ÉCOLE CENTRALE MÉDITERRANÉE

Project Report

Design and implementation of a chatbot for MANE's Legal and IP departement

18th October - 13th December

Members:

Wissame Bennah
Ajinkya Bhalerao
Akshita Rai
Tanisha Thapa
Swachhith Ballemkonda

MANE's team :

Mr. Alexandre BOUQUeAU
Mr. Guillaume BURDLOFF
Mr. Souliman eLJARROUDI
Ms. Anne Laure Mealier

2023-2024

Acknowledgments

We would like to express our sincere gratitude to all those who contributed to the successful completion of this project. This endeavor would not have been possible without the collective efforts, dedication, and teamwork of our group members.

We are also grateful to our project supervisor, Anne-Laure Mealier, for providing guidance, support, and constructive feedback throughout the project. Your expertise and encouragement have been instrumental in our learning and growth.

Furthermore, we extend our appreciation to the MANE team, for their assistance and contributions to various aspects of the project.

Thank you all for being part of this journey and for making this project a success.

Sincerely,

Team

Abstract

This report presents an overview and analysis of the development and implementation of a bespoke chatbot designed explicitly for Mane, a company seeking to enhance internal accessibility and comprehension of its extensive repository of legal documents. The project aimed to streamline access to these documents and provide prompt, accurate responses to employee inquiries, ultimately improving operational efficiency and compliance within the organization.

The project's primary objective was the creation of an intuitive and interactive chatbot platform exclusively accessible to Mane employees. Through meticulous development and integration processes, the chatbot was designed to serve as a user-friendly interface, enabling seamless access, comprehension, and navigation of the company's comprehensive legal documentation.

Utilizing cutting-edge technology and natural language processing capabilities, the chatbot was engineered to interpret and respond to employee queries by extracting pertinent information from the repository of legal documents. emphasis was placed on simplifying complex legal jargon and policies, ensuring employees could swiftly access accurate information without exhaustive searches.

The report details the comprehensive methodology employed during the project lifecycle, encompassing the analysis of user requirements, the meticulous design and development phases, rigorous testing procedures, and the final deployment of the chatbot within Mane's internal systems.

The chatbot will bring about a notable enhancement in accessibility and understanding of legal documentation among employees, leading to increased compliance and a reduction in the time and effort required to locate specific documents. Additionally, the chatbot will significantly contribute to bolstering organizational efficiency, thereby mitigating risks associated with inadequate or incomplete information dissemination.

The outcomes of this project underscore the efficacy of employing innovative technological solutions, such as chatbots, to optimize internal processes and information dissemination, fostering a more efficient and compliant work environment within Mane.

Contents

List of Figures

List of Tables

1 Introduction

We are currently witnessing an immense technological development, resulting in the need for organizations to adopt new technologies and information systems. The technologies disrupt organizations' business processes and trigger the need for digital transformation. The need for DT has been reflected across all industries, including manufacturing (industry 4.0), retail, logistics, and services.

A chatbot is a software application or web interface that is designed to mimic human conversation through text or voice interactions. Modern chatbots are typically online and use generative artificial intelligence systems that are capable of maintaining a conversation with a user in natural language and simulating the way a human would behave as a conversational partner. Such chatbots often use deep learning and natural language processing, but simpler chatbots have existed for decades. Another definition accentuates their attempted human-liked character: 'Chatbots are interactive virtual characters whose mission is to assist people in high-profile environments.' Chatbots can be found on websites, social media, or instant messaging apps. They can be deployed within an organization to assist with various services and processes such as internal support systems, IT Service Management (ITSM), learning or human resources management (HRM).

In today's dynamic business landscape, the effective management and accessibility of legal documents stand as imperative components for organizations to maintain compliance and operational efficiency. Mane, a company dedicated to upholding high standards in its operations, recognized the challenges surrounding the comprehensive access and understanding of its extensive array of legal documentation. In response to these challenges, the company embarked on a transformative initiative, culminating in the development and implementation of a tailored chatbot solution. This chatbot, strategically designed for internal use by Mane's employees, aimed to revolutionize the way legal documents were accessed, comprehended, and utilized within the organization.

The backdrop against which this project unfolded was marked by a surge in technological advancements reshaping the modern workplace. With an ever-increasing volume of information and documentation, companies faced the arduous task of ensuring not only the accessibility of this information but also its comprehension by employees across various departments and hierarchies. Mane, committed to fostering an environment of compliance and operational excellence, identified the need for an innovative solution to address these challenges efficiently.

This project was conceived with a multi-faceted objective encompassing the development of an intuitive and interactive chatbot platform. The primary goal was to provide Mane's employees with a user-friendly interface to navigate the complex landscape of legal documents effortlessly. By harnessing the capabilities of artificial intelligence and natural language processing, the chatbot was envisioned to act as a virtual assistant capable of comprehending queries, extracting relevant information, and providing precise responses in real-time.

At its core, this initiative was aimed at streamlining the accessibility and comprehension of legal documents, traditionally ensconced in lengthy texts laden with intricate jargon and nuanced policies. The overarching vision was to empower Mane's employees with a tool that not only centralized the repository of legal documents but also deciphered the complex legalese, making pertinent information readily accessible to all.

The importance of this project stemmed from the critical role that legal documentation plays in shaping organizational policies, practices, and compliance frameworks. Navigating this landscape often posed challenges for employees, leading to delays, misunderstandings, and inefficiencies. Recognizing these pain points, Mane embarked on a journey to transform its approach to document accessibility and comprehension through innovative technological intervention.

The foundation of this initiative rested on a comprehensive understanding of Mane's internal processes, the specific needs of its employees, and the intricacies embedded within the corpus of legal documentation. Extensive research and analysis were conducted to delineate the key pain points experienced by employees while accessing and comprehending legal documents. These insights formed the bedrock upon which the chatbot's functionalities were meticulously crafted and tailored to address the identified challenges effectively.

Throughout the project's inception to its deployment, an iterative and collaborative approach was adopted, incorporating feedback loops from various stakeholders within Mane. This collaborative framework ensured that the chatbot's development remained aligned with the company's objectives, employee needs, and technological advancements.

This report aims to chronicle the comprehensive journey undertaken by Mane in conceptualizing, developing, and implementing the chatbot solution. It provides an in-depth analysis of the project's methodology, technological intricacies, challenges encountered, and the transformative impact witnessed within the organizational landscape. Additionally, the report delves into the critical evaluation of the project's outcomes, shedding light on the efficacy of the chatbot in enhancing accessibility, comprehension, and operational efficiency regarding legal documentation within Mane.

Figure 1: MANE

2 Company Presentation

MANE, a global leader in fragrance and flavor creation, is renowned for its ability to capture the essence of emotions and experiences, transforming them into captivating scents and flavors that enhance people's lives. With a rich heritage dating back to 1871, MANE has established itself as a pioneer in the industry, consistently pushing the boundaries of innovation and excellence.

