be used to perform aspect-based sentiment analysis with high accuracy and efficiency. With its ability to provide detailed and actionable insights, ABSA using LLaMA 2 can be a valuable addition to any business's analytics arsenal. The LLaMA 2 model has been shown to be very accurate for ABSA. LLaMA 2 model achieved an F1 score of 90.2% on the SemEval-2014 Task 4 ABSA dataset which is significantly higher than the accuracy of other existing ABSA models, such as BERT-PT (75.08%) and LCF-GloVe (80.58%). The LLaMA 2 model's high accuracy in ABSA is due to a number of factors. First, the LLaMA 2 model is trained on a massive dataset of text and code, which gives it a deep understanding of the nuances of human language. Second, the LLaMA 2 model uses a number of advanced techniques, such as self-attention and transformer architecture, which allow it to learn complex relationships between words and phrases. The LLaMA 2 model's high accuracy in ABSA makes it a valuable tool for a variety of applications, such as product review analysis, customer service analysis, and social media monitoring.

VI. FUTURE SCOPE

In envisioning the future trajectory of the project, a key area of expansion lies in dense analysis and understanding of context insights. Current ABSA models are typically developed for English and a few other major languages. In the future, ABSA models can be used for developing more languages and domains. It will make ABSA more accessible to businesses and organizations around the world.

Sarcasm refers to the expression of sentiments in a way that contrasts with the literal meaning of the words used. Sarcasm often involves saying one thing but intending the opposite, and it can be challenging for sentiment analysis models to accurately interpret the underlying sentiment. The use of irony, exaggeration, or mocking language in sarcastic expressions makes it difficult for models like LLaMA-2 to precisely identify and classify sentiments associated with specific aspects of a given text. Handling sarcasm in ABSA requires sophisticated natural language processing techniques and specialized training data that encompass sarcastic expressions to improve the model's ability to accurately analyze sentiments in varying linguistic contexts.

REFERENCES

[1] Keivalya Pandya, Prof. Dr. Mehfuza Holia (2023), Automating Customer Service using LangChain - Building custom open-source GPT Chatbot for organizations, *3rd International Conference on "Women in Science & Technology: Creating Sustainable Career"*

[2] Chia-Ying Li, Yu-Hui Fang, Yu-Hung Chiang, Can AI chatbots help retain customers? An integrative perspective using affordance theory and service-domain logic, *Technological Forecasting and Social Change*, vol 197, 2023,122921

[3] Zhixiang Zeng, Yuefeng Li, Jianming Yong, Xiaohui Tao, Vicky Liu, Multi-aspect attentive text representations for simple question answering over knowledge base, Natural Language Processing Journal, Volume 5, 2023, 100035, ISSN 2949-7191

[4] Khurana, D., Koli, A., Khatter, K. et al. Natural language processing: state of the art, current trends and challenges. Multimed Tools Appl 82, 3713–3744 (2023).

[5] Shabbir, J., & Anwer, T. (2018). Artificial Intelligence and its Role in Near Future. ArXiv, abs/1804.01396

[6] Shawar, B. A., & Atwell, E. S. (2005). Using corpora in machine-learning chatbot systems. International Journal of Corpus Linguistics, 10(4), 489–516. doi:10.1075/ijcl.10.4.06sha

[7] Khder, M.A. (2021). Web Scraping or Web Crawling: State of Art, Techniques, Approaches and Application. International Journal of Advances in Soft Computing and its Applications.

[8] Xing, W., & Ghorbani, A. (2004). Weighted PageRank algorithm. Proceedings. Second Annual Conference on Communication Networks and Services Research, 2004.

[9] Bateman, J. A., Hois, J., Ross, R., & Tenbrink, T. (2010). A linguistic ontology of space for natural language processing. Artificial Intelligence, 174(14), 1027–1071.

[10] Muhammad, A. F., Susanto, D., Alimudin, A., Adila, F., Assidiqi, M. H., & Nabhan, S. (2020). Developing English Conversation Chatbot Using Dialogflow. 2020 International Electronics Symposium (IES).

[11] Li, G., Zhang, Y., Jiang, X., Bai, S., Peng, G., Wu, K., & Jiang, Q. (2013). Evaluation of the ArcCHECK QA system for IMRT and VMAT verification. Physica Medica, 29(3), 295–303.

[12] Sakr, S., Liu, A., & Fayoumi, A. G. (2013). The family of mapreduce and large-scale data processing systems. ACM Computing Surveys, 46(1), 1–44.

[13] "ReactJS: A Modern Web Development Framework" Prateek Rawat, Archana N. Mahajan (2020), International Journal of Innovative Science and Research Technology ISSN No:-2456-2165, Volume 5, Issue 11.

[14] T. Lin and I. Joe (2023), 'An Adaptive Masked Attention Mechanism to Act on the Local Text in a Global Context for Aspect-Based Sentiment Analysis' – IEEE Access(2023), Vol. 11, pp. 43055-43066.

[15] Khan, F. A., Jamjoom, M., Ahmad, A., & Asif, M. (2019). An analytic study of architecture, security, privacy, query processing, and performance evaluation of database-as-a-service. Transactions on Emerging Telecommunications Technologies.

[16] Chandrasekaran, D., & Mago, V. (2021). Evolution of Semantic Similarity—A Survey. ACM Computing Surveys, 54(2), 1–37.

[17] B. N. D. Santos, R. M. Marcacini, S. O. Rezende (2021), 'Multi-Domain aspect extraction using bidirectional encoder representations from transformers' – IEEE Access Vol. 9, pp. 91604–91613.

[18] Pereira Detro, S., Santos, E. A. P., Panetto, H., Loures, E. D., Lezoche, M., & Cabral Moro Barra, C. (2019). Applying process mining and semantic reasoning for process model customisation in healthcare. Enterprise Information Systems, 1–27.

[19] Kirk, J. R. (2022), 'Improving language model prompting in support of semi-autonomous task learning' – arXiv.org, Computer Science, Machine Learning

[20] W. Kuang (2023), 'FederatedScope-LLM: A comprehensive package for fine-tuning large language models in federated learning' – arXiv.org, Computer Science, Artificial Intelligence.

[21] M. H. Zahweh (2023), 'Empirical Study of PEFT techniques for Winter Wheat Segmentation' – arXiv.org, Computer Science, Computer Vision and Pattern Recognition

[22] Hu, E. J. (2021), 'LORA: Low-Rank adaptation of Large Language Models' – arXiv.org, Computer Science, Computation and Language

[23] Dettmers, T., Pagnoni, A., Holtzman, A., & Zettlemoyer, L. (2023), 'QLORA: Efficient Finetuning of Quantized LLMS' – arXiv.org, Computer Science, Machine Learning

[24] Zeng, A. (2023), 'AgentTuning: Enabling Generalized agent abilities for LLMs' – arXiv.org, Computer Science, Computation and Language.

[25] Loh, E., Khandelwal, J., Regan, B., & Little, D.A. (2022). Promotheus: An End-to-End Machine Learning Framework for Optimizing Markdown in Online Fashion E-commerce. Proceedings of the 28th ACM SIGKDD Conference on Knowledge Discovery and Data Mining.