2024 4th International Conference on Ubiquitous Computing and Intelligent Information Systems (ICUIS) | 979-8-3315-2963-5/24/$31.00 ©2024 IEEE | DOI: 10.1109/ICUIS64676.2024.10867024

Applying Generative AI to Create SOP, Reducing API Costs Through Prompt Compression and Evaluating LLM Responses with Tonic Validate RAG Metrics

T. Vetriselvi
Asst. Professor, Dept. of IOT
SCOPE
Vellore Institute of Technology
Vellore, India
vetriselvi.t@vit.ac.in

Mihir Mathur
Dept. of CSE with Data Science, SCOPE
Vellore Institute of Technology
Vellore, India
mihir.mathur2020@vitstudent.ac.in,
mihir1mathur@gmail.com

M. Bhuvaneswari
Asst. Professor, Dept. of IOT
SCOPE
Vellore Institute of Technology
Vellore, India
m.bhuvaneswari@vit.ac.in

Abstract— **Generative Artificial Intelligence (AI) utilizes existing data to create new forms of content, including text, images, and audio. One valuable application of generative AI in an IT Enabled Services (ITES) company is the development of Standard Operating Procedures (SOPs) by processing and consolidating existing SOPs through a Large Language Model. This paper explores how generative AI can generate content within standardized templates, improve the language of SOPs, and make them suitable for specific industries and processes, such as the pharmaceutical procure-to-pay process. It also addresses version control, helps users maintain consistency and compliance, extracts knowledge for onboarding, and provides interactive training on SOPs through Questions and Answers. The proposed workflow employs Azure OpenAI's GPT-3.5 turbo for generating responses, which are evaluated using Tonic Validate Retrieval Augmented Generation (RAG) metrics. Furthermore, the paper introduces Prompt Compression approaches and selects one prompt compression approach to streamline the context retrieved for large language models while preserving the semantic meaning of the outputs. It also details strategies for reducing API call costs by over 10% for prompts of varying token sizes, ensuring high-quality responses from the GPT-3.5 turbo model according to RAG metrics. These criteria of accuracy, precision and cost reduction are considered to recommend the prompt compression approach for Large Language Models used in development of Standard Operating Procedures.**

Keywords—Generative AI, Artificial Intelligence, Large Language Models, Azure OpenAI LLM (GPT-3.5 turbo), Standard Operating Procedures, Prompt Compression, Stemming, NLTK, word tokenize, Retrieval Augmented Generation (RAG), Tonic Validate, API call cost

I. INTRODUCTION

Generative AI, a branch of artificial intelligence, involves systems that create new content similar to existing examples. Leveraging machine learning, particularly neural networks, these systems understand and replicate patterns within training data. [1]. Goldman Sachs estimates that generative AI could increase global Gross Domestic Product (GDP) by 7%. It is estimated that generative AI will impact 40% of all working hours across every industry. According to 97% of global executives, AI foundation models will facilitate connections between different data types, transforming the ways and areas in which AI is applied.

Applications such as OpenAI's ChatGPT [2], Google Bard, and Microsoft Bing AI, which are powered by large language models (LLMs), use machines to generate detailed, human-like content, understand context, infer intent, and demonstrate independent creativity. They can be used for various applications to transform work and reinvent the ways of doing business. Generative AI will maximize efficiency and drive competitive advantage. Responsible AI is important in a culture of innovation and experimentation. In text generation, applications include content creation for articles, product descriptions, and poetry, as well as improving chatbot responses through Natural Language Processing (NLP). Image generation utilizes Generative Adversarial Networks (GANs) for art, design, and style transfer between images. Video synthesis involves deepfake technology, creating realistic videos by superimposing one person's likeness onto another's actions. In drug discovery, generative models assist in generating new molecular structures, while in medical imaging, they provide synthetic images for data augmentation. For anomaly detection, these models identify irregularities in financial transactions, supporting fraud detection. Generative AI also plays a role in multilingual translation, preserving the style and tone of original content. Simulation and training benefit from generative models, which simulate realistic environments for training autonomous vehicles, drones, and robots. Although generative AI is highly versatile, it raises ethical concerns, especially regarding the creation of realistic fake content and its potential misuse. Consequently, it is essential to develop and follow ethical guidelines and regulations as the technology progresses.

Generative AI creates content through a process of learning and replicating patterns from existing data during a training phase [7]. The main method employs neural networks, a type of machine learning model modelled after the human brain. Generative AI enhances the creation of Standard Operating Procedures (SOPs) by analyzing data to identify best practices, generating standardized templates, and suggesting content improvements. It automates updates based on regulatory changes, aids in training, and manages version

979-8-3315-2963-5/24/$31.00 ©2024 IEEE

control. This integration leads to increased efficiency, consistency, and adaptability in SOP management.

1. **Data Collection:** Generative AI models require a substantial amount of data to learn from. For text generation, it includes a collection of books, articles, or other relevant text data, while for image generation, it could be a dataset of images. Standard Operating Procedures were collected and ingested to generate the final SOP using Azure Open AI LLM (GPT-3.5 Turbo).The SOPs used were Invoice Processing documents from 10 countries which were cleansed like removing sensitive data like company's name, ERP information and vendor details etc.
2. **Training the Model:** The chosen neural network is trained using the gathered data. For generative models such as GANs (Generative Adversarial Networks), there are usually two neural networks at play: a generator and a discriminator. The generator creates content, and the discriminator evaluates it. The models engage in a feedback loop during training, with the generator improving its ability to create content that is more difficult for the discriminator to distinguish from real data.
3. **Pattern Recognition:** The neural network learns patterns and relationships within the data during training. For text generation, this might involve understanding syntax, grammar, and semantic relationships. In image generation, the model learns features, textures, and structures.
4. **Content Generation:** Once the model is trained, it can generate new content by creating variations of the learned patterns. For example, in text generation, the model might complete sentences, paragraphs, or even generate entirely new text based on the learned style. In image generation, it can produce new images with similar styles and structures as the training data.
5. **Fine-Tuning (Optional):** Depending on the application, the generated content may undergo additional fine-tuning to meet specific criteria or adhere to certain constraints.