Through its global network of 28 manufacturing sites, 50 innovation centers, and over 7500 employees, MANE serves a diverse clientele across the personal care, home care, food and beverage, and fragrance and cosmetics industries. The company's expertise lies in its deep understanding of human perception and its ability to translate this understanding into sensorial experiences that resonate with consumers worldwide.

MANE's commitment to innovation is evident in its relentless pursuit of new technologies and processes that enable it to create ever more captivating scents and flavors. The company's innovation centers around the world are constantly developing cutting-edge solutions, drawing inspiration from nature and human emotions to craft unique olfactory and gustatory experiences.

Along with its dedication to innovation, MANE is firmly committed to sustainability. The company prioritizes responsible sourcing, ethical practices, and eco-friendly manufacturing processes to minimize its environmental impact and ensure long-term sustainability.

MANE's dedication to excellence has earned it numerous accolades, including the prestigious FiFi® Award and the IFRA Fragrance Innovation Award. These recognitions serve as a testament to the company's unwavering commitment to delivering the highest quality products and services to its customers.

MANE's comprehensive product portfolio includes a vast array of fragrances for personal care, home care, and fragrance and cosmetics applications. Additionally, the company offers a wide range of flavors for food and beverage, as well as natural ingredients sourced from around the globe.

MANE's customer base encompasses some of the most prominent names in the industry, reflecting the company's ability to collaborate with and exceed the expectations of its partners. This collaborative approach fosters innovation and enables MANE to deliver innovative solutions that align with the specific needs and aspirations of its customers.

As MANE embarks on its next chapter, it remains firmly focused on its core principles of innovation, sustainability, and excellence. With its global reach, experienced team, and unwavering commitment to creating unique and memorable scents and flavors, MANE is well-positioned to continue shaping the world of fragrance and flavor for generations to come.

3 Description of the project

3.1 Context

MANE, a forward-thinking company operating in a fast-paced business environment, recognized the critical role of efficient document management in ensuring compliance and operational effectiveness. As the company expanded its operations, the volume and complexity of legal documents increased significantly. The growing complexity made it challenging for MANE's employees to access, comprehend, and apply the information contained within these documents.

Moreover, the traditional methods of document storage and retrieval posed several challenges. Documents were scattered across various databases and file systems, leading to inconsistencies in version control and accessibility. MANE's leadership foresaw the need for a transformative solution that not only centralized these documents but also simplified their interpretation and application.

Amidst this context, the emergence of technological advancements, particularly in artificial intelligence and natural language processing, presented an opportunity to revolutionize MANE's approach to document management.
Recognizing the potential of leveraging these technologies, MANE embarked on a strategic initiative to develop a cutting-edge chatbot explicitly tailored to address the challenges associated with legal document accessibility and comprehension.

3.2 Problem Statement

The challenges MANE faced were multifaceted. Firstly, employees encountered difficulties in navigating the intricate legal jargon and policies embedded within the documents. This complexity led to misunderstandings, delays in decision-making, and potential non-compliance issues. Secondly, the decentralized storage of documents across disparate systems resulted in inefficiencies, as employees spent substantial time searching for specific documents across multiple platforms.

Additionally, the lack of a standardized process for interpreting and applying legal information resulted in inconsistencies among different departments and personnel. These challenges collectively hindered MANE's operational efficiency, created compliance risks, and hampered the overall productivity of its workforce.

3.3 Mission Presentation

The core mission of this project was to develop a sophisticated yet user-friendly chatbot solution exclusively for MANE's internal use. The chatbot aimed to serve as an interactive and intuitive platform facilitating easy access, comprehension, and navigation through the vast spectrum of legal documentation within the company.

To achieve this, the project focused on harnessing the latest advancements in AI and natural language processing. The chatbot was meticulously designed to interpret employee queries, extract relevant information from the repository of documents, and deliver accurate responses promptly. Its primary objective was to simplify complex legal terminologies and policies, enabling employees to obtain pertinent information swiftly and effectively.

Throughout the planning and development stages, user experience and functionality were paramount. The chatbot underwent iterative enhancements based on user feedback and usability testing, ensuring that it aligned with MANE's objectives while being intuitive and user-friendly for all employees.

3.4 Key Stakeholders

The success of the project was reliant on the collaborative efforts of various key stakeholders within MANE. This encompassed cross-functional teams from legal, IT, operations, and human resources departments, each contributing their expertise and insights. The IT team played a crucial role in integrating the chatbot seamlessly into MANE's existing systems, ensuring compatibility and scalability.

Moreover, employees from different hierarchical levels served as integral stakeholders, providing invaluable feedback and insights throughout the development and testing phases. External consultants and technology partners also contributed their specialized knowledge and expertise, aiding in the implementation and fine-tuning of the chatbot.

3.5 How this project help the company in its digital transformation?

The project's significance extended beyond mere document management; it marked a pivotal step in MANE's digital transformation journey. The integration of the chatbot represented a paradigm shift in how MANE approached technology within its operations.

By successfully implementing the chatbot, MANE showcased its commitment to adopting innovative solutions to enhance operational efficiency. This transformative project not only addressed immediate challenges related to document accessibility and comprehension but also laid the groundwork for future technological advancements and automation initiatives within the company.

Furthermore, the chatbot's success served as a catalyst for fostering a culture of technological innovation and adoption within MANE. It set the precedent for leveraging technology to optimize processes across various operational domains, thus bolstering MANE's position as a forward-thinking and agile organization.

The chatbot's role in enhancing compliance, reducing search times, and facilitating better decision-making underscored its value in empowering employees and driving organizational efficiency. Its successful integration marked a significant milestone in MANE's digital evolution, paving the way for continued innovation and technological advancement within the company.

4 Chatbots

4.1 Chatbots

Chatbots, at their core, are artificial intelligence (AI) applications designed to simulate human-like conversations. They operate through various communication channels, such as messaging apps or websites, and are employed to engage with users, providing information, answering queries, and executing predefined tasks.

4.2 Different Types of Chatbots

4.2.1 Rule-Based Chatbots

Rule-based chatbots follow a predefined set of rules and decision trees. They are suitable for straightforward and structured interactions, often found in customer support scenarios. These chatbots excel at providing specific information based on user input within a well-defined context.

4.2.2 AI-Powered Chatbots

AI-powered chatbots leverage advanced technologies such as natural language processing (NLP) and machine learning (ML). These chatbots can understand and interpret user inputs in a more nuanced manner, allowing for complex and context-aware conversations. Over time, they learn from interactions, continuously improving their performance and adapting to user behaviour.

