One Model, Two Voices: Adaptive AI for Diverse Needs

1 Abstract

This study successfully demonstrates the potential of Retrieval-Augmented Generation (RAG) for tailoring responses to diverse audiences. Tuning revealed distinct persona-specific needs: marketing favors more concise responses, while research requires more comprehensive outputs. Retrieval parameters also impact performance, with optimal configurations differing by audience. Research models outperformed marketing counterparts, with Mistral demonstrating superior performance overall. Future work should prioritize improving response quality, addressing source bias, and optimizing inference speed for real-world deployment.

2 Introduction

As Generative AI transforms industries, organizations need flexible systems to deliver tailored information to diverse teams. Company X's rapid growth and commitment to GenAI-based products create a pressing need for improved document search and question-answering (QA) capabilities across its research and marketing functions. While both teams often share similar queries about Generative AI, they require distinct levels of detail and focus. For instance, the question *"How does a large language model learn from text?"* demands technical depth for researchers but a high-level strategic view for marketers. This report explores a proof-of-concept Retrieval Augmented Generation (RAG) system that successfully uses state-of-the-art techniques to address these needs.

3 Methodology

3.1 Architecture

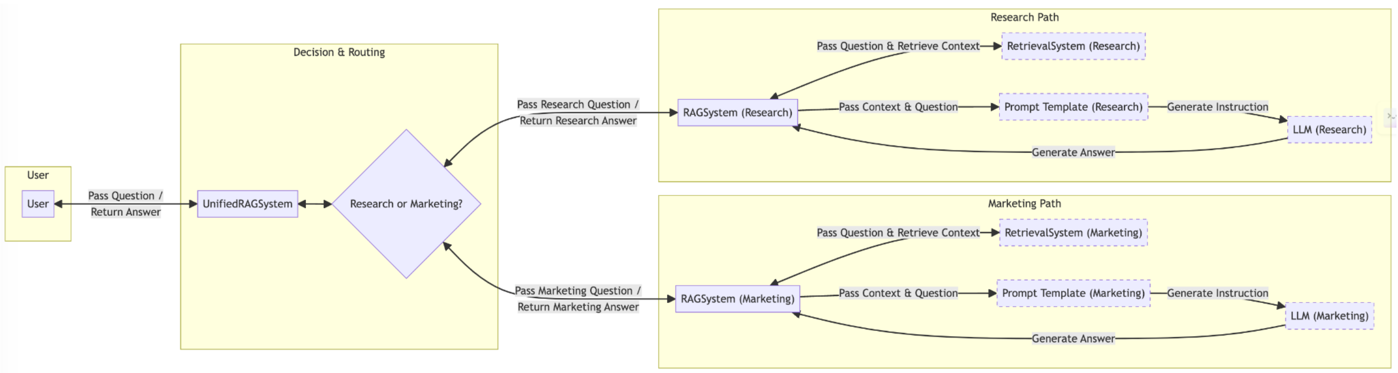

Figure 1: RAG System Architecture

This system uses a RAG architecture, leveraging the LangChain library to seamlessly integrate information retrieval, state-of-the-art language models (LLMs), and persona-specific prompting (see Fig. 1). To address the diverse needs of research and marketing teams, the core components offer a high degree of customization:

- **RetrievalSystem Class:** Employs dense vector search (a method for finding similar documents) to identify relevant materials from a corpus of arXiv papers, GitHub repositories, and Wikipedia entries. Document chunking with overlap ensures comprehensive representation for accurate retrieval. The system supports different embedding models ('multi-qa-mpnet-base-dot-v1', 'all-MiniLM-L6-v2', 'GIST-Embedding-v0') and customization of parameters like chunk size, overlap, search type (similarity or potentially maximum marginal relevance), and top-K documents retrieved.

- **LLM Object:** The system offers flexibility in text generation by supporting the use of either the Mistral or Cohere LLMs. Mistral is employed using the 'Mistral-7B-Instruct-v0.1' model from Hugging Face and quantized with BitsAndBytesConfig for computational efficiency. This reduces model size and improves inference speed, making it valuable for real-world deployment. Meanwhile, Cohere is accessed through the Cohere API. The system allows customization of LLM settings (max new tokens, temperature, top-p sampling, repetition penalty) to tailor response characteristics, influencing factors like length, creativity, focus, and diversity.

- **RAGSystem Class**: Orchestrates the core answer generation process, coordinating retrieval, creating a persona-specific LLM object, guiding that LLM's response generation, and ensuring refined output. It encapsulates a RetrievalSystem instance to obtain relevant context for the LLM and manages the end-to-end LangChain pipeline. The class allows modification of retrieval parameters (via the embedded RetrievalSystem), selection of the LLM (Mistral or Cohere), fine-tuning of LLM generation settings (e.g., max new tokens, temperature), and customization of prompt templates. These enable the tailoring of output style and content for research or marketing audiences.

- **UnifiedRAGSystem Class**: The UnifiedRAGSystem class acts as a dispatcher, routing questions to the appropriate RAGSystem instance (research or marketing) to ensure answers are tailored to the user's persona.

