Journal of Computer and Communications, 2025, 13(5), 84-100
https://www.scirp.org/journal/jcc
ISSN Online: 2327-5227
ISSN Print: 2327-5219

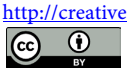
Prototyping Large Language Model from Scratch as 1st Line Customer Engagement & Support Tool

Inn Keat Ng, Oscar Yung Qin Koh, Jia Lin Tan, Tong Ming Lim*

Faculty of Computing and Information Technology, Tunku Abdul Rahman University of Management and Technology, Kuala Lumpur, Malaysia
Email: inkeat0331@gmail.com, jialiitan1024@gmail.com, oscarkoh070809@gmail.com, *limtm@tarc.edu.my

How to cite this paper: Ng, I.K., Koh, O.Y.Q., Tan, J.L. and Lim, T.M. (2025) Prototyping Large Language Model from Scratch as 1st Line Customer Engagement & Support Tool. *Journal of Computer and Communications*, **13**, 84-100.
https://doi.org/10.4236/jcc.2025.135006

Received: February 13, 2025
Accepted: May 24, 2025
Published: May 27, 2025



Abstract

Small and Medium Enterprises (SMEs) in Malaysia face challenges in managing customer engagement due to resource constraints such as high labour cost and heavy reliance on commercial Large Language Models (LLMs) which often produce inconsistent and irrelevant outputs as one of the major challenges. This study develops a domain-specific base LLM tailored for SMEs from scratch, leveraging on the advanced transformer architectures implemented in LLaMA 3.2 with Rotary Positional Embedding and Grouped Query Attention for enhanced scalability and efficiency. A rigorously curated dataset enabled fine-tuning, resulting in significant improvements in generating relevant and human-like responses. While LLaMA 3.2 outperforms GPT-2, challenges in coherence remain. The findings highlight the potential of LLMs in transforming SME operations and offer a framework for scalable, domain-specific solutions.

Keywords

SMEs, Customer Engagement, Large Language Models, LLaMA 3.2, Fine-Tuning

1. Introduction

1.1. Background

In recent years, SMEs in Malaysia have experienced significant growth, contributing 38.2% to the nation's GDP in 2020 [1]. However, they face challenges such as human resources and financial constraints. Limited salaries, job instability, and restricted career advancement lead to high employee turnover, affecting their ability to engage

customers [2]. Financial limitations further hinder SMEs from adopting costly CRM software, which is crucial for understanding customer needs and providing strategic business insights [3]. To address these issues, SMEs have turned to commercial Large Language Models (LLMs) as cost-effective solutions. LLMs automate responses to routine customer queries, reducing the need for additional staff and alleviating human resource constraints. Additionally, LLMs provide customer analysis capabilities, eliminating the need for expensive CRM systems, thus addressing financial challenges while enhancing customer engagement. The use of Retrieval-Augmentation-Generation (RAG) with external domain specific knowledge on pretrained LLM, however, still produces inconsistent and irrelevant factual outcomes. Such phenomenal is called hallucination [4] that still do not have a solution. On many occasions, a domain-specific LLM is generally considered a small language model (SLM), as it is a specialized version of a larger language model to perform well within a specific domain with its own unique vocabulary and requirements, often resulting in a smaller model size compared to a general-purpose LLM.

1.2. Problem Statement

Based on real life feedback, companies in Malaysia that provide involve in selling consumer electronic appliances and property rental businesses has identified several challenges with the commercial Large Language Models (LLMs) when they are used as the general-purpose generative AI chatbots. These issues include:

- Inconsistent responses to customer queries.
- Inability to provide accurate answers, even when relevant knowledge is supplied and proper prompts.
- Susceptibility to hallucinations, leading to responses that combine correct and incorrect information.
- Lack of follow-up questioning when unable to address customer queries effectively.

These persistent challenges hinder the effectiveness of commercial LLMs in customer engagement and underscore the need for improved or alternative solutions.

1.3. Objective

This project aims to study, design, build and validate a domain-specific base LLM model from scratch for customer engagement for two selected SME companies. The proof-of-concept prototype will be able to response with relevant, consistent and human-like outcome. This research will study a number of different model architectures, datasets and hyperparameters in the building of the base LLM model attempting to observe the above issues.

1.4. Project Scope and Contributions

This research focuses on building and evaluating a customer engagement specific Large Language Model (LLM) or Small Language Model from scratch, tailored for supporting and engaging customers in SMEs. The primary function of the do-

main-specific LLM is to generate consistent and contextually relevant responses to user queries and conversations in terms of products and services required by their customers.

The LLM will be trained on domain-specific datasets, including customer engagement literature and related materials. This specialized training aims to address issues like inconsistency, irrelevance, and the limitations of a particular industry for a domain-specific LLM model. The contribution includes studying the architectures of GPT-2 and Llama 3.2, focusing on their differences in design and performance. This analysis highlighted how their transformer models and attention mechanisms operate uniquely, influencing their suitability for various tasks.

Additionally, a domain-specific base LLM model will be designed and trained specifically for SME on customer engagement using related engagement data. The study found that hyperparameter tuning greatly impacts model performance, but training on a sufficiently large dataset is crucial to achieving meaningful improvements.

2. Related Work

Large Language Models (LLMs) have transformed natural language processing (NLP) by offering scalable solutions for complex tasks. This section explores key advancements in LLM architectures, such as GPT-2, Llama, and GEMINI, just to choose a few, and their diverse applications across industries, including finance, food and beverage (F&B), manufacturing, and customer engagement. These models demonstrate adaptability, efficiency, and versatility, highlighting their growing role in addressing industry-specific challenges and advancing modern NLP systems.