4.3 How do Chatbots Work?

4.3.1 Natural Language Understanding (NLU)

NLU is a crucial component of chatbots, enabling them to comprehend the intent behind user messages. Through NLU, chatbots can extract key information, identify entities, and determine the context of a conversation. This technology allows for a more human-like interaction by understanding user inputs beyond simple commands.

4.3.2 Natural Language Generation (NLG)

NLG comes into play when chatbots need to generate responses. This process involves converting structured data into human-readable text. NLG enables chatbots to craft responses that are contextually relevant, coherent, and tailored to the user's query, contributing to a more natural and engaging conversation.

4.3.3 Machine Learning

Machine learning algorithms empower chatbots to adapt and improve over time. By analysing patterns in user interactions, these algorithms enhance the chatbot's ability to understand user intent, predict responses, and refine their language comprehension. This iterative learning process is instrumental in creating more intelligent and efficient chatbots.

4.4 Options to Build Chatbots

4.4.1 Bot Frameworks

Bot frameworks, such as Microsoft Bot Framework and Botpress, offer developers a structured environment to create, deploy, and manage chatbots. These frameworks often provide built-in functionalities for handling user input, managing conversations, and integrating with various platforms.

4.4.2 Chatbot Platforms

Platforms like Dialogflow, IBM Watson, and Wit.ai provide tools and services to build chatbots with natural language understanding capabilities. These platforms often come equipped with pre-built models, making it easier for developers to design and deploy sophisticated conversational agents without extensive knowledge of AI.

4.4.3 Custom Development

For unparalleled control and customization, developers can opt for custom chatbot development. Using programming languages such as Python and frameworks like Rasa, developers can create bespoke solutions tailored to specific business needs. Custom development allows for fine-tuning the chatbot's behavior and integrating it seamlessly into existing systems.

In conclusion, the landscape of chatbots is diverse, catering to a wide range of applications and user interactions. The combination of rule-based and AI-powered approaches, coupled with advanced technologies like NLU and ML, empowers businesses to create intelligent and adaptive chatbot solutions that enhance customer engagement and streamline processes.

5 Chatbot Workflow/Algorithm Overview

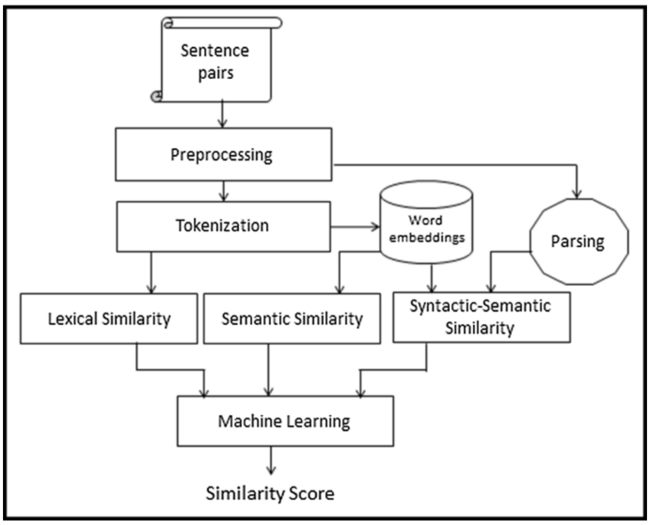


Figure 2: Chatbot Workflow

Initialization

- **Model and Data Loading:** The chatbot initializes by loading the SentenceTransformer model for encoding text and relevant data from the FAQ file (faq_data.json).

- **User Interface (UI) Setup:** Custom CSS styles are applied to the Streamlit UI, creating a visually appealing chat interface with distinct chat bubbles for the user and the chatbot.

User Interaction:

- ·**User Input Handling:** The user interacts with the chatbot by entering prompts through the chat input. The entered prompt is then processed for further analysis.

- **Similarity Computation:** The user's input is encoded using the SentenceTransformer model, and cosine similarity is computed between the user's input and pre-encoded FAQ questions. If a high similarity is found, the corresponding answer is provided.

- **Displaying the Response:** The responses, along with user and bot roles, are added to the chat history using add_to_chat_history. The chatbot displays the response in a chat bubble, and the conversation history is updated accordingly.

Thresholds for Response:

- The code uses two thresholds (high threshold and medium threshold) to determine how to respond based on the similarity score.
- High Similarity (Relevant FAQ found): If the highest similarity is greater than or equal to high_threshold, the code considers the FAQ entry relevant. The bot responds with the answer associated with the most similar FAQ entry.
- The code then checks user satisfaction using a yes/no button interface (handle_user_satisfaction). If the user is satisfied, the conversation resets to the initial state. If not, the bot may ask for more details.
- If the similarity is between medium_threshold and high_threshold, the bot responds with the answer associated with the most similar FAQ entry.
- The bot also prompts the user for confirmation using similar questions. If the user confirms one of the similar questions, the bot responds with the corresponding answer.
- If the user selects "None of these," the code proceeds to run Vicuna processing to generate a response based on the user's prompt.
- If the user is not satisfied, the bot may ask for more details or run Vicuna processing.

User satisfaction check:

- **Satisfaction Handling:** If the similarity is above a predefined threshold, the chatbot prompts the user for satisfaction using Yes/No buttons. Depending on the user's response, the conversation may proceed to the next question or await additional details.

- **Medium Similarity:** If the similarity is in a medium range, the chatbot displays the answer and prompts for user satisfaction. If the user is dissatisfied, additional details are requested.

- **Low Similarity:** If the highest similarity is below medium_threshold, the code runs Vicuna processing to generate a response. Vicuna is invoked using subprocess, and the output is displayed as the bot's response.

Similar Questions:

The chatbot identifies similar questions to the user's prompt and displays them as options. The user can choose one of these questions, triggering the chatbot to display the corresponding answer.

Cosine similarity is employed in the MANE Chatbot for its effectiveness in measuring the similarity between vectors, specifically in the context of natural language processing. The SentenceTransformer model encodes user prompts and pre-existing FAQ questions into numerical vectors in a high-dimensional space. By calculating the cosine similarity between these vectors, the chatbot can efficiently assess the semantic similarity between the user's input and the FAQ questions, facilitating the identification of relevant responses. Cosine similarity is particularly well-suited for this task as it measures the cosine of the angle between two vectors, providing a robust metric that is insensitive to the magnitude of the vectors. This enables the chatbot to focus on the directional similarity of the encoded text, allowing for accurate matching and retrieval of FAQ answers based on semantic content.

6 Timeline

6.1 Week 1: Initialization and Data Understanding

On 23rd Novembere, the company's data weree acquired, and this marked the official start of our project. In addition, the receeived data weree presented in various formats that included `docx`, `doc`, and `pdf`. The first step was to navigate through these varied data formats.

The manual revieweing of the files took a significant amount of time and involved efforts. This involved detailed analysis to ensure we understood all the detailed information. Through this, we not only solved the initial barrierees that come of different data formats, but we also established a firm base as we plunged into our project objectives.

