ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24MAY2413

AI Companion: Revolutionizing Sales and Services During Product Advisor and Consumer Interaction

Ong Tzi Min[1]; Lim Tong Ming[2*]
Faculty of Computing and Information Technology
Tunku Abdul Rahman University of Management and Technology
Kuala Lumpur, Malaysia

Corresponding Author:- Lim Tong Ming[2*]

Abstract:- In today's fast-paced consumer electronics industry, staying ahead of the competition and satisfying customers are top priorities. This research investigates the use of AI-powered tools, particularly conversational AI and chatbots, to improve customer interaction and boost sales in electronic retail. As digital platforms become more dominant over traditional sales channels, these AI tools offer significant benefits by delivering personalized, efficient, and timely customer service. The analysis examines various AI technologies, including Large Language Models (LLMs) and retrieval-augmented generation, which enhance consumer interaction. The study also explores the practical implications and challenges of implementing these technologies, with a focus on how they can streamline operations, improve customer experiences, and drive sales. Different models like DialoGPT, Flan-T5, and Mistral 7B are evaluated for their effectiveness in real-time consumer interactions, highlighting the importance of continuous adaptation and learning within AI systems to meet consumer demands and keep up with technological advancements.

Keywords:- *Chatbot; LLM; Mistral-7B; Flan-T5; DialoGPT; Lang Chain; Transformers.*

I. INTRODUCTION

➢ *Background and Needs of the Industry*

The consumer electronic industry is a highly competitive and rapidly evolving business that demands constant innovation and adaptation to fulfill customers' ever-changing needs and preferences worldwide. The sector includes a wide range of devices such as smartphones, computers, televisions, and other personal electronics that are integral to daily life, driving continuous demand for newer, more advanced products. Consumer electronics chain faces significant challenges in managing diverse product portfolios, optimizing supply chain management, and improving operation efficiency while deep understanding on customer preferences and behaviors. In this fast-paced digital era, consumer interaction in electronic retail has shifted significantly from in-person engagements and over-the-phone support to a predominant reliance on e-commerce platforms and digital communication avenues. The emergence of e-commerce and online retail platforms aggravates the challenges of traditional brick-and-mortar stores in providing in-store shopping services that cannot be replicated online. Today's consumers are more informed and have higher expectations for product information and customer service. They demand convenience, speed, and efficiency in every interaction, whether it's online or in-store. Implementing AI-driven tools like frontliner assistant chatbots can address these needs by providing quick, efficient, and personalized customer service. Such technology not only meets the demands for rapid and customized interaction but also helps consumer electronics chain maintains competitiveness in a market where tailored customer experiences and operational agility are key to influencing purchasing decisions and achieving business success.

➢ *Problems Faced by the Industry*

The consumer electronics industry faces numerous challenges that necessitate the adoption of salesperson chatbot assistants. This industry is fiercely competitive and rapidly evolving, demanding constant innovation and adaptation to satisfy global customers' ever-shifting demands and preferences. Frequent product launches with new features and functionalities constantly introduced contribute to evolving customer expectations. In addition, a vast array of products, ranging from smartphones to home appliances, flood the market, leading to the challenge of managing diverse product portfolios.

Moreover, anticipating customer preferences and behaviors poses a considerable challenge for retailers as they struggle to cater to consumers' needs effectively. Globalization has expanded the market reach of consumer electronics companies, exposing them to diverse customer preferences and cultural nuances. Changing lifestyles, economic conditions, and societal trends influence consumers' behavior. Hence, the companies should have remained agile and adaptable by utilizing AI technologies and tailoring their products to suit varying needs and preferences across different regions.

The inadequacy of salespersons in effectively serving consumers could be a significant hurdle for the consumer electronics industry. Salespersons may struggle to stay abreast of the wide range of products available. For instance, they lack comprehensive product knowledge, such as the latest features, specifications, and updates for each product. Also,

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24MAY2413

lacking adequate training or experience in customer service skills such as problem-solving, active listening, and empathy could lead to difficulties in addressing customer inquiries, handling complaints, or offering personalized recommendations. This could diminish the overall shopping experience and potentially deter repeat business. Thus, adopting an AI-powered chatbot is necessary to improve customer engagement and maintain competitiveness in this rapidly evolving marketplace.

- *In Summary, the Key Challenges Faced by the Consumer Electronics Industry are Outlined Below:*

- Fierce competitive and rapid evolution
- Frequent product launches
- Diverse Product Portfolios
- Customer preferences and behaviors
- Global Market Reach
- Inadequacy of salespersons.

- *Objectives*

This project aims to boost customer engagement by utilizing AI-powered tools for personalized and efficient service. Also, it seeks to enhance the sales staff's performance by equipping them with AI assistants, which improves their product knowledge and develops their customer service skills. A crucial goal of this initiative is to refine our management of a diverse product portfolio, which requires strategic planning to handle our extensive array of products effectively, ranging from smartphones to home appliances.

From a technical standpoint, the project seeks to leverage cutting-edge machine learning algorithms and AI techniques to stay competitive in a rapidly evolving market. This includes adopting sophisticated data analytics methods to gain deeper insights into consumer behavior, thereby enabling us to anticipate and adapt to changing consumer preferences with greater accuracy.

II. LITERATURE REVIEW

Section A provides a comprehensive review of each case study where AI has been implemented in retail, ranging from AI chatbots in e-commerce to AI-driven predictive modelling in fashion sales forecasting. It discusses the outcomes and areas for improvement. Section B discusses the deployment of advanced LLMs and other AI models to enhance customer service interactions, focusing on the technical aspects and the challenges encountered. In Section C, a summary of various studies on the use of LLMs in diverse contexts is presented, outlining the main problems addressed, techniques used, and achievements noted.

- *AI-Powered Sales Support for Retail Industry*

Study by [1] explored AI's role in boosting customer loyalty in 910 companies worldwide, examining AI-powered customer service, predictive modeling, personalization, and NLP. It evaluated these features using six ML algorithms: Logistic Regression, KNN, SVM, Decision Tree, Random Forest, and AdaBoost, with AdaBoost and Logistic Regression showing the highest accuracies of 0.639 and 0.631, respectively. The research emphasized the need for further investigation into model interpretability and addressing long-term business challenges.

In [2], the study highlighted the implementation of SamBot, an AI conversational bot, on the Samsung IoT Showcase website. SamBot enhanced user engagement and satisfaction by using an extensive knowledge base to respond to queries, integrate AI-driven conversational capabilities, and recommend questions. Although user engagement increased, the article noted the necessity for deep learning models to automate knowledge creation and called for further research to optimize the performance of conversational bots in corporate settings.

[3] described a Virtual Customer Service prototype using the A.L.I.C.E chatbot in a batik-themed e-commerce store in Malang. The AIML-powered chatbot aimed to improve user satisfaction by accurately and promptly responding to inquiries, dynamically expanding its knowledge base from website data when needed. With an 87% accuracy rate in relevant responses, the chatbot provided efficient customer service, though improvements in knowledge acquisition and adaptability to customer needs were suggested.

[4] addressed the growing trend of AI-Powered Automated Retail Stores (AIPARS), integrating AI, robotics, and advanced software systems to revolutionize the retail experience. It explored how AI-driven technologies like chatbots, augmented reality, and machine learning reshaped both physical and online retail landscapes, with a focus on enhancing customer engagement, personalization, and operational efficiency. Achievements included significant efficiency improvements, such as a 50% increase in assortment efficiency, 20% stock reduction, and a 30% rise in online sales. However, an outstanding issue was the need to address consumer adoption and ease the transition to automated shopping for widespread success.

[5] discussed AI applications in aiding customer buying processes and providing after-sales support. The implementation of AI-powered image search tools and recommendation systems were significant achievements. However, the integration of AI disrupted traditional retail models, leading to challenges in employee adaptation and customer acceptance of less personalized services.

[6] addressed AI's role in enhancing customer service and marketing strategies in retail. The primary technique discussed was the use of AI for personalized marketing campaigns and predictive analytics to forecast consumer behavior. Achievements included improved customer engagement and sales conversion rates. However, ethical issues regarding data privacy remained a significant concern.