2.1. Models' Architectures of LLMs

The architecture of LLMs underpins their ability to process vast datasets and perform complex tasks. GPT-2, developed by Radford *et al.* (2019), is a foundational decoder-only Transformer model that employs multi-head self-attention and feed-forward layers. Its absolute positional encoding enables efficient token order understanding for general-purpose NLP tasks, though it struggles with long-range dependencies [5]. Building on these limitations, Llama, introduced by Touvron *et al.* (2023), incorporates Rotary Positional Embedding (RoPE) for encoding positional relationships and Grouped Query Attention (GQA) to enhance computational efficiency. These features enable Llama to handle long-context scenarios and complex domain-specific tasks while minimizing resource usage [6]. GEMINI, a hybrid encoder-decoder model, extends these advancements to multimodal tasks by integrating textual and visual data through multimodal attention mechanisms. Its architecture enables simultaneous processing of diverse inputs, supporting complex reasoning and cross-modal interactions [7]. Together, these models showcase the evolution of LLM architectures, with improvements in scalability, efficiency, and versatility for modern applications.

2.2. Domain-Specific Applications

Large Language Models (LLMs) are transforming industries by addressing domain-specific challenges through tailored applications. In finance, models like ChatGPT and LLaMA facilitate first-level customer interactions by answering queries about account balances, transaction histories, and fraud detection. This not only reduces operational costs but also enhances customer satisfaction by providing accurate, real-time responses [5] [8]. In the F&B sector, LLMs enhance operations by predicting inventory needs and offering personalized menu recommendations. Similarly, in manufacturing, LLMs leverage IoT data for real-time monitoring, fault prediction, and preventive maintenance, minimizing downtime and maximizing productivity. The versatility of LLMs is further amplified by their ability to retrieve detailed and contextually relevant information through Retrieval-Augmented Generation (RAG). This integration is pivotal in industries requiring precise knowledge retrieval, such as healthcare for diagnostics and pharmaceuticals, or legal services for document analysis [9] [10].

These domain-specific adaptations of LLMs not only improve efficiency and reduce costs but also allow businesses to tailor their services to meet the unique demands of their industries, demonstrating the expansive potential of these models in transforming functional areas (**Table 1**).

Table 1. Industry-specific applications of large language models (LLMs).

Industry	LLM Application
Finance	Customer service (queries on balances, transactions, fraud detection)
F&B (Food & Beverage)	Inventory prediction, personalized menu recommendations
Manufacturing	IoT data analysis for real-time monitoring, fault prediction, and preventive maintenance
Healthcare	Diagnostics, pharmaceutical research through RAG
Legal Services	Document analysis, retrieval of case laws and legal precedents using RAG

2.3. LLM in Customer Engagement

In customer engagement, LLMs like GPT-4, LLaMA 3.2, and GEMINI 2.0 address dynamic conversational challenges through advanced architectures. GPT-4 employs a decoder-only transformer design with improvements in contextual understanding and multi-turn conversation handling, enabling personalized responses for sentiment analysis, troubleshooting, and complex queries [11] [12].

Llama 3.2, introduced by Meta AI (2024), optimizes transformer efficiency through mechanisms like Grouped Query Attention (GQA) and Rotary Positional Embedding (RoPE) [13]. These innovations improve memory efficiency and long-context processing, making Llama particularly suitable for industries requiring quick, resource-efficient responses. In customer engagement, Llama excels in managing high query volumes by enabling real-time assistance, such as product recommendations, complaint resolution, and feedback analysis. Another notable

model, GEMINI 2.0, employs a hybrid encoder-decoder architecture designed for multimodal applications, integrating textual and visual data. This architecture enables GEMINI to provide enhanced customer service in e-commerce by analysing product descriptions alongside user-uploaded images to resolve issues like mismatched deliveries or damaged goods.

These models contribute significantly to customer engagement by addressing technical challenges like coherence, context retention, and dynamic query handling. Additionally, their ability to integrate with Retrieval-Augmented Generation (RAG) systems further enhance their performance in retrieving accurate, domain-specific knowledge, providing precise and contextually relevant outputs for customer inquiries [14]. Conclusively, the evolution of LLMs, from GPT-2 to advanced models like Llama 3.2 and GEMINI 2.0, showcases continuous architectural innovations. These models tackle scalability, context retention, and multimodal processing challenges, driving transformative impacts in customer engagement, inventory management, and predictive maintenance (**Table 2**).

Table 2. Comparison of customer engagement-focused large language models (LLMs).

Model	Architecture	Key Features	Customer Engagement Applications
GPT-4	Decoder-only Transformer	Enhanced contextual understanding, multi-turn conversation handling	Sentiment analysis, troubleshooting, handling complex queries
Llama 3.2	Optimized Transformer	Grouped Query Attention (GQA), Rotary Positional Embedding (RoPE), memory efficiency	Real-time assistance, product recommendations, complaint resolution
GEMINI 2.0	Hybrid Encoder-Decoder (Multimodal)	Text and image integration for customer service, RAG capabilities	E-commerce support, analysing product descriptions and images, resolving mismatched deliveries

3. Methodology

This section outlines the activities to be carried out for the design and development of the customer engagement base model. **Figure 1** illustrates the complete workflow for the base model development process. The workflow begins with understanding project requirements, where collaboration with stakeholders defines the model's technical and functional needs, ensuring it is resource-efficient and adaptable for real-world deployment. An in-depth study of existing LLMs follows, analysing key components like tokenization, embeddings, and attention mechanisms to inform design choices. Data preparation involves extracting, cleaning, and transforming diverse datasets into structured formats suitable for training.