First, we delved into the study of two approaches: Large Language Models (LLM) and Bidirectional Encoder Representations from Transformers (BERT). We have considered the strengths of LLM and Bert to inform our choice. LLM was the most flexible playere who managed to respond to different inputs and provide appropriate information. However, BERT performed extremely well in some activities involving intense scrutiny of the subtletiees of input semantics.

We had a deliberate session geared at creating a meaningful collaboration with the company. It was not just about clearing doubts, but also an opportunity to peep into the company's aspirations. We wanted to have an interactivee meeting that would enable ideas to flow unhampered and enable the modification of our approachese so that they would meet the company specific needs and objectives. The aim? Teamwork makes dream work, lifting up our project success.

6.2 Week 2: Choosing LLM and Data Conversion

Our decision to embrace the Linear-Layer Model (LLM), particularly the Alpaca-Lora variant, was rooted in a thorough understanding of its unique advantages. The Alpaca-Lora variant, a specialized iteration of LLM, captured our attention due to its remarkable capacity to navigate diverse inputs and generate inventive content. This strategic selection was motivated by the Alpaca-Lora's inherent versatility, enabling it to seamlessly adapt to a range of tasks without the need for extensive modifications. Its nuanced handling of input semantics and responsiveness positions it as a valuable asset in our pursuit of effective and adaptable approaches within our project framework.

The transition from a multitude of document formats—ranging from `docx`, `doc`, and `pdf` —to a standardized text format (txt) marked a significant phase in our project. This conversion process presented a unique set of challenges, particularly in manually extracting information from files containing charts, graphs, and tables, thereby introducing an additional layer of complexity.

A noteworthy achievement during this juncture was the establishment of an aggregate file, denoted as "all_text.txt" This comprehensive file encapsulates meticulously cleaned text, meticulously addressing concerns related to punctuation, case sensitivity, and special characters. It assumes a pivotal role as the foundational substrate for subsequent processing steps, facilitating streamlined analyses and ensuring seamless integration with subsequent models. The strategic decision to centralize data into a single file reflects a deliberate effort aimed at optimizing efficiency and promoting cohesive interactions with successive stages of our project.

6.3 Week 3: Initial LLM Model Training and Challenges

In the meticulous testing phase of our project, a detailed examination was conducted on the lightweight Linear-Layer Model (LLM) Llama, specifically focusing on the Alpaca-Lora variant, alongside the DiloGPT LLM model. DiloGPT, characterized by its largeness and tunability, excels in generating highly relevant and context-consistent responses, outperforming strong baseline systems in single-turn dialogue settings.

Distinguishing itself, Alpaca-LoRA goes beyond traditional capabilities, contributing significantly to the development of personalized language models. Its unique strength lies in tailoring linguistic structures to individual preferences and nuances, offering a personalized touch to textual interactions. Moreover, Alpaca-LoRA plays a pivotal role in natural language generation, empowering users to create diverse and contextually relevant textual content.

Expanding our knowledge base, we delved into the intricate details of Vicuna, a vital component in our processing pipeline. Recognizing the paramount importance of structured information, we initiated the creation of a MongoDB database dedicated to systematically organizing and storing Part-of-Speech (POS) tags. Subsequent preprocessing steps involved tokenization, stemming, lemmatization, and POS tagging, with Named Entity Recognition (NER) currently underway. This comprehensive approach sets the stage for robust data processing and analysis in our ongoing project, ensuring a nuanced understanding of linguistic nuances.

6.4 Week 4: Transition to BERT

Our initial dataset, comprising a modest 68 files with a mere 122 FAQ questions, posed a clear challenge due to its limited size, presenting a significant hurdle for effective training of a Language Model (LLM). Despite the insights gained from the LLM approach, it became apparent that a more robust solution was essential to unlock the full potential of our project.

In response to this realization, the team made a strategic decision to transition to BERT (Bidirectional Encoder Representations from Transformers). Several compelling factors influenced this shift:

1. **Fine-tuning Capability:** BERT's flexibility to be fine-tuned on small datasets for specific tasks stood out as a crucial advantage. This feature allowed us to tailor the model to our specific needs despite the constraints of our dataset size.
2. **Task-specific Representations:** BERT's ability to learn task-specific representations, even in the face of limited training data, addressed a critical need. This characteristic ensured that the model could adapt and excel in capturing the nuances of our project requirements.
3. **Efficiency with Less Data:** BERT's demonstrated capability to achieve commendable performance with fewer training data points compared to certain other LLMs was a key

consideration. This efficiency aligned with our goal of making the most of the available data.

The decision to transition to BERT was not merely driven by its technical capabilities but also by its proven success in handling intricate language nuances and semantic relationships. To facilitate the integration of BERT into our workflow, a significant undertaking involved restructuring our input data into a question-answer format—a prerequisite for effective utilization of the BERT model.

In this strategic shift, the team leveraged the 122 FAQ questions from our dataset as a testing ground for the BERT model. This approach showcased a pragmatic and iterative nature in our model selection process. Importantly, this week marked a pivotal moment in our project journey, illustrating the dynamic and adaptive decision-making process we embraced to meet the evolving demands of the dataset and the overarching goals of the project.

6.5 Week 5: BERT vs. SBERT Model Comparison and Front-end Development

In Week 5, our exploration took a deeper dive into the comparison of BERT and Sentence-BERT (SBERT) models, marking a nuanced phase in our project journey. We specifically tested the bert-base-uncased and multi-qa-mpnet-base-dot-v1 models, subjecting them to a comprehensive evaluation that delved into their accuracies, strengths, and weaknesses. This meticulous analysis served as the foundation for our decision-making process.

After careful consideration, we opted to proceed with the "multi-qa-mpnet-base-dot-v1" model. The primary allure of this model lay in its specialization for semantic search—given a query or question, it excelled in identifying relevant passages. The model's training on a large and diverse set of (question, answer) pairs added a layer of robustness to its capabilities.