[7] evaluated how LLMs could enhance online sales processes. The model used was a proprietary LLM by OpenAI, which demonstrated the ability to perform complex tasks such as passing simulated exams. The primary

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24MAY2413

advantage of this LLM was its versatility in handling diverse sales tasks, leading to increased sales efficiency. However, its reliance on extensive data raised concerns about privacy and the potential for biased outputs. The study showed potential in revolutionizing sales strategies but warned against over-dependence on AI systems without adequate human oversight.

- *LLM Models and Techniques used in Generative AI Chatbot for Sales Activity*

[8] addressed inefficient manual interactions in consumer-facing platforms by implementing AI-driven chatbots using advanced Large Language Models (LLMs) like Mistral 7B and Palm 2, utilizing deep learning and NLP techniques to enhance user interaction. Challenges included ensuring response relevance and accuracy and handling out-of-scope queries. The implementation improved user experience and operational efficiency, but integration complexity and domain-specific data training remained issues.

In [9] study, the BERT and GoBot models were used to enhance intent identification in networking chatbots. The goal was to create a chatbot that could understand user queries and respond appropriately, using NLU and natural language generation (NLG) to generate human-like interactions. The study showed improved accuracy, but scaling the model for larger datasets remained a challenge. More research was needed to optimize performance and integrate intent classification for broader use in customer service.

According to [10], SAS-BERT, which is a BERT-based architecture designed specifically for sales and support conversations, achieved performance comparable to fine-tuned LLMs in shorter training time and with fewer parameters. While promising for highly domain-specific tasks, it relied on internal datasets and faced challenges in email exchange coherence. However, SAS-BERT showed potential for enhancements and broader applications in chat and voice data. Pros included improved performance and cost reduction, while cons involved dataset limitations and coherence issues. The outstanding issues included the need for broader dataset availability and improvements in coherence modeling for email exchanges.

[11] explored sales forecasting in fashion, proposing a novel approach that combined customer feedback with sales data to improve accuracy. Using machine learning models such as Linear Regression, Decision Tree, and Random Forest, along with sentiment analysis via BERT, the research achieved notable enhancements in predictive performance. Integrating customer feedback enriched the models' comprehension of market dynamics, highlighting the potential of merging machine learning with human insight for better predictions. While advantages included enhanced accuracy and market understanding, challenges might arise from data availability and computational complexity. Future research avenues included expanding to other product types, exploring additional sentiment analysis techniques, and addressing daily sales pattern variations for more precise forecasts.

[12] discussed aligning large language models (LLMs) like GPT-4 with human values using the On-the-fly Preference Optimization (OPO) method. This method utilized an external memory to store and update human values, allowing the model to align dynamically without retraining. The main achievement of OPO was its ability to adjust model responses efficiently to contemporary norms, as demonstrated in rigorous evaluations. However, the method was computationally intensive and assumed the external memory always captured appropriate values accurately, which might lead to potential misalignments. The challenge of ensuring computational efficiency and the reliability of stored values in external memory remained an outstanding issue.

[13] explored improving e-commerce chatbots using the Falcon-7B model, a Large Language Model trained on 1,500 billion tokens and characterized by its 16-bit full quantization. This approach significantly boosted the chatbot's natural language understanding and response capabilities. While the model enhanced computational efficiency and performance, it faced challenges in maintaining high accuracy due to the precision reduction from quantization. This balance between efficiency and precision remained a critical area for further development in enhancing e-commerce chatbot interactions.

[14] used only web data, refined through extensive filtering and deduplication, to match the quality of curated corpora. This approach used the Falcon LLM model and the RefinedWeb dataset, which consisted of five trillion tokens. The study demonstrated that this method could achieve zero-shot performance comparable to models trained on high-quality datasets. However, it faced challenges in balancing the computational demands of processing large-scale web data with maintaining high data quality, which was essential for effective LLM training.

The paper by [15] explored using the BLOOM model enhanced by Low-Rank Adaptation for generative AI in smart cities, targeting SMEs with limited resources. This method reduced computational costs and training time, supporting multilingual interactions effectively. While it significantly lowered resource demands and supported diverse languages, maintaining response accuracy and computational efficiency remained a challenge, especially for SMEs operating under resource constraints.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24MAY2413

➢ *Analysis of Past LLM Model Research*

Table 1 LLM Model Analysis

Study	Problems	Technique Used	Achievement
[8]	Inefficient manual interactions on consumer-facing platforms	AI-driven chatbots using LLMs (Mistral 7B, Palm 2), deep learning, NLP	Improved user experience and operational efficiency, integration complexity addressed
[9]	Need for enhanced intent identification in networking chatbots	BERT, GoBot models, NLU, NLG	Improved accuracy in user query understanding and response, need for scaling the model
[10]	Domain-specific challenges in sales and support conversations	SAS-BERT, a BERT-based architecture	Comparable performance to fine-tuned LLMs, cost-effective, faces email coherence issues
[11]	Inaccurate sales forecasting in fashion	Linear Regression, Decision Tree, Random Forest, sentiment analysis via BERT	Enhanced predictive performance, improved market understanding, need for more diverse data
[12]	Misalignment of LLMs with human values	On-the-fly Preference Optimization (OPO) with external memory	Efficient adjustment of model responses to contemporary norms, computational intensity remains a challenge
[13]	Balancing efficiency and precision in e-commerce chatbots	Falcon-7B model, 16-bit full quantization	Improved natural language understanding and computational efficiency, challenges in maintaining high accuracy
[14]	Quality and computational efficiency in training LLMs	Falcon LLM model, RefinedWeb dataset	Zero-shot performance comparable to high-quality datasets, challenges in data quality maintenance
[15]	Resource constraints in implementing generative AI for smart cities SMEs	BLOOM model with Low-Rank Adaptation	Reduced computational costs and training time, supports multilingual interactions, accuracy and efficiency issues

Table I. provides a comprehensive overview of various studies that integrating large language models (LLMs) and AI techniques to mitigate distinct challenges across different domains. It discusses the adoption of AI-powered chatbots, such as Mistral 7B and Palm 2, to enhance user engagement and streamline operations, as well as the use of specific models like SAS-BERT to bolster sales support. According to Table I., each case study showcases benefits like increased user involvement and more accurate predictive analytics, but also points out issues related to scaling the models and ensuring data quality. This review emphasizes the necessity for ongoing advancements in AI to effectively handle the intricacies associated with the retail and customer service sectors today.

III. PROJECT METHODOLOGY

The section begins by discussing the project workflow, which encompasses the stages from business understanding through to data modeling and evaluation, and ultimately, deployment. Subsequently, the second section provides an overview of the hardware and software utilized, including detailed descriptions of the libraries and frameworks employed in the project.