Subsequent phases include selecting an appropriate model architecture based on scalability and domain-specific requirements, followed by the implementation of source code on core functionalities during base model development. Parameter studies explore configurations like context length and vocab size to optimize performance. The base models are built using these insights, evaluated against metric such as loss and further refined through prompt design to ensure relevance, con-

sistency and human-likeness in responses. Human evaluators then assess the generated outputs to identify strengths and improvement areas. The process concludes with finalizing the best-performing base model, which balances efficiency, relevance, consistency and human-likeness, establishing a strong foundation for future applications.

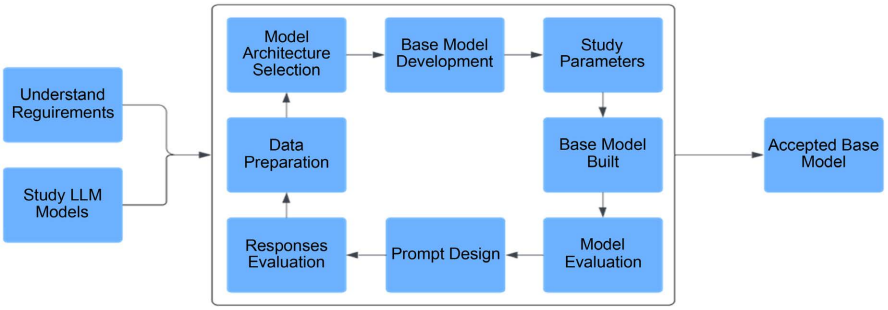


Figure 1. Base model development process workflow.

4. Results and Discussions on the Design and Implemented Model

This section explores the design criteria in term the chosen architecture, parameters and transformer block as well as the development and evaluation of a customer engagement-focused large language model (LLM). Section 4.1 describes the data preparation process, detailing the steps of data collection, cleaning, and embedding alignment to ensure seamless integration with the model architecture. Section 4.2 provides a comparative analysis of Generative Pre-trained Transformer 2 (GPT-2) and Large Language Model Meta AI (LLaMA) 3.2, emphasizing key design advancements and justifying their selection for this study. Section 4.3 delves into the training phase, including hyperparameter tuning and evaluations conducted across different configuration attempts. Section 4.4 evaluates the optimal model from each architecture based on the text generation metrics to test the model generation ability. Finally, Section 4.5 evaluates the models' response performance, focusing on relevance, consistency, and human-like quality in responses generated by the optimal configurations of each architecture.

4.1. Data Preparation

The data preparation process for pretraining the domain-specific language model was meticulously designed to ensure the quality and relevance of the dataset. It began with the collection of English customer engagement book from PDFDrive [15]. Initial preprocessing steps included joining paragraphs and appending metadata such as the data source at the beginning of the text with a special token <|endoftext|> at the end to mark content as a termination mark. The cleaned dataset is stored in an SQLite database. The size was 18 MB and contained approximately 3.7 million tokens. However, when this dataset was used to train utilising the GPT-2 architecture, it produced incoherent responses, exposing limitations in dataset quality, size, and preprocessing rigor.

In order to enrich data for model training, additional data was gathered from multiple reputable sources which include journal articles retrieved from platforms like ScienceDirect, Google Scholar, SpringerLink, and ProQuest. Enhanced preprocessing measures were implemented, such as removing irrelevant content like URLs, spaces, tables, images, and email addresses. A sophisticated cleaning pipeline was introduced, leveraging the LLaMA 3.1 model to filter out noise and refine the text further. Following this automated cleaning, a manual review was conducted to ensure the highest quality of data. Unstructured elements like tables disrupted the model's context understanding, leading to incoherent outputs. Removing such content improved training focus, enhancing the model's ability to generate fluent and contextually accurate responses. Metadata and updated special tokens were appended to the cleaned text, and the refined dataset was stored in a new SQLite database. The updated dataset size increased to 21 MB, with a reduced token count of 1.6 million due to the elimination of noisy data. When used for training the GPT-2 model, this enhanced dataset demonstrated significant improvements in producing responses that were more relevant, consistent, and fluent.

The tokenizers for both LLaMA 3.2 and GPT-2 were selected to ensure efficiency and compatibility with their respective architectures, while eliminating the tokenizer as a potential factor that could impact performance. The LLaMA tokenizer, based on SentencePiece-based Byte Pair Encoding (BPE), offers greater flexibility for handling multilingual and complex text structures, ensuring efficient tokenization and minimizing redundancy. In contrast, the GPT-2 tokenizer uses a fixed BPE vocabulary, providing consistent tokenization aligned with the model's original design, preserving stability and efficiency. By using each model's own tokenizer, the tokenization process is optimized to align closely with the architecture, ensuring better compatibility and performance.

For the LLaMA 3.2 model, subsequent refinements were made to optimize the training dataset. Special tokens were updated—replacing <|endoftext|> with <|end_of_text|> and introduce <|begin_of_text|> at the start of the text. The final dataset, stored in a separate SQLite database, was 21 MB in size and consisted of 1.5 million tokens. This reduction in token count was achieved by adopting a different tokenizer, based on BPE from Hugging Face, which ensured compatibility with LLaMA 3.2's architecture. The dataset prioritized domain relevance, with 61K tokens (3.91%) sourced from ScienceDirect articles, offering research-driven language, and 1.5 million tokens (96.09%) from PDF books, which provided structured, high-quality text for training (**Table 3**).

Table 3. Cleaned pre-training data.

Data Source	Proportion	Number of Tokens
Article	3.91%	61 K
Book	96.09%	1.5 M
		1.57 M

4.2. Proposed Base Model Design and Architecture

To explore the evolution of transformer-based architectures, this study selected two distinct models for comparison which are GPT-2 and LLaMA 3.2. These models were chosen to represent two critical stages in the development of transformer architectures, with GPT-2 serving as a foundational model widely adopted for natural language processing tasks and LLaMA 3.2 showcasing more recent advancements in efficiency and scalability (Figure 2). Both models have publicly available open-source implementations, providing accessible reference code for research and general explanation [16]. The selection of these architectures enables a comprehensive examination of the progression in design principles and the incorporation of innovative features over time. Below is a summarized comparison of their architectural designs in Table 4.