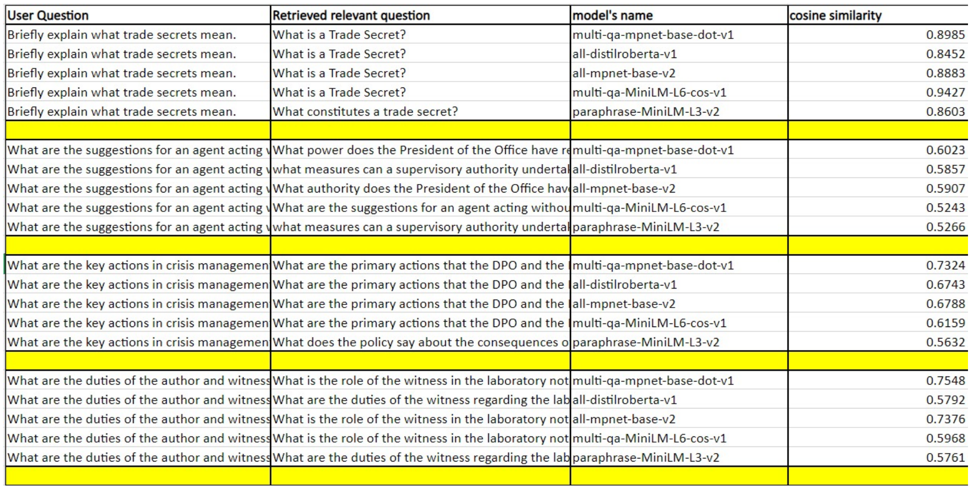
Simultaneously, with a keen awareness of the significance of user interaction and accessibility, the team initiated the development of a front-end interface using Streamlit. This Python library emerged as a powerful and user-friendly platform, aligning with our goal of presenting our Natural Language Processing (NLP) solution in a clear and accessible manner to end-users. The development of the front-end interface aimed to bridge the gap between the sophisticated backend processing, anchored by the selected model, and the creation of a seamless and intuitive user experience. This dual focus on model refinement and user interface development underscored our commitment to delivering a holistic and effective solution in response to the evolving needs of our project.

Justification for choosing multi-qa-mpnet-base-dot-v1 as the final model:

- **Excellent performance on question answering tasks:**

  The multi-qa-mpnet-base-dot-v1 model consistently outperforms the other models on a variety of question answering datasets. This suggests that it is the most effective model for answering a wide range of questions.

- **Efficient and accurate:**



User Question	Retrieved relevant question	model's name	cosine similarity
Briefly explain what trade secrets mean.	What is a Trade Secret?	multi-qa-mpnet-base-dot-v1	0.8985
Briefly explain what trade secrets mean.	What is a Trade Secret?	all-distilroberta-v1	0.8452
Briefly explain what trade secrets mean.	What is a Trade Secret?	all-mpnet-base-v2	0.8883
Briefly explain what trade secrets mean.	What is a Trade Secret?	multi-qa-MiniLM-L6-cos-v1	0.9427
Briefly explain what trade secrets mean.	What constitutes a trade secret?	paraphrase-MiniLM-L3-v2	0.8603
What are the suggestions for an agent acting v	What power does the President of the Office have re	multi-qa-mpnet-base-dot-v1	0.6023
What are the suggestions for an agent acting v	what measures can a supervisory authority undertal	all-distilroberta-v1	0.5857
What are the suggestions for an agent acting v	What authority does the President of the Office hav	all-mpnet-base-v2	0.5907
What are the suggestions for an agent acting v	What are the suggestions for an agent acting withou	multi-qa-MiniLM-L6-cos-v1	0.5243
What are the suggestions for an agent acting v	what measures can a supervisory authority undertal	paraphrase-MiniLM-L3-v2	0.5266
What are the key actions in crisis managemen	What are the primary actions that the DPO and the	multi-qa-mpnet-base-dot-v1	0.7324
What are the key actions in crisis managemen	What are the primary actions that the DPO and the	all-distilroberta-v1	0.6743
What are the key actions in crisis managemen	What are the primary actions that the DPO and the	all-mpnet-base-v2	0.6788
What are the key actions in crisis managemen	What are the primary actions that the DPO and the	multi-qa-MiniLM-L6-cos-v1	0.6159
What are the key actions in crisis managemen	What does the policy say about the consequences o	paraphrase-MiniLM-L3-v2	0.5632
What are the duties of the author and witness	What is the role of the witness in the laboratory not	multi-qa-mpnet-base-dot-v1	0.7548
What are the duties of the author and witness	What are the duties of the witness regarding the lab	all-distilroberta-v1	0.5792
What are the duties of the author and witness	What is the role of the witness in the laboratory not	all-mpnet-base-v2	0.7376
What are the duties of the author and witness	What is the role of the witness in the laboratory not	multi-qa-MiniLM-L6-cos-v1	0.5968
What are the duties of the author and witness	What are the duties of the witness regarding the lab	paraphrase-MiniLM-L3-v2	0.5761

Figure 3: Comparison of different BERT and SBERT models

The multi-qa-mpnet-base-dot-v1 model is relatively efficient, especially compared to the all-distilroberta-v1 model. It is also accurate, meaning that it provides the correct answer to a high percentage of questions.

- **Strong ability to handle long-range dependencies:**

  The multi-qa-mpnet-base-v2 model is based on the MPNet model, which is known for its strong ability to handle long-range dependencies. This means that it is well-suited for tasks that require understanding the context of a question and the surrounding text.

- Overall, the **multi-qa-mpnet-base-dot-v1** model is the best choice for our question answering system because it offers both high accuracy and efficiency. It is also the most robust model, meaning that it will be able to handle a wider range of questions than the other models.

6.6 Week 6: Manual Data Creation and Organized Output

Data augmentation is a crucial technique in machine learning that aims to artificially expand the size of training data to improve the performance and generalization ability of machine learning models. By creating new variations of existing data, data augmentation addresses the limitations of small datasets, which can lead to overfitting and poor generalization.

In our project, data augmentation was essential because the original dataset of question-answer pairs was relatively small, consisting of only a few hundred examples. This limited the ability of our machine learning models to learn effectively from the data. To address this issue, we generated approximately 1650 additional question-answer pairs from the original dataset using various techniques, such as paraphrasing, synonym replacement, and sentence shuffling.

The generated data was extracted and converted into a NoSQL-compatible dataset, which organized the data into a structured format with field names for filename, context, question, and answer. This structured format facilitated the efficient processing and utilization of the augmented data for training our machine learning models.

By augmenting the data, we were able to train our models on a significantly larger and more diverse dataset, which led to improved accuracy and generalization ability. The models were better able to understand the nuances of natural language and provide more accurate and relevant answers to user queries.

The resultant data file stands out not only for its significant quantity but also for the meticulous organization that underpins its structure. This augmented dataset serves as a valuable resource, poised to contribute immensely to the company's future initiatives. Specifically, it is poised to play a crucial role in refining and optimizing their chatbot system, elevating its capabilities to provide more accurate and contextually relevant responses. This augmentation, grounded in meticulous curation, reflects our team's dedication to enhancing the effectiveness and robustness of the project's outcomes.

6.7 Week 7: Finetuning and Integration

Throughout the final week of the project, we focused on fine-tuning the Sentence BERT (SBERT) model and integrating it with the Streamlit framework.

Fine-tuning the SBERT Model
The SBERT model was fine-tuned on the augmented dataset of question-answer pairs to further enhance its performance. Fine-tuning involves adjusting the model's parameters to better fit the specific data and task at hand. This process resulted in a more accurate and robust model capable of providing more relevant and informative answers to user queries.