A. Flow of the Project Works

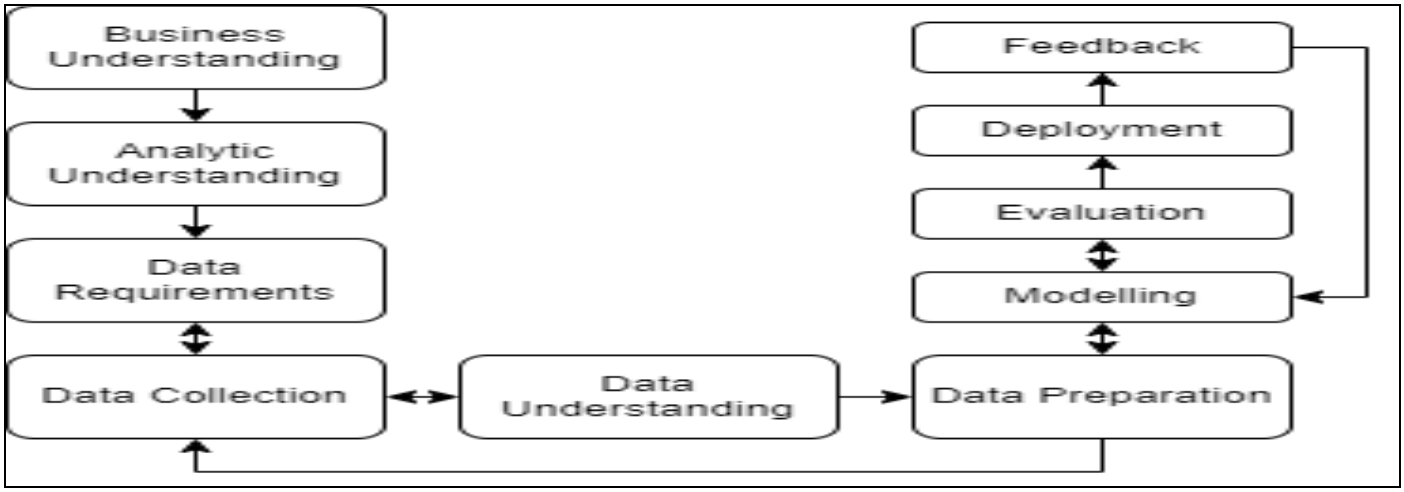

Fig 1 Project Workflow

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24MAY2413

Fig. 1 depicts the workflow of this project. In the business understanding phase, understanding specific needs and goals of a certain domain such as a service domain for which the AI assistant is being developed is the initial steps to be carried out. The process involves determining the objectives of the business, the role of the AI assistant, and the ways in tackling specific industry problems. Following the business understanding, analytic understanding phase focuses on comprehending the analytics needed to achieve the defined business goals. In the case of a conversational AI within the specific sector, this entails the scope of questions the AI should answer, the interactions nature, and the intended results of these interactions.

The data requirements phase will identify the data requirements from analytic insights. The data type needed to train the model will be defined in this process. For instance, if the business focuses on the service domain, the data needed includes customer service transcript, common queries, customer feedback and domain-specific knowledge based. With clear data requirements, the subsequent steps involve the collection of data required. For example, the dataset gathered should contain text conversations, customer feedback, service manuals, and other relevant information regarding the service domain.

Next, this phase involves a comprehensive examination of the collected data to comprehend its structure, quality, and content. It encompasses exploring the data to unveil initial insights and identify potential needs for data cleaning or preprocessing. After data understanding, the following step is to prepare it for modeling. Data preparation may encompass activities such as cleaning, normalization, feature extraction, and other transformations essential to render the data suitable for training AI models. Data formatting should be implemented to ensure the data is suitable for model training.

In the modeling stage, AI models are constructed and trained using the prepared data. For a conversational AI assistant, this might involve training a natural language processing model to comprehend and generate text resembling human language. Encoding and decoding process may be applied to the model too. When model training is completed, an evaluation will be performed ensuring that the result meets the business requirements mentioned in the initial phase. For development of AI assistant, the evaluation could include assessing the accuracy of chatbot response, its capability to handle diverse service-related queries, and testing it within a controlled environment.

Following successful evaluation and refinement, the model is deployed in a real-world environment where it engages with end-users. Deployment strategies may vary depending on whether the AI assistant is utilized on a website, in a mobile app, or through other service channels. After deployment, user feedback is gathered to evaluate the AI assistant's performance. This includes qualitative feedback from users and quantitative data based on the assistant's responses and interactions. The feedback loop prompts a revisit to the modeling. This iterative process ensures continuous improvement of the AI assistant, maintaining alignment with business goals, user requirements, and industry advancements.

B. Hardware and Software Used

The hardware involves the NVIDIA RTX 3090 graphics cards where it provides the necessary computational power to handle intensive data processing and model training tasks. This GPU accelerates machine learning workflows and handles the demands of large neural networks.

On the software side, the application was primarily developed using Jupyter Notebook, which is a platform that enables real-time code execution, documentation, and visualization. These features are crucial for iterative development and testing in machine learning projects. The development process was aided by various important libraries and frameworks which can be showed in Table 2.

Table 2 Libraries and Frameworks Used

Libraries	**Description**
Streamlit	Facilitates interactive web application development with a user-friendly interface.
Request and BeautifulSoup	Web Scrapping, with Requests handling HTTP requests and BeautifulSoup parsing HTML/XML.
Pandas	Streamlines data manipulation and analysis with powerful data structures.
Torch	A key tool for building and training neural networks efficiently.
Langchain	Enhances AI conversational abilities by chaining language models.
Transformers	Provides a suite of pre-trained models from Hugging Face for advanced NLP tasks.
FAISS	Optimizes the similarity search and clustering of dense vectors for fast data retrieval.
Google Translate API	Facilitates real-time translation between English and Malay, improving accessibility.

Additionally, Docker played a critical role in the project by enabling the creation, deployment, and operation of applications within containers. This approach guaranteed that software and configurations were consistent across various development and production environments, preventing problems related to dependencies and conflicts.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24MAY2413

IV. DATA AND PRELIMINARY WORKS

This section provides a detailed exploration of the AI models used in the project, covering data preparation, model architecture, training, and validation processes. It discusses the implementation of DialoGPT and Flan-T5, highlighting their configurations, experimental setups, and performance outcomes in enhancing conversational AI applications.

A. Data Source, Nature of Data, and Data Sizes

The generative AI models utilize datasets sourced from HuggingFace, a renowned platform for machine learning models and datasets. Specifically, the dataset in focus is "knkarthick/dialogsum", a rich collection of dialogues that reflect various aspects of daily life [16]. These dialogues, which simulate face-to-face interactions, span an array of topics such as education, employment, healthcare, retail, recreation, and travel. The conversations typically occur between individuals in familiar settings, like friends and colleagues, or in more formal interactions, such as those between customers and service providers. This diverse range of dialogues provides a comprehensive foundation for training conversational AI, enabling the models to handle a wide spectrum of topics and social contexts.

The "knkarthick/dialogsum" dataset consists of 14,460 dialogues with corresponding manually labeled summaries and topics. In Fig. 2, the dataset dictionary involves 12,460 training datasets, 500 validation datasets, and 1500 testing datasets.

```
DatasetDict({
    train: Dataset({
        features: ['id', 'dialogue', 'summary', 'topic'],
        num_rows: 12460
    })
    validation: Dataset({
        features: ['id', 'dialogue', 'summary', 'topic'],
        num_rows: 500
    })
    test: Dataset({
        features: ['id', 'dialogue', 'summary', 'topic'],
        num_rows: 1500
    })
})
```

Fig 2 Details of "Knkarthick/Dialogsum" Dataset

The "knkarthick/dialogsum" dataset consists of four columns: id, dialogue, summary, and topic. A detailed explanation of each column is listed in the table 3.

Table 3 Description of Each Column in the Dataset

Column	Description
Id	Unique Identifier
Dialogue	Conversation between two people
Summary	A summary of conversation between two people
Topic	The topic of the conversation

B. Data Preparation, Preprocessing, and Data Sampling

In this project, the AI models used are DialoGPT and Flan-T5. These two models require different data formatting before model training so that it is well suited to the model. The data preprocessing for DialoGPT involves splitting the dialogues in the dialogue column into individual lines after loading the "knkarthick/dialogsum" dataset. A context window of 7 is created, where each response is considered with the previous 7 interactions. This allows the model to understand the conversation flow so that it can generate relevant responses. Then, the dataframe created will undergo data splitting where the "test_size" parameter is set to 0.1, indicating 10% of data will be allocated for validation set and the remaining 90% will be used for training.

