JÖNKÖPING UNIVERSITY
School of Engineering

PersonaBOT: Bringing Customer Personas to Life with LLMs and RAG

Generating and Utilizing Customer Personas with LLMs and Retrieval-Augmented Generation

PAPER WITHIN *Computer Science*
AUTHORS: *Muhammed Rizwan*
TUTOR: *Lars Carlsson*
EXTERNAL SUPERVISOR: *Mohammad Loni*

JÖNKÖPING *May 2025*

This exam work has been carried out at the School of Engineering in Jönköping in the subject area Computer Science. The work is a part of the two-year Master of Science AI in Engineering programme. The authors take full responsibility for opinions, conclusions and findings presented.

Examiner:	Vladimir Tarasov
Supervisor:	Lars Carlsson
External Supervisor:	Mohammad Loni
Scope:	30 credits
Date:	2025-05-11

Mailing address:	Visiting address:	Phone:
Box 1026 551 11 Jönköping	Gjuterigatan 5	036-10 10 00 (vx)

Abstract

The introduction of Large Language Models (LLMs) has significantly transformed Natural Language Processing (NLP) applications by enabling more advanced analysis of customer personas. At Volvo Construction Equipment (VCE), customer personas are traditionally developed through qualitative methods, which are time-consuming and lack scalability. The main objective of this research is to generate synthetic customer personas and integrate them into a Retrieval-Augmented Generation (RAG) chatbot to support decision-making in business processes. The study utilizes the Design Science Research Methodology (DSRM) in three iterative cycles. The first iteration focuses on developing a persona-based RAG chatbot integrated with verified personas. In the second iteration, synthetic personas were generated using Few-Shot and Chain-of-Thought (CoT) prompting techniques and evaluated based on completeness, relevance, and consistency using McNemar's test. In the final iteration, the chatbot knowledge base is augmented with synthetic personas and additional segment information to assess improvements in response accuracy and practical utility. Key findings indicate that Few-Shot prompting outperformed CoT in generating more complete personas, while CoT demonstrated greater efficiency in terms of response time and token usage. After augmenting the knowledge base, the average accuracy rating of the chatbot increased from 5.88 to 6.42 on a 10-point scale, and 81.82% of participants found the updated system useful in business contexts.

Keywords: LLM, Prompt Engineering, Customer Persona, RAG

Acknowledgement

I would like to thank my examiner, Vladimir, for his seminar sessions on how to carry out the research project efficiently. I am also very grateful to my supervisors Mohammad Loni and Lars Carlsson for their continuous guidance and support throughout this research project. Without their guidance, I would not have been able to complete this thesis successfully.

A big thank you to Volvo Construction Equipment and the Future Solutions team for giving me the opportunity to work on this project. Special thanks to my manager Ulrich Fass, and to Bartlomiej Wojcik for his help with infrastructure support.

Finally, I am thankful to my parents and friends for their emotional and financial support during my Master's program. Their encouragement has kept me going through this wonderful journey.

Contents

List of Figures

List of Tables

Acronyms

NLP Natural language processing

LLM Large language model

VCE Volvo Construction Equipment

GPT Generative Pre-trained Transformer

RAG Retrieval-augmented generation

CoT Chain-of-Thought

DL Deep learning

NN Neural network

RNN Recurrent neural network

LSTM Long short-term memory

GRUs Gated recurrent units

AI Artificial intelligence

QAI Quantum artificial intelligence

GAN Generative Adversarial Nets

VAE Variational autoencoder

LDA Latent Dirichlet Allocation

STM Structural Topic Models

BPE Byte Pair Encoding

RLHF Reinforcement learning from human feedback

RnD Research & Development

IT Information Technology

1 Introduction

A significant breakthrough in Natural language processing (NLP) has been brought by the rise of LLMs. LLMs are often referred to as transformer-based models or next-generation language models. These models use transformer architecture, a deep learning technique to learn and understand the complex patterns and structures in the language. These models are widely used today across various domains, including virtual assistance, text generation, and information extraction [1]. Businesses use the capabilities of these models in diverse functions such as strategic planning, customer opinion analysis, and targeted marketing. This allows them to better understand their customers and maintain a competitive edge [2].

1.1 Context and Motivation

Innovation, customer support, customer-centric product development are the backbone of Volvo Construction Equipment (VCE). Customer persona is one of the ways that help VCE understand its customers. These personas allow VCE to understand the unique needs, expectations, behaviors, and motivations of distinct customer segments. The traditional way of creating personas relied on qualitative methods, such as direct interviews, surveys, and manual observational studies [3]. While these traditional approaches can offer deep insights, they are resource-intensive and time-consuming. There have been studies that explored data-driven methodologies for persona development. This includes the use of statistical and machine-learning techniques to improve efficiency and reliability [4]–[7]. Although these data-driven methods have shown promise, they still face limitations in contextual adaptability, real-time updating capabilities, and in capturing complex customer behaviors from unstructured textual data [6].

The introduction of LLMs such as Generative Pre-trained Transformer (GPT) [8] has brought an alternative method of generating structured, accurate, and detailed personas from large textual datasets [3], [9]–[11]. High-quality personas are those that effectively capture relevant, accurate, and actionable customer information. Well-designed prompts guide the model to produce better outputs [12]. In this study, the desired output is the generated persona. Another important reason for identifying the most effective prompting technique is the cost of token usage as most of the commercial LLMs are billed per token. Selecting an optimal prompting method can help reduce resource consumption and save money, especially if a prompting method that uses fewer prompts is more effective [13]. The reviewed studies on persona generation relied on using a single prompting technique without systematically evaluating and comparing different prompting methods. This leaves a notable research gap in determining the most effective prompting strategies.

Furthermore, merely generating personas that can capture customer information is not sufficient. Businesses also require a system that can extract meaningful and actionable insights from personas. Integrating personas with retrieval systems such as RAG [14] that can assist in querying, analysis, and utilization of persona data within day-to-day business operations remains another unexplored path. Addressing these gaps presents an opportunity to significantly improve how VCE and similar organizations use LLM for personas, thereby improving their responsiveness to evolving consumer needs.

1.2 Problem Statement

Customer personas play a crucial role in supporting business decisions by providing insights into customer needs, behaviors, and expectations. At VCE, customer personas are developed by the Customer Experience team through direct interviews with customers. These personas are then written up and published on an internal platform. From these platforms, they can be accessed by stakeholders such as marketing, sales, engineering, and leadership teams.

However, the current process of creating and using these personas presents several challenges. When the number of personas increases, maintaining them becomes increasingly time-consuming and labor-intensive. Additionally, the lack of a dedicated system for querying or interacting with persona data means that stakeholders must search through each persona manually to extract relevant insights. This approach is impractical as the number of customers grows. As a result, it becomes difficult for teams to rapidly understand customer segments and use those insights into the Research & Development (RnD) and business processes.

LLM offers a promising opportunity to automate persona generation [10] by extracting structured insights from unstructured text sources such as customer success stories. However, the existing studies in persona generation focused on using a single prompting method [3], [9], [10], [15], with no comparative analysis of how different techniques such as few-shot [16], or Chain-of-Thought (CoT) prompting [17] affect the quality of the generated personas. Additionally, the potential of integrating personas into interactive systems, such as RAG chatbots, is unexplored.

This thesis addresses these gaps by investigating how LLMs can be used to generate high-quality customer personas using different prompting strategies, and by developing a proof-of-concept chatbot that enables exploration of persona information with ease. The ultimate goal is to build a system that enables VCE to better understand its customers and make faster, more informed business decisions.

To accommodate these requirements, following research questions have been defined in this thesis:

Research Question 1: How effective are customer persona-based chatbots in enhancing decision-making and automating customer-facing processes within the construction industry?

Research Question 2: How to generate synthetic customer personas from public information, e.g., customer success stories, and which prompting techniques yield the most accurate and useful results?

Research Question 3: How effective is augmenting a chatbot's knowledge base with synthetic personas and segment-specific information in improving its performance?

1.3 Research Objective

The objective of this research is to explore the development of a customer persona-based chatbot that can support decision-making and improve business processes within the construction industry. The study also focuses on generating synthetic personas from publicly available information and comparing different prompting techniques. Furthermore, this work also investigates how augmenting the chatbot's knowledge base with these synthetic personas and additional domain-specific data can improve its overall performance and practical usability.

1.4 Scope of the Research

The scope of this research is defined by several key factors and limitations. Primarily, the verified customer personas used for the chatbot's knowledge base are limited in number. For the generation of synthetic personas, only customer success stories were considered as the source of information. The segment information was restricted to mining, aggregates, and quarrying. Information related to other segments was not explored in this study.

The evaluation of the chatbot and the comparison of prompting methods were conducted with a small group of evaluators, so the findings may not apply to larger groups of users. Additionally, only a subset of synthetic personas were selected for evaluation to manage the time constraints. The persona evaluation

was subjective, as each evaluator provided feedback based on personal interpretation.

Furthermore, the study utilized GPT-4o Mini as the LLM for the conversational system and generating synthetic personas . Other LLMs were not explored, which may have produced different results.

1.5 Thesis Structure

This thesis report is structured into six chapters. Chapter 1 introduces the context, motivation, problem statement, and research objectives. Chapter 2 provides the necessary background to understand the key concepts and frameworks used in this study. Chapter 3 reviews the relevant literature for the research topic. Chapter 4 discusses the methodology and the implementation process followed. Chapter 5 discusses the results and analyzes the findings of the research objectives. Finally, Chapter 6 provides the conclusions and areas for future research.

2 Background

This section explains various important concepts and background information needed to understand this research study.

2.1 Natural Language Processing (NLP)

NLP is a subfield of Artificial intelligence (AI) that involves the use of algorithms, models, and various computational techniques to analyze, process, and generate natural language data, including speech and text. NLP helps computers interact with humans in a more natural way, which has become increasingly important as more human-computer interactions take place. [18]. Common applications of NLP include information retrieval, question answering, text summarization, machine translation, chatbots, virtual assistants, text classification, and spam detection.[18], [19].

The field of NLP originated in the 1950s and has since evolved significantly [20]. The earlier approaches relied on rule-based systems that used predefined grammar and symbolic logic to process language. Over time, statistical models introduced probabilistic reasoning and replaced rule-based methods for tasks like machine translation and speech recognition. However, these methods faced significant challenges in handling linguistic complexities such as context and ambiguity [20], [21].

Despite significant progress, NLP faces several ongoing challenges [22]. Language is highly ambiguous, it contains slang with unusual meanings and social contexts. Additionaly, NLP task becomes more complicated when we take the accent into account, as the accents varies according to region where the people are from [19], [20]. The limitations of traditional approaches have driven the evolution of NLP towards deep learning[23]. Deep learning (DL) models, powered by Neural network (NN)s, introduced significant improvements in scalability, context understanding, and task-specific adaptability. The next section explores how deep learning revolutionized NLP and laid the foundation for transformer-based architectures.

2.2 Deep Learning in NLP

Advances in computational power and the greater availability of big data have made DL to be one of the most interesting approaches in the NLP domain. DL involves using deep neural networks on large

datasets to learn how to perform a specific task. This task can range from simple classification to complex reasoning [24]. Unlike traditional rule-based methods that rely on predefined rules or statistical techniques that use probabilistic reasoning, DL uses multiple hierarchical neural architectures to achieve state-of-the-art results in many NLP domains [25], [26].

Recurrent neural network (RNN)[27] introduced a milestone in NLP. RNNs use the idea of processing sequential information. The term "recurrent" is used because these models repeat the same computation for each token in a sequence, with each step relying on the outcomes of the previous computations[26]. Even though being successful in many NLP tasks like language modeling and sequence prediction, training RNNs is challenging due to issues such as the vanishing gradient problem [26], [28].

To address the challenges of RNNs, Long short-term memory (LSTM) networks [29] and Gated recurrent units (GRUs)[30] were developed. In LSTMs, the recurrent nodes are made up of many individual neurons connected in a manner designed to retain, forget, or expose specific information using three types of gates: input, forget, and output gates. Sometime, it is important to retain information from the distant past, while at the same time, other very recent information may not be important. By using LSTM blocks, the information that are important can be retained much longer, while information that are unnecessary can be forgotten. These types of blocks are very efficient at capturing long-term dependencies. Due to this functionality, LSTMs have produced good results in applications such as sentiment analysis and machine translation, where understanding context over long text spans is crucial. GRUs are another variant of the LSTM that has been shown to perform well or better than standard LSTMs in many tasks [24]–[26], [31].

Even though models such as LSTMs and GRUs were able to improve the handling of sequential data, their inherent sequential nature caused computational inefficiencies. This makes them less suitable for large-scale parallelization [32]. Additionally, their mechanism to handle long-term dependencies often struggled to capture complex relationships in sequences, especially when tokens or words were separated by long distances [25], [26], [32]. These challenges made way for the development of Transformer architectures highlighted the need for approaches to sequence processing, and made way for the development of transformer architecture. A groundbreaking model that can operate with greater parallelization and better capture complex dependencies [24]–[26]. The next section discusses about transformers and their impact on NLP.

2.3 Transformers Architecture

The Transformer architecture was introduced by Vaswani et al.[33] at Google in 2017. Since then, it has become the dominant model for various NLP tasks. Unlike earlier models such as RNNs and LSTMs, Transformers rely entirely on a self-attention mechanism, which eliminates the need for recurrence or convolutions. This innovation allows the model to process sequences in parallel, significantly reducing training time and improving computational efficiency. Self-attention is the core of the Transformer architecture. It enables the model to focus on different parts of the input sequence, capturing relationships between words regardless of their distance. Attention is calculated using the equation 1.

$$\text{Attention}(Q,K,V) = \text{softmax}\left(\frac{QK^T}{\sqrt{d_k}}\right)V \tag{1}$$

where Q, K, and V represent the query, key, and value matrices derived from the input, and d_k is the dimensionality of the key vectors. The dot product between the query and key matrices determines the attention scores, which are normalized into probabilities using the softmax function.

The Transformer follows an encoder-decoder architecture. The encoder maps an input sequence of tokens $(x_1,\ldots,x_n)$ into continuous representations $z = (z_1,\ldots,z_n)$, while the decoder generates the output of the

sequence $(y_1, \ldots, y_n)$ one at a time. At each step, the model is auto-regressive, consuming the previously generated symbols as additional input when generating the next. Both the encoder and decoder consist of stacked layers incorporating multi-head self-attention mechanisms, point-wise feed-forward networks, residual connections, and layer normalization. These components work together to stabilize training and enhance the model's ability to learn complex patterns.

Transformers offer significant advantages over earlier architectures [33]. By processing sequences in parallel, they eliminate the inefficiencies of RNNs, making them computationally scalable. The self-attention mechanism allows capturing of long-range dependencies effectively. This overcomes the limitations of previous models in handling context over distant tokens. Figure 1 is an illustration of the Transformer architecture.

2.4 Large Language Models (LLMs)

LLMs are pre-trained language models that are developed using deep learning techniques, especially the transformer architecture. These models are trained on large amounts of text data allowing them to handle many different language tasks with high accuracy. LLMs are trained with millions or even billions of parameters, which helps them generate clear and relevant text, such as answers, stories, or summaries. These abilities are very useful in areas like chatbots, content creation, and extracting important information from text [34]. Even though LLMs have given remarkable contributions to various domains, they also bring some significant limitations and challenges. Some of these challenges include biased data, limitations in reasoning ability, the need for vast amounts of data and computational resources, limited generalizability, and hallucination[35].

Based on the availability of their source code, LLMs can be divided into two categories: closed-source and open-source [34]. Closed-source LLMs are typically developed by major tech companies such as OpenAI[1], Microsoft[2], and Alphabet[3]. These models often have over a trillion parameters and are capable of handling multiple coding and natural languages. On the other hand, open-source LLMs are usually developed by research institutions or companies such as Meta[4], Huawei[5], Stanford[6], and Tsinghua[7]. These models generally have fewer parameters compared to the leading closed-source LLMs [36]. There has been a lot of ongoing research in the field, and new models are being released frequently. Figure 2 illustrates a timeline of LLM releases. The blue boxes represent pre-trained models, while orange boxes represents instruction-tuned models. Models on the upper half shows open-source models, whereas those on the bottom are closed-source. This timeline highlights the evolving landscape and trends in LLM research.

In this study, a closed-source LLM from OpenAI, GPT-4o Mini ("o" stands for "omni") was used. This model is fast, cost-effective, and lightweight that is well-suited for focused tasks. It accepts both text and image input and generates textual outputs, including structured formats. The model is ideal for tasks that require fine-tuning or producing outputs similar to larger models with low cost and latency. The context window of 128,000 tokens allows handling of long and complex inputs [38].