This system's strength lies in its ability to deliver tailored answers for distinct audiences. Flexibility is evident in its use of persona-specific configurations within both the language models (Mistral or Cohere) and the retrieval components. Distinct prompt templates, along with separate RetrievalSystem and RAGSystem instances for research and marketing, allow fine-grained optimization for each audience's unique needs.

3.2 Hyperparameter Tuning, Testing, and Evaluation

Tuning and testing relied on BLEURT as the primary metric to drive the hyperparameter optimization process. While BLEURT has limitations, it offers an automated way to assess the similarity between a generated response and a human-written reference.

A staged tuning process targeted optimal configurations for both marketing and research audiences. Initial tuning used a subset of the validation dataset (10 samples) for efficient parameter exploration. Following each optimization stage (LLM, retrieval, and prompt template), the best configurations were inherited and evaluated on the full validation dataset (75 samples). Final configurations were further assessed through manual evaluation of selected questions.

Optimization focused on three core RAG system areas. For the LLM, max new tokens, temperature, top-p, and repetition penalty, were adjusted and the performance of tuned Mistral and Cohere models were compared. For retrieval, different embedding models and varied document chunk sizes and overlaps were tested. Due to time constraints, document search type (e.g., similarity vs. maximum marginal relevance) and top-K retrieval parameters were not tuned. Finally, template variations (Table 3) were tested, including phrasing adjustments, instructions to only use provided context, and the use of synthetic examples (generated using GPT-4 conditioned on the validation question-answer data) to enable few-shot learning.

4 Results

RAG System	**LLM** **Max New Tokens, Temp, Top-P, Repetition Penalty**	**Retrieval** **Embedding Model, Chunk Size, Chunk Overlap**	**Template**	**BLEURT**	%
Marketing					
Cohere Marketing Base	150, 0.6, 0.95, 0.25	Multi-qa-mpnet-base-dot-v1, 128, 0	1	0.433	-
Cohere Marketing LLM Opt	100, 0.6, 1, 0	Multi-qa-mpnet-base-dot-v1, 128, 0	1	0.447	3.2%
Mistral Marketing Base	300, 0.6, 0.95, 1.2	Multi-qa-mpnet-base-dot-v1, 128, 0	1	0.464	-
Mistral Marketing LLM Opt	250, 0.6, 0.95, 1	Multi-qa-mpnet-base-dot-v1, 128, 0	1	0.467	0.6%
Mistral Marketing LLM Retrieval Opt	**250, 0.6, 0.95, 1**	**all-MiniLM-L6-v2, 128, 64**	**1**	**0.474**	**1.5%**
Mistral Marketing LLM Retrieval Template Opt	250, 0.6, 0.95, 1	all-MiniLM-L6-v2, 128, 64	3 w/ examples	0.464	(2.1%)
Research					
Cohere Research Base	150, 0.6, 0.95, 0.25	Multi-qa-mpnet-base-dot-v1, 128, 0	1	0.444	
Cohere Research LLM Opt	150, 0.8, 0.95, 1	Multi-qa-mpnet-base-dot-v1, 128, 0	1	0.450	1.4%
Mistral Research Base	300, 0.6, 0.95, 1.2	Multi-qa-mpnet-base-dot-v1, 128, 0	1	0.471	
Mistral Research LLM Opt	300, 0.8, 0.95, 1	Multi-qa-mpnet-base-dot-v1, 128, 0	1	0.474	0.6%
Mistral Research LLM Retrieval Opt	**300, 0.8, 0.95, 1**	**GIST-Embedding-v0, 512, 128**	**1**	**0.493**	**4.0%**
Mistral Research LLM Retrieval Template Opt	300, 0.8, 0.95, 1	GIST-Embedding-v0, 512, 128	1	0.486	-1.4%

Table 1: BLEURT Scores by Persona and Model Configuration (Full Validation Set). Final configurations are bolded. Retrieval settings set at top-K = 3 using similarity search across all configurations.

Table 1 summarizes the performance of various RAG system configurations, demonstrating a clear advantage for persona-tailored optimization. Research models consistently outperformed marketing models, likely due to the research-focused dataset composition (e.g., Arxiv papers comprised 90.6% of retrieval system passages for the top marketing model, while Wikipedia only comprised 6.3% - see Table 2). Mistral-based RAG systems excelled across both domains, with the choice of LLM (Mistral vs. Cohere) being the largest contributor to performance differences across tested configurations. Marked differences between personas emerged in optimal configurations: marketing systems favor shorter responses and the all-MiniLM-L6_v2 embedding model, while research benefits from longer responses and the GIST-Embedding-v0 model. Of all the Further analysis of these findings, including the unexpected impact of template optimization on performance, is provided in subsequent sections.

4.1 LLM Choice and Configuration

Mistral-based LLM configurations consistently outperformed Cohere across both marketing and research audiences, demonstrating the importance of persona-specific tuning. Marketing configurations favored shorter responses (achieved through lower 'max new tokens), while research configurations benefitted from longer, more comprehensive responses (enabled by higher 'max new tokens'). Tuning yielded greater BLEURT score improvements for Cohere (Marketing: 3.2%, Research: 1.4%) than Mistral (Marketing: 0.6%, Research: 0.6%). This suggests that Cohere's default parameters may have been further from optima.