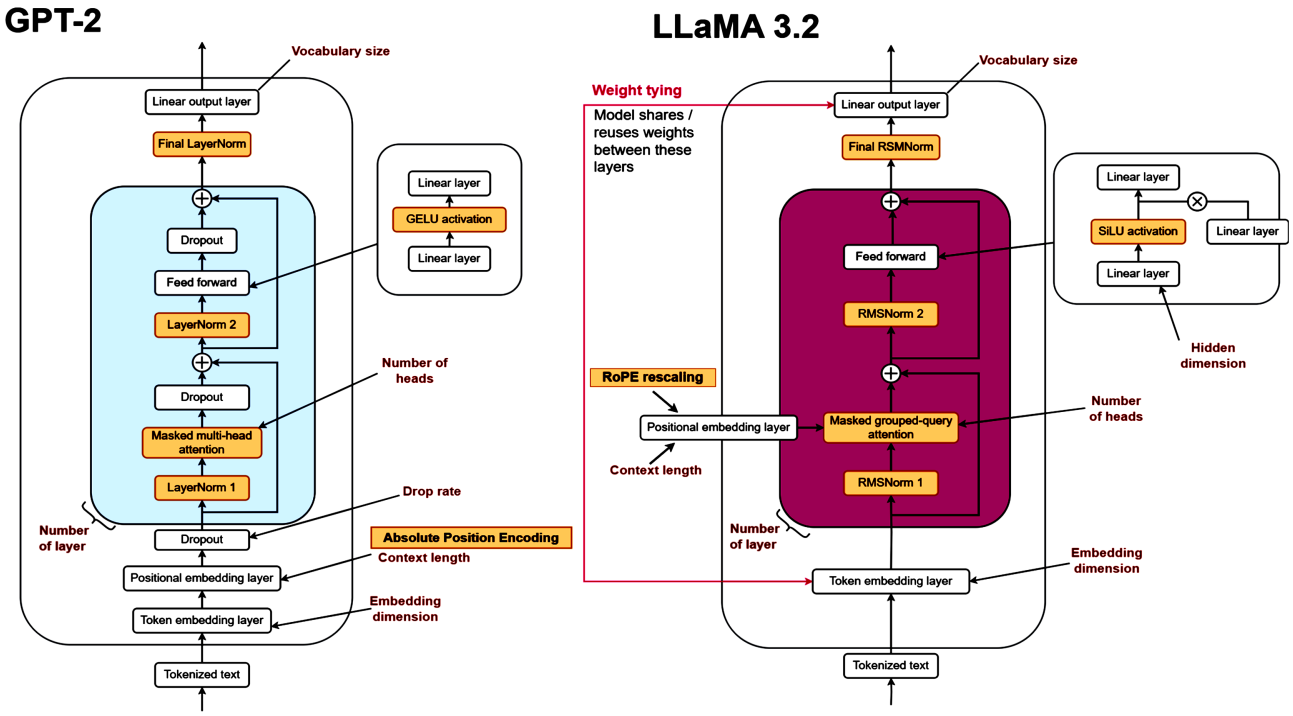


Figure 2. Base model architectural design of GPT-2 and LLaMA 3.2.

The comparison between GPT-2 and LLaMA 3.2 reveals significant advancements in transformer-based architectures, showcasing the evolution from foundational models to more sophisticated, efficient designs. GPT-2, as an earlier iteration, is robust for general-purpose natural language processing tasks, particularly with shorter contexts. Its architecture employs absolute positional encoding, multi-head attention, and Layer Normalization, ensuring stable learning and coherent outputs. However, it faces challenges with long-range dependencies due to its static positional encoding and less flexible attention mechanism. Despite these limitations, GPT-2 remains a reliable baseline for tasks requiring moderate computational resources.

Table 4. Comparison of architecture design of generative pre-trained transformer 2 (GPT-2) and large language model meta AI (LLaMA) 3.2.

Aspect	Generative Pre-trained Transformer 2 (GPT-2)	Large Language Model Meta AI (LLaMA) 3.2
Tokenizer	Byte Pair Encoding (BPE) with 50,257 tokens using Tiktoken	Expanded BPE tokenizer with 100,000 tokens and 28,000 additional tokens for multilingual support
Positional Encoding	Absolute Positional Encoding: captures token order effectively in short sequences but struggles with long-range dependencies	Rotary Positional Embedding (RoPE): captures both absolute and relative positional relationships, enabling superior handling of long contexts
Attention Mechanism	Multi-head Attention attends to different sequence parts simultaneously, ensuring diverse contextual relationships	Grouped-Query Attention (GQA): reduces computational cost by grouping attention heads while maintaining high-quality attention
Normalization	Layer Normalization stabilizes training by normalizing input across layers	Root Mean Square Normalization: improves computational efficiency and ensures stable training dynamics
Feed-Forward Network	Standard feed-forward network with Gaussian Error Linear Unit activation: ensures smooth gradients and non-linear transformations	Feed-forward network with Sigmoid Linear Unit activation: improves feature interaction and expressiveness
Strengths	Robust for shorter contexts and general-purpose NLP tasks	Superior scalability, efficiency, and performance for domain-specific and long-context tasks
Limitations	Static positional encoding limits long-range dependency handling	More computationally intensive but optimized for efficient use of resources

LLaMA 3.2 builds upon these foundations with a more advanced and optimized design. It incorporates Rotary Positional Embedding to handle both absolute and relative positional relationships, enhancing its ability to process longer contexts. The grouped-query attention mechanism significantly reduces computational costs without compromising attention quality, while Root Mean Square Normalization ensures efficient and stable training dynamics. The feed-forward network, enhanced by the Sigmoid Linear Unit activation, further improves the model's ability to capture complex patterns and maintain training stability. These innovations make LLaMA 3.2 a superior choice for tasks requiring scalability, efficiency, and robust contextual understanding.