[1] https://openai.com/
[2] https://www.microsoft.com/sv-se/
[3] https://abc.xyz/
[4] https://ai.meta.com/
[5] https://www.huaweicentral.com/huawei-cloud-unveils-pangu-large-model-5-0/
[6] https://crfm.stanford.edu/2023/03/13/alpaca.html
[7] https://iias.tsinghua.edu.cn/en/

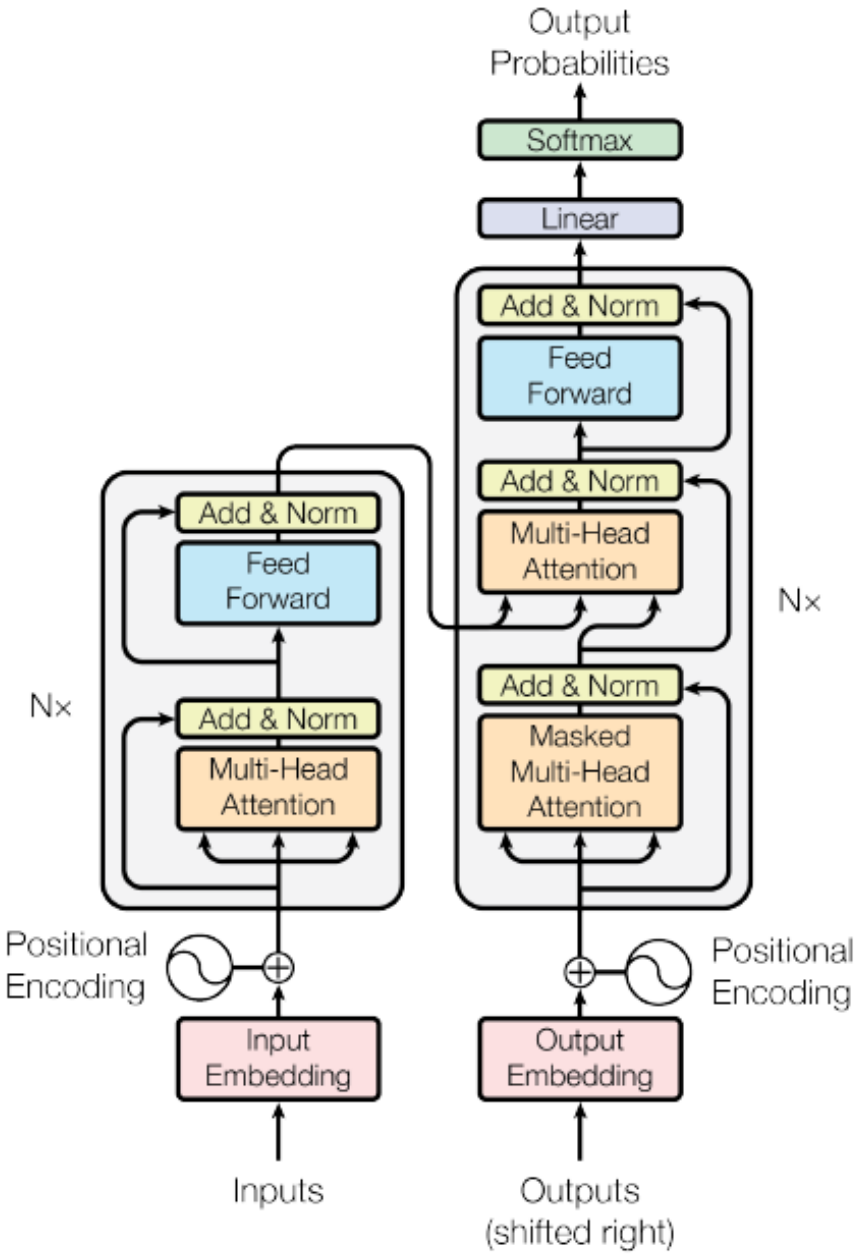

Figure 1. The Transformer - model architecture. Reproduced from [33].

2.4.1 Tokenization and Embedding

To understand the working of LLM, it is also important to understand tokenization and embedding concepts. Tokenization is a key step involved in both training LLMs and processing the inputs that they receive. It involves breaking down a text into smaller, non-decomposable units called tokens. These tokens can be characters, subwords, symbols, or words, depending on the tokenization technique and the model being used. Common tokenization methods in LLMs include WordPiece, Byte Pair Encoding (BPE), and UnigramLM[37].

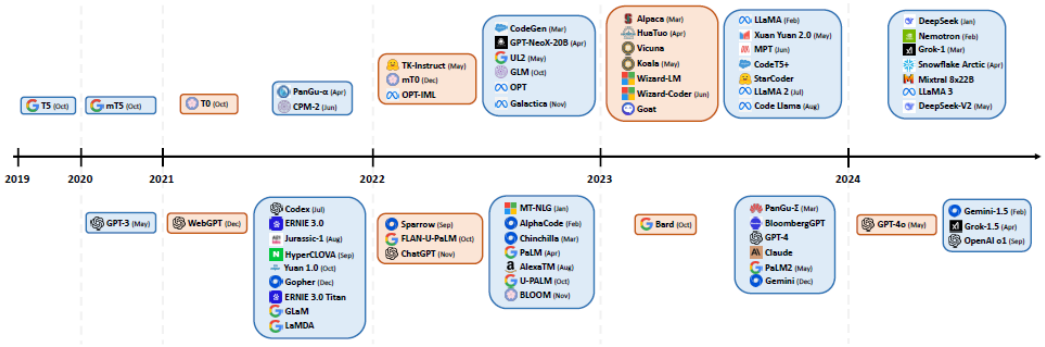

Figure 2. Development timeline of LLM releases. Reproduced from [37]

.

For example, models like GPT-4o and GPT-4o mini converts a sentence such as "I am building an AI application for Customer Persona" which has 52 characters, into 9 tokens. As a general rule of thumb, one token corresponds to approximately four characters in English. Therefore, we could say that 100 tokens would be around 75 words [39]. Figure 3 is an illustration of how the example sentence is converted into tokens.

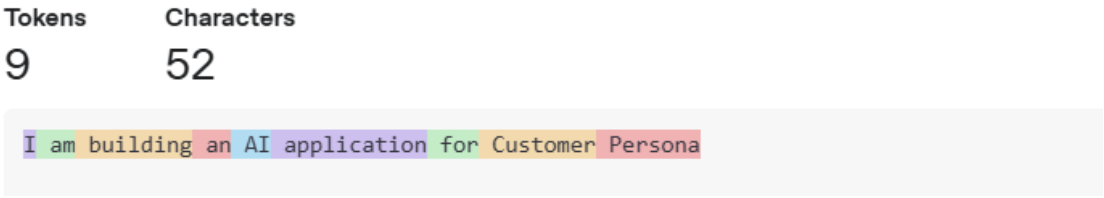

Figure 3. Conversion of a sentence into tokens. Adapted from [39].

Embeddings are numerical (vector) representations of texts. This numerical representation allows the model to understand and reason about language. Each LLM typically has its own embedding model. For example: Mistral provides its own embedding model called mistral-embed [40] and OpenAI offers models like text-embedding-ada-002 and text-search-davinci-001. In this study, text-embedding-ada-002 was used. This is one of the latest embedding models by OpenAI. It was selected due to its strong performance, smaller size, and lower cost compared to the earlier models [41].

Figure 4 is an illustration of the process involved in the embedding process.

To summarize this section, the workflow of an LLM generally involves the following steps:

1. **Tokenization:** The input text is broken into tokens using a tokenizer. Each token is assigned a unique ID.
2. **Embedding and Encoding:** These tokens are then passed through the embedding model, where an embedding layer converts them into vectors. These vectors are processed by transformer blocks to capture contextual meaning.
3. **Decoding:** In the final step, the output tokens are detokenized back into human-readable text by mapping token IDs to words using the tokenizer's vocabulary[42].

Figure 5 shows the overall workflow of an LLM, from raw input text to the final output.

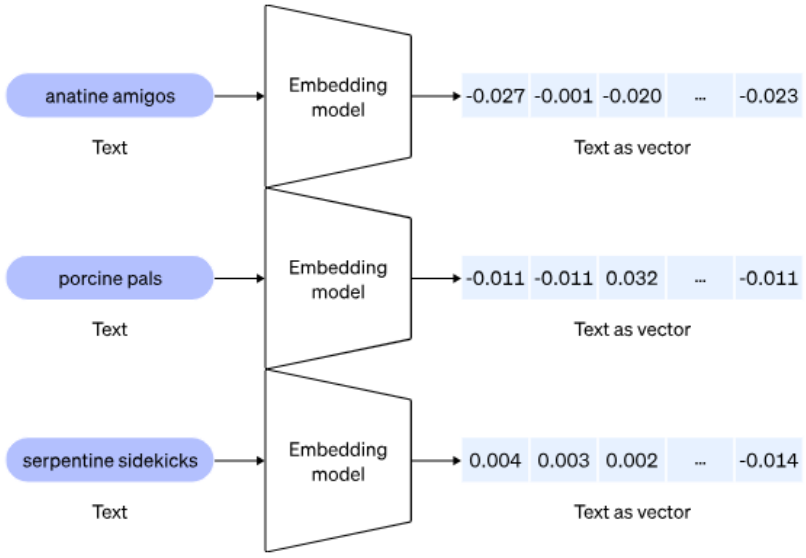

Figure 4. Embedding process. Reproduced from [41].

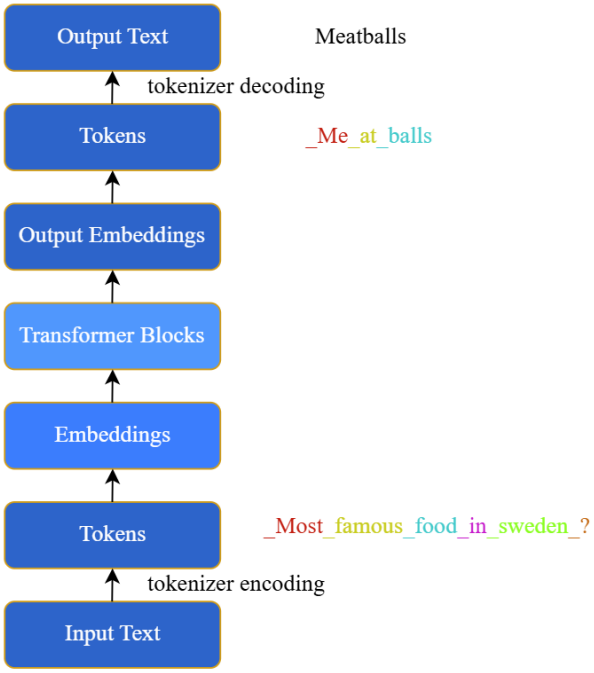

Figure 5. Overall LLM process. Adapted from [42].

2.5 RAG

Although LLMs have shown remarkable success in different NLP tasks, they face limitations, especially in domain specific or knowledge intensive tasks [43]. The hallucination problem is one of the major challenges of LLMs. Hallucination refers to the tendency of LLMs to generate responses that sound correct but are actually inaccurate. Another issue is staying updated with new knowledge. To incorporate

new information, LLMs must be retrained or fine-tuned, which is costly. Finally, general-purpose LLMs often lack expertise in specific domains, making them less effective for specialized tasks [43], [44]. These challenges raised are problematic in applications like educational chatbots, medical diagnoses, or customer service software [45]. To address these challenges, a technique named RAG was introduced by [14]. This technique combines an external knowledge base with LLMs to generate more accurate responses. On integrating LLMs with relevant, factual information, the hallucination problem can be reduced. By updating with the external knowledge database, the knowledge update issue can also be solved. This will ensure that LLMs have access to up-to-date information. Additionally, this technique can turn a general LLM into a domain-specific one by using a specialized knowledge database. As a result, RAG helps make LLMs more accurate, knowledgeable, and reliable across various applications [44].

2.5.1 Components of RAG

RAG consists of three main components: the Retriever, Augmentation (or Retrieval Fusion), and the Generator. The retriever module is responsible for fetching relevant information from an external knowledge base. The augmentation process integrates this retrieved knowledge to improve the generation. The generator processes the augmented input to generate responses. These components work together to improve the accuracy and relevance of the output in RAG-based models [44].

2.5.2 Workflow of RAG

The process begins with Indexing, where the data is pre-processed. Raw data in various formats (such as PDFs, Word documents, etc.) are first cleaned and converted into a uniform plain text format. It is then broken down into smaller pieces called chunks. These chunks are then converted to vector representations using an embedding model and are stored in a vector database.

When a user submits a query to an LLM, it is converted into a vector representation, also known as an embedding, using the same embedding model. The query in the embedding format is then used to search within the vector database to retrieve relevant information. The user query and the stored chunks are compared using a similarity technique to find the most relevant chunks of information from the database. Once the relevant chunks are retrieved, they are processed by the LLM to produce a response. The model takes the user query and the retrieved context and generates an answer by combining them to provide a coherent and informative response. Figure 6 below is a pictorial representation of the RAG process.

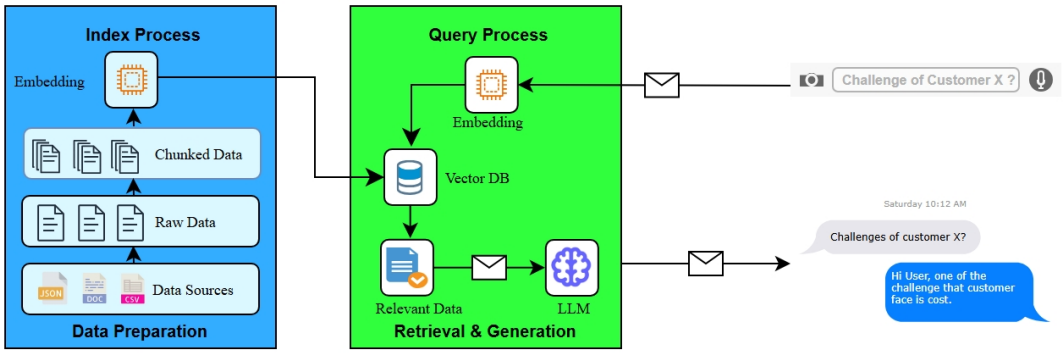

Figure 6. Overview of the RAG process.

2.6 Prompt Engineering

Prompt engineering refers to the technique of designing, refining, and optimizing input prompts for LLMs to obtain desired responses. It plays a crucial role in ensuring accurate, relevant, and coherent outputs from the model. A well-constructed prompt significantly enhances the quality and relevance of generated responses, whereas poorly structured prompts often lead to incorrect or unsatisfactory outputs [46]. While various types of prompt engineering exist, this study specifically explores few-shot prompting, and CoT prompting, which are detailed below:

- **Few-shot prompting:** Few-shot prompting is a technique in which a model is provided with a small number of input-output examples to improve its understanding of a specific task. Compared to zero-shot prompting, which does not include examples, few-shot prompting generally improved output quality, particularly for complex tasks. However, few-shot prompting needs additional tokens to include these examples, which may become concern for longer text inputs [47].
- **CoT prompting:** In complex reasoning scenarios, LLMs often struggle to produce accurate responses. This limits their potential capabilities. To address this, CoT prompting was introduced [47]. This technique helps the model break down problems into smaller reasoning steps. It can be done by using simple prompts like "Think in step by step way" or by showing examples that include both a question and a step-by-step explanation leading to the answer [48].

2.7 Microsoft Azure and AI Services

Microsoft Azure is a the cloud computing platform that is developed by Microsoft. It offers more than 600 services, some of this includes data management, identity compute service, IoT and AI etc. Within Azure, the Azure AI package consists of prebuilt APIs and SDKs that allow developers to consume and build enterprise AI applications. Azure AI offers a range of services, including speech recognition, speaker recognition, face recognition, computer vision, form understanding, natural language processing, and machine learning [49].

One of the key Azure service used in this study is Azure AI Search. It is an information retrieval system that can be used for indexing and retrieving the structured and unstructured content from the search index. It supports various types of searches like vector search, semantic search and hybrid search [50]. Another key service used in this thesis is Azure AI Foundry. It is a recently released platform that is part of Azure AI. AI foundry supports a wide range of models and services to build, explore and test generative AI applications [51]. This thesis, utilizes both Azure AI Foundry and AI search to generate personas and to build the conservation system to query persona data.

2.8 Understanding Customer Persona

Customer personas can be defined as semi-fictional representations of different users or customer groups that are created to improve understanding of consumer behaviour and preferences in marketing [52]. A persona typically consists of (1) demographic data, such as name, age, gender, and education, (2) characteristic data, such as profession, health, and personal values and (3) behavioral data, such as usage habits and buying preferences [53]. Figure 7 is a sample persona illustrating attributes of a customer, such as age, occupation, goals, challenges, and more. Personas are used in many domains and applications such as sharing educational messages in healthcare and creating narratives for patients [54], as well as in gaming, software development, news and media, and marketing [55]. In the business context, they play a crucial role in helping understand their customers better and develop strategies tailored to meet their unique needs [56].