4.2 Retrieval Vector Embedding Model

Retrieval optimization played a critical role in enhancing RAG system performance. The first step focused on selecting the optimal embedding model, which determines the effectiveness of document retrieval for the LLM. For marketing, switching from the default multi-qa-mpnet-base-dot-v1 model to the all-MiniLM-L6-v2 model led to a minor 0.65% BLEURT improvement (from 0.458 to 0.461) in the validation subset. For research, switching from the default multi-qa-mpnet-base-dot-v1 to the GIST-Embedding-v0 model led to a greater improvement of 1.97% BLEURT (from 0.451 to 0.460) in the validation subset.

4.3 Retrieval Chunk Size and Overlap

With optimal embedding models identified, further refinements were made to chunk size and overlap parameters. For marketing, increasing chunk overlap from the default of 0 to 128 led to a notable 3.56% BLEURT improvement (from 0.449 to 0.465) in the validation subset. This suggests that for marketing purposes, providing greater overlap between chunks helps the LLM identify important concepts and patterns. Analysis of a heatmap cross-tabulating chunk size and overlap (Fig. 3) shows that while higher chunk sizes (e.g., 512) are generally preferred, the optima for marketing is for a relatively low chunk size (128) with high relative overlap (64). This suggests that for marketing tasks, the system benefits from a balance of concise chunks with sufficiently large overlaps to provide context and aid in pattern recognition. Given the high overlap preference for marketing, future studies should experiment with a range of top-K parameters (chunks retrieved) to ensure sufficient contextual information is provided to the LLM.

Research configurations demonstrated specific preferences for chunk size and overlap. Increasing chunk size from 128 to 612, with a chunk overlap of 128, resulted in a 2.7% BLEURT improvement (from 0.472 to 0.485) in the validation subset. Analysis in Fig. 3 reinforces this finding, revealing a clear trend of superior BLEURT scores with chunk sizes around 512 and moderate overlaps (64 to 128). This indicates that research-focused systems benefit from larger retrieved passages and a degree of redundancy to ensure the LLM can adequately process complex and interrelated concepts.

4.4 Prompt Template Design

Prompt template design had a significant impact on the RAG system's performance within the marketing domain. Across all variations tested, Template 3 (which included instructions to only use information in the context) achieved the highest BLEURT scores, with all its versions (including and excluding few-shot examples) outperforming other templates, as shown in Fig. 3. Notably, adding few-shot examples to Template 3 consistently improved BLEURT scores in the validation data subset used for these experiments. For instance, transitioning from the default template (Template 1, no examples) to Template 3 with examples resulted in a substantial 4.1% BLEURT increase (from 0.439 to 0.457) in the validation subset.

In contrast to marketing, research configurations demonstrated less sensitivity to template variations, as shown in Fig. 3. The default template (Template 1, no examples) proved optimal for the research domain. While including instructions to focus solely on the provided context was less beneficial for research tasks, the inclusion of few-shot examples led to mixed results. This highlights the need for potential fine-tuning of research-focused examples to ensure they consistently enhance response quality.

4.5 Final Evaluation

Validation of the full dataset revealed a surprising finding: the default marketing template achieved higher overall BLEURT scores than Template 3 with few-shot examples, despite the initial promise of the latter on a smaller subset. This underscores the crucial importance of comprehensive validation. Qualitative evaluation of responses using GPT-4 (Fig. 4) confirmed BLEURT scores and suggested that the LLM Retrieval Opt model offered the optimal balance of accuracy, comprehensiveness, domain relevance, clarity, and adherence to desired response lengths. In contrast, the LLM Retrieval Template Opt model, while excelling in conciseness, sometimes lacked the depth and marketing-specific insights observed in the LLM Retrieval Opt configuration. Consequently, the 'Mistral Marketing LLM Retrieval Opt' and 'Mistral Research LLM Retrieval Opt' were selected as final models.

To further assess the selected models, a qualitative evaluation of marketing and research RAG system responses to two representative test questions was conducted against their corresponding gold standard responses (Table 5, Table 5). The evaluation confirms that the system does tailor responses to the distinct needs of each persona. Retrieved contexts and generated responses alike demonstrate the desired patterns: marketing outputs are shorter and less technical, focusing on domain-specific applications (e.g., customer feedback, chatbots in Table 5), while research outputs are longer, delving into technical specifics and concepts (e.g., mentioning datasets like SuperNaturalInstructions in Table 5). BLEURT scores appear to reflect response length and specificity. In Table 5, both marketing and research responses are much lengthier than reference responses, introducing irrelevant methods not present in the gold standard responses, potentially limiting their BLEURT scores.