Some of the applications of generative AI [2, 3, 5] are tabulated below.

TABLE 1: FEW APPLICATIONS OF GENERATIVE AI

Content Creation: ▪ Generate articles, blogs, social media posts. ▪ Produce advertising copy and marketing materials. ▪ Create poetry, stories, and creative writing.	**Virtual Assistants:** ▪ Provide customer support via chatbots & voice assistants. ▪ Offer personalized recommendations and assistance. ▪ Assist in task management, scheduling, and reminders.
Design and Art: ▪ Generate visual designs like logos and graphics. ▪ Create artwork, including paintings and illustrations. ▪ Develop 3D models and virtual environments.	**Entertainment and Gaming:** ▪ Develop video game characters, levels, and scenarios. ▪ Produce movie scripts and plotlines. ▪ Compose music and create sound effects.
Data Augmentation and Simulation: ▪ Generate synthetic data for training machine learning models. ▪ Simulate realistic scenarios for research and development. ▪ Enhance data privacy by creating anonymized datasets.	**Language Translation and Natural Language Processing:** ▪ Translate text between languages. ▪ Summarize long articles and documents. ▪ Perform sentiment analysis and topic modeling.
Drug and Material Design: ▪ These methods are used to assist in designing new molecules, predicting their properties, and optimizing drug candidates. ▪ To explore chemical space and discover novel compounds with desired properties.	**Synthetic Data Generation:** ▪ Generate synthetic data for training machine learning models, which is valuable when real data is limited or sensitive. ▪ For example, generate realistic medical images for training diagnostic algorithms.
Generative Engineering and Design: ▪ Optimize engineering designs by exploring different configurations ▪ Create innovative architectural designs, product prototypes, and even entire city layouts.	**Personalization and Recommendation Systems:** ▪ Personalize recommendations for users, e.g. generate personalized movie or music playlists based on individual preferences. ▪ Create customized product designs or fashion recommendations.

This paper is organized in 5 chapters. Chapter 2 provides related work though literature review of various papers to explore the scope and development of Generative AI solutions. Chapter 3 outlines the architecture for creating Standard Operating Procedures using Large Language Models (LLMs). Following it is Chapter 4, which outlines a specific workflow for generating responses utilizing Azure Open AI LLM (GPT-3.5 Turbo), and then evaluating those responses using Tonic Validate Metrics, and Prompt compression along with API call cost reduction. Finally, Chapter 5 presents the conclusion and future steps.

II. RELATED WORK

A summary of artificial intelligence, machine learning, deep learning, artificial neural networks, and large language models (LLMs) is important to understand [8]. It also examines the applications and implications of generative AI in business and education, concluding with a discussion of the current challenges in implementing generative AI.

Generative AI models like ChatGPT, Midjourney, and Deep Brain have emerged as groundbreaking technological advancements, able to generate original content, including written material, visuals, and multimedia [9]. These are a significant step towards achieving artificial general intelligence, with huge potential across various industries, like education, business, content creation, healthcare, etc. While there are immense opportunities, generative AI poses challenges in ethics, technology, regulations, and the economy, mainly due to the absence of human-centered AI (HCAI). For generative AI to be successful, it should focus on a human-centric approach that encompasses empathy, transparency, ethical considerations, and effective governance.

Generative AI stands out as a technological breakthrough with the potential to profoundly reshape businesses and society [10]. However, acknowledging its vast potential, it is crucial to recognize existing limitations that span across various industries, including issues related to privacy, ethics, and data ownership. Overcoming these limitations is essential for businesses to successfully incorporate generative AI into their operations. This paper offers viewpoints from sectors including marketing, healthcare, human resources, education, financial services, retail, office culture, production, and sustainability management.

The emergence of advanced language models, exemplified by GPT-3 and its successor GPT-4, marks a groundbreaking milestone in the realm of artificial

979-8-3315-2963-5/24/$31.00 ©2024 IEEE

intelligence. GPT-3, developed by OpenAI, boasts 175 billion parameters, enabling it to capture intricate patterns in vast datasets. The subsequent version, GPT-4, is assumed to enhance this groundwork, potentially extending the limits of language comprehension and creation to a greater extent. **The fascinating evolution of Generative AI** is summarized [6] in the table below.