4.3. Base Model Training and Configuration Evaluation

During the training phase of the two different model architectures, the study of the hyperparameters for each model is carried out to determine the optimal parameters that improve the model to understand the context of the data and queries. Metrics such as training loss, validation loss, and the quality of generated responses for every epoch are recorded in a log txt file and analysed to identify configurations that lead to improved performance. For each configuration combination, the model is initially trained for 10 epochs to ensure sufficient exploration of the hyperparameter space.

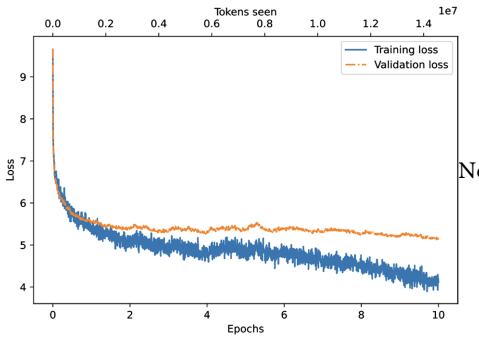
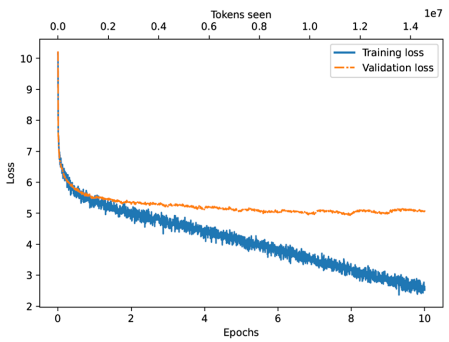
The Adaptive Moment Estimation AdamW optimizer is consistently used across all training sections with a fixed learning rate of 0.0005 and a weight decay of 0.1 to ensure stable and efficient model training. The learning rate controls the size of the updates made to the model's weights during each optimization step, providing a balance between rapid convergence and training stability. Meanwhile, weight decay acts as an L2 regularization term, reducing the risk of overfitting by penalizing excessively large weights. This constant setting ensures a uniform optimization approach across all training experiments, facilitating consistent evaluation of the impact of other hyperparameters on model performance.

With the consistent use of the optimizer mentioned above, two model architectures, GPT-2 and LLaMA3.2, were trained with multiple attempts using different parameter configurations. This section highlights a few configuration settings that have been tested for each model architecture.

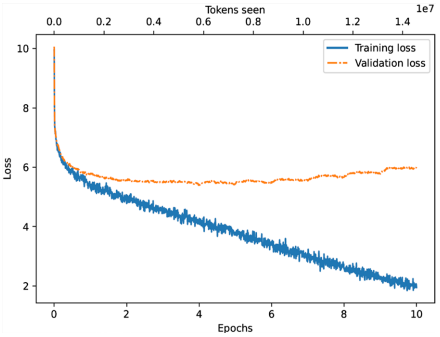
4.3.1. GPT-2 Optimal Configuration and Model Trained Assessment

With the implementation of the GPT-2 architecture, the model is trained using the GPT-2 tokenizer from the Tiktoken library, with a fixed vocabulary size of 50,257 tokens. These three selected models illustrate the impact of adjusting key hyperparameters on performance and result present in Table 5. Of the many configurations tested, these three models yielded the most significant performance variations. The study used the configuration of the existing GPT-2-Small model [17] as a starting point, with subsequent adjustments made to various parameters.

Table 5. GPT-2 model configuration evaluation.

Model Configuration	Train Loss	Validation Loss		Response Quality
GPT-2 Model 1: context_length: 256 emb_dim: 1024, n_heads: 16, n_layers: 12, drop_rate: 0.1,	4.108	5.128		Not a complete sentence.
GPT-2 Model 2: context_length: 256 emb_dim: 768, n_heads: 6, n_layers: 8, drop_rate: 0.2,	2.523	5.092		Responses included repeated words and lacked context or readability.

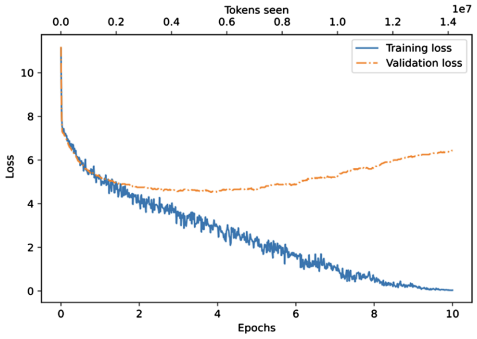
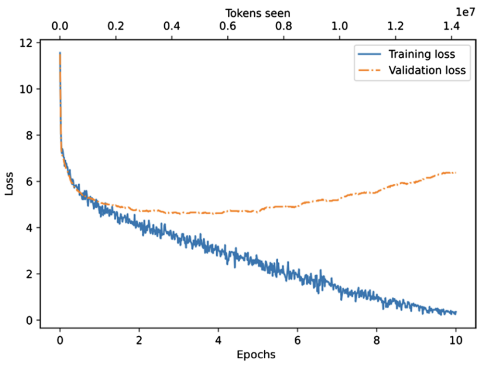
Continued

GPT-2 Model 3: context_length: 512 emb_dim: 768, n_heads: 8, n_layers: 8, drop_rate: 0.2,	1.944	5.966		Not readable phrases generated.