5 Discussion and Conclusions

5.1 Key Findings

This proof-of-concept implementation successfully demonstrates the feasibility of persona-specific RAG system configuration, with several important themes and learnings:

- **Persona-Specific LLM Settings:** The study reveals distinct optimal configurations for marketing and research. Marketing favors shorter responses with lower repetition penalties, while research requires longer, more comprehensive responses. This underscores the importance of understanding both the stylistic and informational needs of different user groups.
- **Chunk Size and Overlap Importance:** Chunk size and overlap significantly influence the LLM's ability to synthesize information, with marketing benefiting from greater context overlap and research requiring larger passages with moderate redundancy. This highlights the need to align information retrieval with persona-specific needs.
- **Research Outperformance and Template Sensitivity:** The research models' superior performance, along with marketing's greater template sensitivity, likely stems from the research-heavy dataset composition. This emphasizes the crucial importance of balanced datasets that encompass the full range of potential user inquiries.
- **Mistral Outperforms Cohere:** Mistral-based RAG configurations consistently achieved higher BLEURT scores than Cohere across both personas. Several factors could contribute to this, including potential differences in pre-training data alignment. Further investigation is needed to isolate the primary reasons.
- **Full Dataset Validation is Essential:** The template selection discrepancy highlights the risks of overfitting hyperparameters to a subset of data during optimization. Robust validation processes with full datasets are essential.

5.2 Recommendations

The proof-of-concept demonstrates the RAG system's potential. A phased plan is recommended for a successful rollout: first, address immediate limitations, then conduct a pilot study, and finally, implement further refinements before deployment.

Based on the system's current limitations, immediate research should focus on four key priorities: Firstly, addressing response quality where BLEURT scores remain unsatisfactory, particularly the gap between marketing and research performance, through hyperparameter optimization of retrieval search (e.g., testing search types, top-K parameters), persona-specific Mistral fine-tuning, and prompt template refinement. Secondly, tackling response length imbalances, by prioritizing conciseness during fine-tuning and setting explicit length targets in evaluation. Thirdly, mitigating the research bias in the retrieval system by diversifying data sources for both audiences. Fourthly, expanding evaluation beyond BLEURT with human feedback and metrics like BERTScore, while enlarging the validation dataset to reflect the diversity of real-world questions.

Once the above issues are demonstrably improved, a closed pilot study should be conducted to achieve several goals crucial for subsequent investment and development decisions: Firstly, quantifying benefits by evaluating efficiency gains (e.g., time saved by engineers finding GenAI implementation answers and by marketers drafting GenAI-related blog posts), alongside the RAG system's impact on GenAI communication quality and clarity for both audiences. Secondly, assessing system load through tracking usage patterns to understand real-world query volume and peak demand periods, to inform optimal deployment strategies and cost projections. Thirdly, identifying and quantifying risks by soliciting user feedback to pinpoint potential biases in generated responses, factual inaccuracies, and any data privacy concerns. Finally, pinpointing areas for improvement by gathering qualitative feedback on pain points, desired features, and overall satisfaction levels to guide targeted refinement of the RAG system.

Should the company decide to further develop the system post-pilot study, further development priorities should focus on addressing the following areas, along with any additional areas of improvement identified during the pilot study: Firstly, enhancing inference speed using techniques like flash attention, as long inference times compromise real-time usability. Secondly, as outdated data in the retrieval system creates the risk of factual inaccuracies, future research should explore automating updates to the vector store and leveraging APIs to external data providers for real-time information access. Finally, developing a risk mitigation strategy is essential before deployment, to address potential biases in generated responses, factual inaccuracies, and data privacy concerns.

For deployment, a hybrid deployment strategy is likely optimal due to the anticipated distribution of queries, in which the company is expected to have a regular daily load with load increases during product releases. Reserving LLM capacity for average daily load and using pay-per-use for peak demand would balance cost-effectiveness and scalability to accommodate fluctuating usage.

This study underscores the potential of persona-specific RAG systems to address the diverse GenAI needs of different teams. By addressing identified limitations and exploring further refinements, this system can become a valuable asset for streamlining GenAI workflows and promoting responsible innovation within the company.