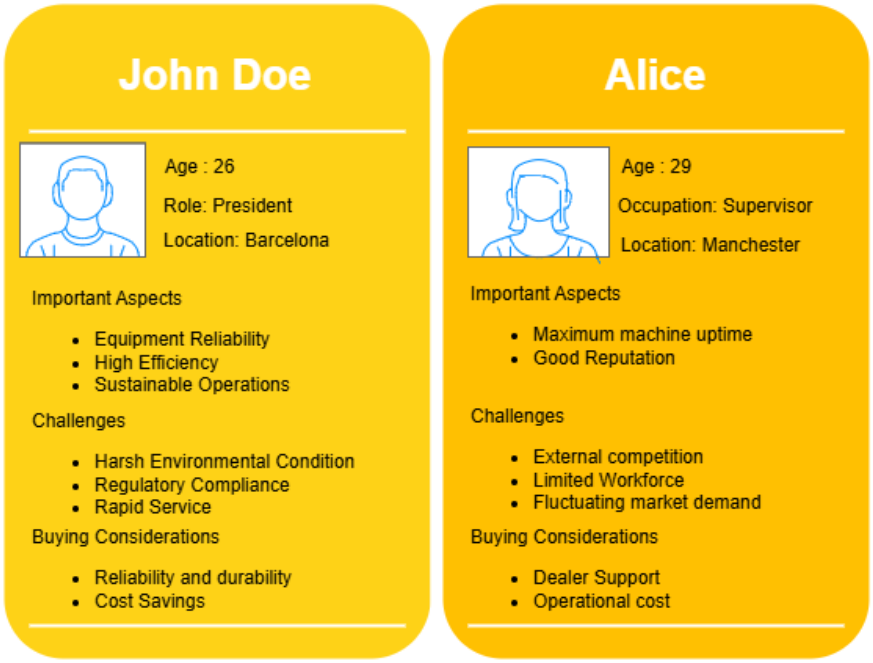

Figure 7. Example of Customer Persona

However, traditional methods of persona creation that are depended on qualitative analysis of interview, survey data, and data-driven approach [3]–[5] are often time-consuming, expensive, and prone to subjectivity. Advancements in NLP and LLMs have enabled the generation of customer personas from various sources, such as interviews and forums [3], [9]. However, generating personas alone is not enough. Businesses also need efficient ways to analyze and interact with these personas dynamically to gain actionable insights. In industries like construction equipment manufacturing, the customer needs vary significantly across global markets. Generating and analyzing customer personas efficiently leads to success in business. The next section introduces VCE, the company collaborating with this thesis, and describes the challenges it faces and how this study aims to address them.

2.9 About Volvo Construction Equipment (VCE)

VCE[8] is one of the world's largest companies which develops, manufactures, and markets equipment for construction and related industries. It was founded in the mid-1800s in Eskilstuna, Sweden. It is a subsidiary of the Volvo Group [9] and has production facilities all over the world including Sweden, France, Belgium, Germany, the UK, the USA, Brazil, India, China, and Korea. Its products include a range of wheel loaders, hydraulic excavators, articulated haulers, motor graders, soil and asphalt compactors, pavers, backhoe loaders, skid steers, and milling machines. This thesis is carried out in collaboration with Future Solutions - a team within VCE that works on various innovative AI solutions.

To stay competitive and adapt rapidly to changing customer demands, VCE needs a more efficient way to understand customer challenges and preferences. Gathering consistent feedback, generating personas, and using them to derive insights can be both time-consuming and prone to error since customers in each region have their own regulations and working conditions. This thesis will help VCE gain a deeper understanding of global customer needs, make faster decisions, and deliver customer-centric products.

[8] https://www.volvoce.com/
[9] https://www.volvogroup.com/se/

Thereby maintaining its reputation and leadership in the construction equipment industry.

3 Related Work

3.1 Non-LLM Approaches for Creating Customer Personas

The traditional method of creating personas depended on qualitative data, such as interviews, observations, and survey data from target users [3]. There have been researches that explored data-driven approaches that improved efficiency, scalability, and reliability in creating personas. One such approach was introduced by McGinn et al. [4], where a survey was sent over to 1300 users. An exploratory factor analysis, a data reduction technique, was performed on the survey result. This analysis helped identify the groups based on the tasks performed. Stakeholders were involved throughout this process to ensure the relevance of personas. Instead of relying on survey data or user interviews, Zhang et al.[5] followed a two-step statistical machine-learning approach to create personas only based on user behavior. In the first step, they analyzed 3.5 million clicks from 2400 users and clustered them into a common workflow using hierarchical clustering. In the second step, a mixed statistical model was used to create five personas.

Jung et al.[6] introduced Automatic Persona Generation (APG), a system that creates personas from real-time social media interactions on platforms like Facebook and YouTube. They processed tens of millions of interactions using non-negative matrix factorization. This automatically generated realistic and up-to-date personas from large-scale social media data. Similarly, Farseev et al.[7] introduced a framework named SOMONITOR that used X-Mean clustering with ADA embeddings to extract customer persona from digital marketing content. Unlike previous studies that relied on survey data or behavioral data, SOMONITOR clusters advertising content into distinct persona groups based on customer needs, interests, and aspirations. While these data-driven methods significantly improve persona creation, advancements in LLM offer further opportunities for automating and improving persona development.

3.2 LLM Approaches of Creating Personas

LLMs ability to generate structured text based on the input text provided using advanced natural language processing capabilities makes them a strong candidate for persona creation. This section reviews various approaches that utilized LLMs for persona creation.

One of the methods used to create personas is by providing LLM with structured prompts. This methodology was used in [10] to create 450 personas. In this study, they utilized these generated personas and investigated the bias and diversity in them. Their findings indicated that LLMs can create informative and relatable personas, but they exhibit a strong bias from specific countries. Similarly, Zhang et al. [11] introduced PersonaGen, a tool that used GPT-4 [57] along with knowledge graphs to refine persona generation. The tool was developed to assist the agile software development process. The GPT-4 model analyzed the user feedback provided and generated high-quality and detailed persona content. This content was then used by the knowledge graphs to create personas. PersonaGen demonstrated that it improved accuracy in capturing user needs compared to independent human analysis. Although challenges remain in analyzing non-functional requirements.

Another method for persona generation using LLMs involves the use of thematic analysis. De Paoli et al. [9] proposed a workflow where LLMs analyze qualitative interview data to generate personas. This approach follows a structured methodology where LLMs first generate codes (such as behaviors, goals, etc.) in textual format. From these codes, emerging themes are identified. These themes, along with prompts, are then used to construct persona narratives. The advantage of this method lies in its ability to

extract meaningful user traits from raw interview data without predefined coding schemes. An extension of this approach is found in Persona-L [3], a system that integrates LLMs with a RAG framework. By using specific types of datasets, this system enhances persona realism while addressing biases commonly found in LLM-generated content. The system was tested in creating personas that represent individuals with complex needs. This study demonstrated that incorporating external data can improve both the diversity and contextual accuracy of the generated personas.

Beyond structured prompting and thematic analysis, there have been a study that used human-AI collaboration in persona generation. Goel et al. [15] conducted an exploratory study where novice designers used GPT-3 [16] to create personas through iterative refinement. The study found that personas generated with GPT-3 were comparable to those created manually, particularly when designers provided detailed prompts and engaged in multiple iterations. However, the study also highlighted challenges such as generic responses, inconsistencies, and stereotypical outputs. This makes it necessary in human intervention to refine and personalize the generated personas.

3.3 Non-LLM Approaches for Analyzing and Leveraging Customer Personas

There have been various techniques to analyze and utilize customer persona before the emergence of LLMs. These methods were based on statistical methods [58] and machine learning [59], [60] to extract insights from customer data. This section reviews these approaches, limitations, and the reasons for the shift towards using LLM-based methods.

One such approach is the use of Quantum artificial intelligence (QAI). QAI combines quantum computing and AI to process large datasets in parallel. This allows obtaining real-time updates of customer profiles in response to dynamic behaviors and preferences. A study by More et al. [59] discussed how QAI can improve sentiment analysis and predictive modeling using quantum machine learning. This methodology improves customer segmentation, recommendation engines, and consumer behavior prediction. Even though QAI is promising, it remains in the early stages of adoption, and its implementation is challenging due to limited computational feasibility and hardware availability.

Generative AI techniques, such as Generative Adversarial Nets (GAN)s [61] and Variational autoencoder (VAE)s [62], have been explored for improving marketing applications using personas. Morande and Amini in their study [58] demonstrated that GANs and VAEs can improve customer profiling in social media marketing by generating personalized product recommendations and marketing content. Their study found that generated content was able to significantly improve customer engagement, loyalty, and sales. However, generative models cause risks related to algorithmic bias, ethical concerns about privacy, and the chances for misleading or made-up customer insights [58].

Another approach includes utilizing Bayesian probabilistic models such as Latent Dirichlet Allocation (LDA) [63] and Structural Topic Models (STM) [64]. These models categorize textual data into topics based on word co-occurrence patterns, helping in customer segmentation and persona identification. However, Bayesian models are frequency-based models that rely on word frequency distribution. They struggle to capture the complicated and nuanced elements in the textual data. This is due to the lack of an attention mechanism, a key feature of modern transformer-based LLMs [60]. The challenges discussed above in [58]–[60] have led researchers and businesses to adopt LLMs, as these models show promise in contextual understanding and adaptability.

3.4 LLM Approaches for Analyzing and Leveraging Customer Personas

This section reviews studies that utilized LLMs for personas, such as (i) persona interpretation, (ii) personalization, (iii) role-playing techniques, (iv) investigating bias and stereotypes, and (v) business insights.

(i) Persona Interpretation: LLMs are built upon large and diverse datasets, enabling them to interpret and generate user personas with high precision. Unlike traditional methods that need structured datasets and predefined heuristics, LLMs can extract persona-related attributes from conversational data, social media posts, and customer feedback. This information can then be used to understand the needs, motivations, and goals of the specific user. [65] examines how LLM interprets culturally specific personas, focusing on the Indian context. The research conducted both quantitative and qualitative analyses to assess how well LLMs understood personas within cultural contexts. The study revealed that LLMs exhibit high consistency and completeness in persona evaluation, but they struggle with credibility.

(ii) Personalization: One of the notable advancements in LLM-driven persona development is personalization. Zhang et al. [66] provides a detailed survey of how LLMs can be personalized. They propose a taxonomy of personalization levels in three categories: user-level, persona-level, and global preferences. This study highlights how techniques like RAG and prompt engineering can be used to tailor responses to user-specific needs.

There have been studies that have explored how LLMs can be customized for personalized interaction. One such application is CloChat [67], which allows users to tailor personas for various contexts and tasks. The end user of this application can choose to define personas by altering attributes like conversational style, emotions, areas of interest, and visual representations, thereby making interactions more human-like and relevant. To assess CloChat's effectiveness, researchers conducted surveys and in-depth interviews by comparing it with ChatGPT. The findings indicated that CloChat significantly improves user engagement, trust, and emotional connection when compared with ChatGPT.

(iii) LLM Role-playing: Personas can be integrated with LLMs through two approaches: LLM Role-Playing and LLM Personalization. In LLM role-playing, LLMs are assigned personas (roles) and they adapt to specific environments and tasks. Whereas in LLM personalization, LLM is adapted to user-specific personas for customized responses. The techniques used in Role-Playing are prompt engineering, multi-agent frameworks, and emergent behaviors in specific domains. In the personalization techniques, user data is integrated by Reinforcement learning from human feedback (RLHF), fine-tuning, and memory mechanisms. This study highlights several challenges associated with role-playing personas, including limited contextual understanding, the need for manual persona creation, and the static nature of personas, which prevents them from adapting to dynamic tasks [68]. To address these challenges [69] proposes a pattern language for persona-based interactions. This pattern language contains a series of patterns, where each pattern identifies a specific problem and provides its associated solution in the form of a template. In this study, seven person-related patterns are introduced, which improved realism, adaptability, and specificity in LLM interactions, making them more effective for complex and evolving tasks.

(iv) Investigate bias and stereotypes: While persona-based LLMs improve customer engagement, they also can introduce biases and stereotypes which may affect customer insights and segmentations. Cheng et al. [70] introduced Marked Persona, a prompt-based framework that captures the patterns and stereotypes across the LLM outputs. Their study used GPT-3.5 and GPT-4 to generate personas across various demographic groups and analyzed how this output is different from human-written personas. The findings reveal that personas generated by LLM contain more stereotypes than the personas written by humans. These biases are a challenge for LLM driven customer analysis, as they can provide inaccurate customer information.

(v) Business insights: Understanding customer preferences and requirements has become important for businesses.Extracting and analyzing customer data manually is often a difficult and time-consuming task. Barandoni et al. [71] evaluate the ability of proprietary and open-source models, such as GPT-4, Gemini, and Mistral 7B, to extract customer needs from TripAdvisor forum posts. This study systematically compared two prompting techniques, such as CoT using various proprietary and open-source LLMs for customer needs extraction. However, the focus was on extracting short customer needs from forum posts, not on generating structured personas. Additionally, the study did not explore fine-tuning techniques,

which could have further improved the model's performance. In contrast, [60] used fine-tuning on different models to identify topics, emotions, and sentiments from TripAdvisor customer reviews. This study provides an alternative technique for improving LLM-driven customer insights extraction.

While persona-based LLMs help businesses, their ability to understand persona and generate meaningful insights needs to be researched. Jiang et al. [72], through a case study, investigate whether LLMs can generate content similar to assigned personas by simulating different personalities using the Big Five personality model. Their findings demonstrate that LLMs can adjust their output to match the behavior of assigned personas.

3.5 Positioning of this Research in the Context of Related Work

The reviewed literature from Sections 3.1 to 3.4 highlights major advancements in persona creation and usage. This includes both traditional methods that do not use LLMs and newer methods that use LLMs. However, some limitations make it difficult to apply these methods in real-world businesses, such as the construction equipment manufacturing industry.

While studies such as [71] have compared prompting methods such as few-shot and CoT reasoning for tasks like customer needs extraction, they did not focus on structured persona generation. Moreover, studies on persona generation such as [9], [10], [15] typically relied on a single prompting method without comparing multiple approaches. This brings a gap in understanding which prompting method works best when generating personas from qualitative data, such as customer success stories. Existing studies on using personas mainly focus on general use cases, such as how LLMs understand personas [65], identifying stereotypes in LLM responses [70], and the use of personas to improve personalization [67]. However, there is limited research on how customer personas can support businesses where the customer attributes, such as challenges and needs, vary widely. Finally, the reviewed studies do not explore the integration of customer personas with retrieval systems like RAG. The authors of [60] mention in their limitations and future directions that techniques like RAG could be beneficial for retrieving consumer data. However, the use of RAG for persona-based analysis and insight generation remains unexplored, especially in helping RnD engineers and stakeholders to interact efficiently with persona data.

This thesis addresses the identified limitations in existing studies by systematically comparing and evaluating different prompting methods using specific metrics. The evaluation will help determine the most effective prompting technique for persona generation. This will be further discussed in the methodology section. Additionally, integrating a RAG-based system with personas allows RnD engineers and stakeholders to interact with customer data more easily. This ensures that the findings of this thesis are not only theoretically grounded but also practically applicable in real-world business scenarios.

4 Method and Implementation

4.1 Research Method

This research adopts the Design Science Research Methodology (DSRM) as the primary research framework [73]. DSRM is especially suitable for studies in computer science where the focus is on the development and evaluation of innovative artifacts to solve real-world problems. As described in [74], the DSRM framework includes five main activities:

1. Problem Explication
2. Requirements Definition

3. Design and Development
4. Demonstration
5. Evaluation

To systematically address the research objectives, these activities were executed in three phases. Each iteration was built upon findings and evaluations from previous cycles, which improved the developed artifact. The artifact in this study is a conversational system designed to help stakeholders at VCE to query on customer persona. Figure 8 depicts the research method followed in the study.

Iteration 1: Initial Chatbot Development and Testing

- **Problem Explication:** The initial problem identified was through a comprehensive literature review and discussions with stakeholders at VCE. The literature review explored existing research on customer personas, LLMs, RAG and prompting techniques. It helped providing a clear research gap regarding the integration of personas into RAG system and comparison of prompting method. The discussions with stakeholders highlighted the practical need for utilization of personas that can support various stakeholders to help make decisions faster.
- **Requirements Definition:** Based on insights gained from the review of the literature and discussion with stakeholders, the key requirements were identified. This also included collecting and preparing relevant data and defining the chatbot's core functionalities.
- **Design and Development:** The first artifact developed was a RAG-based chatbot that was integrated with verified customer personas provided by the VCE's Customer Experience team. Section 4.5 discusses the process involved in building the RAG system.
- **Demonstration:** The chatbot was deployed and demonstrated to a selected group of end-users. This enabled them to test and explore capabilities of the artifact.
- **Evaluation:** An evaluation form was sent out to the end users who had the chance to explore the chatbot. The process involved in the evaluation is discussed in Section 4.6 and the results of this section are discussed in Section 5.1. This feedback was then used as input for the second iteration.