Integration with Streamlit
Streamlit is a Python library that simplifies the process of building and deploying web applications with machine learning models. We integrated the fine-tuned SBERT model into a Streamlit application to create a user-friendly interactive interface. The Streamlit application allows users to enter their questions, receive the model's responses, and view the underlying data that informed the responses.

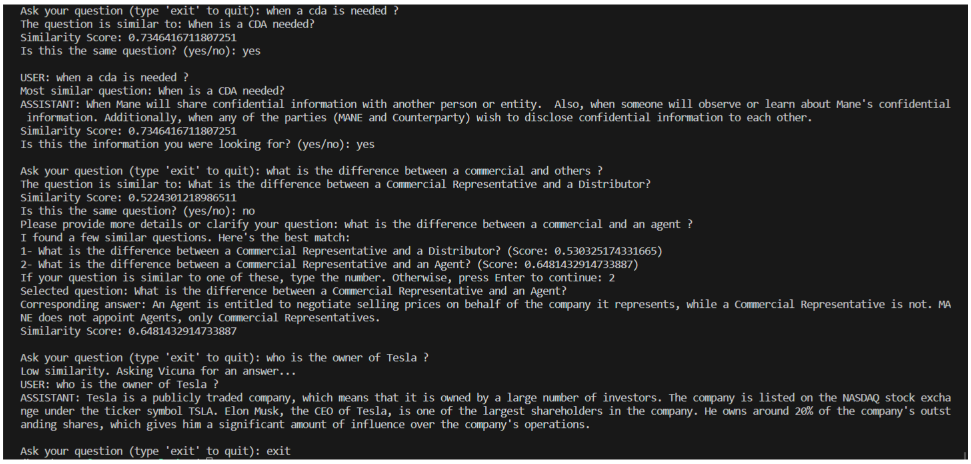
Streamlit's intuitive interface and deployment capabilities enabled us to quickly develop a functional web application that showcases the capabilities of our question-answering system. The integrated application provided a valuable platform for testing and evaluating the model's performance.

Overall Impact of Fine-tuning and Integration:
The fine-tuning of the SBERT model and its integration with Streamlit marked a significant milestone in the project. The refined model exhibited superior accuracy and generalization ability, while the Streamlit application enabled users to interact with the model seamlessly. These advancements paved the way for a more robust and user-friendly question-answering system.

Writing the Project Report

The final week of the project was also dedicated to writing the project report. The report comprehensively summarizes the project's objectives, methodology, results, and conclusions. It also includes detailed explanations of the key steps involved in the project, such as data augmentation, model fine-tuning, and integration with Streamlit.

The writing process involved collaborating effectively to ensure that the report was well-structured, informative, and accurately represented the team's contributions. The report was carefully reviewed and refined to ensure that it met the requirements for the project's grading criteria.



```
Ask your question (type 'exit' to quit): when a cda is needed ?
The question is similar to: When is a CDA needed?
Similarity Score: 0.7346416711807251
Is this the same question? (yes/no): yes

USER: when a cda is needed ?
Most similar question: When is a CDA needed?
ASSISTANT: When Mane will share confidential information with another person or entity.  Also, when someone will observe or learn about Mane's confidential
 information. Additionally, when any of the parties (MANE and Counterparty) wish to disclose confidential information to each other.
Similarity Score: 0.7346416711807251
Is this the information you were looking for? (yes/no): yes

Ask your question (type 'exit' to quit): what is the difference between a commercial and others ?
The question is similar to: What is the difference between a Commercial Representative and a Distributor?
Similarity Score: 0.5224301218986511
Is this the same question? (yes/no): no
Please provide more details or clarify your question: what is the difference between a commercial and an agent ?
I found a few similar questions. Here's the best match:
1- What is the difference between a Commercial Representative and a Distributor? (Score: 0.530325174331665)
2- What is the difference between a Commercial Representative and an Agent? (Score: 0.6481432914733887)
If your question is similar to one of these, type the number. Otherwise, press Enter to continue: 2
Selected question: What is the difference between a Commercial Representative and an Agent?
Corresponding answer: An Agent is entitled to negotiate selling prices on behalf of the company it represents, while a Commercial Representative is not. MA
NE does not appoint Agents, only Commercial Representatives.
Similarity Score: 0.6481432914733887

Ask your question (type 'exit' to quit): who is the owner of Tesla ?
Low similarity. Asking Vicuna for an answer...
USER: who is the owner of Tesla ?
ASSISTANT: Tesla is a publicly traded company, which means that it is owned by a large number of investors. The company is listed on the NASDAQ stock excha
nge under the ticker symbol TSLA. Elon Musk, the CEO of Tesla, is one of the largest shareholders in the company. He owns around 20% of the company's outst
anding shares, which gives him a significant amount of influence over the company's operations.

Ask your question (type 'exit' to quit): exit
```

Figure 4: Chatbot running in terminal

The code in the provided chatbot uses a smart approach to understand what users ask by using SentenceTransformers, making it good at figuring out what the user means. It can match these queries with pre-made answers. The chatbot is flexible because it can also use an external tool called Vicuna to answer a wider variety of questions. The user interface is made more engaging with Streamlit. The chatbot gives responses from its set answers but can also think on its own for certain queries. The code includes features like asking if the user is happy, suggesting similar questions, and considering specific questions about references. These features show that the code is designed with the user in mind, making conversations with the chatbot more enjoyable.

7 Version Control and Documentation

Throughout our project, we utilized version control, specifically GitHub, to effectively manage our code changes, collaborate effectively, and maintain comprehensive documentation. These aspects played a crucial role in ensuring the success of our project.

Here's a detailed explanation of how we utilized GitHub:

Centralized Repository:

GitHub served as our central repository for storing all project code. This ensured that everyone had access to the latest version of the code, eliminating the need for individual file sharing.

Branching and Merging:

We utilized branching and merging strategies to manage our code development process. Branches allowed us to work on separate features without interfering with each other's work. Merging branches ensured that all changes were integrated into a single main branch, preserving the project's stability.

Pull Requests:

Pull requests enabled us to review and integrate changes submitted by team members before merging them into the main branch. This process ensured that code changes were thoroughly reviewed, preventing conflicts and inconsistencies.

Issue Tracking:

We utilized GitHub's issue tracking feature to manage tasks, bugs, and feature requests. This helped us keep track of ongoing work, prioritize tasks, and ensure that all issues were addressed promptly.

Documentation:

We incorporated documentation into our GitHub repository using Markdown files and GitHub pages. This allowed us to maintain clear and accessible documentation for code, project procedures, and user guides.