6 Appendix

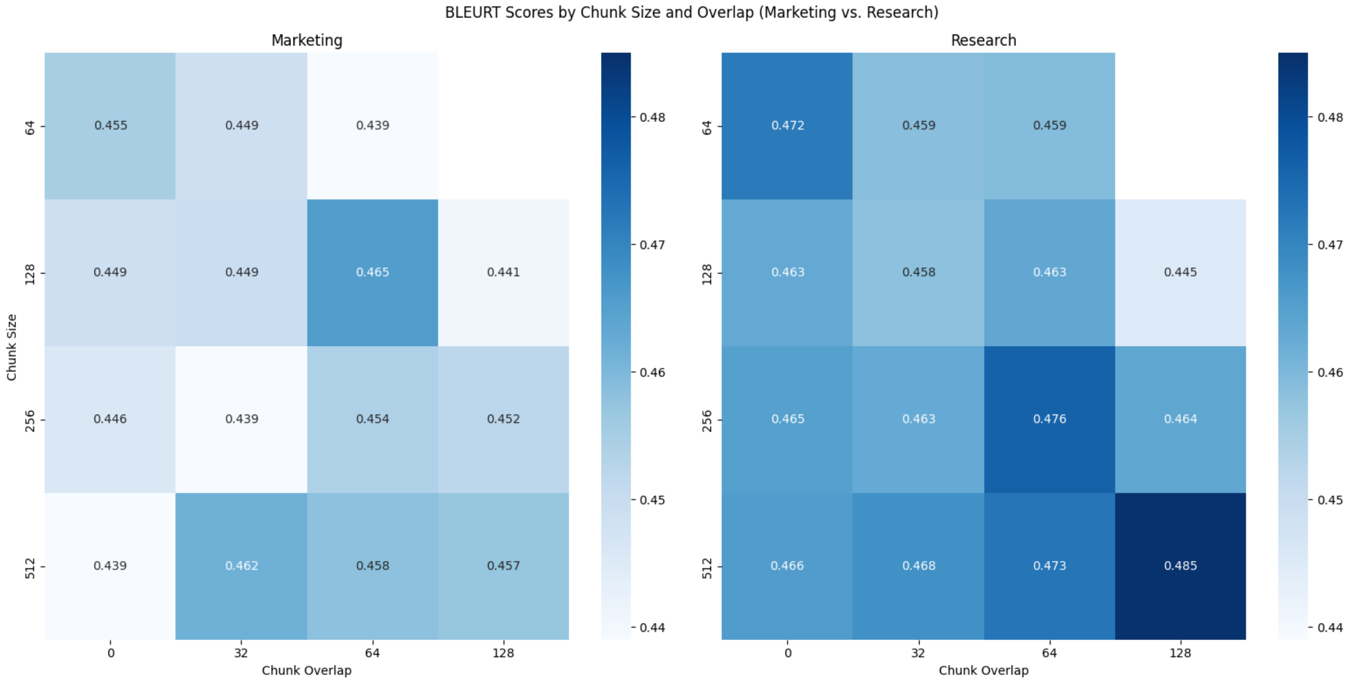

Figure 2: BLEURT Scores by Persona, Retrieval Chunk Size and Overlap (Validation Subset)

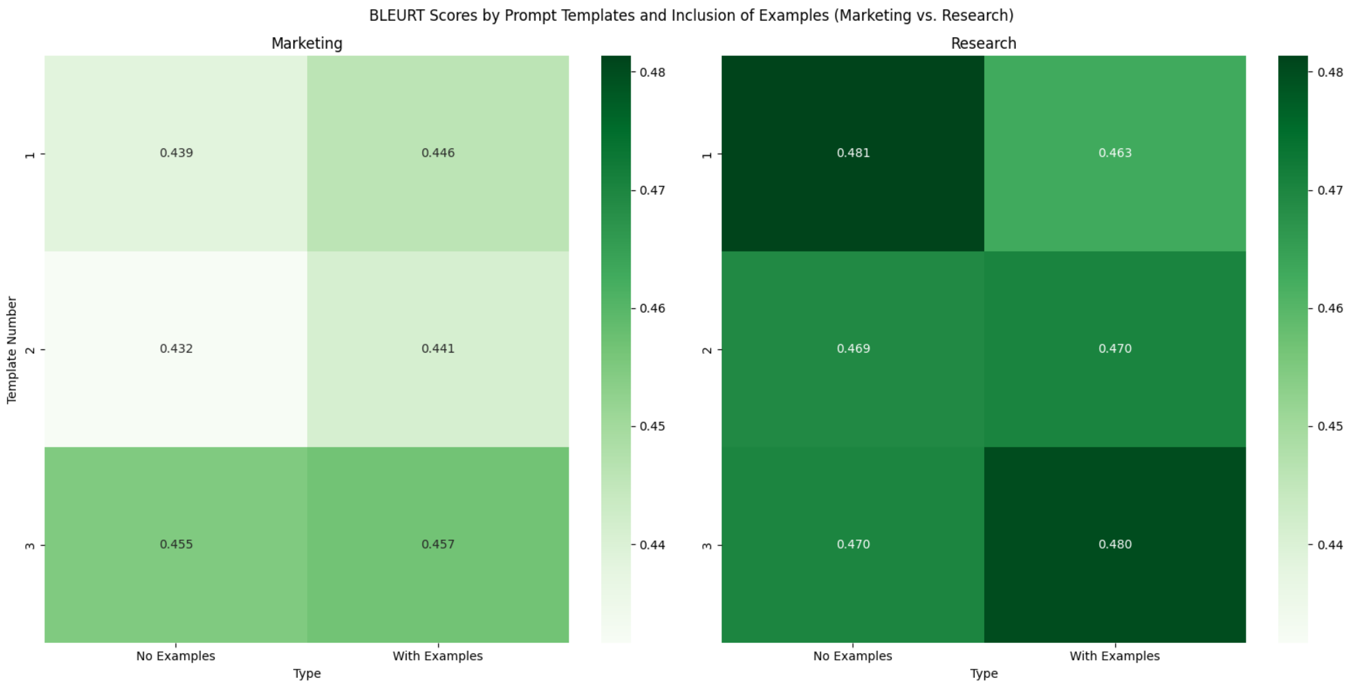

Figure 3: BLEURT Scores by Persona, Core Prompt Template and Inclusion of Examples in Template (Validation Subset)