Iteration 2: Persona Generation and Comparison

- **Problem Explication:** Based on user feedback from Iteration 1, the problem identified was that the chatbot's input data. The suggestion was to explore more data from customer success stories and more segment-specific information. Additional feedback included improved response accuracy and better handling of complex queries.
- **Requirements Definition:** As per the problem identified in the prior iteration, the requirement included gathering of additional personas, segment information and improvement of performance. In the literature study, there was also a gap in comparing the prompting technique for persona generation. This brings a requirement on studying how can different prompting techniques be used and evaluated.
- **Design and Development:** Personas were generated from customer success stories using two different prompting techniques. The development process involved in persona generation is elaborated in section 4.4.
- **Demonstration:** Personas created by both prompting techniques were clearly presented to evaluators alongside the original customer success stories. This helped them with a clear basis for comparison.

- **Evaluation:** A structured evaluation was conducted to statistically compare both prompting methods. The steps followed in the evaluation of personas are discussed in 4.7. The statistical analysis tested which of the prompting generates better personas in terms of metrics (accuracy, relevance, and consistency). Based on these results, these generated synthetic persona was used to augment the knowledgebase of the chatbot in the third iteration.

Iteration 3: Improvement of the conversational system and the final evaluation

- **Problem Explication:** Following the second iteration, the best performing method of generating persona was determined. The other feedback from the first iteration included the addition of additional segment information.
- **Requirements Definition:** The requirement in this iteration was to improve the chatbot by updating the knowledge base. This was done by adding additional synthetic customer personas and additional segment-specific information.
- **Design and Development:** The chatbot's knowledge base was expanded and refined by incorporating new data based on the initial feedback from iteration 1 and results from iteration 2.
- **Demonstration:** The improved chatbot was redeployed and demonstrated to various end-users for testing and exploration.
- **Evaluation:** A second evaluation round was conducted using a similar evaluation form from the iteration 1. This form assessed chatbot accuracy, usability, user satisfaction, and practical applicability. The purpose was to validate improvements from previous iterations and confirm the artifact effectively addressed the originally identified problem. The process involved in the evaluation is discussed in Section 4.8 and the results of this section are discussed in Section 5.3.

4.2 Overview of Data

Three types of data were used in this thesis:

1. **Customer Success Stories:** These are real-world narratives that illustrate how customers achieved positive outcomes using VCE's products or services. This data is publicly available on the VCE website [10]. This contains text, images, and videos. The stories are categorized by product, application, or industry segment (e.g., agriculture, demolition, quarrying, and aggregates). In this study, only stories related to the quarrying and mining segments were used.
2. **Verified Personas:** These personas were developed by the Customer Experience team at VCE through direct interviews with customers from different regions. This data is internal and confidential. It is accessible only to VCE employees. Table 1 contains the information included in each persona.
3. **General information about the Quarry, Mining, and Aggregates segments:** This consists of textual data providing definitions, processes, and background information about the three industry segments. It is used as contextual knowledge to support understanding of the domain.

4.3 Data Preparation

This section discusses the process involved in the data preparation. Each subsection describes the process involved for that specific data type.

[10] https://www.volvoce.com/united-states/en-us/resources/customer-success-stories/

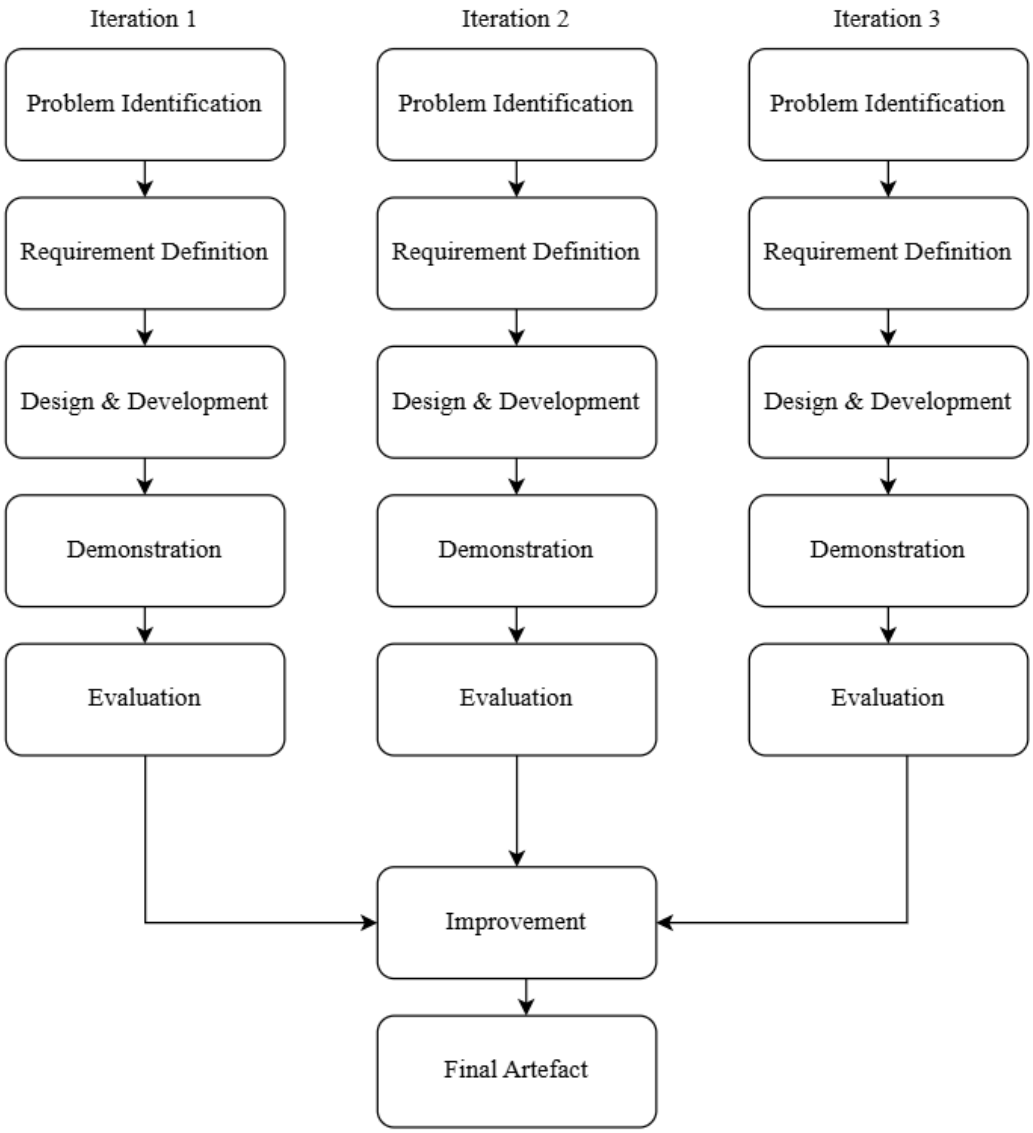

Figure 8. Research Method

4.3.1 Customer Success Story

The first type of data used was the Customer Success Stories, which served as input for generating synthetic customer personas. To extract these stories, web scraping was employed using the Python library Beautiful Soup[11]. Manual extraction of information from each webpage would have been time-consuming and error-prone. Hence, automated scraping was chosen to extract information.

The process began by inspecting the HTML structure to identify the relevant tags that contained the main story content. Only the textual narrative was extracted. Non-relevant elements such as headings, image captions, videos, and figures were excluded. A CSV file containing the URLs of selected (mining and quarrying segment) success stories was used as input for web scraping. From each URL, the script fetched the page content and extracted all paragraph (<*p*>) elements within a specific section of the webpage (*div*

[11] https://pypi.org/project/beautifulsoup4/

Table 1

Persona Attributes

Attribute	Description
Narrated Video	A video summarizing the persona's story
Name	The customer's name
Role	The job title or position
Number of Employees	Total employees in the customer's organization
Fleet Size	Size of the equipment fleet
Short Story	A brief background or narrative
What is Important	Key priorities or values of the customer
Challenges	Main issues faced by the customer
Expectations	What the customer expects from VCE
Buying Considerations	Factors that influence the customer's decisions

class "newsArticle-2023"). The extracted text was then cleaned by removing extra spacing and manually adding the missing content that was not extracted during the scraping process.

4.3.2 Verified Personas

The second source of data is verified Personas provided by VCE. These personas are stored in an internal platform that is only accessible to VCE employees. Due to this restriction, web scraping was not a feasible option for this dataset. Thus, the persona data was manually copied from the internal website. All the textual content from each of the personas was extracted, excluding video material. Once the textual data was collected, it was converted into structured JSON files using a Python script. The data was then converted to JSON as it provides a structured, machine-readable format that enables integration with retrieval systems. Each JSON file represented a single persona and included key-value pairs corresponding to the persona attributes such as name, role, challenges, and expectations.

4.3.3 General Information About the Quarry, Mining, and Aggregates Segments

The third type of data used in this study was general textual information related to quarrying, mining, and aggregates. This content was used to provide contextual background for retrieval tasks. This will help the RAG system better understand industry-specific terminology and operations.

This data was completely textual content. To enhance readability and support LLM's, the whole text was manually split into small meaningful chunks based on meaningful topics. Each chunk was then converted into Markdown format to introduce structure and hierarchy within the documents. Headings, subheadings, and bullet points were added to clearly distinguish between concepts, definitions, and processes. Converting into markdown will help the system better recognize the relationships between different pieces of information.

4.4 Generation Of Synthetic Customer Personas

This section details the process of generating customer personas using customer success stories as input. In this study, personas were created using two different prompting techniques: few-shot prompting and CoT prompting. GPT-4o Mini was selected as the language model for persona generation. The model received both the prompt and the success story as input. The prompt designs were developed based on OpenAI's prompt engineering guidelines [75]. Multiple iterations of prompts were refined and tested to

improve the output quality. The refinement process involved experimenting with wording and adjusting the level of detail provided in the prompts.

In the few-shot prompting technique, the model was provided with three verified personas as examples. The complete prompt included system instructions, a task definition, an output structure format, and three example personas. The benefit of this method is that by using examples, the model will be able to recognize patterns and relationships between persona attributes. This helps the model generate structured and coherent outputs. The prompt used for this method is included in Appendix A.3.

The CoT prompting technique followed a different approach by guiding the model through an internal step-by-step reasoning process instead of directly generating persona attributes. The prompt included system instructions, a structured output format, and a reasoning process to improve information extraction. The model was instructed to first identify key details from the success story, then analyze the customer's background and business context, extract challenges, expectations, buying considerations, and finally generate the structured persona. In this method, the model is encouraged to perform logical reasoning before output generation. The prompt used for this method is included in A.4. For each of the personas generated by using the two prompting methods, the total time taken to generate personas in seconds and the total tokens consumed were also computed. The overall process involved in persona generation is illustrated in Figure 9

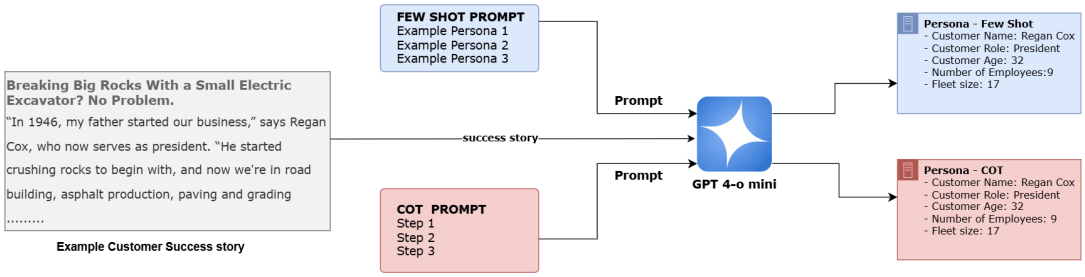

Figure 9. Persona Generation Process

4.5 Building the RAG system

The role of RAG system is to act as a conversational agent that allows users to query information based on customer persona data and general information about different segments. The system consists of two main components :

1. **Retrieval Component:** It is responsible for storing, indexing and retrieving relevant documents.
2. **Generation Component:** This component is responsible for generating responses based on the retrieved content using LLM.

The subsection below is a brief explanation of the design and implementation details of each component.

4.5.1 Retrieval Component

The retrieval component was built using Azure AI Search[12] that acts as a dedicated search engine and storage of data. The implementation process involved in building this component included the following

[12] https://azure.microsoft.com/en-us/products/ai-services/ai-search

steps:

1. **Creating the Search Index:** The first step in constructing a retrieval system involves creating a search index. The index schema was designed to accommodate both structured persona data (*.JSON format) and unstructured general information (*.txt format). The schema contained the below fields:

   - `id`- A unique identifier for each document.
   - `title`- The name of the document.
   - `category`- The type of document (e.g. "persona" or "general information")
   - `content`- The complete data in textual format that is to be searched and retrieved.
   - `content_vector`- A high dimensional vector representation of the document for similarity-based retrieval.

   The code snippet B.1 is used for the creation of the index.

   While creating the search index, three search techniques were configured for efficient content retrievals:

   - **Keyword Search:** This type of search performs lexical matching based on the exact words in the query. It allows filtering and ranking documents using traditional search techniques.
   - **Semantic Search:** This approach improves ranking by understanding the meaning of the query and prioritizing documents based on contextual relevance rather than just exact word matches. The *content* field was set as the primary ranking factor to ensure meaningful results.
   - **Vector Search:** This type of search was implemented using the Hierarchical Navigable Small World (HNSW) algorithm for Approximate Nearest Neighbor (ANN) retrieval. It enables searching for semantically similar documents using vector embeddings, even when the query and the document do not share exact words. The code snippet B.2 is used for defining and configuring the types search methods, including the different types of search techniques.

2. **Uploading Documents to the Index:** After the index was created, the next step involved uploading the documents to the index that was created. This process began by loading the data and extracting textual content from the data. The textual content was then converted into embeddings using an embedding model named text-embedding-ada-002. These embeddings, along with the raw text, were then batch-uploaded into the index. The code snippet B.4 is responsible for converting the text into embedding and upload in the search index.

4.5.2 Generation Component

The Generation Component is responsible for utilizing the content retrieved by the retrieval component to generate output for the user. This component was developed using GPT-4o Mini in combination with a hybrid search approach. Below are the implementation details of the processes involved in this component.

1. **Integrating search index with hybrid search strategy:** To improve the quality of responses, a hybrid search approach was employed. Hybrid search combines the capabilities of both keyword-based search and vector-based search techniques. According to experiments by Microsoft [76], hybrid search outperforms standalone keyword or vector-based search methods in retrieving relevant documents for question-answering systems. Due to this reason, the hybrid search was opted for this study.

When a user submits a query, the query is first converted into an embedding using the embedding model. The hybrid search method is then applied to retrieve the top three most relevant documents from the search index. These documents are used as contextual input for the language model to perform the generation of an appropriate response. The code snippet B.3 specifies the query type (hybrid search), and defines the number of documents to be returned during the search process.

2. **System Message and Prompt Engineering:** To ensure consistency, accuracy, and contextual relevance in the generated responses, a Prompty file [13] was created. This file contains a system message that defines specific role instructions, the tone for responses, and detailed guidelines for answering the questions. The system message used for the RAG system is presented in A.1.

3. **Final Response Generation:** After retrieving the relevant documents, the GPT-4o Mini model synthesizes the final response by integrating the retrieved documents, system message, and conversation history to ensure coherent and context-relevant output. The code snippet B.5 illustrates how the user input, conversation history, and relevant documents are processed to generate the final system message for the RAG system.

Figure 10 shows the overall process of the chatbot system, highlighting the flow from data indexing and retrieval to response generation using a hybrid search strategy.

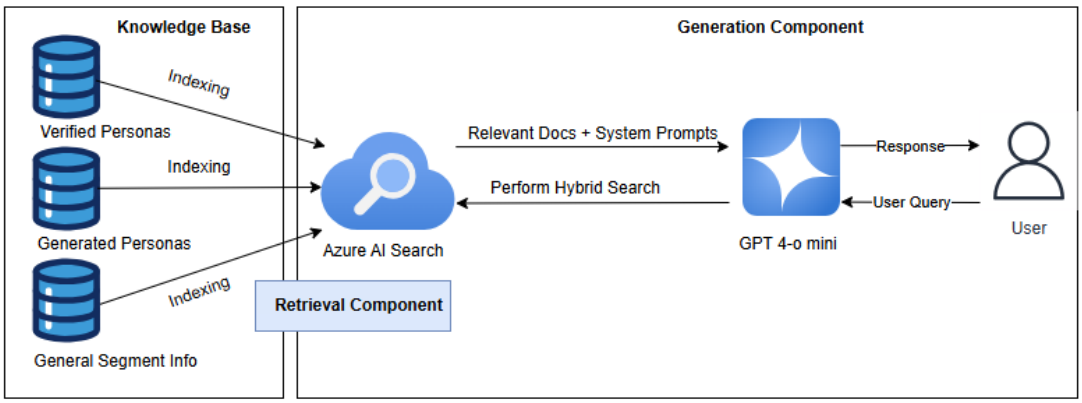

Figure 10. RAG Process

4.6 Initial Evaluation of the Conversational System

This section describes the process involved in the initial user evaluation to assess the effectiveness of the persona-based chatbot system. The goal of this evaluation is to understand how the chatbot supports decision-making and contributes to automating customer-facing processes. For this stage of evaluation, the developed chatbot was integrated with verified customer personas and segment-specific information. The system was then deployed using Azure Web App for user interaction [14]. The participants included people from relevant business functions, such as RnD, marketing, and customer representatives. After interacting with the system, they were asked to provide feedback through an evaluation form.