TABLE 2: FASCINATING EVOLUTION OF GENERATIVE AI

Year	Milestone	Year	Milestone
1805	First Neural Network/ Linear Regression	1925	First Recurrent Neural Network Architecture
1943	Neural Nets introduced	1958	Multi-Layer Perceptron MLP (No Deep Learning)
1965	First Deep Learning	1967	Deep Learning by Stochastic Gradient Descent
1972	Published Artificial Recurrent Neural Networks	1979	Deep Convolution Neural Network
1980	Auto Encoders released	1986	Back Propagation invented
1988	Image Recognition Convolutional Neural Network	1990	Generative Adversarial Networks/ Curiosity
1991	First Transformers, Vanishing Gradient	1995	Release of LeNet-5, pioneering 7-level CNN
1997	Long Short-Term Memory (LSTM) introduced	2001	Introduction of NPLM
2014	Variational Autoencoder (VAE)	2014	Release of Generative Adversarial Networks (GAN)
2014	Release of Gated Recurrent Unit (GRU)	2015	Release of Diffusion Models
2016	Release of Wave Net	2017	Release of Transformers
2018	Release of Generative Pretraining Transformers (GPT) by OpenAI	2018	Google releases BERT, Bidirectional Encoder Representations from Transformers.
2019	Release of StyleGAN	2020	Release of wave2vec 2.0 by Meta AI
2021	Release of DALL.E	2022	Release of Latent Diffusion Models
2022	Release of DALL.E 2 by OpenAI	2022	Release of Midjourney
2022	Release of Stable Diffusion	2022	Release of ChatGPT by OpenAI
2022	Release of AudioLM by Google	2023	Release of GPT-4 by OpenAI
2023	Release of Falcon LLM	2023	Release of BARD by Google
2023	Azure Open AI by Microsoft	2023	Release of AutoGPT
2023	Release of LongNet	2023	Release of VoiceBox by Meta AI
2023	Release of LLaMA by Meta AI	2024	…. the exponential evolution continues

Generative AI is poised to significantly boost the global economy, with projections [11] suggesting it could contribute an annual $2.6 to $4.4 trillion across 63 use cases, enhancing the overall influence of artificial intelligence by 15 to 40 percent. This economic benefit is primarily expected in areas such as customer service, marketing, sales, software development, and research and development. Sectors like banking, technology, and life sciences stand to experience notable gains, with the banking sector alone potentially adding $200 billion to $340 billion each year. With generative AI's capacity to automate tasks that currently take up 60 to 70 percent of workers' time, it is anticipated to transform the workforce, potentially automating up to half of all job activities by the period between 2030 and 2060. While generative AI can significantly enhance labour productivity, investments are required to support workers in adapting to changing job requirements. Integrating generative AI with various other technologies has the potential to add between 0.2 and 3.3 percentage points to annual productivity growth.

The synergy of AI-driven Optical Character Recognition (OCR) and Generative AI [12] revolutionizes document handling by automating processes to enhance both speed and precision. OCR captures and transforms physical documents into editable text, interpreting both printed and handwritten content, thereby facilitating the shift to digital mediums. When paired with OCR, AI algorithms classify texts, simplifying the categorization of documents. It also enhances error detection by pinpointing and rectifying inaccuracies from OCR. Together, OCR and Generative AI facilitate the extraction of data, analysis of sentiments, and translation of languages, catering to the needs of global enterprises. Furthermore, AI crafts adaptable document templates, demonstrating the profound influence of AI-empowered OCR and Generative AI on contemporary document administration and workflow optimization.

Generative AI transforms marketing by creating personalized, efficient, and data-driven experiences [13]. As organizations embrace this technology, they must balance its benefits with responsible use and ethical considerations. Generative AI impacts marketing activities through Personalized Content and Offerings, Efficiency and Productivity, Insights Generation and Sales Lead Generation. While Generative AI offers immense potential, organizations must address barriers such as data privacy, ethical considerations, and model interpretability.

AI applications are undergoing a significant transformation, moving from discriminative to generative models, which opens novel applications and fields, including historically hesitant ones to adopt such technologies [14]. Five core challenges are outlined: bias, transparency, hallucinations, misuse, and societal impact. There is immense potential of generative AI impacting diverse industries. It highlights the significance of clear technology practices, fostering confidence in generative AI among the public, and creating procedural frameworks for the implementation of systems based on Generative AI.

The impact of generative AI is also happening on managerial work, emphasizing its potential at strategic, functional, and administrative levels [15]. Generative AI is playing an increasing role in enhancing data-driven decision-making, aiding entrepreneurs, and managers in making informed judgments. The focus on knowledge management emphasizes its contribution to information sharing and idea generation within organizations. In administrative tasks, generative AI is expected to support work time organization and scheduling, though concerns about introducing a control mechanism are noted. The paper highlights the necessity for academic research to evaluate the enduring effects on workforce productivity, especially in terms of enhancement and the consequences of replacing tasks through automation.

Generative AI tools hold considerable promise in the educational sector [16], particularly when applied through the 4PADAFE instructional design framework. The study seeks to investigate the ways in which these tools, in tandem with the framework, can improve educational experience. The results suggest that the fusion of generative AI tools with the instructional design matrix is vital for creating expansive MOOC virtual classrooms. This amalgamation enables instructors to personalize educational material, engage learners in creative ways, and support individualized learning paths.

ChatGPT is an AI-driven conversational agent that utilizes sophisticated natural language processing techniques, like expansive language models, to produce responses that closely mimic human conversation across a wide array of subjects [17]. Despite concerns about maintaining academic honesty and potential barriers to enhancing students' critical thinking skills, the article presents a framework called DATS. This framework outlines an approach for refining the application of Generative AI in academic environments. It incorporates the insights of crucial stakeholders, including developers, administrators, teachers, and students, aiming to overcome these challenges and ensure its effective implementation.

Generative AI is also integrated across sectors, with significant impact on digital transformation and creativity [18]. Applications in academia, engineering, communications, healthcare, agriculture, government, and business offer increased productivity and personalized experiences. However, ethical challenges, safety concerns, and impacts on critical thinking are acknowledged. Bibliometric analysis highlights a significant expansion of Generative AI applications from 2014 to September 2023, driven by tools like ChatGPT.

Prompt compression is a cutting-edge method [19] that effectively streamlines input prompts while retaining crucial information. It aims to craft more succinct prompts for language models, enhancing their efficiency and focus. The Prompt Compression Toolkit (PCToolkit) stands as a complete package for Large Language Models. It boasts state-of-the-art Prompt Compressors, a variety of Datasets and Metrics, a Modular Structure, and Intuitive Interfaces. PCToolkit's compressors have been tested on a wide array of linguistic tasks, including Reconstruction, Summarization, Mathematical Problem Solving, Question Answering, Few-Shot Learning, Synthetic Tasks, Code Completion, Boolean Expressions, Multiple Choice Questions, and Lie Detection. Thus, prompt compression streamlines input prompts for better performance in language models across a wide range of tasks.