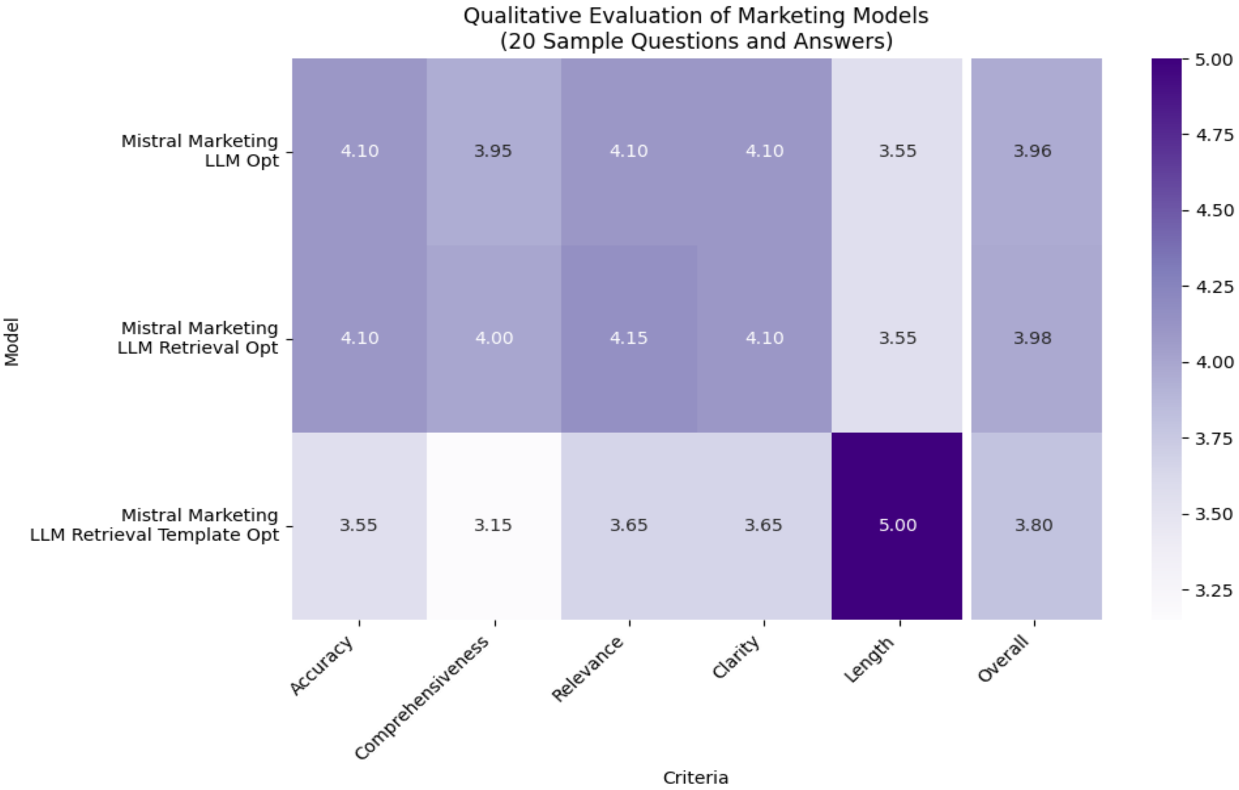

Figure 4: Qualitative Evaluation of Top 3 Marketing Models (0-5 Scale, 20 Sample Answers)

RAG System	**Document Chunk Size / Overlap**	**Distribution of Splits in Retrieval Vector Store**			
		Arxiv	**Web (GitHub)**	**Wikipedia**	**Total**
Mistral Marketing LLM Retrieval Opt	128, 64	20,661 (90.6%)	705 (3.1%)	1,440 (6.3%)	22,806 (100.0%)
Mistral Marketing LLM Retrieval Template Opt					
Mistral Research LLM Retrieval Opt	512, 128	3,897 (88.9%)	156 (3.6%)	326 (7.4%)	4,379 (100.0%)
Mistral Research LLM Retrieval Template Opt					

Table 2: Distribution of Document Sources in Retrieval Vector Store for Final RAG Configurations, by Persona

Research	
Few Shot Examples Generated by GPT-4	Example 1: Question: How do neural networks learn from data? Answer: Neural networks learn by adjusting internal parameters to reduce differences between actual and predicted outcomes. This process involves calculating the gradient of the loss function, allowing the model to update weights in a direction that minimizes the loss, making predictions increasingly accurate. Example 2: Question: What are the implications of AI in enhancing cybersecurity? Answer: AI enhances cybersecurity by enabling automated threat detection and response, analyzing patterns in data to identify potential threats. Despite its benefits, AI's use in cybersecurity raises concerns about privacy and ethical implications, necessitating careful consideration.
Template 1	[INST] Deliver an in-depth technical exploration, focusing on the underpinnings, theoretical models, and practical applications. Highlight recent advancements and comparative analysis with prior models. <Few Shot Examples Included Depending on Variant> Context: {context} Question: {question} [/INST]
Template 2	[INST] Provide a detailed investigation that covers foundational theories, model architectures, and application scenarios. Emphasize innovation in the field and conduct a review comparing new models to traditional approaches. <Few Shot Examples Included Depending on Variant> Context: {context} Question: {question} [/INST]
Template 3	[INST] Using only the information provided in the context, conduct a thorough analysis that delves into the fundamental theories, model designs, and their practical implications. Discuss innovations and perform a critical comparison with established methodologies. <Few Shot Examples Included Depending on Variant> Context: {context} Question: {question} [/INST]
Marketing	
Few Shot Examples Generated by GPT-4	Example 1: Question: How do neural networks learn from data? Answer: Neural networks learn from data by making guesses, learning from mistakes, and improving predictions over time. This self-improvement makes technologies smarter and more efficient. Example 2: Question: What are the implications of AI in enhancing cybersecurity? Answer: AI revolutionizes cybersecurity, offering faster, more accurate threat detection and response. It automates security monitoring, providing stronger protection against digital threats.
Template 1	[INST] Based on the context, provide a succinct and high-level overview that answers the question for a marketing audience. Focus on broad implications, market trends, and how the information could be used in marketing strategies. <Few Shot Examples Included Depending on Variant> Context: {context} Question: {question} [/INST]
Template 2	[INST] Offer a brief yet comprehensive summary tailored for a non-technical audience, highlighting the key benefits, industry trends, and strategic value of the technology. <Few Shot Examples Included Depending on Variant> Context: {context} Question: {question} [/INST]
Template 3	[INST] With the context given, craft a clear, concise summary aimed at a general audience. Highlight the practical benefits, current trends in the industry, and the strategic importance of the technology. Ensure the response is grounded solely in the context provided. <Few Shot Examples Included Depending on Variant> Context: {context} Question: {question} [/INST]