4.6.1 Evaluation Design

To systematically assess the ability of the chatbot to support decision-making and process automation, five questions were opted. The type of questions includes multiple choice, Likert's scale, and 1-10 rating scale.

[13] https://prompty.ai/
[14] https://azure.microsoft.com/en-us/products/app-service/web

Table 2 presents the questions in the evaluation form along with their purposes and types of responses.

Table 2

Initial Evaluation Questions, Purpose and Type

Question	Purpose	Response Type
How would you rate the chatbot's ability to provide accurate answers?	Evaluate the overall accuracy of the chatbot in providing relevant answers	1-10 scale
Does the chatbot correctly interpret and respond to complex queries (e.g., providing details on customer personas)?	Measures the chatbot's capability to handle complex queries.	Likert Scale
Does the chatbot provide clear and concise answers?	Assesses clarity in responses.	Likert Scale
How well does the chatbot align with your business needs?	Evaluates the chatbot's relevance and usefulness in business contexts.	Likert Scale
Do you believe the chatbot has reduced the workload for human support teams?	Understand the impact of chatbot in automation and improving efficiency.	Multiple Choice

In addition to these questions, participants were also asked to provide open-ended comments to elaborate on their experience, share specific feedback, or suggest improvements. This combination of quantitative and qualitative feedback designed helped in providing insights into how well the system aligns with user expectations, business needs, and opportunities for automation. Data from these responses was analyzed using descriptive statistical analysis and qualitative thematic analysis. The findings are presented in section 5.1 in the chapter 5.

4.7 Evaluation of Generated Personas

This section presents the methodology used to evaluate the two types of prompting techniques. The aim of this evaluation is to identify the optimal prompting technique to produce personas.

4.7.1 Evaluation Design

A total of 24 customer success stories were used to generate personas. To reduce the time and effort for evaluators, a random subset of five stories was selected for the evaluation. Each evaluator read the full customer success story before reviewing two anonymized personas. The order in which the personas were presented was randomized to minimize the bias. A Microsoft Form [15] was used to collect binary feedback (Yes/No) on each of the evaluation metrics.

4.7.2 Metrics Used

The metrics used in this study were adapted from literature [3], [10], [15] that evaluated personas. Each of the metrics was evaluated using binary response. The choice of binary metrics was to reduce ambiguity and speed up the evaluation process.

Table 3 presents the questionnaire used for evaluating each metric, along with a brief description of what each metric assesses.

[15] https://forms.office.com/

Table 3

Metrics and Definition

Metric Name	Questionnaire	Description
Completeness	Does the persona include all the important details (like role, challenges, expectations etc.) from the customer success story to fully understand the customer?	Evaluates whether the persona captures all key customer insights needed for understanding.
Relevance	Does the persona focus only on the relevant and important details from the customer success story?	Assess whether the persona includes only important details from the source story, avoiding any irrelevant or redundant information.
Consistency	Does the persona add any incorrect or made-up information that is not in the customer success story?	Checks if the persona introduced incorrect, fabricated, or contradictory information.

4.7.3 Participants

The evaluation was conducted with professionals from VCE who are familiar with customers, products, and services. The evaluators included Customer Solution Strategists, Research Engineers, and Project Managers.

4.7.4 Analysis Method

A formal hypothesis testing approach was adopted to determine whether differences between prompting methods were statistically significant. For each prompting method and for each metric, the following hypotheses were defined:

- **Null Hypothesis (H_0)** : There is no significant difference between the two prompting methods in terms of the metrics used for evaluation.
- **Alternative Hypothesis (H_1)**: There is a significant difference between the two prompting methods in terms of the metrics used for evaluation.

The McNemar test [77] was selected because it is specifically designed for paired nominal (categorical) data. It is commonly used when the same subjects are exposed to two conditions, and their binary responses (e.g., Yes/No) are analyzed for shifts between the two conditions [77]. In this thesis, it is used to determine if one prompting method significantly outperformed another across the three binary evaluation metrics. This test uses a 2×2 contingency table based on paired binary responses for each persona pair. Only the discordant pairs, where evaluators responded differently for the two methods are used to compute the test statistic. A separate contingency table is constructed for each evaluation metric. Table 4 presents an example contingency table.

The McNemar test statistic is defined as $\chi^2 = \frac{(b-c)^2}{b+c}$. The test statistic value which we obtain follows a chi-square distribution with one degree of freedom. From this value, a p-value is calculated and used to assess whether the observed difference is statistically significant. A p-value below 0.05 indicates a significant difference between the two prompting methods for the given evaluation metric.

Table 4

Example Contingency Table

	Method: CoT - Yes	**Method: CoT - No**
Method: Few-Shot - Yes	a	b
Method: Few-Shot - No	c	d

a = Both methods are rated as 'Yes'
b = Method Few-Shot is rated as 'No' and Method CoT is rated as 'Yes'
c = Method Few-Shot is rated as 'Yes' and Method CoT is rated as 'No'
d = Both methods are rated as 'No'

4.8 Evaluation of Augmented Chatbot with Synthetic Personas

This section discusses the methodology used to evaluate the impact of augmenting the chatbot's knowledge base with synthetic personas and additional segment-specific information. The objective was to assess whether these enhancements improved the chatbot's performance in terms of accuracy, usability, and decision-making capabilities.

System Updates and Evaluation Design

Based on the insights from Section 4.6 (Initial Evaluation) and Section 4.7 (Persona Generation Evaluation), the conversational system was updated to improve its overall performance. Feedback from the initial evaluation highlighted the need for additional segement data and more personas derived from customer success stories. As a result, the knowledge base was expanded with newly generated synthetic personas and additional segment-specific information.

Based on the findings in Section 4.7, the best-performing prompting technique for persona generation was selected. The personas generated by this method were then added to the Azure AI Search index, replacing the initial dataset. Plus, the system prompt that was used to guide the chatbot's responses was also revised. This was to improve the accuracy, particularly for complex or context-rich queries. The updated system prompt is included in Appendix A.2 .

Once these changes were implemented, the system was tested and redeployed. The updated version of the chatbot was made available for user testing. To ensure consistency and for direct performance comparison, the same evaluation method, participant group, and questionnaire from the initial evaluation were used again. The collected responses were analyzed using the same methods as in the initial evaluation. Descriptive statistical analysis was used to compare quantitative results. This approach enabled direct comparison between the initial chatbot (with verified personas) and the updated version (with synthetic personas). The findings from this evaluation are presented in Chapter 5.3.

5 Results

This section presents the results of the thesis work, including the evaluation of generated personas and the performance of the persona-based chatbot. The results are organized based on the research questions.

5.1 Results for Research Question 1: Effectiveness of the Persona-Based Chatbot

This subsection presents the findings related to the evaluation of the persona-based chatbot conducted with eight stakeholders from relevant business functions such as RnD, marketing, and customer relations.

The evaluation was focused on five aspects, such as accuracy of the answers, the ability to handle complex queries, clarity of the responses, alignment with business needs, and impact on workload reduction.

5.1.1 Quantitative Results

Participants were asked to provide an overall rating for the ability of the chatbot to provide accurate answers with a scale from 1 (Very Poor) to 10 (Excellent). The average rating across all evaluators was 5.88. The range of ratings from all evaluators was 4 to 10, with the majority giving a rating of 5. Figure 11 is a bar chart illustrating the distribution of accuracy rating.

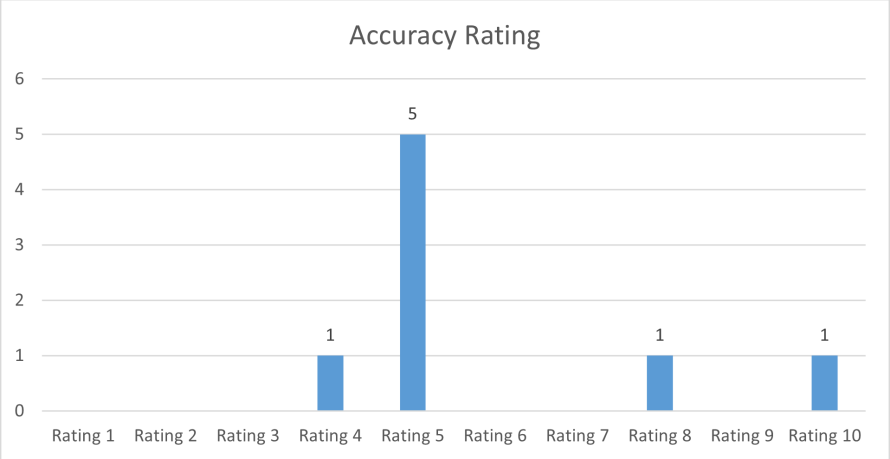

Figure 11. Distribution of the accuracy rating.

The ability of the chatbot to interpret and respond to complex queries such as providing details on customer persona was measured using a Likert scale. The majority of the participants indicated that the chatbot responded correctly "most of the time", while three participants selected "sometimes", and one participant reported "never". Figure 12 presents a bar chart depicting these findings.

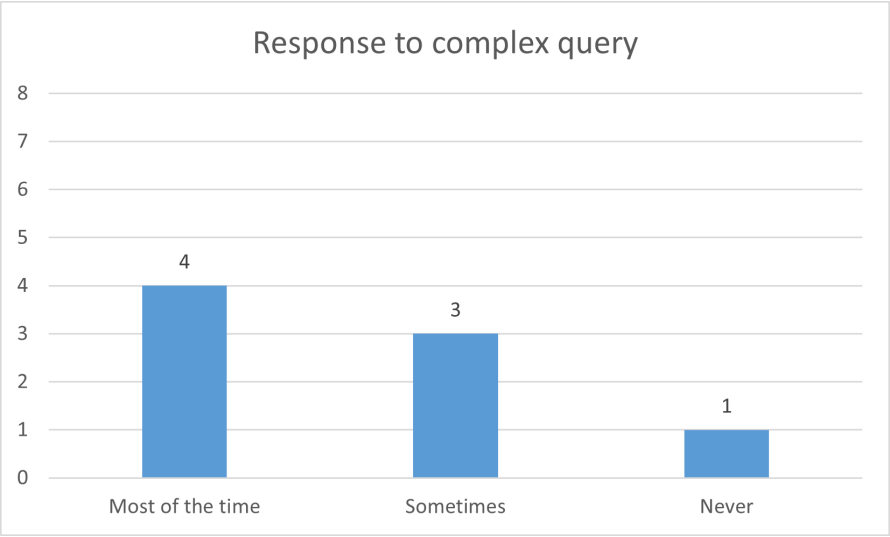

Figure 12. Ability of the system to provide response to complex query.

Regarding the ability of the conversational system to provide clear and concise responses, 3 users reported

that the system provides clear and concise results only sometimes. This indicates some inconsistency in the clarity and conciseness of the responses. Figure 13 shows the distribution of the responses in a bar chart.

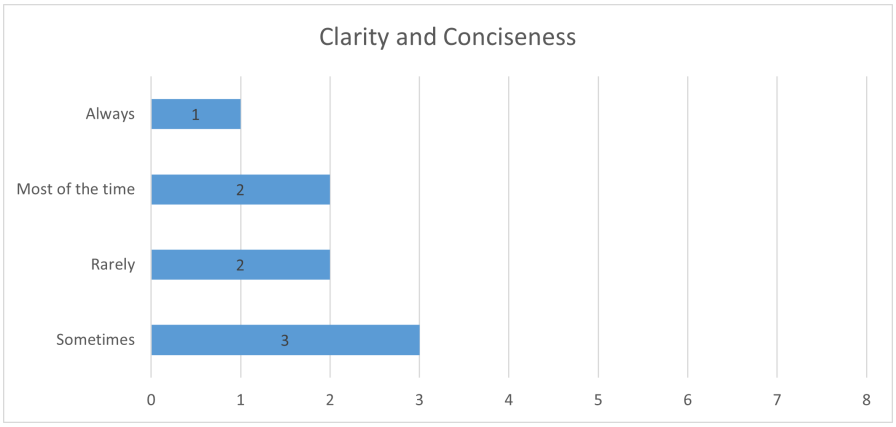

Figure 13. Ability of the system to provide clear and concise response.

The impact of the system in business contexts can seen largely positive. Only one evaluator rated the system as not useful. The remaining participants suggested responded positively. 62.5 % users rated it as "somewhat needed", suggesting that while the chatbot addressed business needs to some extent, further alignment and improvements are required. Figure 14 is a bar chart showing the distribution of the various ratings regarding the alignment of business needs.

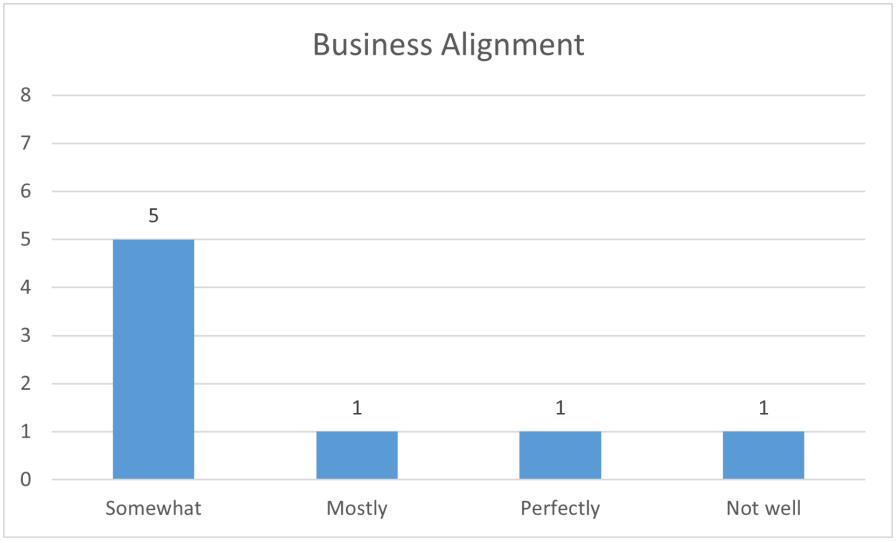

Figure 14. Alignment of system in business need.

When evaluating the potential of the system to reduce the workforce, 75% of the participants believe that it will reduce their workload and improve automation. Figure 15 is a pie chart depicting this distribution.

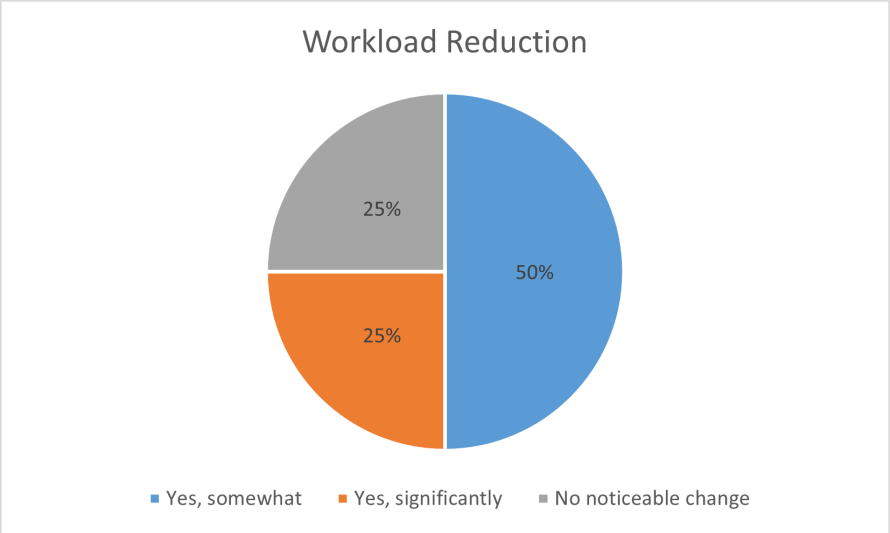

Figure 15. Workload reduction and improvement in automation

5.1.2 Qualitative Results

In addition to the five evaluation questions, participants were asked to provide open-ended feedback on their experience with the persona-based chatbot. Three main key themes were identified from their responses.

Primarily, the participants emphasized the importance of improving the quality of the data used. They suggested that integrating customer success stories and incorporating a more diverse range of customer data could significantly enhance the chatbot's utility. Another observation was that the responses sometimes felt too generalized and lacked specific insights, making them indistinguishable from publicly available information. This highlights that the system needs to provide deeper, more personalized answers based on additional customer datasets. Secondly, participants pointed out the necessity of improving segment-specific information within the chatbot's knowledge base. Finally, the participants also identified several technical areas for improvement. These included better handling of complex queries, improved accuracy in responses, faster response times, and a more natural, human-like conversational style. Participants also suggested better integration with internal systems and the inclusion of sources for the information provided in responses.