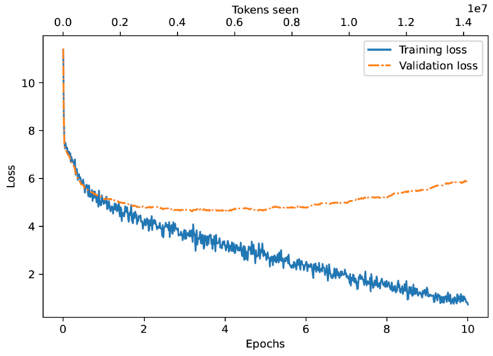
4.3.2. LLaMA 3.2 Optimal Configuration and Model Trained Assessment

With the implementation of the LLaMA model, the tokenizer is sourced from the Hugging Face library, specifically designed to align with the LLaMA 3.2 architecture and used for all attempts using this architecture design in Table 6. The configurations were adjusted from the baseline settings of the existing LLaMA 3.2 1B model [18]. After extensive experimentation with various configurations, only three models of LLaMa 3.2 design were selected to highlight the key performance variations and offer a basis for valuable comparative analysis.

Table 6. LLaMA 3.2 model configuration evaluation.

Model Configuration	Train Loss	Validation Loss		Response Quality
LLaMa Model 1: context_length: 512, emb_dim: 1024, n_heads: 16, n_layers: 12, n_kv_groups: 8, rope_base:200_000, rope_freq: { factor: 8.0 low_freq_factor: 0.5, high_freq_factor: 4.0}	0.037	6.438		Readable sentence and contextual in sentence level.
LLaMa Model 2: context_length: 512, emb_dim: 768, n_heads: 8, n_layers: 8, n_kv_groups: 4, rope_base":10000.0, rope_freq: { factor: 32.0, low_freq_factor: 0.5, high_freq_factor: 8.0}	0.357	6.375		Readable sentence but not answering the query correctly.

Continued

LLaMa Model 3: context_length": 512, emb_dim": 768, n_heads: 8, n_layers: 8, n_kv_groups": 8, rope_base: 400_000.0, rope_freq: { factor: 32.0, low_freq_factor: 1.0, high_freq_factor: 4.0}	0.734	5.844		Readable response, moderate reliability that response to certain intent of the query

The best configurations for the model were determined through extensive testing and evaluation, focusing on achieving an optimal balance between computational efficiency, model performance, and generalization capability. For the context length, 512 tokens emerged as the optimal choice, as it balanced memory usage and performance. An embedding dimension of 768 was identified as ideal, as it provided sufficient capacity for representing tokens while avoiding excessive memory consumption. The number of transformer layers was also optimized, with 8 layers delivering sufficient depth for learning complex patterns in the data.

The grouped-query attention mechanism performed best with 8 key-value groups, effectively balancing memory efficiency and attention quality. In terms of Rotary Positional Embeddings (RoPE), a base value of 400,000 provided the most effective encoding of positional relationships, particularly for mid-range token dependencies. Lower values, such as 10,000, reduced the model's ability to handle long-term positional relationships, while higher values showed no additional benefits. The RoPE frequency factor of 32.0 was optimal for rescaling positional encodings. Additionally, low and high frequency scaling factors of 1.0 and 4.0, respectively, were ideal for encoding positional information across different frequency ranges. These configurations collectively enabled the model to achieve the best performance while maintaining computational efficiency.

The comparative analysis of the two model architectures, GPT-2 and LLaMA 3.2, highlights significant differences in their performance and suitability for the task. The GPT-2 models consistently struggled with overfitting across all configurations, resulting in poor response quality. Despite adjustments to hyperparameters, such as embedding dimensions, attention heads, and context lengths, the GPT-2 architecture failed to generalize effectively, producing incomplete or repetitive responses that lacked semantic coherence. This indicates that the GPT-2 architecture may not be optimal for capturing the complexity and dependencies required in this application. In contrast, LLaMA 3.2 exhibited improved loss stability and better alignment with the intent of the queries, producing readable and more contextually coherent responses. These differences clearly highlight LLaMA

3.2's ability to generalize more effectively and handle complex dependencies, making it the better-performing model.

4.4. Proposed Metric-Based Evaluation

This evaluation carried on the optima model from each architecture only which are 3rd model from Table 5 and Table 6 respectively to compare the generation ability from different architectural design. These text generation metrics are proposed to provide a quantitative assessment of the model's output. The ROUGE (Recall-Oriented Understudy for Gisting Evaluation) scores, including ROUGE-1, ROUGE-2, and ROUGE-L, offer insights into the overlap of unigrams, bigrams, and the longest common subsequence between the generated responses and reference answers. A higher ROUGE score generally indicates better content overlap. The METEOR (Metric for Evaluation of Translation with Explicit Ordering) score considers synonyms and word order, offering a more nuanced evaluation of the semantic similarity between the generated and reference text. By generating fifty questions and answer pairs from the base data and averaging these scores, a comparative analysis of their performance in generating relevant and coherent responses conducted in Table 7. This process allows for an objective comparison of the models' text generation capabilities.

Table 7. Average ROUGE and METEOR scores for optimal models from each architecture.

Average Score	ROUGE-1	ROUGE-2	ROUGE-L	METEOR
GPT-2	0.0461	0.0085	0.0375	0.0976
LLaMA 3.2	0.0500	0.0021	0.0262	0.1051

The evaluation of the language models, GPT-2 and LLaMA 3.2, reveals distinct performance characteristics across various text generation metrics. ROUGE-1, which quantifies the overlap of single words between the generated and reference texts, indicates a slight advantage for LLaMA 3.2. LLaMA 3.2's ROUGE-1 score of 0.0500, compared to GPT-2's 0.0461, suggests that LLaMA 3.2 demonstrates a somewhat stronger ability to replicate individual words present in the reference text. This finding implies that LLaMA 3.2 may be more inclined to use the exact vocabulary of the reference, potentially resulting in outputs that closely align with the source text at a word-for-word level.