Table 3: Prompt Template Variations, by Persona

Question	'What purpose do large language models serve in the field of natural language processing?'
Marketing	
Context Provided	'large language models efficient. Through our work, our aim is to help the community create more\n\n[10] Y. Chen, R. Wang, H. Jiang, S. Shi, and R.-L. Xu. Exploring the use of large language models\n\nThis implies that large models are more generaliz-\nable to compute texts in various domains and task'
Document Sources	['ArXiv', 'ArXiv', 'ArXiv']
Generated Answer	'Large language models are a powerful tool in the field of natural language processing (NLP). They are designed to efficiently process and analyze large amounts of text data across various domains and tasks. By being able to generalize their computations to a wide range of textual data, large language models can be used in a variety of NLP applications, such as sentiment analysis, machine translation, and text classification. This makes them a valuable resource for businesses and organizations looking to improve their NLP capabilities and gain insights from large textual datasets. In marketing, large language models can be used to analyze customer feedback and reviews, track social media sentiment, and improve customer service by automating responses to common queries. Overall, large language models are a key component of modern NLP and offer significant benefits for businesses looking to leverage the power of language data.'
Reference Answer	'Large language models serve the purpose of improving performance in various natural language processing tasks, such as speech recognition, machine translation, natural language generation, optical character recognition, handwriting recognition, grammar induction, and information retrieval.'
BLEURT Score	0.35903
Research	
Context Provided	'models are useful for a variety of tasks, including speech recognition (helping prevent predictions of low-probability (e.g. nonsense) sequences), machine translation, natural language generation (generating more human-like text), optical character recognition, handwriting recognition, grammar induction, and information retrieval.Large language models, currently their most advanced form, are a combination of larger datasets (frequently using scraped words from the public internet), feedforward neural\n\nA large language model (LLM) is a language model notable for its ability to achieve general-purpose language generation and other natural language processing tasks such as classification. LLMs acquire these abilities by learning statistical relationships from text documents during a computationally intensive self-supervised and semi-supervised training process. LLMs can be used for text generation, a form of generative AI, by taking an input text and repeatedly predicting the next token or word.LLMs are\n\nthese different dimensions without having to\nrely on ground truth human annotations. We\nposit that such a framework can crucially con-\ntribute to faster evaluation cycles of RAG archi-\ntectures, which is especially important given\nthe fast adoption of LLMs.\n1\nIntroduction\nLanguage Models (LMs) capture a vast amount\nof knowledge about the world, which allows them\nto answer questions without accessing any exter-\nnal sources. This idea of LMs as repositories of\nknowledge emerged shortly after the introduction'
Document Sources	['Wikipedia', 'Wikipedia', 'ArXiv']
Generated Answer	'Large language models (LLMs) are a type of machine learning model that are used in the field of natural language processing. They are particularly useful for tasks such as text generation, speech recognition, machine translation, and information retrieval. LLMs acquire their abilities by learning statistical relationships from large datasets of text documents, often using techniques such as self-supervised and semi-supervised training. They are capable of generating human-like text, predicting the next token or word in a sequence, and identifying patterns in language.One of the main advantages of LLMs is that they do not require ground truth human annotations in order to learn and perform their tasks. This allows for faster evaluation cycles and more efficient training. Additionally, LLMs can be trained on large datasets of text documents, which allows them to capture a vast amount of knowledge about the world and answer questions without accessing any external sources.Recent advancements in LLM technology have seen the development of even larger and more advanced models, such as the transformer architecture. These models are able to process and understand language in a more sophisticated and nuanced way, and have been shown to achieve state-of-the-art performance on a variety of natural language processing tasks.Compared to prior models, LLMs have shown significant improvements in both accuracy and efficiency. They are able to generate more human-like text and perform more complex tasks, such as language translation and text summarization.'
Reference Answer	'Large language models (LLMs) serve the purpose of enabling general-purpose language generation and other natural language processing tasks such as classification. They achieve this by learning statistical relationships from text documents during computationally intensive self-supervised and semi-supervised training. LLMs can be used for text generation by predicting the next token or word, making them valuable for tasks like speech recognition, machine translation, and information retrieval. Additionally, LLMs have superseded previous models like recurrent neural networks, showcasing their efficiency and effectiveness in NLP tasks.'
BLEURT Score	0.53851