Move on to Flan-T5 model, the dialogues in the dataset are also split into individual lines. A data frame is created by extracting the questions and answers from these dialogues and each line of dialogue beginning with "#Person1#" will be considered as a question while the line beginning with "#Person2" will be treated as the answer. The dataframe is then split into training and testing datasets where the "test_size" parameter is set to 0.2, indicating 20% of data for validation and 80% of data for training purposes. These datasets are converted into Dataset objects, as outlined in Fig. 3, which is a format suitable for training machine learning models. Moreover, each question is processed by adding a prefix, "Please answer this question:", to set the context for the model. The tokenization is also applied to both question and answer columns for converting the text into a numerical format that the model can understand.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24MAY2413

```
DatasetDict({
    train: Dataset({
        features: ['question', 'answer', '__index_level_0__'],
        num_rows: 45191
    })
    test: Dataset({
        features: ['question', 'answer', '__index_level_0__'],
        num_rows: 11298
    })
})
```

Fig 3 Dataset Dictonary Created for Flan-T5 Model

C. Model Considered on the Experimentation Conducted

➢ *DialoGPT and Flan-T5 Architecture and Works Conducted*

In this project, the AI models used are DialoGPT and Flan-T5. The DialoGPT model is a powerful tool for generating realistic text using a transformer framework with layers ranging from 12 to 48. It is designed to work with extensive web-text data and can generate text from scratch or based on user prompts. When working with multi-turn dialogue sessions, the model treats the session as a single continuous text sequence, using a sequence of conditional probabilities to generate each dialogue turn. By concatenating each turn into a long text sequence and computing the probability of a target response given the dialogue history for each token, the model optimizes dialogue generation and maintains context throughout the session [17]. This approach is implemented using the PyTorch-transformer library and is highly efficient and effective.

The Flan-T5 architecture is a modified version of the T5 model that includes the concept of "instruction tuning" in the pre-training phase. This approach involves training the model with a diverse set of natural language tasks, which helps the model develop a general understanding of various tasks before fine-tuning on specific downstream tasks. The Flan-T5 architecture benefits from T5's encoder-decoder framework, as outlined in Fig. 4, which processes input sequences through an encoder to create embeddings that are decoded into output sequences. By integrating instructions directly into the training process, Flan-T5 becomes proficient at following complex instructions in different applications, making it more versatile and effective across a broader range of language understanding and generation tasks [18].

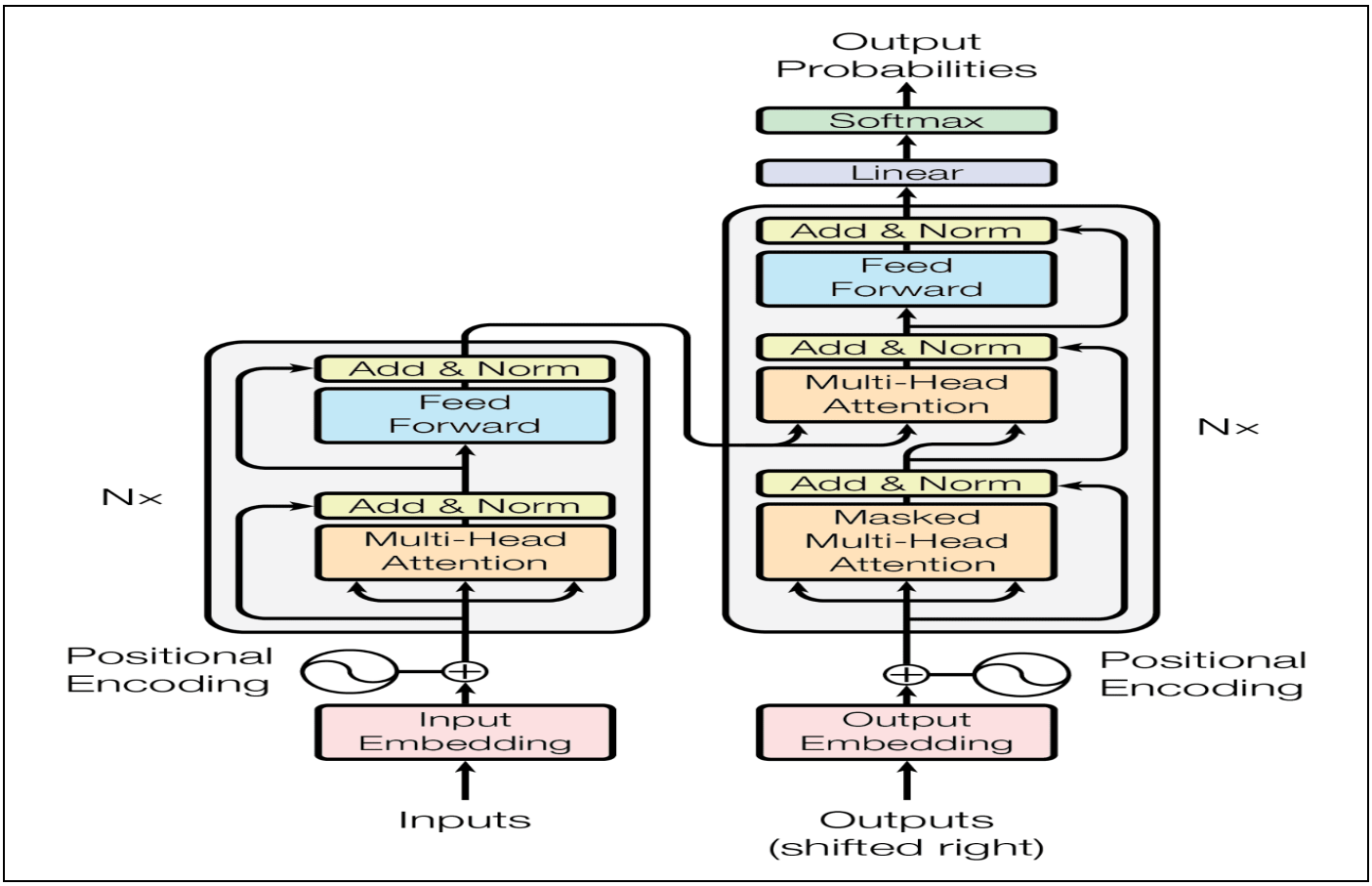

Fig 4 Flan-T5 Achitecture [18]

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24MAY2413

➢ *DialoGPT and Flan-T5 Implementation*

There are various pre-trained DialoGPT models which are "DialoGPT-large", "DialoGPT-medium", and "DialoGPT-small". "DialoGPT-large" is not able to run due to computing limit while "DialoGPT-small" results in poor performance on question-answering tasks. Hence, the model chosen is "microsoft/DialoGPT-medium". Several key parameters shown in Fig. 5 are crucial for configuring the DialoGPT model. Firstly, the "model_name_or_path" is set to "microsoft/DialoGPT-medium" to determine the base model architecture. The "block_size" is typically set to 512 tokens for balancing between context length and computational efficiency. The "learning_rate" of 5e-5(0.00005) ensures model optimization, dictating the rate at which the model learns. The num_train_epochs of value 4 indicates that the training dataset is iterated over 4 times. More epochs can lead to better learning but also increase the risk of overfitting.

```
# Args to allow for easy convertion of python script to notebook
class Args():
    def __init__(self):
        self.output_dir = 'DialogsumGPT-Medium'
        self.model_type = 'gpt2'
        self.model_name_or_path = 'microsoft/DialoGPT-medium'
        self.config_name = 'microsoft/DialoGPT-medium'
        self.tokenizer_name = 'microsoft/DialoGPT-medium'
        self.cache_dir = 'cached'
        self.block_size = 512
        self.do_train = True
        self.do_eval = True
        self.evaluate_during_training = False
        self.per_gpu_train_batch_size = 4
        self.per_gpu_eval_batch_size = 4
        self.gradient_accumulation_steps = 1
        self.learning_rate = 5e-5
        self.weight_decay = 0.0
        self.adam_epsilon = 1e-8
        self.max_grad_norm = 1.0
        self.num_train_epochs = 4
        self.max_steps = -1
        self.warmup_steps = 0
        self.logging_steps = 1000
        self.save_steps = 3500
        self.save_total_limit = None
        self.eval_all_checkpoints = False
        self.no_cuda = False
        self.overwrite_output_dir = True
        self.overwrite_cache = True
        self.should_continue = False
        self.seed = 42
        self.local_rank = -1
        self.fp16 = False
        self.fp16_opt_level = 'O1'

args = Args()
```

Fig 5 Parameters setup for DialoGPT Model

In Fig. 6, the model's state is being saved at regular intervals to preserve the training progress and allow for the model to be resumed or evaluated from these checkpoints. Also, the 106460 of global steps indicates that the model has completed 106,460 optimization steps and it is used to keep track of training progress. The average loss over the batch of training is approximately 0.5693. This value indicates how well the model is learning. A lower loss signifies better learning and model performance.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24MAY2413

```
rate computed by the scheduler, "
Saving model checkpoint to DialogsumGPT-Medium/checkpoint-105000
Saving optimizer and scheduler states to DialogsumGPT-Medium/checkpoint-105000
orch/optim/lr_scheduler.py:261: UserWarning: To get the last learning rate compu

rate computed by the scheduler, "
 global_step = 106460, average loss = 0.569269755702146
Saving model checkpoint to DialoGPTMedium/DialoGPT_model
```

Fig 6 Log Entries for Training Process

The log entries for the training process indicate that there are 11,830 examples in the training dataset and each training iteration involves processing a batch of 4 examples at a time. The perplexity score of the model shown in the final log entry is used to measure how well a language model predicts a given sequence of words. A lower perplexity value usually indicates better performance. In this project, a perplexity of 1.1861 suggests that, on average, the language model is making predictions with relatively low uncertainty. After model training, the fine-tuned model will be used for chat interactions. The model will first receive input from the user, encode the user input, generate a response based on user input, and decode it to human-readable text.