Conversely, the ROUGE-2 metric, which assesses the overlap of word pairs, presents a different perspective. In this case, GPT-2 achieves a score of 0.0085, while LLaMA 3.2 scores 0.0021. The higher ROUGE-2 score for GPT-2 indicates that this model is more effective at reproducing sequences of two-word combinations from the reference text. This suggests that GPT-2 might be better at preserving the local word order and short-range dependencies of the source text, which can be crucial for maintaining the fluency and naturalness of the generated output.

Evaluating the longest common sequence of words, ROUGE-L, further refines the comparative analysis. GPT-2 outperforms LLaMA 3.2 in this metric, with scores of 0.0375 and 0.0262, respectively. ROUGE-L is particularly sensitive to the overall fluency and coherence of the generated text, as longer common sequences imply a greater preservation of the reference text's structure. These results suggest that GPT-2 might be better at generating text that maintains longer stretches of coherent phrases and sentences, potentially indicating a stronger ability to capture the overall flow and organization of the source material.

In contrast to the ROUGE metrics, METEOR provides a more nuanced evaluation by considering synonyms and word order, thereby assessing the semantic similarity between the generated and reference texts. LLaMA 3.2 achieves a METEOR score of 0.1051, surpassing GPT-2's 0.0976. This outcome highlights LLaMA 3.2's superior capability in generating text that is semantically like the reference, effectively capturing the meaning and intent of the source text, even if different words or sentence structures are employed. This suggests that LLaMA 3.2 excels at producing outputs that are not only relevant but also convey the same information as the reference, demonstrating a deeper understanding of the underlying semantics.

4.5. Proposed Human Evaluation of Response Quality

This section will include the focus on evaluation of the relevance, consistency and human-like text of the response generated from the optimal model for each architecture. To test the model's performance under different scenarios, a set of prompts was carefully designed to enhance query understanding, ensuring that the model could respond effectively across diverse prompting styles. This approach allowed for a consistent evaluation framework while exploring the impact of prompt variations.

The evaluation focuses on three critical aspects of the generated responses: relevance, consistency, and human-likeness. Relevance is assessed both at the sentence level and within the context of the entire response paragraph. At the sentence level, each response's alignment with the query's specific intent is measured, while at the paragraph level, the evaluation considers how well the response maintains logical flow, coherence, and adherence to the broader context. Consistency evaluates the uniformity and coherence of the responses across different prompting styles, ensuring the model generates consistent outputs regardless of prompt variations. Lastly, human-likeness examines how natural and conversational the responses appear, assessing the fluency, tone, and structure of the generated text.

The evaluations of GPT-2 and LLaMA 3.2 highlight significant differences in their performance, strengths, and limitations, providing valuable insights into their effectiveness for complex natural language processing tasks. GPT-2 consistently struggles with overfitting, which leads to incomplete, repetitive, and contextually irrelevant outputs. These issues persist regardless of architectural adjustments or configuration changes, reflecting fundamental constraints in its design.

Efforts to enhance its performance through prompt variations, such as role-based prompts or re-reading strategies, have also proven ineffective. Consequently, GPT-2 demonstrates limited capacity to manage extended contexts or represent complex dependencies, rendering it unsuitable for tasks requiring nuanced understanding and reliable coherence (Table 8).

Table 8. Model response evaluation.

		GPT-2	LLaMA 3.2
Relevancy	Sentence level	★★	★★★★★
	Paragraph level	★	★
Consistency		★	★★
Human-Likeness		★★	★★★★

In contrast, LLaMA 3.2 demonstrates notable advancements, especially at the sentence level. It consistently delivers responses that are coherent, relevant, and human-like across various configurations. These achievements are supported by its advanced architectural features, such as Grouped-Query Attention (GQA) and Rotary Positional Embedding (RoPE), which improve stability and reduce susceptibility to overfitting. Additionally, deeper architectures in LLaMA 3.2 enhance its ability to capture intricate linguistic patterns, reinforcing its effectiveness in generating fluent individual sentences. This robust sentence-level performance underscores the significant progress achieved in modern language model architectures.

Despite these strengths, LLaMA 3.2 faces challenges at the paragraph level. While individual sentences are high in quality, they often lack alignment and logical flow when combined into multi-sentence outputs. This limitation reflects a gap in the model's ability to handle global context and dependencies, which are critical for producing cohesive and contextually rich paragraphs. Attempts to address this through prompt engineering have shown minimal success, indicating that these challenges stem from deeper architectural and training limitations. Furthermore, the narrow focus of its pre-training dataset, which emphasizes article-style content over conversational data, reduces the model's adaptability to dynamic queries or natural conversational contexts.

Both models exhibit fluency and grammatical correctness, but their ability to comprehend and respond to query intent varies significantly. GPT-2 frequently produces responses that prioritize high-attention words from its training data without addressing the broader context, resulting in fragmented and mechanical outputs. In comparison, LLaMA 3.2 achieves better sentence-level relevance but struggles with maintaining coherence and logical progression across multiple sentences. These shortcomings highlight the importance of refining training datasets and advancing global context modelling to bridge these gaps.

In summary, GPT-2's performance is hindered by overfitting and inherent architectural constraints, limiting its applicability for complex tasks. Meanwhile, LLaMA 3.2 demonstrates significant improvements in sentence-level performance but requires further development to enhance paragraph-level coherence and adaptability. These findings emphasize the need for ongoing innovations in architecture and dataset design to achieve more holistic and contextually robust natural language processing systems.