Table 4: Sample Responses, Reference Answers and BLEURT Scores (Question 1 of 2), by Persona

Question	’What methods are typically employed to create training data for embedding models that use task-specific instructions?’
Marketing	
Context Provided	ing the importance of instructions to create task-\naware embeddings. Our analysis shows that instruc-\n\nembeddings applicable to a wide range of tasks. It\ncombines several conventional benchmarks (e.g.,\n\nfinetuned embedding models. Given an input text\nx and a task instruction Ix, INSTRUCTOR encodes’,
Document Sources	[’ArXiv’, ’ArXiv’, ’ArXiv’]
Generated Answer	’To create training data for embedding models that use task-specific instructions, several methods are typically employed. These methods include fine-tuning pre-trained models, using task-specific datasets, and generating synthetic data using techniques such as data augmentation and transfer learning. Fine-tuning pre-trained models involves adapting an existing model to a specific task, while using task-specific datasets involves collecting and labeling data specifically for the task at hand. Generating synthetic data can be done by applying techniques such as data augmentation and transfer learning, which involve creating new data by modifying existing data or using data from related tasks. These methods can be used to create task-aware embeddings that are applicable to a wide range of tasks, and can be used in marketing strategies to improve the accuracy and effectiveness of natural language processing and machine learning applications.’
Reference Answer	’Training data for embedding models that use task-specific instructions is typically created by formulating a wide variety of tasks as text-to-text problems, distinguishing good/bad candidate outputs given an input text. This is done by combining datasets with natural language instructions and constructing positive and negative pairs for training.’
BLEURT Score	0.37323
Research	
Context Provided	’datasets with instructions across diverse task cate-\ngories and domains: Multitask Embeddings Data\nwith Instructions (MEDI).\nData\nConstruction\nWe\nbuild\nMEDI\nby\ncombining\n300\ndatasets\nfrom\nSuper-\nNaturalInstructions\n(super-NI;\nWang\net\nal.,\n2022b) with 30 datasets from existing collections\ndesigned for embedding training.\nThe super-NI datasets come with natural lan-\nguage instructions, but positive and negative pairs\nare not provided. We construct these pairs by using\n\nAbstract\nWe introduce INSTRUCTOR, a new method\nfor computing text embeddings given task in-\nstructions: every text input is embedded to-\ngether with instructions explaining the use case\n(e.g., task and domain descriptions). Unlike\nencoders from prior work that are more special-\nized, INSTRUCTOR is a single embedder that\ncan generate text embeddings tailored to differ-\nent downstream tasks and domains, without any\nfurther training. We first annotate instructions\nfor 330 diverse tasks and train INSTRUCTOR\n\nof (x, y+) and {(x, y−\ni)}k\ni=1. Further following Ni\net al. (2021), we compute the same loss with x\nand y swapped and add it to the previous loss (i.e.,\nbidirectional in-batch sampled loss).\n2.3\nMEDI: Multitask Embedding Data with\nInstructions\nThere are no existing datasets that consist of a\nvariety of tasks for embedding training with in-\nstructions. We thus construct a collection of 330\ndatasets with instructions across diverse task cate-\ngories and domains: Multitask Embeddings Data’
Document Sources	[’ArXiv’, ’ArXiv’, ’ArXiv’]
Generated Answer	’To create training data for embedding models that use task-specific instructions, several methods can be employed. One common approach is to manually annotate instructions for each task and then use this annotated data to train the embedding model. This can be a time-consuming and labor-intensive process, but it allows for greater control over the quality and relevance of the training data.Another approach is to use existing datasets that have been designed for embedding training, such as those from the super-NaturalInstructions (super-NI) collection. These datasets can be used to train the embedding model without the need for manual annotation, but they may not include task-specific instructions.In some cases, transfer learning may also be employed to create training data for embedding models that use task-specific instructions. This involves taking a pre-trained embedding model and fine-tuning it on a new task using a small amount of annotated data. This can be an effective approach when the new task is similar to the original task used to train the pre-trained model, but it may not be as effective when the new task is significantly different.Recent advancements in the field of embedding models have focused on developing methods that can generate text embeddings tailored to different downstream tasks and domains without the need for further training. One such method is INSTRUCTOR, which combines 300 datasets from Super-Natural’
Reference Answer	’To create training data for embedding models that use task-specific instructions, a common method is to combine datasets from different sources, such as the SuperNaturalInstructions dataset with existing collections designed for embedding training. The SuperNaturalInstructions dataset provides natural language instructions, which can be paired with positive and negative examples to form training samples. Additionally, for tasks like classification or similarity, training samples can be constructed by selecting text sequences associated with different classes or similarities. This diverse training data is essential for instruction-based finetuning, which enables the embedding model to learn from a wide range of tasks and domains.’
BLEURT Score	0.48057

Table 5: Sample Responses, Reference Answers and BLEURT Scores (Question 2 of 2), by Persona