Move on to the Flan-T5 model, due to computational resources issues, the “flan-t5-large”, “flan-t5-xl” and “flan-t5-xxl” models are not applicable in this project. Henceforth, the “google/flan-t5-base” model is chosen as it can achieve high performance even with limited training data. In Fig. 7, several key parameters are specified to set up training arguments. From the figure below, the learning rate of 3e-4 strikes a balance between efficiency and accuracy as although it may lead to slow convergence, it can provide more accurate results. With a batch size of 8, it determines how many samples the model processes before updating its internal parameters. A larger batch size can lead to faster training and smoother convergence but requires more memory. The 3 number of epochs indicates that it balances between sufficient exposure to the training data and the risk of overfitting if too many epochs are used.

```python
# Global Parameters
L_RATE = 3e-4
BATCH_SIZE = 8
PER_DEVICE_EVAL_BATCH = 4
WEIGHT_DECAY = 0.01
SAVE_TOTAL_LIM = 3
NUM_EPOCHS = 3

# Set up training arguments
training_args = Seq2SeqTrainingArguments(
    output_dir="./flant5-base",
    evaluation_strategy="epoch",
    learning_rate=L_RATE,
    per_device_train_batch_size=BATCH_SIZE,
    per_device_eval_batch_size=PER_DEVICE_EVAL_BATCH,
    weight_decay=WEIGHT_DECAY,
    save_total_limit=SAVE_TOTAL_LIM,
    num_train_epochs=NUM_EPOCHS,
    predict_with_generate=True,
    push_to_hub=False
)
```

Fig 7 Parameters Setup for Flan-T5 Model

The result of model training is shown in the Fig. 8. The training loss decreases from epoch 1 to epoch 3, indicating that the model is improving on the training data. The validation loss increases slightly from epoch 1 to epoch 3. This could suggest overfitting, especially if the increase continues in subsequent epochs.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24MAY2413

```
trainer.train()
```

[16947/16947 1:02:06, Epoch 3/3]

Epoch	Training Loss	Validation Loss	Rouge1	Rouge2	Rougel	Rougelsum
1	2.867400	2.721883	0.075950	0.012525	0.073050	0.075006
2	2.635000	2.648996	0.075996	0.012553	0.073064	0.075040
3	2.424800	2.633693	0.071536	0.011102	0.069828	0.069021

Fig 8 Training Result

V. MODEL CONSIDERED AND ADOPTED FOR THE OBJECTIVES OF THIS PROJECT

We have identified several limitations in fine-tuning large language models like Flan-T5 and DialoGPT in chatbot development. Sometimes, a chatbot fine-tuned using DialoGPT may provide inaccurate or illogical responses because it relies on patterns learned from data rather than a true understanding of language or context. This is due to a lack of genuine comprehension. On the other hand, Flan-T5 is a versatile model capable of handling various tasks but may not be suitable for highly specialized domains or tasks, such as question-answering in the consumer electronics domain. Additionally, the model's performance is heavily influenced by the quality and diversity of the training data. If the training data is biased or lacks variety, the model may not perform well in practical situations or could generate biased results. Henceforth, we discovered another method which utilizes the Retrieval Arguments Generation (RAG) technique for chatbot response rather than fine-tuning the models.

- *Retrieval Arguments Generation (RAG) and Mistral Architecture*

RAG combines retrieval-based and generative approaches that can generate accurate and detailed responses by dynamically retrieving the most relevant documents during a conversation. This approach is beneficial in domains like technical support, where providing correct and detailed information is crucial. While fine-tined models rely on static pre-training data, RAG inherently supports continuous learning, retrieving the most relevant and recent documents and increasing its knowledge over time without the need for retraining.

Regarding model selection, Mistral 7B, a 7.3B parameter model introduced in September 2023, has surpassed DialoGPT and Flan-T5 in chatbot development. Mistral 7B excels in efficiency, outperforming competitors on various benchmarks, especially in handling code and English language tasks. It has advanced features such as Grouped-query Attention and Sliding Window Attention, which improve processing speed and resource management, making it ideal for real-time applications like chatbots [19]. In contrast, Flan-T5 and DialoGPT may lack these specific optimizations, which are crucial for interactive applications demanding prompt responsiveness.

- *Integration of Mistral Architecture and RAG*

To incorporate the Mistral 7B model with RAG, we have initiated the chatbot development from scratch, beginning with data collection and progressing to final deployment. Initially, we compiled a list of smartphones from a retail website, then proceeded to scrape comprehensive details from a review website, ensuring the data was sufficient and accurate. We extracted URLs from review pages, extracted relevant smartphone details, and then saved this information into a CSV file. Next, we ensured that all desired smartphone names were captured by cross-referencing the predefined smartphone list. For a deeper analysis, we further extracted detailed review content from each product's pages and organized information into a structured JSONL file, categorized into six sections: Specifications, Design and Build Quality, Lab Tests, Software and Performance, Camera and Video Quality, and Pros and Cons, as illustrated in Fig. 9. This setup enables the chatbot to deliver detailed and tailored responses to a wide range of user queries. This efficient approach involves time delays between requests to prevent server overload and maintains a clean and organized dataset for subsequent analysis.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24MAY2413