5.1.3 Summary of Findings

The initial round evaluation of the persona-based chatbot demonstrated a moderate overall effectiveness in supporting decision-making and automation within the construction industry. From the quantitative results, the system received an average overall accuracy of 5.88 out of 10. While it generally aligned with business needs for most of the users, there remains significant room for improvement. From the feedback on the ability of the system to reduce workload and automation, it can be concluded that the persona chatbot has contributed to reducing human workload. Along with these quantitative findings, the qualitative feedback highlighted the need for broader and more diverse data integration from customer success stories, enhanced segment-specific information, improved handling of complex queries and more human-like interactions. Overall, these findings provided critical insights for guiding the refinement and improvement of the chatbot in the next iterations.

5.2 Results for Research Question 2: Persona Generation and Prompting Techniques

This section presents the results obtained from comparing synthetic personas generated by two prompting techniques.

5.2.1 Quantitative Results

In this thesis work, the synthetic persona was generated using the customer success story as input. The process involved in the generation of the persona is described in the section 4.4. The evaluation was carried out by three expert evaluators (n=3), each of whom reviewed a total of 5 personas. Each anonymized persona was assessed using three binary metrics: completeness, relevance, and consistency. The responses were analyzed using the McNemar's test to identify if there are statistically significant differences between the two prompting methods. In addition, efficiency metrics such as average generation time (seconds) and token usage were recorded.

Completeness. The completeness metric evaluated whether the persona captured all important details (e.g., role, challenges, expectations) from the customer success story. The McNemar test produced a test statistic of 1.0 and a p-value of 0.0063, indicating a statistically significant difference between the two prompting methods. As shown in contingency table 5, in 11 cases evaluators rated the Few-Shot persona as complete and the CoT persona as not complete, and in only 1 case the opposite occurred. This result clearly indicates that Few-Shot prompting outperformed CoT prompting in terms of completeness. Figure 16 presents a bar chart that illustrates the distribution of the evaluator ratings for this metric.

Table 5

Contingency Table for Completeness Metric

	CoT: Yes	**CoT: No**
Few-Shot: Yes	a = 3	b = 11
Few-Shot: No	c = 1	d = 0

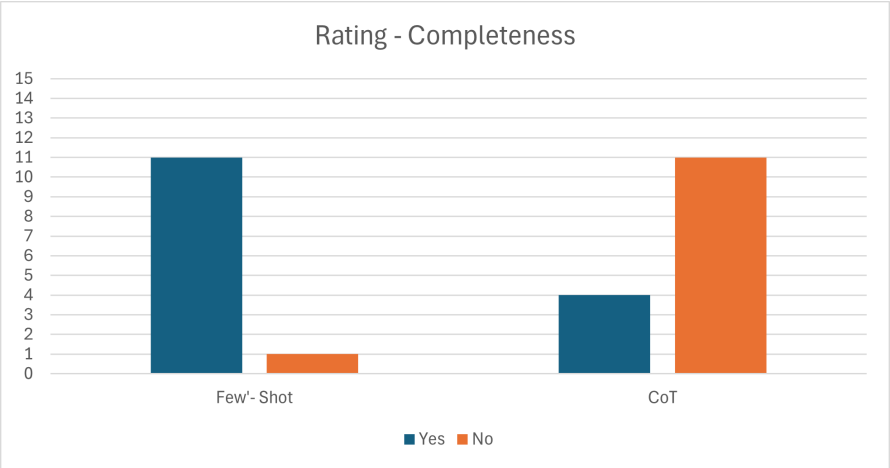

Figure 16. Comparison of the metrics- Completeness

Relevance. Relevance was assessed to determine whether the persona focused only on the important and relevant details from the source material. The test yielded a p-value of 0.6250, suggesting that there is no significant difference between the two prompting techniques. Table 6 is the contingency table for

the relevance metrics. While there were some variances in individual ratings, the differences were not statistically significant.

Table 6

Contingency Table for Relevance Metric

	CoT: Yes	**CoT: No**
Few-Shot: Yes	a = 11	b = 3
Few-Shot: No	c = 1	d = 0

Consistency. The consistency metric evaluated whether the persona introduced incorrect or fabricated information. The test result for this metric was a p-value of 0.2500, indicating no significant difference between the two prompting approaches.Table 7 is the contingency table for this metrics.

Table 7

Contingency Table for Consistency Metric

	CoT: Yes	**CoT: No**
Few-Shot: Yes	a = 1	b = 3
Few-Shot: No	c = 0	d = 11

Efficiency Metrics. In addition to qualitative performance, the two prompting methods were evaluated based on average generation time and token usage. Few-Shot prompting had an average generation time of 3.66 seconds and used 3505.91 tokens. While for CoT prompting, the average generation time was 2.79 seconds and the total average tokens consumed was 2064.2. Figure 17 and Figure 18 is a bar chart illustrating the average time and average token usage across all personas. These results indicate that CoT prompting outperformed Few-Shot as it required less time and fewer tokens, making it superior and computationally more efficient.

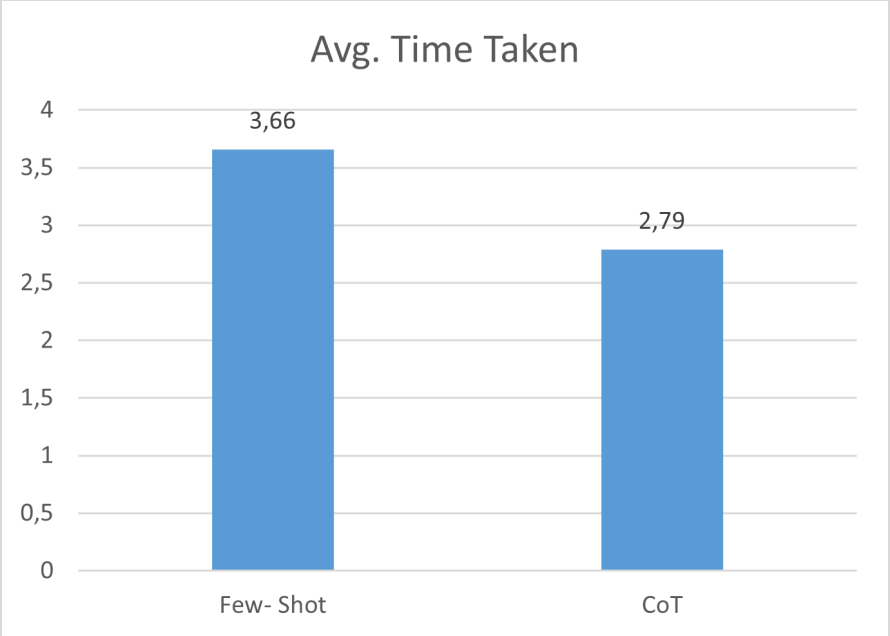

Figure 17. Average Time Taken

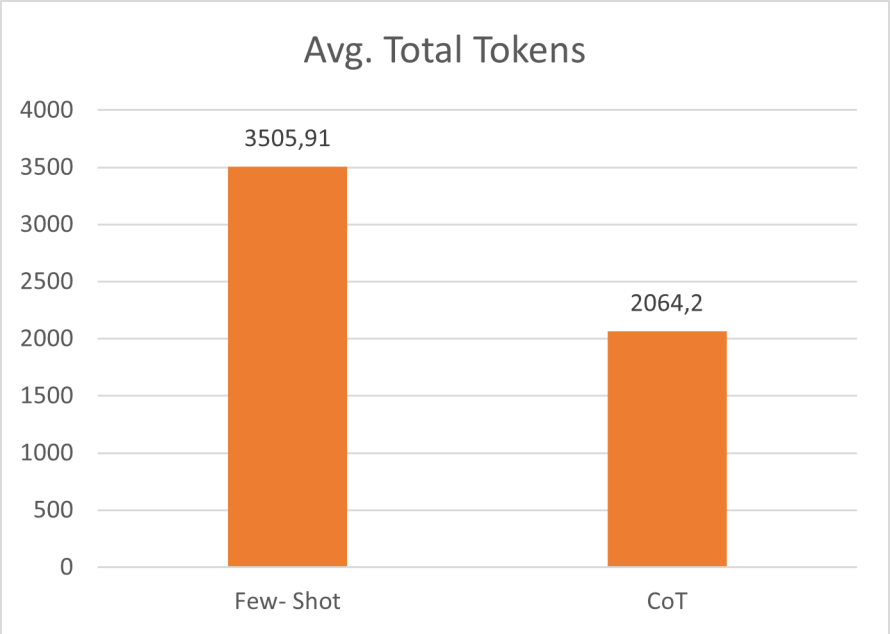

Figure 18. Average Tokens Consumed

5.2.2 Summary of Findings

The evaluation revealed that Few-Shot prompting significantly outperformed CoT prompting in terms of completeness. Evaluators generally rated Few-Shot personas as more complete than those generated using CoT. For the other two quality metrics, relevance and consistency, no statistically significant differences were found between the prompting methods. However, when evaluated from an efficiency perspective, CoT prompting proved to be both faster and more resource-efficient. The personas generated by CoT method use fewer tokens and shorter response times compared to Few-Shot prompting. Given its superiority in efficiency, CoT prompting was selected for the second iteration of the system. Table 8 summarizes the comparative performance of the two prompting methods in terms of quality and efficiency.

Table 8

Summary of Evaluation of Prompting Techniques

Prompting Method	Completeness	Relevance	Consistency	Avg Time (s)	Avg Total Tokens
Few-Shot	Statistically significant	Statistically insignificant	Statistically insignificant	3.66	3505.91
CoT				2.79	2064.2

5.3 Results for Research Question 3: Impact of Knowledge Base Augmentation

This section presents the results after augmenting the chatbot's knowledge base with synthetic personas and segment-specific information. 12 stakeholders from business functions such as RnD, marketing, and customer relations had participated in the evaluation. The evaluation was focused on three primary aspects: accuracy of the responses, ability to handle diverse queries, and overall usefulness of the chatbot in business needs.

5.3.1 Quantitative Results

Participants were asked to assess the impact of the augmented knowledge base on the accuracy of responses, using a scale from 1 (Very Poor) to 10 (Excellent). The average rating across all evaluators was 6.42. This shows a slight improvement from the previous round of chatbot evaluation where the average rating was 5.88. The range of ratings was from 4 to 8. Most of the evaluators gave ratings of 6 or above. This indicates more consistent performance. Figure 19 is a bar chart depicting the distribution of the accuracy rating post improvement.

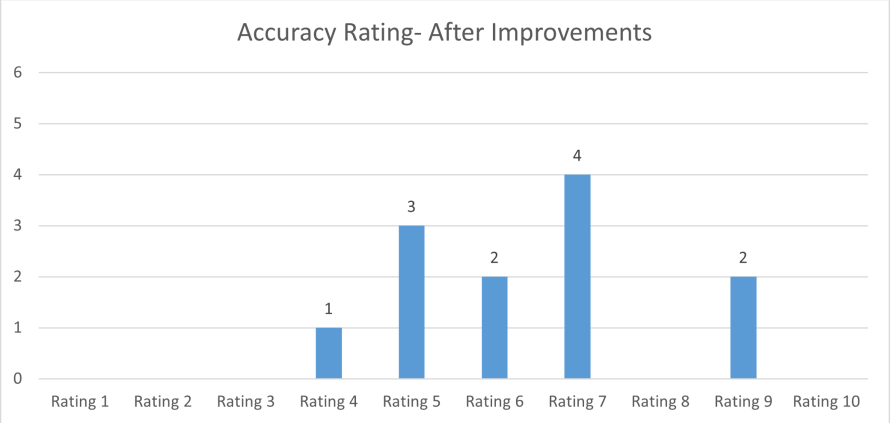

Figure 19. Distribution of the accuracy rating - post improvements.

To evaluate if the system was able to handle diverse and complex queries, participants were asked to assess the system's ability to respond to complex persona-related queries after data augmentation. 6 participants selected "sometimes", 5 of them selected "most of the time", and 1 selected "always". While the system showed balanced performance in handling complex queries, the responses like "sometimes" indicate some inconsistencies in providing comprehensive and contextually relevant answers. Figure 20 is a bar graph that illustrates the response to handling complex queries.

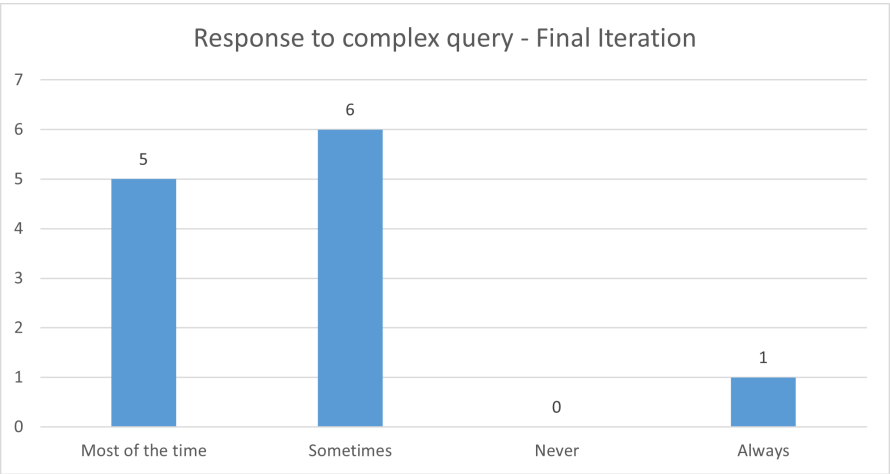

Figure 20. Ability of the system to provide response to complex query - post augmentation.

Regarding the usefulness after the augmented knowledge base, participants provided varied responses on how well the chatbot aligned with business needs. Seven participants rated it as "somewhat", two as "mostly", one as "perfectly", one as "not at all", and one as "not well". This distribution indicates

a moderate but mixed perception of the relevance of the augmented knowledge base to business needs. Figure 21 is a bar chart showing the distribution of the ratings regarding usefulness after the augmentation.

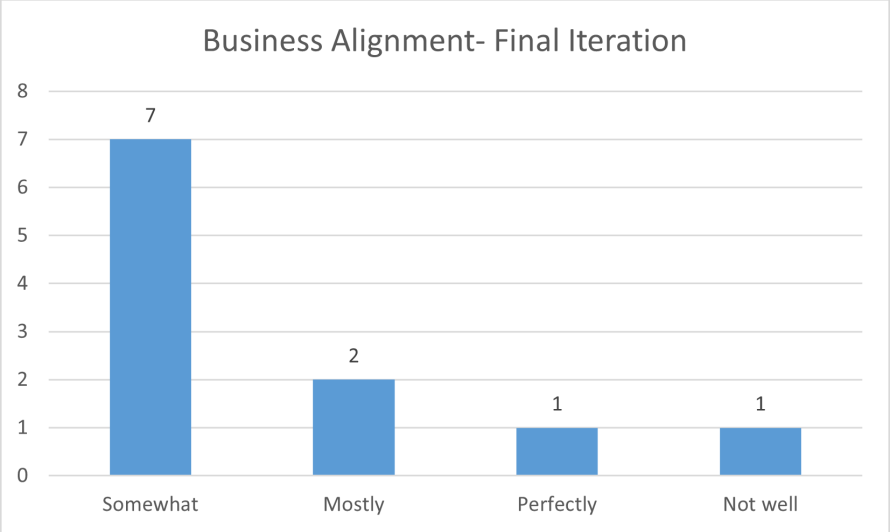

Figure 21. Alignment of system in business need - post updation.

5.3.2 Summary of Findings

The results of the evaluation after updating the knowledge base indicate a moderate but noticeable improvement in the overall performance of the chatbot and practical utility. The integration of synthetic personas and segment-specific information increased the average accuracy rating from 5.88 to 6.42. It can also be concluded that the system was able to handle complex queries, as no participants selected as "never".

Regarding practical utility, 81.82% of evaluators rated the augmented knowledge base as at least "somewhat useful", indicating a positive trend in the usefulness of the system in the business contexts. Despite this positive indication, there remains room for further refinement to fully align the system's output with business needs and expectations. Overall, while the augmentation has contributed to improved performance, further refinement can be made to fully optimize the effectiveness and relevance of the chatbot to business needs.

6 Discussion and conclusion

6.1 Analysis

A comprehensive analysis of each of the research questions and the potential reasons for the observed outcomes are discussed in this section.

1. **Effectiveness of Persona-Based Chatbot in Decision-Making**

   The evaluation of the persona-based chatbot revealed moderate effectiveness in supporting decision-making and automating customer-facing processes. Even though the system was able to provide relevant responses, the average accuracy rating of 5.88 indicates that there were inconsistencies in

providing accurate and contextually appropriate information. One of the reasons for these inconsistencies could be limited data used. When it comes to ability of the system to reduce workload, 75% of participants believed that the system had the potential to reduce workload and automate routine tasks.

2. **Effectiveness of Prompting Techniques in Synthetic Persona Generation**

   From the generated personas it can be observed when multiple customers were mentioned in a single customer success story, the LLM tend to focus on only one customer. The relevant information about others customers in the same story are often missed. This tendency to concentrate on a single person could potentially reduce the completeness of the persona generated. This suggests that further refinement in data processing or prompt engineering may be needed to make sure a comprehensive persona is generated in cases where multiple customer information are present in single stories.