5. Conclusion and Future Works

This study presents key findings from the evaluation of LLaMA 3.2 and GPT-2 in customer engagement tasks, emphasizing their strengths and areas for improvement. LLaMA 3.2, with its advanced architectural features such as grouped-query attention, RoPE positional encoding, and RMSNorm, consistently outperformed GPT-2 in generating contextually relevant, consistent, and human-like responses. However, despite its superior performance, LLaMA 3.2 exhibited limitations, particularly in its ability to fully understand query intent and maintain logical flow in responses. These shortcomings were attributed to mild overfitting during training, which led to an overreliance on surface-level patterns and a formal tone that detracted from its human-like quality.

On the other hand, GPT-2, though constrained by its simpler architecture and limitations in data preprocessing, provided valuable insights into model development strategies. Both models benefited from iterative improvements, including stricter data preprocessing methods and optimized configurations. However, LLaMA 3.2 consistently demonstrated superior performance, reaffirming its potential as a more effective tool for customer engagement tasks.

Future work can focus on addressing the identified limitations of LLaMA 3.2 by incorporating a broader and more diverse range of conversational data, improving data-cleaning pipelines for enhanced preprocessing consistency, and adopting incremental training techniques tailored to dataset quality and characteristics. These initiatives are critical to further enhancing the model's ability to produce contextually relevant, cohesive, and human-like responses, ensuring its applicability in dynamic customer engagement scenarios.

Conflicts of Interest

The authors declare no conflicts of interest regarding the publication of this paper.

References

[1] Department of Statistics Malaysia (2020) Small and Medium Enterprises (SMEs) Performance 2020. https://www.dosm.gov.my/portal-main/release-content/small-and-medium-enterprises-smes-performance-2020

[2] Abas, M.F., Pardiman, P. and Supriyanto, S. (2024) Unlocking Human Potential: A Literature Review on HR Challenges and Innovations in SME Entrepreneurship. *Jurnal Manajemen Bisnis*, **11**, 785-799. https://doi.org/10.33096/jmb.v11i2.837

[3] Obradovic, V. (2022). CRM Software as a Service and Importance of the Approach for SMEs. *International Journal of Electrical Engineering and Computing*, **6**, 42-47. https://doi.org/10.7251/ijeec2206042o

[4] Ji, Z., Yu, T., Xu, Y., Lee, N., Ishii, E. and Fung, P. (2023) Towards Mitigating LLM Hallucination via Self-Reflection. *Findings of EMNLP* 2023, Singapore, 6-10 December 2023, 1827-1843. https://doi.org/10.18653/v1/2023.findings-emnlp.123

[5] Radford, A., Wu, J., Child, R., Luan, D., Amodei, D. and Sutskever, I. (2019) Language Models Are Unsupervised Multitask Learners. https://cdn.openai.com/better-language-models/language_models_are_unsupervised_multitask_learners.pdf

[6] Touvron, H., Lavril, T., Izacard, G., Martinet, X., Lachaux, M.-A., Lacroix, T., Scialom, T., *et al.* (2023) LLaMA: Open and Efficient Foundation Language Models. arXiv: 2302.13971. https://doi.org/10.48550/arxiv.2302.13971

[7] Ramachandran, A. (2024) Unveiling Google's Gemini 2.0: A Comprehensive Study of Its Multimodal AI Design, Advanced Architecture, and Real-World Applications. https://www.researchgate.net/publication/387089907_Unveiling_Google

[8] Zhang, K. and Smith, R. (2023) LLM: From Transformers to ChatGPT. ChatGPT: Optimizing Language Models for Dialogue. https://kpzhang.github.io/report/ChatGPT-KZ-Feb2023.pdf

[9] Zhao, S., Qiao, L., Luo, K., Zhang, Q.W., Lu, J. and Yin, D. (2024) SNFinLLM: Systematic and Nuanced Financial Domain Adaptation of Chinese Large Language Models. arXiv: 2408.02302v1. https://arxiv.org/abs/2408.02302v1

[10] Satta, G. and Farhangian, M. (2024) Adaptation of Large Language Models to Assistant Chatbots for Industrial Plants. Master's Thesis, University of Padova. https://thesis.unipd.it/bitstream/20.500.12608/64608/1/Farhangian_Mohammadardalan.pdf

[11] OpenAI (2023) GPT-4 Technical Report. https://cdn.openai.com/papers/gpt-4.pdf

[12] Brown, T., Mann, B., Ryder, N., Subbiah, M., Kaplan, J., Dhariwal, P., Amodei, D., *et al.* (2020) Language Models Are Few-Shot Learners. arXiv: 2005.14165. https://doi.org/10.48550/arxiv.2005.14165

[13] Meta AI (2024) Llama 3.2: Revolutionizing Edge AI and Vision with Open, Customizable Models. https://ai.meta.com/blog/llama-3-2-connect-2024-vision-edge-mobile-devices/

[14] Lewis, P., Perez, E., Piktus, A., Petroni, F., Karpukhin, V., Guu, K., Riedel, S., *et al.* (2021) Retrieval-Augmented Generation for Knowledge-Intensive NLP Tasks. arXiv: 2005.11401. https://doi.org/10.48550/arXiv.2005.11401

[15] PDF Drive (n.d.) Free Books on Artificial Intelligence and More. https://www.pdfdrive.com/

[16] Raschka, S. (2024) Build a Large Language Model (From Scratch). Manning Publications.

[17] Hugging Face (n.d.) OpenAI GPT-2 Documentation. https://huggingface.co/docs/transformers/en/model_doc/gpt2

[18] Hugging Face (2024) Meta-Llama/Llama-3.2-1B. https://huggingface.co/meta-llama/Llama-3.2-1B