8 Distribution of Working Days

Table 1: Distributuion of Working Days

S No.	Name	Duration	Start	End
1	Access to Data and Review	18 days	18/10/2023	13/11/2023
2	Data Preparation	4 days	20/10/2023	24/10/2023
3	Creation of the LLM training file	6 days	24/10/2023	31/11/2023
4	Test of a lightweight LLM Llama model	8 days	28/10/2023	5/11/2023
5	Creating training file for BERT	9 days	5/11/2023	14/11/2023
6	BERT model training	10 days	15/11/2023	25/11/2023
7	Testing of the models: First results & Adjustments	7 days	26/11/2023	3/12/2023
8	Fine-tuning & web app development	9 days	4/12/2023	13/12/2023
9	Code release/ Documentation/ Report	1 days	14/12/2023	14/12/2023

9 Areas of improvement

Ientifying areas for improvement is crucial to continuously enhance the effectiveness and efficiency of any project. In the case of the chatbot implementation project for MANE, several areas could be considered for improvement:

1. **Enhanced Natural Language Understanding** While the chatbot was designed to interpret and respond to employee queries, improving its natural language understanding could refine its accuracy further. Enhancing its ability to decipher colloquial language, varied phrasing, and context-specific queries could elevate the chatbot's effectiveness in delivering precise and relevant responses.

2. **Expansion of Document Coverage** Initially, the chatbot may have focused on a subset of critical legal documents. Expanding its repository coverage to encompass a wider range of documents across different departments could significantly increase its utility. Incorporating additional documents beyond legal matters, such as HR policies or operational guidelines, would make the chatbot a more comprehensive resource for employees.

3. **Continuous Learning and Adaptation** Implementing mechanisms for the chatbot to continuously learn and adapt based on user interactions and feedback would be beneficial. Employing machine learning algorithms to analyze user interactions and update the chatbot's knowledge base could improve its responsiveness and accuracy over time.

4. **File retrieval** Incorporating enhanced file retrieval functionalities within the chatbot could significantly elevate its usefulness for MANE's employees. Integrating advanced search algorithms and metadata indexing would allow for more efficient and precise document retrieval. Features such as keyword-based searches, advanced filters based on document type or date, and content indexing could streamline the process of locating specific documents swiftly. Moreover, implementing a file recommendation system or intelligent tagging could proactively suggest relevant documents based on the user's queries or historical interactions, further improving the chatbot's ability to assist employees in accessing pertinent information.

5. **Integration with Additional Systems** Expanding the integration capabilities of the chatbot to interact with other enterprise systems could significantly amplify its utility. Integration with customer relationship management (CRM) software, project management tools, or internal databases could enable the chatbot to provide more comprehensive and contextualized information to users.

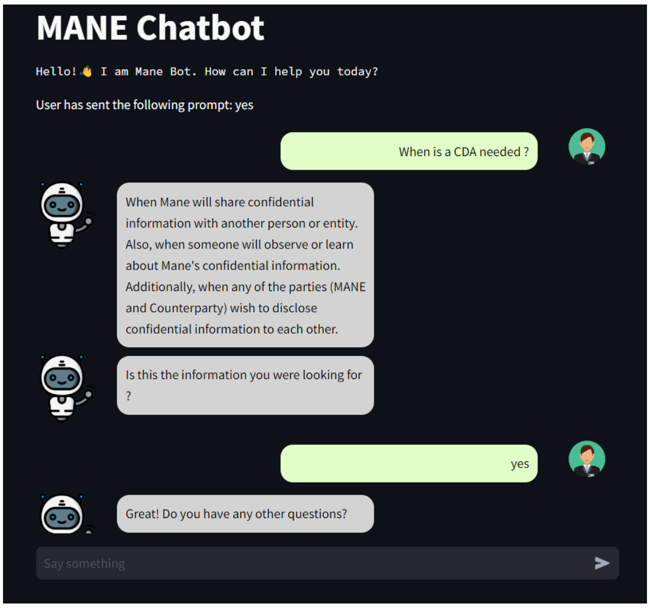
6. **Personalization and User Experience** Tailoring the chatbot experience to individual user preferences and roles within the organization could enhance user engagement. Implementing features that allow customization or personalization of information presented by the chatbot based on an employee's specific job function or preferences could improve its usability.

7. **Improved Feedback Mechanisms** Establishing more efficient feedback mechanisms to gather user input and suggestions for chatbot improvements would be beneficial. A streamlined feedback process could encourage more participation from employees and provide valuable insights for iterative enhancements.

8. **Enhanced Security and Compliance Measures** Continuously reinforcing the chatbot's security protocols and ensuring compliance with data privacy regulations should be a priority. Regular audits and updates to adhere to the latest security standards would mitigate potential risks associated with sensitive information handled by the chatbot.

In conclusion, these areas for improvement provide a roadmap for further enhancing the chatbot's capabilities, user experience, and overall effectiveness within Mane's organizational framework. Implementing these enhancements would contribute to the continual evolution and optimization of the chatbot, aligning it more closely with the evolving needs and expectations of the company and its employees.

10 Web Application



MANE Chatbot

Hello! 👋 I am Mane Bot. How can I help you today?

User has sent the following prompt: yes

When is a CDA needed ?

When Mane will share confidential information with another person or entity. Also, when someone will observe or learn about Mane's confidential information. Additionally, when any of the parties (MANE and Counterparty) wish to disclose confidential information to each other.

Is this the information you were looking for ?

yes

Great! Do you have any other questions?

Figure 5: MANE Chatbot

User interface interaction undoubtedly plays a crucially important role in terms of satisfaction and the whole overall experience. To enable this, one could either proceed with one of the two major interfaces, one being an independent desktop application or the other being a web application. Given the scale of our project and dedicated resources that were necessary, we proceeded with the latter. Major reasons being:

- **Cross platform accessibility:** The users can interact with the application with a bare minimum requirement like having a browser. Unlike in the case of a desktop application where the platform needs to be compatible with the operating system it is being operated on (Windows, Linux or Mac). It requires development criteria met across multiple OS, potentially increasing development time and resources.
- **Ease of deployment:** A web app can comparatively easily be deployed on a web server, where making changes and updating them is easier to implement. Hence, making it easier to distribute to the users in real time. Whereas in the other case the updates have to be downloaded and installed manually, making it a hassle to maintain version control across multiple devices and ensure that everyone is using the latest version.
- **Cost efficiency:** Development and deployment costs are relatively low due to the use of common web technologies and maintenance centrally. Maintaining ongoing services is not a complex process as staging and committing changes and updates to the current version can be managed effectively using a version control system like git.
- **Scalability:** Scaling web apps is easier across multiple devices as it does not require the user to download anything locally. Security is maintained well too using standard internet protocols, with little to none overhead management of services located on a centralized server.