III. PROPOSED WORK

The proposed architecture of this model is to learn about Generative AI and its applications which an IT enabled service provider can use to add value to its customers and stakeholders. Generative AI is a valuable tool used in creating and enhancing Standard Operating Procedures (SOPs) in various ways, and answering the questions based on the generated SOP. In this paper, generative AI has been applied to SOP writing (Figure 1) as per the approach below:

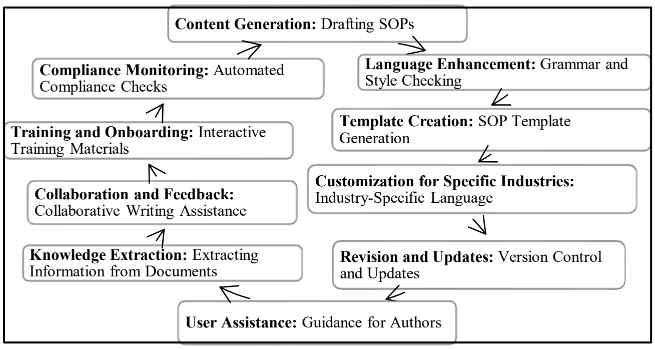
Content Generation: Drafting SOPs
Language Enhancement: Grammar and Style Checking
Template Creation: SOP Template Generation
Customization for Specific Industries: Industry-Specific Language
Revision and Updates: Version Control and Updates
User Assistance: Guidance for Authors
Knowledge Extraction: Extracting Information from Documents
Collaboration and Feedback: Collaborative Writing Assistance
Training and Onboarding: Interactive Training Materials
Compliance Monitoring: Automated Compliance Checks

Figure 1: Schematic Flow of creating SOPs using Generative AI

1. **Content Generation:** Generative model assists in the initial drafting of SOPs by generating content based on provided inputs. This can save time for human authors and ensure a consistent structure.
2. **Language Enhancement:** Generative AI has been employed to improve the language, grammar, and overall writing style of SOPs, ensuring clarity and adherence to organizational standards.
3. **Template Creation:** AI models using Azure Open AI LLM (GPT-3.5 Turbo) were used to analyze existing SOPs to create standardized templates, making it easier for teams to follow a consistent format when drafting new procedures.
4. **Customization for Specific Industries:** Generative AI models were trained on industry-specific terminology and standards, allowing for the generation of SOPs to sectors, like pharmaceutical industry for Procure to Pay processes.
5. **Revision and Updates:** Generative AI can assist in updating and revising SOPs, ensuring that documents remain accurate and compliant with evolving regulations or best practices.
6. **User Assistance:** AI-powered tools offer real-time suggestions and guidance to human authors as they create or edit SOPs, helping to maintain consistency and compliance.
7. **Knowledge Extraction:** Generative models analyze existing documents to extract relevant information, which can be used in the creation or revision of SOPs.
8. **Collaboration and Feedback:** AI facilitates collaboration by suggesting edits, providing feedback, and managing contributions from multiple authors during the SOP creation process.
9. **Training and Onboarding:** Generative AI was used to create interactive and engaging training materials based on SOPs, aiding in employee onboarding and continuous training efforts.
10. **Compliance Monitoring:** AI algorithms can be also used to check SOPs for compliance with industry standards, regulations, and internal policies, reducing the risk of non-compliance.

Generative AI was used at various stages in the content creation, using pharmaceutical procure to pay industry specific terminologies, and in knowledge extraction, training and onboarding new users for these Standard Operating Procedures. This model outlines a specific workflow (Figure 2) for generating responses using a Large Language Model (LLM), utilizing Azure Open AI LLM (GPT-3.5 Turbo) and

979-8-3315-2963-5/24/$31.00 ©2024 IEEE

then evaluating those responses using Tonic Validate RAG Metrics. This was done as per the following steps:

1. **Create Embeddings:** Convert the text data into numerical representations (embeddings). This step helps capture semantic information about the text.

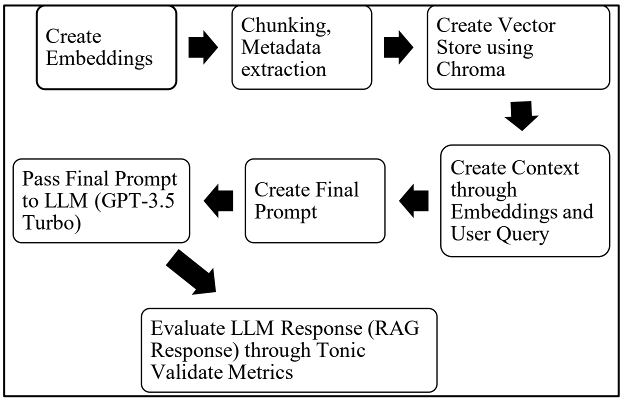
Create Embeddings
Chunking, Metadata extraction
Create Vector Store using Chroma
Create Context through Embeddings and User Query
Create Final Prompt
Pass Final Prompt to LLM (GPT-3.5 Turbo)
Evaluate LLM Response (RAG Response) through Tonic Validate Metrics

Figure 2: Specific Flow to Generate Responses using LLM (GPT 3.5 Turbo) and Evaluating Responses using Tonic Validate RAG metrics