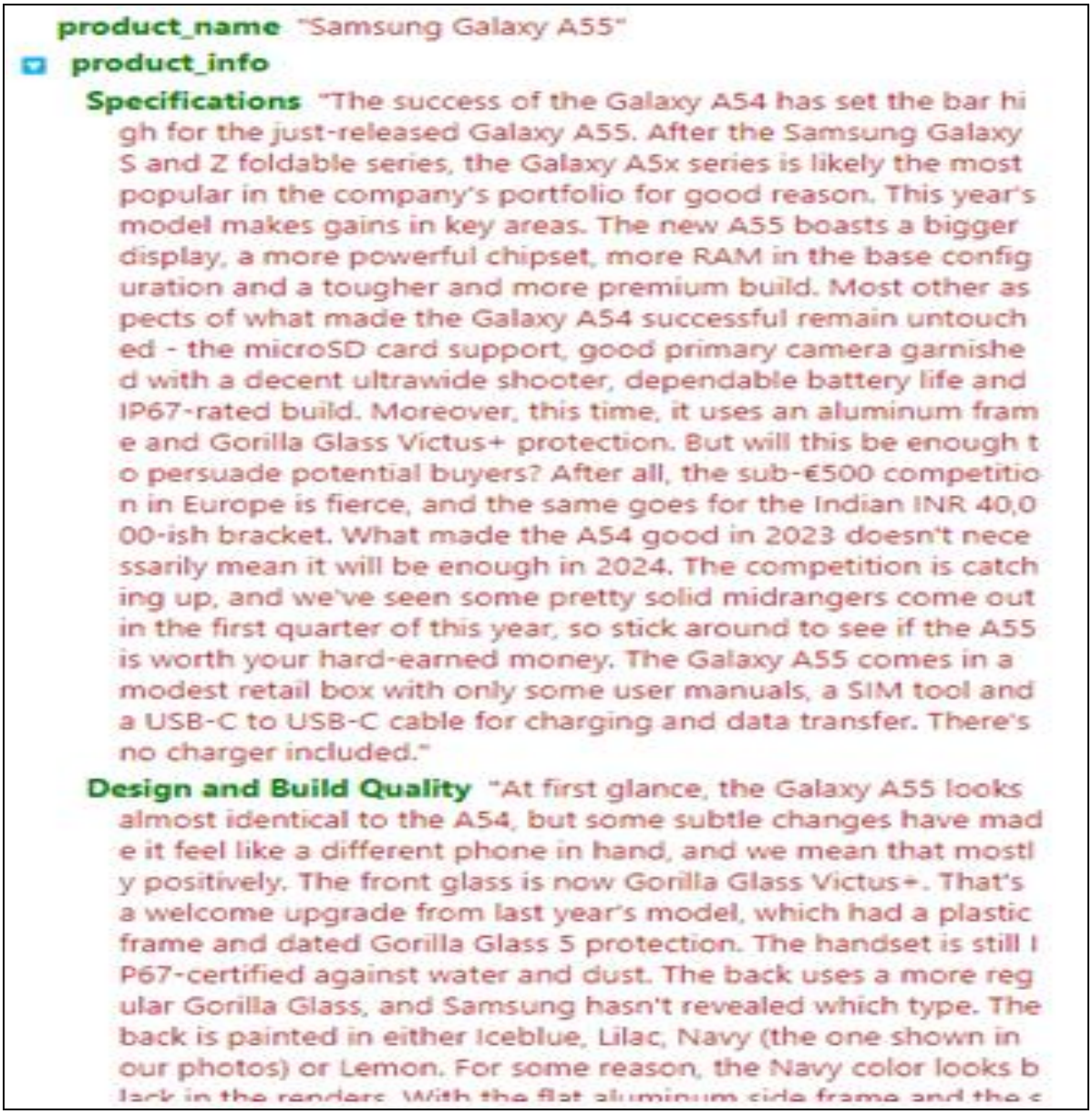

Fig 9 Sample Screenshot of Collected Product Details

The collected dataset consists of three columns: ‘id,’ ‘product_name,’ and ‘product_info.’ The Table IV. provides a detailed explanation of each column.

Table 4 Description of each Column in the Dataset

Column	Description
Id	Unique Identifier
product_name	The name of smartphone
product_info	Detailed description of each smartphone where it includes characteristic, performance measure and expert evaluation.

The model development phase uses Mistral-7B, a transformer-based model integrated with the Langchain library for enhanced conversational capabilities. The model configuration is shown in Fig. 10, and the maximum token limit is set to 8000 for producing extensive and detailed content. The GGUF extension path, which aims at enhancing the model’s performance with additional advanced features, has surpassed the GGML extension path. The 2048 context(n_ctx) indicates that the model can consider up to 2048 tokens from the input context when generating responses. A lower temperature of 0.6 increases predictability of model output. The callback manager supports token-wise streaming where once the model generates a token, they are processed and displayed, enabling real-time interaction and responsiveness. The verbose parameter provides detailed logging during operation for debugging and understanding of model behavior.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24MAY2413

```
llm = LlamaCpp(
    max_new_tokens=8000,
    model_path="mistral-7b-instruct-v0.2.Q5_K_M.gguf",
    n_ctx = 2048,
    temperature = 0.6,
    callback_manager=callback_manager,
    verbose=True,  # Verbose is required to pass to the callback manager
)
```

Fig 10 Mistral-7B Model Parameter Setup

- The model architecture shown in Fig. 11 consists of three main stages: indexing, retrieval, and generation.

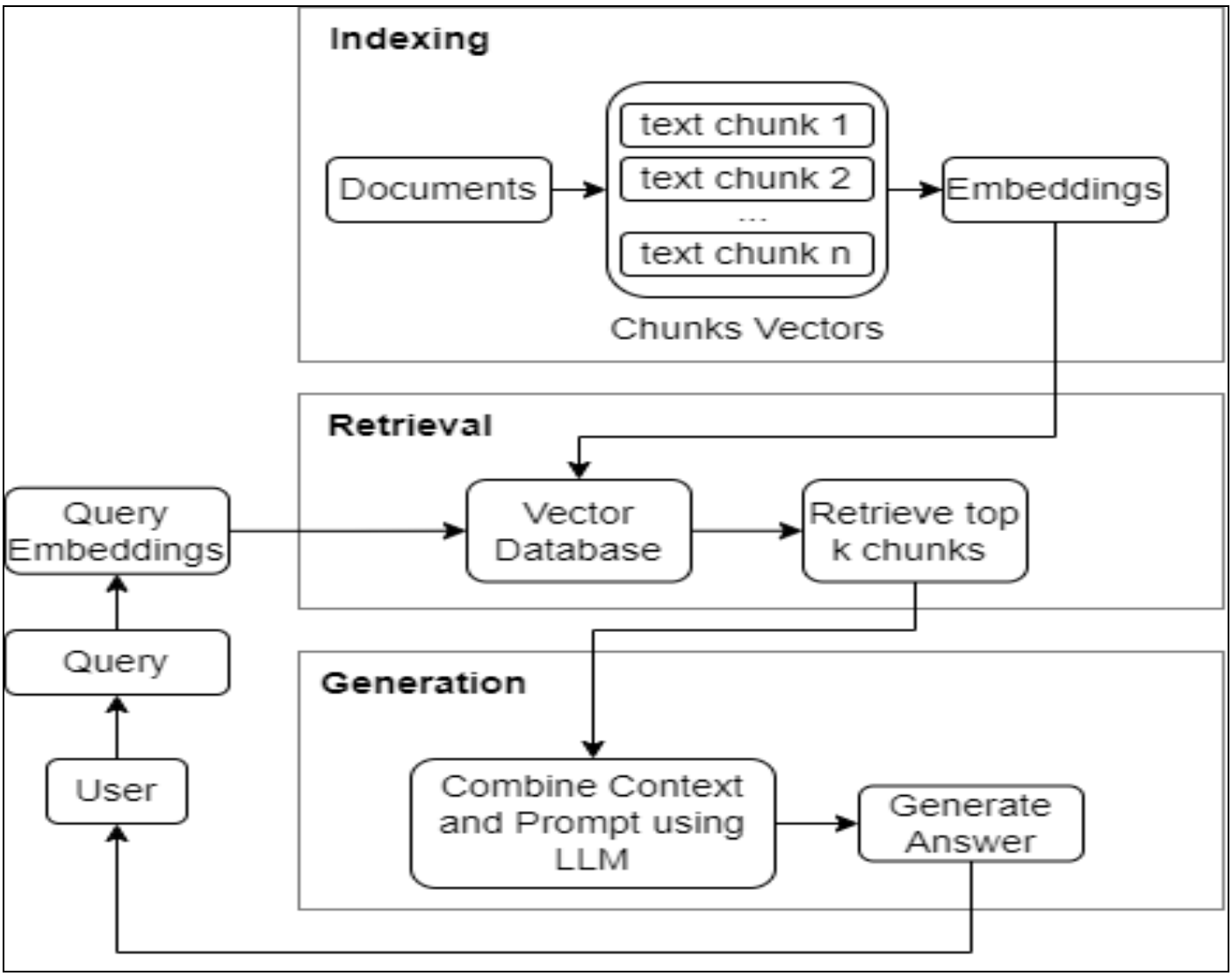

Fig 11 Model Architecture

During the initial phase of indexing, JSONL formatted review data is loaded and fragmented into smaller chunks. These chunks undergo vector encoding to generate embeddings, representing the semantic meaning of the text. FAISS (Facebook AI Similarity Search) is then employed to create efficient vector stores for quick retrieval of relevant information, which is critical for answering user queries effectively. In the subsequent retrieval phase, the model retrieves the top k chunks that are most relevant to the user's question based on their semantic similarity. Fig. 12's "search_kwargs" parameter dictates the retrieval of the two closest segments based on cosine similarity to the input query vector.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24MAY2413

```
mistral_chain = ConversationalRetrievalChain.from_llm(
    llm=st.session_state['mistral'],
    chain_type='stuff',
    retriever=st.session_state['vector_store'].as_retriever(search_kwargs={"k": 2}),
    memory=memory
)
```

Fig 12 Top 2 Chunks Retrieval

Finally, the Mistral-7B model integrated with Langchain's 'LLMChain' for generating responses, which involves crafting prompts dynamically based on the query and context. This includes a two-step language model chaining approach: initial retrieval and summarization of relevant information followed by generating a detailed response. This ensures that the chatbot can provide precise and contextually relevant answers. This three-stage process ensures that the model is able to retrieve and generate accurate and relevant responses to user queries.