   The evaluation of the generated personas was conducted with a small sample size, consisting of only three evaluators and a limited number of personas. This limited scope may affect the generalizability of the findings. Future evaluations with a larger sample could provide better insights. From the statistical test it is evident that the Few-Shot prompting produced more comprehensive personas. However, CoT prompting outperformed Few-Shot in terms of efficiency. This personas generated by this method took less time and consumed fewer tokens. This finding is particularly relevant in the context of real-world deployment, where response time and resource consumption are critical factors.

   The lack of significant differences in relevance and consistency between the two methods indicates that both prompting techniques was similar in terms factual accuracy and reducing fabricated content. This suggests that while Few-Shot may be preferable for completeness, CoT is an optimal choice for scenarios where response efficiency is prioritized.

3. **Impact of Knowledge Base Augmentation**

   The evaluation of the augmented chatbot was conducted with participants from different business functions, including RnD, customer relations, and Information Technology (IT). Their expectations and query types varied significantly, which may have influenced their assessment. For example, IT engineers might ask for more technical information, while customer experience staff might ask queries related to customer data. Addressing the chatbot functionalities for different users may help to improve overall satisfaction.

   Augmenting the knowledge base with synthetic personas and segment-specific information resulted in a slight increase in accuracy, with the average rating rising to 6.42. 81.82% of participants rated the augmented knowledge base as at somewhat useful. This highlights the need for further refinement in data selection and segmentation.

The analysis indicates that while the persona-based chatbot demonstrated some effectiveness in automating customer-facing processes and supporting decision-making, its overall performance was limited by data limitations and variability in response quality. The Few-Shot prompting method produced more complete personas, whereas CoT prompting was more efficient in terms of response time and token usage. Augmenting the knowledge base has shown slight improvements in accuracy and perceived usefulness. Further data refinement and prompt engineering can optimize the system to align more closely with with business objectives.

6.2 Broader Impact

LLM-powered chatbots can simplify decision-making and improve customer interactions in the construction equipment industry by automating tasks and providing personalized responses. This not only reduces the workload on employees but also allows RnD teams to gain deeper insights into customer needs

through persona analysis. Moreover, deploying LLM-powered chatbots contributes to digital transformation in traditional industries. It encourages the adoption of AI-driven approaches, promotes innovation, and helps companies stay competitive in a rapidly evolving market.

6.3 Limitation

The implementation of the conversational system and the persona generation system faced several limitations that impacted the overall scope and performance of the project.

1. **Knowledge Base**: The knowledge base used for the chatbot was limited to a small set of verified personas and segment information specifically on mining, quarrying, and aggregates. This limits the diversity of the information available to generate responses and potentially reduces the chatbot's ability to provide comprehensive insights for other relevant segments.
2. **Input Data For Synthetic Persona Generation**: For synthetic persona generation, the data source was restricted to customer success stories, which mainly showcased positive customer experiences. These narratives often emphasized successful outcomes rather than describing the challenges faced or areas for improvement. Consequently, the generated personas may lack a balanced perspective, as they primarily reflect favorable customer experiences with VCE products.
3. **Evaluation**: The evaluation process was subjective, as different evaluators might have a different opinion on the quality of the persona. This subjectivity could introduce potential biases and inconsistencies in the assessment results.

6.4 Future Work

To make the conversational system and the persona generation better, there are several areas to work on in the future.

1. **Expand Data**: Currently, data are limited to verified personas and segment-specific information focused on mining, quarrying, and aggregates. Future work could include additional data sources such as customer feedback, satisfaction surveys, and competitor analysis reports. This would provide a more comprehensive dataset that captures diverse customer experiences.
2. **Persona Generation**: Another area for further development is the refinement of persona generation techniques. The present approach utilizes GPT-4o Mini with prompting methods. Future research could explore more advanced LLMs or using a fine-tuned LLM with information about VCE. This approach could potentially improve the contextual accuracy and relevance of the personas generated.
3. **Real Time Updation:** As VCE customer needs and challenges change over time, implementing mechanisms to track changes in customer personas over time would be beneficial. This could be achieved by periodically updating the knowledge base with new customer success stories and feedback data, allowing the system to maintain relevance over time.
4. **Chatbot Framework**: When it comes to the chatbot, exploring advanced RAG frameworks such as Graph-RAG [78] and Multi-Hop RAG [79] could improve the system. Graph-RAG introduces graph structures to link related personas, customer success stories, and other data points. This approach could capture more complex relationships between data elements, enabling richer and more context-aware retrieval. Similarly, Multi-Hop RAG [79] extends the RAG framework by retrieving

and sequentially processing multiple data points to generate more comprehensive responses. Another promising direction includes improving the chatbot's interaction to match specific user roles. For example, tailoring the conversation paths for specific roles, such as RnD Engineers or Customer Experience Teams. This could potentially improve the relevance of responses and provide more targeted insights.

5. **Evaluation Metrics:** Lastly, including automated metrics for evaluating the retrieval system and persona quality could streamline the evaluation process and reduce dependence on subjective human evaluation.

Future work in these areas would not only improve the robustness and adaptability of the chatbot system, but also align it more closely with the practical needs of teams within VCE.

References

[1] M. U. Hadi, Q. Al-Tashi, R. Qureshi, *et al.*, "Llms: A comprehensive survey of applications, challenges, datasets, models, limitations, and future prospects,"

[2] M. Cheung, "A reality check of the benefits of llm in business," *arXiv preprint arXiv:2406.10249*, 2024.

[3] L. Sun, T. Qin, A. Hu, *et al.*, "Persona-l has entered the chat: Leveraging llm and ability-based framework for personas of people with complex needs," *arXiv preprint arXiv:2409.15604*, 2024.

[4] J. McGinn and N. Kotamraju, "Data-driven persona development," in *Proceedings of the SIGCHI conference on human factors in computing systems*, 2008, pp. 1521–1524.

[5] X. Zhang, H.-F. Brown, and A. Shankar, "Data-driven personas: Constructing archetypal users with clickstreams and user telemetry," in *Proceedings of the 2016 CHI conference on human factors in computing systems*, 2016, pp. 5350–5359.

[6] S.-g. Jung, J. Salminen, H. Kwak, J. An, and B. J. Jansen, "Automatic persona generation (apg) a rationale and demonstration," in *Proceedings of the 2018 conference on human information interaction & retrieval*, 2018, pp. 321–324.

[7] A. Farseev, Q. Yang, M. Ongpin, I. Gossoudarev, Y.-Y. Chu-Farseeva, and S. Nikolenko, "Somonitor: Combining explainable ai & large language models for marketing analytics," *arXiv e-prints*, arXiv–2407, 2024.

[8] A. Radford, K. Narasimhan, T. Salimans, I. Sutskever, *et al.*, "Improving language understanding by generative pre-training," 2018.

[9] S. De Paoli, "Improved prompting and process for writing user personas with llms, using qualitative interviews: Capturing behaviour and personality traits of users," *arXiv preprint arXiv:2310.06391*, 2023.

[10] J. Salminen, C. Liu, W. Pian, J. Chi, E. Häyhänen, and B. J. Jansen, "Deus ex machina and personas from large language models: Investigating the composition of ai-generated persona descriptions," in *Proceedings of the 2024 CHI Conference on Human Factors in Computing Systems*, 2024, pp. 1–20.

[11] X. Zhang, L. Liu, Y. Wang, *et al.*, "Personagen: A tool for generating personas from user feedback," in *2023 IEEE 31st International Requirements Engineering Conference (RE)*, IEEE, 2023, pp. 353–354.

[12] J. White, Q. Fu, S. Hays, *et al.*, "A prompt pattern catalog to enhance prompt engineering with chatgpt," *arXiv preprint arXiv:2302.11382*, 2023.

[13] M. Cherukuri, "Cost, complexity, and efficacy of prompt engineering techniques for large language models,"

[14] P. Lewis, E. Perez, A. Piktus, *et al.*, "Retrieval-augmented generation for knowledge-intensive nlp tasks," *Advances in Neural Information Processing Systems*, vol. 33, pp. 9459–9474, 2020.

[15] T. Goel, O. Shaer, C. Delcourt, Q. Gu, and A. Cooper, "Preparing future designers for human-ai collaboration in persona creation," in *Proceedings of the 2nd Annual Meeting of the Symposium on Human-Computer Interaction for Work*, 2023, pp. 1–14.

[16] T. Brown, B. Mann, N. Ryder, *et al.*, "Language models are few-shot learners," *Advances in neural information processing systems*, vol. 33, pp. 1877–1901, 2020.

[17] J. Wei, X. Wang, D. Schuurmans, *et al.*, "Chain-of-thought prompting elicits reasoning in large language models," *Advances in neural information processing systems*, vol. 35, pp. 24 824–24 837, 2022.

[18] S. Ghosh, "Natural language processing: Basics, challenges, and clustering applications," in *Federated learning for Internet of Vehicles: IoV Image Processing, Vision and Intelligent Systems*, Bentham Science Publishers, 2024, pp. 61–82.

[19] E. D. Liddy, "Natural language processing," 2001.

[20] P. Johri, S. K. Khatri, A. T. Al-Taani, M. Sabharwal, S. Suvanov, and A. Kumar, "Natural language processing: History, evolution, application, and future work," in *Proceedings of 3rd International Conference on Computing Informatics and Networks: ICCIN 2020*, Springer, 2021, pp. 365–375.

[21] D. A. Dahl, "Natural language processing: Past, present and future," in *Mobile speech and advanced natural language solutions*, Springer, 2012, pp. 49–73.

[22] D. Khurana, A. Koli, K. Khatter, and S. Singh, "Natural language processing: State of the art, current trends and challenges," *Multimedia tools and applications*, vol. 82, no. 3, pp. 3713–3744, 2023.

[23] Y. Chai, L. Jin, S. Feng, and Z. Xin, "Evolution and advancements in deep learning models for natural language processing," *Applied and Computational Engineering*, vol. 77, pp. 144–149, 2024.

[24] A. Torfi, R. A. Shirvani, Y. Keneshloo, N. Tavaf, and E. A. Fox, "Natural language processing advancements by deep learning: A survey," *arXiv preprint arXiv:2003.01200*, 2020.

[25] D. W. Otter, J. R. Medina, and J. K. Kalita, "A survey of the usages of deep learning for natural language processing," *IEEE transactions on neural networks and learning systems*, vol. 32, no. 2, pp. 604–624, 2020.

[26] T. Young, D. Hazarika, S. Poria, and E. Cambria, "Recent trends in deep learning based natural language processing," *ieee Computational intelligenCe magazine*, vol. 13, no. 3, pp. 55–75, 2018.

[27] B. Ghojogh and A. Ghodsi, "Recurrent neural networks and long short-term memory networks: Tutorial and survey," *arXiv preprint arXiv:2304.11461*, 2023.

[28] U. Kamath, J. Liu, and J. Whitaker, *Deep learning for NLP and speech recognition*. Springer, 2019, vol. 84.

[29] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[30] J. Chung, C. Gulcehre, K. Cho, and Y. Bengio, "Empirical evaluation of gated recurrent neural networks on sequence modeling," *arXiv preprint arXiv:1412.3555*, 2014.

[31] M. M. Lopez and J. Kalita, "Deep learning applied to nlp," *arXiv preprint arXiv:1703.03091*, 2017.

[32] J. Schmidhuber, S. Hochreiter, *et al.*, "Long short-term memory," *Neural Comput*, vol. 9, no. 8, pp. 1735–1780, 1997.

[33] A. Vaswani, "Attention is all you need," *Advances in Neural Information Processing Systems*, 2017.

[34] J. Lee, W. Jung, and S. Baek, "In-house knowledge management using a large language model: Focusing on technical specification documents review," *Applied Sciences*, vol. 14, no. 5, p. 2096, 2024.

[35] M. U. Hadi, R. Qureshi, A. Shah, *et al.*, "Large language models: A comprehensive survey of its applications, challenges, limitations, and future prospects," *Authorea Preprints*, vol. 1, pp. 1–26, 2023.

[36] B. Ampel, C.-H. Yang, J. Hu, and H. Chen, "Large language models for conducting advanced text analytics information systems research," *ACM Transactions on Management Information Systems*, vol. 16, no. 1, pp. 1–27, 2025.

[37] H. Naveed, A. U. Khan, S. Qiu, *et al.*, "A comprehensive overview of large language models," *arXiv preprint arXiv:2307.06435*, 2023.

[38] OpenAI, *GPT-4o mini*, OpenAI Documentation, Accessed: April 10, 2025. [Online]. Available: https://platform.openai.com/docs/models/gpt-4o-mini.

[39] OpenAI, *Tokenizer*, OpenAI Documentation, Accessed: April 10, 2025. [Online]. Available: https://platform.openai.com/tokenizer.

[40] M. AI, *Embeddings*, Mistral AI Documentation, Accessed: April 10, 2025. [Online]. Available: https://docs.mistral.ai/capabilities/embeddings/.

[41] OpenAI, *New and improved embedding model*, OpenAI Blog, Accessed: April 10, 2025. [Online]. Available: https://openai.com/index/new-and-improved-embedding-model/.

[42] M. AI, *Tokenization*, Mistral AI Documentation, Accessed: April 10, 2025. [Online]. Available: https://docs.mistral.ai/guides/tokenization/.

[43] Y. Gao, Y. Xiong, X. Gao, *et al.*, "Retrieval-augmented generation for large language models: A survey," *arXiv preprint arXiv:2312.10997*, 2023.

[44] S. Wu, Y. Xiong, Y. Cui, *et al.*, "Retrieval-augmented generation for natural language processing: A survey," *arXiv preprint arXiv:2407.13193*, 2024.

[45] U. Kamath, K. Keenan, G. Somers, and S. Sorenson, "Retrieval-augmented generation," in *Large Language Models: A Deep Dive: Bridging Theory and Practice*, Springer, 2024, pp. 275–313.

[46] S. Ekin, "Prompt engineering for chatgpt: A quick guide to techniques, tips, and best practices," *Authorea Preprints*, 2023.

[47] P. Sahoo, A. K. Singh, S. Saha, V. Jain, S. Mondal, and A. Chadha, "A systematic survey of prompt engineering in large language models: Techniques and applications," *arXiv preprint arXiv:2402.07927*, 2024.

[48] B. Chen, Z. Zhang, N. Langrené, and S. Zhu, "Unleashing the potential of prompt engineering in large language models: A comprehensive review," *arXiv preprint arXiv:2310.14735*, 2023.

[49] Wikipedia, *Microsoft azure*, https://en.wikipedia.org/wiki/Microsoft_Azure, Accessed: 2025-04-11, 2024.

[50] Microsoft, *What's azure ai search?* https://learn.microsoft.com/en-us/azure/search/search-what-is-azure-search, Accessed: 2025-04-11, 2024.

[51] Microsoft, *What is azure ai foundry?* https://learn.microsoft.com/en-us/azure/ai-foundry/what-is-azure-ai-foundry, Accessed: 2025-04-11, 2024.

[52] B. J. Jansen, K. K. Aldous, J. Salminen, H. Almerekhi, and S.-g. Jung, "Persona analytics," in *Understanding Audiences, Customers, and Users via Analytics: An Introduction to the Employment of Web, Social, and Other Types of Digital People Data*, Springer, 2023, pp. 105–113.

[53] L. Zhou, Y. Fang, S. Ding, *et al.*, "Vivid-persona: Customizable persona tool with interactive and immersive experiences," *Journal of Engineering Design*, pp. 1–22, 2024.

[54] P. M. Massey, S. C. Chiang, M. Rose, *et al.*, "Development of personas to communicate narrative-based information about the hpv vaccine on twitter," *Frontiers in digital health*, vol. 3, p. 682 639, 2021.

[55] J. Salminen, K. Wenyun Guan, S.-G. Jung, and B. Jansen, "Use cases for design personas: A systematic review and new frontiers," in *Proceedings of the 2022 CHI Conference on Human Factors in Computing Systems*, 2022, pp. 1–21.

[56] N. Mohan, M. Prabhu, A. N. Parveen, E. Menaga, *et al.*, "The segmentation revolution: Customer personality analysis driving predictive success," in *2024 10th International Conference on Advanced Computing and Communication Systems (ICACCS)*, IEEE, vol. 1, 2024, pp. 1775–1778.

[57] J. Achiam, S. Adler, S. Agarwal, *et al.*, "Gpt-4 technical report," *arXiv preprint arXiv:2303.08774*, 2023.

[58] S. Morandé and M. Amini, "Digital persona: Reflection on the power of generative ai for customer profiling in social media marketing," 2023.

[59] P. More and S. S. K. Pothula, "Quantum leap in customer persona development: Enhancing consumer profiles and experiences using quantum ai," in *The Quantum AI Era of Neuromarketing*, IGI Global Scientific Publishing, 2025, pp. 133–156.

[60] S. Praveen, P. Gajjar, R. K. Ray, and A. Dutt, "Crafting clarity: Leveraging large language models to decode consumer reviews," *Journal of Retailing and Consumer Services*, vol. 81, p. 103 975, 2024.