2. **Chunking, Metadata Extraction:** Break down the text into chunks or segments and extract relevant metadata. This process helps organize and structure the information, especially when dealing with text extracted from PDFs.
3. **Create Vector Store using Chroma:** Utilize Chroma to create a vector store, likely representing the embeddings of the text. This vector store can be a repository of numerical representations that capture the semantic meaning of the text data.
4. **Create Context through Embeddings and User Query:** Combine the embeddings created in step 1 with the user's query to generate context. This context provides a foundation for generating relevant responses.
5. **Create Final Prompt:** Combine the generated context from step 4 with the user's query to form the final prompt. This prompt serves as input to the LLM (GPT-3.5 Turbo), guiding it to generate responses based on the given context and user query.
6. **Pass Final Prompt to LLM:** Feed the final prompt into the Azure Open AI LLM (GPT-3.5 Turbo) to generate a response. The LLM uses its pre-trained language understanding to produce contextually relevant text based on the input prompt.
7. **Evaluate LLM Response (RAG Response) through Tonic Validate Metrics:** Utilize Tonic Validate Metrics to evaluate the LLM (GPT-3.5 Turbo) response. Tonic is a tool for data testing, validation, and generation. In this context, it assesses the quality of the LLM response using predefined metrics, likely focusing on aspects such as coherence, relevance, and other criteria specified by the Tonic Validate Metrics.

IV. RESULT ANALYSIS

Prompt Compression is a technique used to make the retrieved context for language models more concise and focused. It plays a crucial role in scenarios where efficient information retrieval and processing are essential. The iterative prompt compression approaches considered are listed below. The final approach was selected based on the purpose, limitation of each and good Tonic Validate RAG metrics (overall score > 0.5) with suggested good responses to user queries with compressed form, thus ensuring semantic meaning of the prompt is retained.

1. LLMLingua Framework:

Purpose: To boost the performance of large language models (LLMs) by minimizing the length of prompts.
Method: LLMLingua detects and eliminate non-essential tokens from input prompts by well-trained small language models (GPT2-small or LLaMA-7B).
Effectiveness: While prompts compressed at the token level might be difficult for people to comprehend, they are highly efficient when processed by large language models (LLMs).
Application: LLMLingua's prompt compression method can be useful for accelerating inference in closed LLMs.

2. RAG-Based Applications:

Context: In applications based on Retrieval Augmented Generation (RAG) architecture, prompt compression is particularly beneficial.
Benefits:

- **Conciseness:** Compressed prompts provide a more focused context for language models.
- **Efficiency:** Efficient information retrieval and processing are crucial in RAG-based systems.

3. LongLLMLingua (for Long-Context Scenarios):

Purpose: Designed for scenarios with evolving information over time (e.g., retrieval-augmented question-answering tasks in chatbots or summarizing online meetings).
Application: LongLLMLingua enables efficient interaction with LLMs by condensing lengthy prompts without losing essential information.

4. Stop Word Removal and Punctuation Removal

Purpose: Stop words are common words in a language that usually have little impact on the overall significance of a sentence. Removing them can help in reducing the noise in the text and focusing on the more meaningful words, which are often called content words. By eliminating stop words, we decrease the size of the tokenized text while preserving its essential meaning. Removing punctuation helps in standardizing the text and focusing on the words themselves, rather than the punctuation marks. It can simplify the tokenization process and improve the efficiency of downstream text processing algorithms.
Limitation: Stop word removal and punctuation removal can sometimes lead to a loss of context or important grammatical information. Stop words may seem insignificant individually, but collectively they contribute to the overall structure and meaning of a sentence. Neglecting the domain-specific stop words can lead to loss of important context or meaning in specialized domains. Similarly, punctuation marks convey important cues about sentence boundaries, emphasis, and tone. Removing them indiscriminately can distort the original meaning of the text.

5. Porter Stemmer and Punctuation Removal

Purpose: The Porter stemming algorithm is a widely used method for reducing words to their base or root form, known as the stem. Its purpose is to normalize words by removing suffixes, which helps in reducing the token size by collapsing variations of words into a single representative form. For example, "running", "runs", and "ran" would all be stemmed to "run". This process reduces redundancy and variation in the text, making it easier to analyze semantically related words as a single entity. Porter stemming and punctuation removal contribute to reducing the token size of a prompt by simplifying the text representation and removing unnecessary variations and characters. To perform stemming using NLTK (Natural Language Toolkit), import the 'PorterStemmer' from the 'nltk.stem' module and word_tokenize from nltk.tokenize module into Python script.

Limitation: By applying Porter Stemmer, we can lose the semantic meaning of the original prompt and token reduction percentage is also much less.

- **Over-stemming**: The Porter stemming algorithm can sometimes be too aggressive in its stemming process, resulting in over-stemming. Over-stemming occurs when it removes suffixes that are part of the stem, leading to the loss of important information and generating incorrect word stems. For example, "university" might be stemmed to "univers" instead of the desired "universi".
- **Under-stemming**: Conversely, under-stemming can occur when the algorithm fails to remove suffixes that should be stemmed. This can result in different variations of the same root word not being collapsed into a single stem. For example, "running" and "runs" might not be stemmed to the same root form, leading to redundancy in the stemmed vocabulary.
- **Language-specific Limitations:** The Porter stemming algorithm is designed primarily for English text and may not perform as effectively for languages with different morphological structures or orthographic conventions. For languages with complex inflectional or derivational morphology, such as Arabic or Finnish, the Porter stemming algorithm may not produce accurate stems.
- **Ambiguity**: The algorithm may produce ambiguous stems for certain words, especially those with multiple meanings or interpretations. Ambiguity in stemming can lead to confusion and inaccuracies in downstream text analysis tasks, such as information retrieval or text classification.
- **Lack of Context Awareness:** The Porter stemming algorithm operates on individual words without considering the context in which they appear. This lack of context awareness can result in incorrect stem assignments, particularly in cases where the meaning of a word depends on its surrounding words or phrases.
- **Performance Trade-offs:** While the Porter stemming algorithm is computationally efficient and easy to implement, its simplicity comes with trade-offs in terms of accuracy and precision. More advanced stemming algorithms, such as the Snowball stemmer, address some of the limitations of the Porter stemmer but may require higher computational resources.