Furthermore, frequently updating the model's knowledge base with the most recent information available online is crucial in ensuring the chatbot responses are up to date. Utilizing a web scraping module employing BeautifulSoup and Requests facilitates fetching and parsing content from specific URLs, enabling the AI to not only generate responses based on historical data but also continuously update its knowledge base, thereby adapting to new information and queries. Additionally, user feedback is collected and stored, allowing for iterative improvements to the chatbot based on user interactions. Lastly, the integration of the 'googletrans' library facilitates the translation of responses into Malay, as depicted in Fig. 13 and 14.

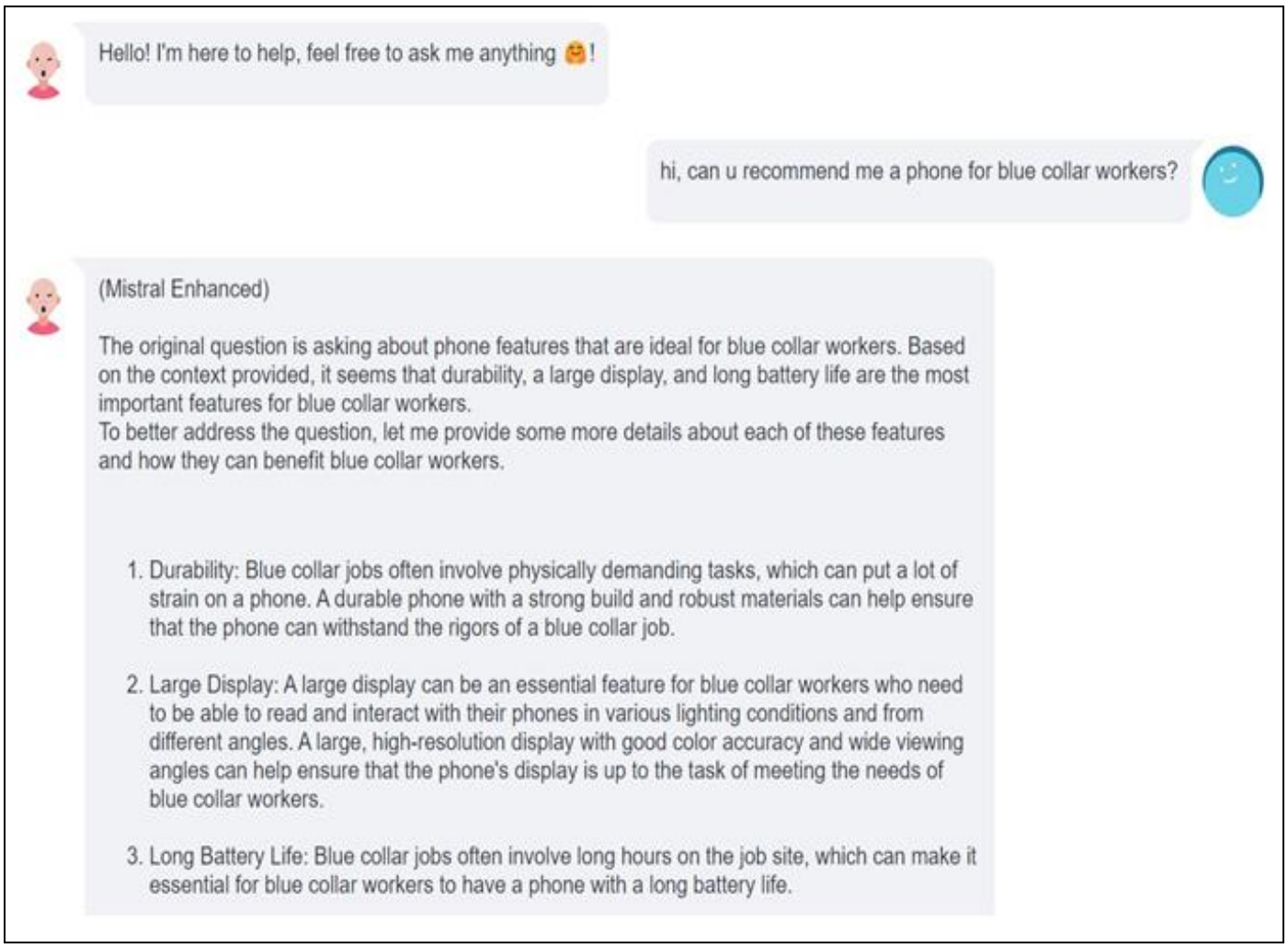

Fig 13 Sample Chat Log History

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24MAY2413

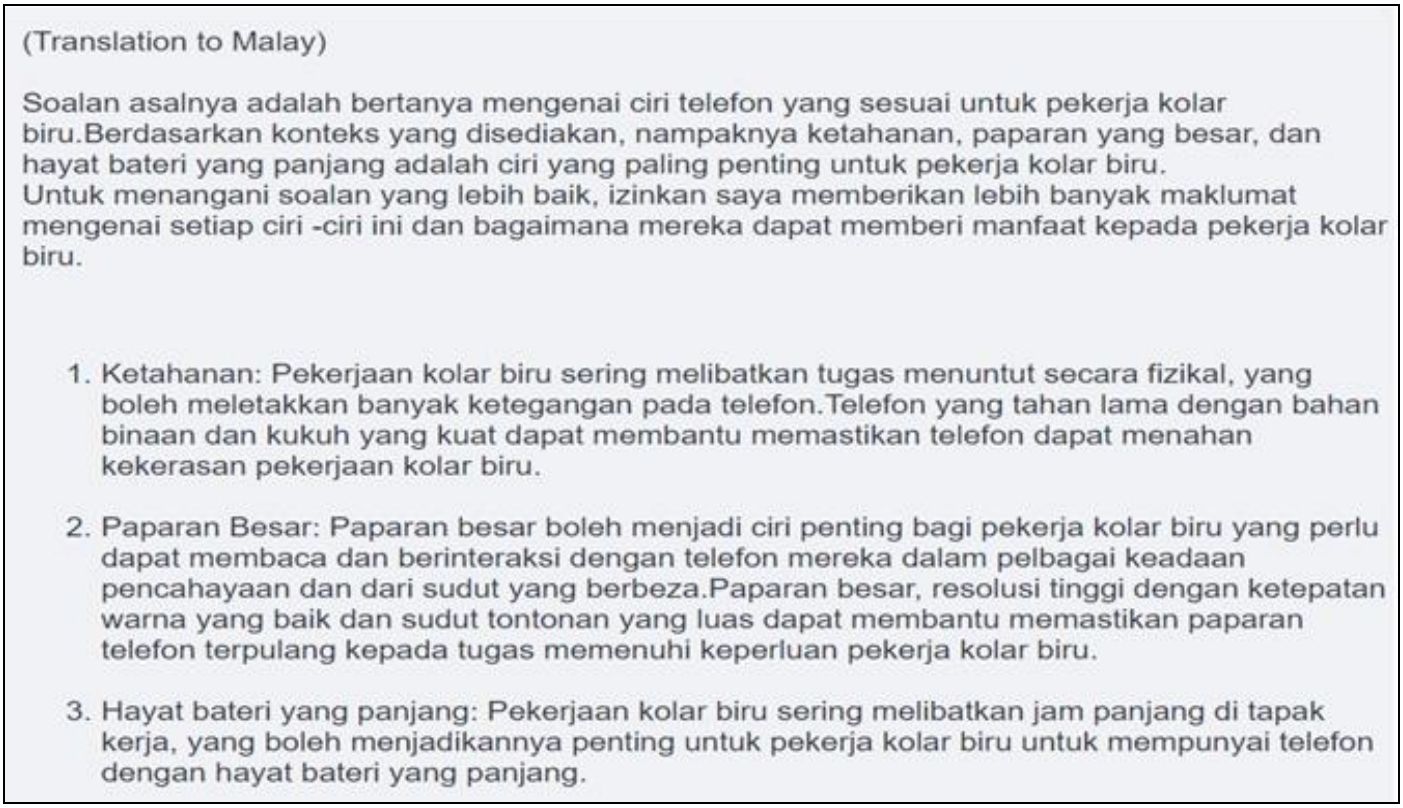

Fig 14 Malay Translation

VI. CONSLUSION AND FUTURE WORKS

In conclusion, the integration of Mistral-7B with the RAG technique presents a promising avenue for the development of advanced chatbots. This powerful combination allows for the retrieval of relevant information from a dynamic knowledge base, leading to the generation of accurate and contextually relevant responses. This is a significant advancement over traditional models like Flan-T5 and DialoGPT, which often struggle with understanding and context. However, the project is currently limited by a lack of processing power, which could potentially impact the chatbot's scalability and efficiency. The high computational demand of the Mistral-7B model may also restrict its deployment in resource-constrained environments, affecting real-time performance and accessibility.

Looking ahead, it is of utmost importance to tackle these computational limitations. We must actively explore optimized deployment strategies to alleviate resource constraints and boost performance, such as model quantization, distributed computing, or the utilization of cloud-based solutions. Furthermore, we should prioritize efforts to continually update the model's knowledge base through web scraping and user feedback collection. This ensures that the chatbot remains current and evolves over time, enhancing its effectiveness and user experience.

ACKNOWLEDGMENT

We would like to extend our sincere appreciation to all the individuals who have contributed to this research. We would also like to express our gratitude to the Faculty of Computing and Information Technology at Tunku Abdul Rahman University of Management and Technology for their support and provision of resources essential for this study. We recognize the crucial role played by the developers of the AI models and frameworks, particularly those at OpenAI, Google, and Hugging Face, whose groundbreaking work has significantly propelled our research. Lastly, we would like to acknowledge the invaluable feedback from our peers and colleagues, which has greatly enhanced the quality of this work.

REFERENCES

[1]. N. Patel and S. Trivedi, "Leveraging Predictive Modeling, Machine Learning Personalization, NLP Customer Support, and AI Chatbots to Increase Customer Loyalty | Empirical Quests for Management Essences," researchberg.com, Aug. 2022, Available: https://researchberg.com/index.php/eqme/article/view/46.

[2]. A. Pradana, O. Goh, Sing, and Y. Kumar, "SamBot - Intelligent Conversational Bot for Interactive Marketing with Consumer-centric Approach," International Journal of Computer Information Systems and Industrial Management Applications, vol. 6, pp. 265–275, 2014, Available: https://mirlabs.org/ijcisim/regular_papers_2017/IJCISIM_61.pdf.

ISSN No:-2456-2165 https://doi.org/10.38124/ijisrt/IJISRT24MAY2413