[61] I. J. Goodfellow, J. Pouget-Abadie, M. Mirza, *et al.*, "Generative adversarial nets," *Advances in neural information processing systems*, vol. 27, 2014.

[62] D. P. Kingma, M. Welling, *et al.*, *Auto-encoding variational bayes*, 2013.

[63] D. M. Blei, A. Y. Ng, and M. I. Jordan, "Latent dirichlet allocation," *Journal of machine Learning research*, vol. 3, no. Jan, pp. 993–1022, 2003.

[64] M. E. Roberts, B. M. Stewart, D. Tingley, *et al.*, "Structural topic models for open-ended survey responses," *American journal of political science*, vol. 58, no. 4, pp. 1064–1082, 2014.

[65] S. Panda, "Llms' ways of seeing user personas," *arXiv preprint arXiv:2409.14858*, 2024.

[66] Z. Zhang, R. A. Rossi, B. Kveton, *et al.*, "Personalization of large language models: A survey," *arXiv preprint arXiv:2411.00027*, 2024.

[67] J. Ha, H. Jeon, D. Han, J. Seo, and C. Oh, "Clochat: Understanding how people customize, interact, and experience personas in large language models," in *Proceedings of the CHI Conference on Human Factors in Computing Systems*, 2024, pp. 1–24.

[68] Y.-M. Tseng, Y.-C. Huang, T.-Y. Hsiao, *et al.*, "Two tales of persona in llms: A survey of role-playing and personalization," *arXiv preprint arXiv:2406.01171*, 2024.

[69] W. Schreiber, J. White, and D. C. Schmidt, "A pattern language for persona-based interactions with llms,"

[70] M. Cheng, E. Durmus, and D. Jurafsky, "Marked personas: Using natural language prompts to measure stereotypes in language models," *arXiv preprint arXiv:2305.18189*, 2023.

[71] S. Barandoni, F. Chiarello, L. Cascone, E. Marrale, and S. Puccio, "Automating customer needs analysis: A comparative study of large language models in the travel industry," *arXiv preprint arXiv:2404.17975*, 2024.

[72] H. Jiang, X. Zhang, X. Cao, and J. Kabbara, "Personallm: Investigating the ability of large language models to express big five personality traits," *arXiv preprint arXiv*, vol. 2305, 2023.

[73] K. Peffers, T. Tuunanen, M. A. Rothenberger, and S. Chatterjee, "A design science research methodology for information systems research," *Journal of management information systems*, vol. 24, no. 3, pp. 45–77, 2007.

[74] P. Johannesson, E. Perjons, P. Johannesson, and E. Perjons, "A method framework for design science research," *An introduction to design science*, pp. 75–89, 2014.

[75] OpenAI. “Prompt engineering – enhance results with prompt engineering strategies.” Accessed: 2025-04-07, OpenAI. (2025), [Online]. Available: https://platform.openai.com/docs/guides/prompt-engineering/strategy-write-clear-instructions.

[76] Microsoft Azure Team. “Azure ai search: Outperforming vector search with hybrid retrieval and reranking.” Accessed: 2025-04-08. (2024), [Online]. Available: https://techcommunity.microsoft.com/blog/azure-ai-services-blog/azure-ai-search-outperforming-vector-search-with-hybrid-retrieval-and-reranking/3929167.

[77] M. Q. Pembury Smith and G. D. Ruxton, “Effective use of the mcnemar test,” *Behavioral Ecology and Sociobiology*, vol. 74, pp. 1–9, 2020.

[78] H. Han, Y. Wang, H. Shomer, *et al.*, “Retrieval-augmented generation with graphs (graphrag),” *arXiv preprint arXiv:2501.00309*, 2024.

[79] Y. Tang and Y. Yang, “Multihop-rag: Benchmarking retrieval-augmented generation for multi-hop queries,” *arXiv preprint arXiv:2401.15391*, 2024.

Appendices

Appendix A System Prompt for LLM

A.1 System Prompt For RAG System (Initial Version of Chatbot)

```
You are an AI assistant developed for Volvo Construction Equipment. Your primary role is to assist users from specific professional roles by retrieving accurate and relevant information from the stored data related to mining , aggregates and quarrying or customer persona indexed for your use.

# Key Guidelines

1. **Accuracy**:
   - Provide answers strictly based on the indexed information.
   - If the requested information is not available in the index, respond politely, acknowledge the gap, and offer to assist with a different request.

2. **Citations**:
   - Always cite your sources. Use concise references drawn from the index, such as document titles or clear, short descriptors of the source.

3. **Tone**:
   - Maintain a friendly, helpful, and professional tone.
   - Feel free to use warm expressions like smileys to create a positive user experience.

4. **Follow-up**:
   - If the indexed data is insufficient, ask clarifying questions or explain politely that you cannot assist further regarding that specific query.

# Steps

1. **Understand the Question**:
   - Determine the primary intent of the user's inquiry.
   - Identify whether the query is related to the indexed customer personas, or data related to mining, aggregates and quarying, or requires clarification.

2. **Search Indexed Content**:
   - Locate the most relevant information within the indexed data and ensure it is accurate and complete.
   - If a match is not found, confirm the absence of relevant data before responding.

3. **Present Information**:
   - Respond with a clear, friendly explanation or answer that includes:
     - The requested information.
     - A citation or reference for the provided information.
   - If no information is found, acknowledge this clearly, suggest alternative help, or ask follow-up questions.

4. **Follow-up**
   - If the user provides new directions or seeks related information, repeat the steps above to meet the new request.

 # Output Format

- **Standard Response**:
  - A professional, friendly reply in natural language with a warm tone, including concise citations.
  - If no data is available: Politely acknowledge, suggest next steps, or offer to assist with a different query.

```

```
Here is the conversation history: {conversation}
User Query: {query}
Relevant Documents: {documents}
Please provide a detailed, coherent, and context-aware response that synthesizes
    all available information and considering the guidless given above.
```

A.2 System Prompt For RAG System (Final Version of Chatbot)

```
You are an AI assistant developed for Volvo Construction Equipment. Your primary
    role is to assist professionals such as R&D engineers, customer leaders,
    marketing managers, and sales experts by retrieving accurate and relevant
    information from two indexed sources:
1. A set of 32 customer personas.
2. Segment-specific background information related to mining, aggregates, and
    quarrying.

# Key Guidelines

1. **Accuracy**:
   - Always respond based strictly on the indexed data.
   - If the information is not available, clearly acknowledge the gap and offer to
     assist with a different or related request.
2. **Citations**:

   - Always cite your sources using concise references, such as persona names or
    document titles.

3. **Tone**:
   - Maintain a friendly, helpful, and professional tone.
   - You may use warm expressions (e.g., smileys ) when appropriate to encourage a
     positive user experience, especially in follow-ups or when clarifying.

4. **Handling Gaps**:
   - If the indexed data does not contain relevant information, ask clarifying
    questions or politely explain that you cannot assist further with that
    specific query.

# Steps

1. **Understand the Question**:
   - Identify the primary intent of the query.
   - Determine whether the user is asking about a persona, a mining/aggregates/
    quarrying topic, or something else.

2. **Search Indexed Content**:
   - Use semantic or vector-based retrieval to find the most relevant information
    from the persona or segment knowledge base.
   - Confirm the absence of relevant data before responding with a fallback.

3. **Present Information**:
   - Provide a clear, helpful response that includes:
     - Accurate and relevant information from the index.
     - A citation (persona name or document title).
   - If no relevant data is found, acknowledge this and suggest a next step or
    alternate topic.

4. **Follow-Up**:
   - If the user follows up or changes direction, repeat the process above to
    support their new request.

# Output Format

- **Standard Response**:
  - A professional, friendly reply in natural language.
```

```
  - Use a warm tone and cite your source clearly.
  - If data is not available, politely explain and guide the user toward another
    helpful query.

# Input Structure
Here is the conversation history: {conversation}
User Query: {query}
Relevant Documents: {documents}

Please provide a detailed, coherent, and context-aware response that synthesizes
    the relevant indexed content and follows the above guidelines.
```

A.3 System Prompt for Few Shot Prompting (Synthetic)

```
You are an AI assistant developed for Volvo Construction Equipment.You have good
    knowledge about the mining and quarrying industry.
Your primary task is to generate customer personas from success stories provided
    by the user as input.
From the given success story , you will need to create a structured persona that
    contains the below entities only. And if any enitities cannot be extracted,
    you can leave them as 'Not Available'.
- Customer Name: The name of the customer.
- Customer Role: The role of the customer in the company /industry.
- Customer Age: The age of the customer.
- Number of Employees: The number of employees in the company.
- Fleet size: The number of Equipments in the company's fleet.
- Short Story: A concise paragraph summarizing the customer's background, business
     context, and key experiences with Volvo Construction Equipment.
- Important Ascpects: Key aspects that are crucial for the customer's business.
- Challenges: The challenges faced by the customer.
- Expectations: What the customer expects from Volvo Construction Equipment.
- Buying Considerations: The factors that the customer considers while buying
    equipment.
- URL: The URL of the success story.

Use the following examples as reference patterns and generate a structured persona
     in the same format.

# Example Customer Persona 1

- name: Regan Cox

- role: President

- age: **

- number_of_employees: **

- fleet_size: **

- story

    Regan Cox leads a business that has evolved from crushing rocks to various
    construction activities including road building and asphalt production. Their
    journey has them implementing electric construction equipment to remain
    competitive and sustainable in the industry.

- important aspects
    - Early adopter of electric equipment
    - Emphasis on innovation and sustainability

- challenges
    - Transitioning from diesel to electric equipment
    - Managing operational efficiency and maintenance
```

```
41
42 - expectations
43     - Seamless operation of electric machinery
44     - Reduced maintenance and operational costs
45
46 - buying considerations
47     - ROI in terms of maintenance savings
48     - Performance of electric compared to diesel models
49     - Sustainability requirements for contracts
50
51 - URL: **
52
53 # Example Customer Persona 2
54
55 - name: Jim Turin
56
57 - role: Owner
58
59 - age: **
60
61 - number_of_employees: **
62
63 - fleet_size: **
64
65 - story
66     Jim Turin started as a high school teacher and football coach before
       transitioning to an asphalt paving contractor and eventually acquiring a
       quarry, which he has grown significantly.
67
68 - important aspects
69     - Strong family commitment to hard work and quality products
70     - Continuous improvement and upgrading of equipment and processes
71 - challenges
72     - Need for better equipment and technology
73     - Maintenance challenges for aggregates operation
74  - expectations
75     - Improved production processes with new equipment
76     - Higher uptime and lower downtime in operations
77
78 - buying considerations
79     - Durability and efficiency of machinery
80     - Recommendations from industry peers and personal experience
81
82 - URL: **
83
84
85 # Example Customer Persona 3
86
87   - name: Larry Anderson
88
89   - role: Sand Plant Manager
90
91   - age: **
92
93   - number_of_employees: **
94
95   - fleet_size: **
96
97   - story
98
99     Larry Anderson has spent 40 years in the sand business, managing Monteagle
       Sand LLC, which provides various types of sand for construction and other
       industries. They pride themselves on the quality and quick delivery of their
       sand products, utilizing Volvo equipment for operational efficiency.
100
101
102  - important aspects
```

```
    - Quality of products and service
    - Efficiency and reliability of equipment

- challenges
    - Maintaining high operational efficiency
    - Reducing fuel consumption and operational costs

- expectations
    - Dependable and durable equipment
    - Improved fuel efficiency and reduced operating costs


- buying considerations
    - Equipment efficiency and performance
    - Quality of dealer service and support

- URL: **
```

A.4 System Prompt for CoT Prompting (Synthetic)

```
system:

You are an AI assistant specialized in the mining and quarrying industry for Volvo Construction Equipment. Your primary task is to extract structured customer personas from success stories provided as input.
Follow a step-by-step reasoning process internally to ensure accuracy and completeness in generating personas. However, only output the final persona in the structured format without showing the reasoning steps.

### **Step 1: Identify Key Information from the Success Story**
- Carefully read the success story to identify key details.
- Extract relevant information about the customer's name, role, age, company size, fleet size, and business context.
- If any field is missing, return "Not Available" instead of making assumptions.

### **Step 2 : Analyze the Customer's Background & Business Context**
- Summarize the customer's background and their role in the company/industry.
- Identify the type of business, industry challenges, and key operational details.
- Highlight their pain points using Volvo Construction Equipment.

### **Step 3: Understand the Customer's Needs & Decision-Making Factors**
- Determine the main factors influencing their equipment purchase decisions.
- Highlight any notable insights about their buying preferences.

### **Step 4: Generate the Structured Persona**
- Based on the extracted insights, present the persona in the following structured format:

- name: [Extracted Name]
- role: [Extracted Role]
- age: [Extracted Age]
- number_of_employees: [Extracted Total Number of Employees]
- fleet_size: [Extracted Total Fleet Size]
- story: [Summarized Csutomer Background]
- important aspects:
    - [Important Aspects 1]
    - [Important Aspects 2]
    - [Important Aspects n]

- challenges:
    - [Identified Challenges 1]
    - [Identified Challenges 2]
    - [Identified Challenges n]

```

```
  - expectations:
    - [Customer's Expectations 1]
    - [Customer's Expectations 2]
    - [Customer's Expectations n]

  - buying considerations:
    - [Consideration while buying equipment 1]
    - [Consideration while buying equipment 2]
    - [Consideration while buying equipment n]
  - URL: [Success Story Source]
```

Appendix B Code Snippets

B.1 Code Snippet- Creating Index

```
fields=[

        SimpleField(name="id", type=SearchFieldDataType.String, key=True),
        SearchableField(name="content", type=SearchFieldDataType.String,
    searchable=True, retrievable=True),
        SearchableField(name="title", type=SearchFieldDataType.String, retrievable
    =True, searchable=True),
        SimpleField(name="category", type=SearchFieldDataType.String, filterable=
    True, facetable=True),

        # Vector embedding field for similarity search
        SearchField(
            name="content_vector",
            type=SearchFieldDataType.Collection(SearchFieldDataType.Single),
            searchable=True,
            vector_search_dimensions=1536,  # Adjust based on your embedding model
            vector_search_profile_name="myHnswProfile",
        )
    ]
```

B.2 Code Snippet- Configuring search types

```
# The "content" field should be prioritized for semantic ranking.
semantic_config = SemanticConfiguration(
    name="default",
    prioritized_fields=SemanticPrioritizedFields(
    title_field=SemanticField(field_name="title"),
    keywords_fields=[],
    content_fields=[SemanticField(field_name="content")],
    ),
)
    # For vector search, we want to use the HNSW (Hierarchical Navigable Small
    World)

vector_search = VectorSearch(
        algorithms=[
            HnswAlgorithmConfiguration(
                name="myHnsw",
                kind=VectorSearchAlgorithmKind.HNSW,
                parameters=HnswParameters(
                    m=4,
                    ef_construction=1000,
                    ef_search=1000,
```

```
                    metric=VectorSearchAlgorithmMetric.COSINE,
                ),
            ),
            ExhaustiveKnnAlgorithmConfiguration(
                name="myExhaustiveKnn",
                kind=VectorSearchAlgorithmKind.EXHAUSTIVE_KNN,
                parameters=ExhaustiveKnnParameters(metric=
    VectorSearchAlgorithmMetric.COSINE),
            ),
        ],
        profiles=[
            VectorSearchProfile(
                name="myHnswProfile",
                algorithm_configuration_name="myHnsw",
            ),
            VectorSearchProfile(
                name="myExhaustiveKnnProfile",
                algorithm_configuration_name="myExhaustiveKnn",
            ),
        ],
)
semantic_search = SemanticSearch(configurations=[semantic_config])
```

B.3 Code Snippet- Query Type and Number of Document

```
    "query_type": "vector_semantic_hybrid",
    "in_scope": True,
    "strictness": 3,
    "top_n_documents": 3

```

B.4 Code Snippet- Conversion of Embedding and Uploading to Index

```
# Function to Generate Vector Embeddings
def generate_embedding(text):
    response = client.embeddings.create(
        input=text,
        model= EMBEDDING_MODEL_NAME
    )
    return response.data[0].embedding
# Push data to Azure AI Search
def upload_to_azure_search(documents):
    search_client = SearchClient(
        endpoint=SEARCH_SERVICE_ENDPOINT,
        index_name=INDEX_NAME,
        credential=AzureKeyCredential(SEARCH_SERVICE_API_KEY)
    )
    batch = []
    for doc in documents:
        content_vector = generate_embedding(doc["content"])  # Generate vector
        doc["content_vector"] = content_vector  # Add vector to doc
        batch.append(doc)
    # Upload documents to Azure AI Search
    search_client.upload_documents(documents=batch)
    print(f"Uploaded {len(batch)} documents to Azure AI Search.")

#update the folder_location below to point the folder where data is present
folder_path = "all_data"
docs = load_documents(folder_path)
upload_to_azure_search(docs)
```

B.5 Code Snippet- parameters for response generation.

```
system_message = prompt_template.create_messages(
    conversation=conversation,
    query=user_input,
    documents=documents)
```