6. Semantic Search of User Query Response in Prompt:

Purpose: The purpose of semantic search in user query response is to deliver more relevant, accurate, and personalized results by understanding the meaning, context, and intent behind the user's queries and the content being searched.

Limitation: It is not a generic solution for prompt compression. It faces limitations related to contextual understanding, semantic gap, training data, trade-offs between compression and relevance, and user expectations.

7. Article Removal and Selected Punctuation Marks Removal:

Purpose: Article removal, which involves eliminating definite and indefinite articles such as "the", "a", and "an", serves several purposes in prompt compression:

- **Reducing Token Size:** Articles, while commonly found in text, typically don’t hold substantial meaning on their own. Their exclusion can shrink the prompt’s token count, resulting in a more streamlined and succinct input that retains the necessary information.
- **Focus on Nouns and Verbs:** By removing articles, the emphasis shifts to the nouns and verbs in the prompt, which typically convey the core information and action. This streamlined focus can make the prompt more actionable and easier to process for downstream tasks such as summarization, question answering, or search.
- **Improving Readability:** Eliminating articles can improve the readability of the compressed prompt by removing unnecessary words that may clutter the text. This can make the prompt clearer and more understandable, especially in contexts where brevity is valued, such as mobile interfaces or voice assistants.
- **Enhancing Search Relevance:** In search applications, removing articles can help improve search relevance by reducing noise and focusing on the keywords that are more likely to match the user's query. This can lead to more accurate search results and better user satisfaction.
- **Normalization:** Removing articles helps normalize the prompt by standardizing the text and making it more consistent across different prompts and queries. This can simplify text processing tasks such as tokenization, stemming, and similarity comparison.
- **Preserving Semantic Meaning:** While articles may not always contribute significantly to the semantic meaning of a sentence, their removal should be done cautiously to ensure that the essential meaning and grammatical structure of the prompt are preserved. In some cases, articles may be necessary for conveying specificity or definiteness in the prompt, and their removal could alter the intended interpretation.

The success of this workflow depends on the quality of embeddings, the effectiveness of the vector store, the performance of the LLM, and the appropriateness of the evaluation metrics. Additionally, the specifics of the workflow can be adapted based on the requirements and characteristics of the application or system. Tonic Validate [20] is designed to assess, track, and monitor the performance of the LLM (GPT-3.5 turbo) and RAG applications. It provides a set of custom-built metrics that use LLMs to evaluate different aspects of the RAG system. Tonic Validate has six default metrics that are applicable to most RAG systems:

1. **Answer Similarity Score:** Evaluates alignment between the standard answer and the response produced by the large language model (LLM). [score between 0 and 5].

2. **Retrieval Precision:** Assesses whether the retrieved context is relevant for answering a given question [= Count of relevant retrieved context / Count of retrieved context].
3. **Augmentation Precision:** Assesses if the relevant context is present in the LLM answer [= Count of relevant retrieved context in LLM answer / Count of relevant retrieved context].
4. **Augmentation Accuracy:** Assesses if all the relevant context is included in LLM generated answer [= Count of retrieved context in LLM answer / Count of retrieved context].
5. **Answer Consistency:** Ensures consistency across multiple responses generated by the LLM [= Number of key points in answer derived from context/ Total number of key points in the answer]
6. **Question-Context Overlap:** Measures the overlap between the question and the retrieved context.

RAG (Retrieval Augmented Generation) metrics are a set of metrics designed for assessing various aspects, such as the retrieval system's proficiency in pinpointing pertinent and concentrated context passages, the language model's capability to utilize these passages faithfully, and the inherent quality of the generated content, all without depending on human-annotated ground truth.

TABLE 3: RETRIEVAL AUGMENTED GENERATION METRICS

Score	Input	What does it measure?	Evaluated components
Answer similarity score	▪ Question ▪ Reference answer ▪ LLM answer	Similarity between the reference answer and the LLM-answer.	All components
Retrieval precision	▪ Question ▪ Retrieved context	Relevance of the retrieved context to the posed question.	▪ Chunker ▪ Embedder ▪ Retriever
Augmentation precision	▪ Question ▪ Retrieved context ▪ LLM answer	Determines presence of relevant context within the LLM-answer	▪ Prompt builder ▪ LLM
Augmentation accuracy	▪ Retrieved context ▪ LLM answer	Checks if the entire context is reflected in the LLM-answer.	▪ Prompt builder ▪ LLM
Answer consistency Or Answer consistency binary	▪ Retrieved context ▪ LLM answer	Evaluates if the LLM answer contains information which is not sourced from the given context..	▪ Prompt builder ▪ LLM

By doing prompt compression, iteratively it was identified that by using stop word removal, article removal and punctuation removal, good quality compressed prompts in minimal time were obtained, while maintaining the semantic content of original prompt and ensuring good quality LLM responses as per the results tabulated below:

TABLE 4: AVERAGE PROMPT COMPRESSION TIME FOR TOKEN SIZE REDUCTION – STOP WORD REMOVAL

Token Size Range	Average Token Reduction % = [(Original Prompt Token-Compressed Prompt Token)]/(Original Prompt Token)*100	Average Compression Time (in seconds)
0-100	48.64	0.0019
100-200	36.15	0.0018
200-300	38.27	0.0020
300-400	35.81	0.0022
400-500	40.91	0.0022
500-600	36.23	0.0024
600-700	41.21	0.0023
700-800	40.09	0.0025
800-900	39.47	0.0031
900-1000	35.97	0.0026

TABLE 5: AVERAGE PROMPT COMPRESSION TIME FOR TOKEN SIZE REDUCTION – PORTER STEMMER

Token Size Range	Average Token Reduction % = [(Original Prompt Token-Compressed Prompt Token)]/(Original Prompt Token)*100	Average Compression Time (in seconds)
0-100	1.8229	0.0073
100-200	13.8499	0.0126
200-300	4.8563	0.0108
300-400	3.0914	0.0118
400-500	8.5721	0.0148
500-600	10.3712	0.0144
600-700	3.9449	0.0157
700-800	6.4099	0.018
800-900	4.4324	0.0264
900-1000	5.2892	0.0227

TABLE 6: AVERAGE PROMPT COMPRESSION TIME FOR TOKEN SIZE REDUCTION – SEMANTIC SEARCH

Token Size Range	Average Token Reduction % = [(Original Prompt Token-Compressed Prompt Token)]/(Original Prompt Token)*100	Average Compression Time (in seconds)
0-100	53.6260	0.1326
100-200	33.7836	0.1238
200-300	63.6303	0.7133
300-400	55.7321	0.4138
400-500	49.4996	0.4599
500-600	47.6336	0.6134
600-700	60.3487	0.6714

700-800	51.9015	1.1549
800-900	61.1815	1.2601
900-1000	74.4424	1.2926

The main purpose of doing Prompt Compression is to reduce the cost of making an API call by reducing token size and maintaining the semantic content of the prompt. The cost of making an API call is calculated by the formula:

*Cost of making an API call = (Input Cost * Prompt Tokens) + (Output Cost * Completion Tokens) ... Equation 1*

Here,

- Input Cost = $0.0015 per1000 tokens
- Output Cost = $0.002 per 1000 tokens
- Prompt Tokens is the number of tokens passed to LLM- (GPT-3.5 turbo)
- Completion Tokens is the number of tokens of the response generated by LLM (GPT 3.5-turbo)

TABLE 7: AVERAGE PROMPT COMPRESSION TIME AND COST REDUCTION FOR PROMPT TOKEN SIZE – ARTICLE REMOVAL AND SELECTED PUNCTUATION MARKS REMOVAL

Token Size Range	**Average Token Reduction % = [(Original Prompt Token-Compressed Prompt Token)]/(Original Prompt Token)*100**	**Average Comparison Time (in seconds)**	**Average Cost Reduction % (in Dollars)**
0-100	12.6	0.0001	10.2
100-200	9.53	0.0028	9.8
200-300	8.58	0.0009	11.2
300-400	10.3	0.0004	12.3
400-500	9.13	0.0009	11.1
500-600	11.2	0.0010	10.5
600-700	6.58	0.0008	11.4
700-800	10.9	0.0010	12.2
800-900	10.3	0.0014	12.2
900-1000	9.74	0.0010	11.5
1000-2000	8.21	0.0028	12.2
2000-3000	7.63	0.0028	12.1
3000-4000	6.25	0.0037	12.4

Various Prompt Compression approaches were used and the above approach was finalized because as compared to the other approaches, the semantic meaning of the prompt is retained at a higher level, while also reducing the cost of API call, as summarized in the table below.

TABLE 8: COMPARISON OF PROMPT COMPRESSION APPROACHES

Approach	Results	Limitations
Gptrim Python Library	Token Reduction > 20%	Irrelevant responses, due to loss of semantic meaning
Stop Words and selected Punctuation Marks removal	Token Reduction > 35%	Cannot exactly define position (or reference) of one object with respect to another
Porter Stemmer and selected punctuation marks removal	Average cost reduction ~ 4.57%	More cost reduction can be achieved
Articles, selected Stop Words and Punctuation Marks removal	Average cost reduction > 10%	Goal achieved in cost reduction and context retention

To evaluate the LLM (GPT-3.5 turbo) performance, 4 Standard Operating Procedures were created with the help of LLM, which were related to Invoice Processing in 4 different countries. 4 SOPs were ingested as input, whose vector store was created. To analyze the accuracy of compressed prompt, user queries were passed to it. The overall Tonic Validate RAG metrics score above 0.5 suggests that the semantic meaning of the original prompt is retained in the compressed prompt itself.

TABLE 9: COMPARISON OF USER QUERIES ON TONIC VALIDATE RAG METRICS ON THE SELECTED PROMPT COMPRESSION APPROACH

User Query	**Total Score (out of 1)**	**Answer Similarity Metric (out of 5)**	**Retrieval Precision Metric (out of 1)**	**Augmentation Accuracy Metric (out of 1)**	**Augmentation Precision Metric (out of 1)**	**Answer Consistency Binary (out of 1)**
1	0.8	5	1	1	1	0
2	1.0	5	1	1	1	1
3	0.6	5	0	1	0	1
4	0.5	5	1	0	0	1
5	0.2	5	0	0	0	0

This table shows LLM responses of Queries 1 and 2 were almost accurate and were in match with the retrieved context. However, Query 5 was from tabular data, so it was not able to create its vector store, thus gave a low score. In general, this LLM (GPT-3.5 turbo) model's performance is good.

The goal of this paper to achieve a cost reduction of more than 10%, has therefore been achieved as per Table 7 and 8 above for prompts of varying token sizes and also ensuring good quality LLM (GPT-3.5 turbo) responses from compressed prompt as per Tonic Validate RAG metrics in Table 9 above.