[3]. Y. Afandi, Maskur, and T. R. Arjo, "Use of Chatbot on Online Store Website as Virtual Customer Service to Improve Sales," Proceedings of 2nd Annual Management, Business and Economic Conference (AMBEC 2020), 2021, doi: https://doi.org/10.2991/aebmr.k.210717.012.

[4]. R. Pillai, B. Sivathanu, and Y. K. Dwivedi, "Shopping intention at AI-powered automated retail stores (AIPARS)," Journal of Retailing and Consumer Services, vol. 57, no. 1, p. 102207, Nov. 2020, doi: https://doi.org/10.1016/j.jretconser.2020.102207.

[5]. L. Cao, "Artificial intelligence in retail: applications and value creation logics," International Journal of Retail & Distribution Management, vol. 49, no. 7, Mar. 2021, doi: https://doi.org/10.1108/ijrdm-09-2020-0350.

[6]. V. Kumar, A. R. Ashraf, and W. Nadeem, "AI-powered marketing: What, where, and how?," International journal of information management, pp. 102783–102783, Apr. 2024, doi: https://doi.org/10.1016/j.ijinfomgt.2024.102783.

[7]. C. Frey and M. Osborne, "Generative AI and the Future of Work: A Reappraisal." Available: https://ora.ox.ac.uk/objects/uuid:f52030f5-23eb-4481-a7f1-8006685edbae/files/stb09j741f.

[8]. R. Bhattacharyya, S. Chandra, K. Manikonda, B. Depuru, and B. Kumar, "An AI-Driven Interactive Chatbot: A Well-Trained Chatbot that Communicates with the Users and Reduces the Manual Interaction," International Journal of Innovative Science and Research Technology, vol. 9, no. 2, 2024. [Online]. Available: https://ijisrt.com/assets/upload/files/IJISRT24FEB203.pdf. [Accessed Apr. 28, 2024].

[9]. Dr Geetha N, Mr Vivek G, and V. T. A, "Intent Classification using BERT for Chatbot application pertaining to Customer Oriented Services," Jan. 2021, doi: https://doi.org/10.4108/eai.7-12-2021.2314563.

[10]. A. Varma and C. Bhat, "Sas-bert: Bert for Sales and Support Conversation Classification Using a Novel Multi-objective Pre-training Framework," Ssrn.com, Oct. 05, 2023. https://papers.ssrn.com/sol3/papers.cfm?abstract_id=4750949. [Accessed Apr. 28, 2024].

[11]. V. T. Luong, N. T. Nguyen, and O. T. Tran, "Improving Sales Forecasting Models by Integrating Customers' Feedbacks: A Case Study of Fashion Products," www.atlantis-press.com, Feb. 05, 2024. https://www.atlantis-press.com/proceedings/icech-23/125997561

[12]. C. Xu et al., "Align on the Fly: Adapting Chatbot Behavior to Established Norms," arXiv.org, 2023. https://arxiv.org/abs/2312.15907. [Accessed Apr. 28, 2024].

[13]. Y. Luo, Z. Wei, G. Xu, Z. Li, Y. Xie, and Y. Yin, "Enhancing E-commerce Chatbots with Falcon-7B and 16-bit Full Quantization," Journal of Theory and Practice of Engineering Science, vol. 4, no. 02, pp. 52–57, Feb. 2024, doi: https://doi.org/10.53469/jtpes.2024.04(02).08.

[14]. G. Penedo et al., "The RefinedWeb Dataset for Falcon LLM: Outperforming Curated Corpora with Web Data, and Web Data Only The Falcon LLM team." [Online]. Available: https://arxiv.org/pdf/2306.01116. [Accessed Apr. 28, 2024].

[15]. N. T. Tuan, P. Moore, D. H. V. Thanh, and H. V. Pham, "A Generative Artificial Intelligence Using Multilingual Large Language Models for ChatGPT Applications," Applied Sciences, vol. 14, no. 7, p. 3036, Jan. 2024, doi: https://doi.org/10.3390/app14073036.

[16]. "knkarthick/dialogsum · Datasets at Hugging Face," huggingface.co, Mar. 05, 2024. https://huggingface.co/datasets/knkarthick/dialogsum [Accessed Nov. 28, 2023].

[17]. Y. Zhang et al., "DialoGPT: Large-Scale Generative Pre-training for Conversational Response Generation," arXiv:1911.00536 [cs], Nov. 2019, Available: https://arxiv.org/abs/1911.00536

[18]. A. Vaswani et al., "Attention Is All You Need," arXiv.org, Jun. 12, 2017. https://arxiv.org/abs/1706.03762.

[19]. A. Q. Jiang et al., "Mistral 7B," arXiv.org, Oct. 10, 2023. https://arxiv.org/abs/2310.06825.