V. CONCLUSION

This model gives an overview that as technology advances, the integration of Generative AI into SOP writing processes can significantly streamline and enhance the efficiency of creating, updating, and maintaining procedural documentation. While Generative AI can be a powerful tool, human oversight is crucial, especially in critical areas such as compliance and safety. Additionally, organizations should carefully review and validate content generated by AI to ensure accuracy and relevance. As seen in the proposed model, the RAG metrics evaluation shows that the LLM

(GPT-3.5 turbo) is giving accurate responses as per the retrieved context. However, creating a vector store of tabular data is still a challenge to solve. By doing prompt compression, using stop word removal and punctuation removal, good quality compressed prompts in minimal time were obtained, while maintaining the semantic content of original prompt and ensuring good quality LLM responses. The paper achieves its goal to reduce cost of making an API call by more than 10%, for prompts of varying token sizes and ensures good quality LLM (GPT-3.5 turbo) responses from compressed prompt as per Tonic Validate RAG metrics. Higher accuracy in terms of good Tonic Validate scores can be achieved by trying to eliminate more stop words and punctuation marks if possible, without altering the semantic meaning of the original prompt. This will reduce the token size of the prompt and in turn will also reduce the API cost associated with the large language model. Precision can be improved by modifying the chunking strategy which will fetch more relevant chunks to answer the user query, thus increasing the retrieval and augmentation precision. Continued research and model development is needed in this emerging field.

REFERENCES

[1] https://www.techtarget.com/searchenterpriseai/definition/generative-AI (What is Generative AI? Everything you need to know, accessed in January 2024)

[2] https://openai.com/blog/chatgpt (OpenAI: Introducing ChatGPT, accessed in January 2024)

[3] https://research.ibm.com/blog/what-is-generative-AI? (IBM: What is Generative AI, accessed in January 2024)

[4] https://cloud.google.com/use-cases/generative-ai (Google Cloud: Generative AI examples, accessed in January 2024)

[5] https://learn.microsoft.com/en-us/training/paths/introduction-generative-ai/ (Microsoft Azure AI, accessed in February 2024)

[6] https://www.analyticsvidhya.com/blog/2023/07/the-fascinating-evolution-of-generative-ai/ (Analytics Vidya: The Fascinating Evolution of Generative AI, accessed in February 2024)

[7] https://www.linkedin.com/pulse/evolution-generative-ai-deep-dive-life-cycle-training-aritra-ghosh/ (LinkedIn: The Evolution of Generative AI: A Deep Dive into the Life Cycle and Training of Advanced Language Models, accessed in February 2024)

[8] Kalota, F. A Primer on Generative Artificial Intelligence. Educ. Sci. 2024, 14, 172.

[9] Fui-Hoon Nah, F., Zheng, R., Cai, J., Siau, K. and Chen, L., 2023. Generative AI and ChatGPT: Applications, challenges, and AI-human collaboration. Journal of Information Technology Case and Application Research, 25(3), pp.277-304.

[10] Ooi, K.B., Tan, G.W.H., Al-Emran, M., Al-Sharafi, M.A., Capatina, A., Chakraborty, A., Dwivedi, Y.K., Huang, T.L., Kar, A.K., Lee, V.H. and Loh, X.M., 2023. The potential of Generative Artificial Intelligence across disciplines: Perspectives and future directions. Journal of Computer Information Systems, pp.1-32.

[11] Chui, M., Hazan, E., Roberts, R., Singla, A. and Smaje, K., 2023. The economic potential of generative AI.

[12] Abdelaziz, T.A.I. and Fazil, U., 2023. Applications of integration of AI-based Optical Character Recognition (OCR) and Generative AI in Document Understanding and Processing. Applied Research in Artificial Intelligence and Cloud Computing, 6(11), pp.1-16.

[13] Kshetri, N., Dwivedi, Y.K., Davenport, T.H. and Panteli, N., 2023. Generative artificial intelligence in marketing: Applications, opportunities, challenges, and research agenda. International Journal of Information Management, p.102716.

[14] Banh, L. and Strobel, G., 2023. Generative artificial intelligence. Electronic Markets, 33(1), p.63.

[15] Korzynski, P., Mazurek, G., Altmann, A., Ejdys, J., Kazlauskaite, R., Paliszkiewicz, J., Wach, K. and Ziemba, E., 2023. Generative artificial intelligence as a new context for management theories: analysis of ChatGPT. Central European Management Journal.

[16] Ruiz-Rojas, L.I., Acosta-Vargas, P., De-Moreta-Llovet, J. and Gonzalez-Rodriguez, M., 2023. Empowering education with generative artificial intelligence tools: Approach with an instructional design matrix. Sustainability, 15(15), p.11524.

[17] Liu, M., Ren, Y., Nyagoga, L.M., Stonier, F., Wu, Z. and Yu, L., 2023. Future of education in the era of generative artificial intelligence: Consensus among Chinese scholars on applications of ChatGPT in schools. Future in Educational Research, 1(1), pp.72-101.

[18] Al Naqbi, H., Bahroun, Z. and Ahmed, V., 2024. Enhancing Work Productivity through Generative Artificial Intelligence: A Comprehensive Literature Review. Sustainability, 16(3), p.1166.

[19] Li, J., Lan, Y., Wang, L. and Wang, H., 2024. PCToolkit: A Unified Plug-and-Play Prompt Compression Toolkit of Large Language Models. arXiv preprint arXiv:2403.17411.

[20] https://docs.tonic.ai/validate/about-rag-metrics/tonic-validate-rag-metrics-summary (Tonic AI Validate, accessed in February 2024)