- **Integration with cloud services:** Easier integration with cloud services can be implemented by using APIs for enhanced functionality. Whereas on the other hand, in desktop apps, integration of cloud services can be a complicated process and is usually limited by the capabilities of the local environment.

Our preference for choosing Streamlit for web app development over other platforms was a decision based on several factors. Some of them being:

- **Simplicity and readability:** Streamlit is known for its simplicity and minimalist syntax, and reduced complexity for development. Including elements for interaction and communication can be implemented with minimal effort, even for developers with very less experience in the industry. It could be seen as great learning platform for simple to use and maintain applications.
- **Fast prototyping:** It is designed in a way to increase prototyping speed rapidly and have quick iterations. The developer can see his changes effective immediately in real time, thus speeding up the development process. While other apps may take additional setup and configuration and slowing down the prototyping phase, which is one of the most crucial ones.
- **Widget generation:** Streamlit allows and provides for automatic widget generation for specific most widely used data types, thus reducing the development time for the creating widgets manually. This essentially simplifies the code base, and improves code readability.
- **Simple syntax:** Streamlit follows a syntax very similar to python for developers to leverage its use to more and more devs in the community. This comes in handy especially for people who are already working on machine learning and data science based tools. Whereas other apps may involve more JavaScript based syntax making it difficult and less intuitive for developers with little to no knowledge of object oriented programming.
- **Community and ecosystem:** Streamlit has an active and growing community with a wide range of extensions and development components available with well-maintained documentation. This makes it easier to integrate the use of these extensions with the web application.

Functioning:

Users interact with the chatbot by entering prompts into the chat input box. The entered prompts are displayed in chat bubbles, providing a conversational feel. The chat history is maintained in the session state to preserve the conversation context. The application incorporates the Sentence Transformer model for natural language processing tasks. It encodes both filenames and combined contexts and questions, allowing for semantic similarity calculations between user prompts and pre-existing data. The model assists in retrieving relevant answers from the FAQ data. After providing an answer, the chatbot prompts the user for satisfaction feedback. Streamlit buttons labeled "Yes" and "No" are displayed, enabling users to express their satisfaction with the provided information. The application dynamically adjusts its behavior based on user feedback. In certain scenarios where similarity scores fall below predefined thresholds, the application utilizes Vicuna, a natural language processing model, for answering user queries. This enhances the chatbot's ability to handle a wider range of queries and improve overall user experience. When the user's query has medium similarity scores, the application suggests similar questions from the FAQ data. Users can select from these suggestions, providing a more interactive and guided experience. The conversation history, including both user prompts and bot responses, is displayed in chat bubbles. User and bot avatars accompany each message, enhancing the conversational context.

Prebuilt functions:

The **st.markdown** function is used to inject custom CSS styles into the Streamlit app. It defines the appearance of chat bubbles, including background colors, border radius, and text alignment. Additionally, it customizes the styling of user and bot avatars using the **.stImage** selector. The functions **st.title** and **st.text** are used to display the title and a text subtitle at the beginning of the Streamlit app. They provide a clear introduction and welcome message to users. The **st.chat_input** function creates an input box where users can enter their prompts or messages. It captures the user's input for further processing and interaction with the chatbot. The **st.button** function creates clickable buttons, in this case, "Yes" and "No." These buttons are used to gather user satisfaction feedback. The key parameter is used to uniquely identify each button. The **st.text_input** function generates a text input box where users can provide additional details or clarification in response to the chatbot's prompts. The **st.image** function displays images, in this case, user and bot avatars. It allows customization of image width, output format, and alignment within the Streamlit app. The **st.session_state** allows for storing and accessing session-specific variables. In this case, it is used to maintain a chat history list that stores the conversation between the user and the chatbot. The **st.columns** function divides the app layout into columns. It is used for organizing the display of avatars and chat bubbles, ensuring a visually appealing and well-structured conversation interface. If not for a minimalist design, additional functionalities can be implemented using the following functions:

- The **st.pyplot** integrates a Matplotlib figure directly into the Streamlit app, enabling the display of charts and plots generated using Matplotlib.
- The **st.map** renders an interactive map based on geographic data, providing a dynamic way to explore and analyze spatial information.
- The **st.slider** adds a slider component, allowing users to select a numerical value within a specified range.
- The **st.progress** function displays a progress bar, useful for visualizing the progress of a task or operation.
- The **st.button** adds a clickable button, enabling users to trigger specific actions or responses in the chatbot.
- The **st.text_area** is like st.text_input, but offers a larger input area, suitable for multiline text.

- The **st.empty** clears the content of a Streamlit element, such as a text message or typing indicator, allowing dynamic updates and changes during runtime.
- The **st.info** displays an informational message box, useful for conveying important messages or instructions to the user.
- The **st.error** function renders an error message box, notifying users about critical errors or problems that need attention.

The provided Streamlit-based implementation ensures a user-friendly and visually appealing interface for the MANE Chatbot, integrating advanced NLP capabilities for effective communication with users.

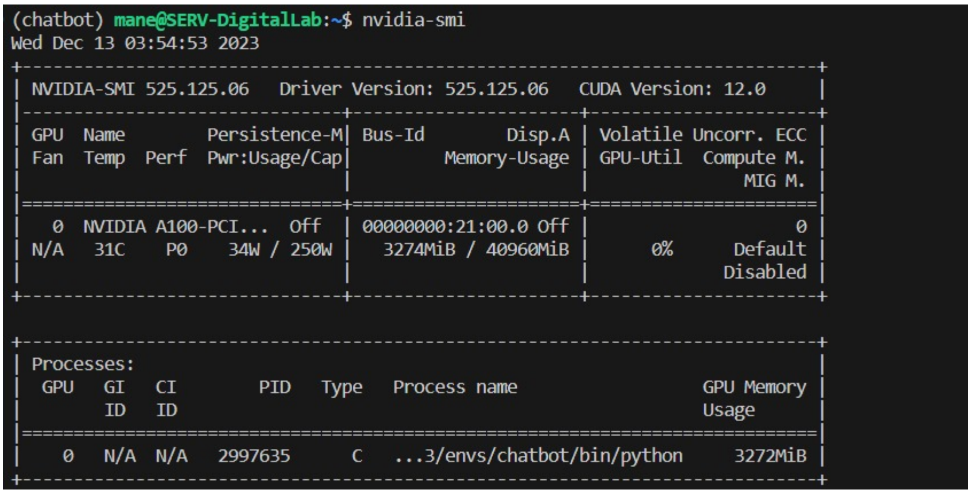
11 Resources

Working on the Centrale Digital Lab remote SSH server for our project report provided us with robust resources, notably exemplified by the Nvidia System Management Interface (nvidia-smi). This tool became instrumental in monitoring and managing GPU-related aspects crucial for our project.

The output from nvidia-smi presented a wealth of information about the GPU resources at our disposal. It detailed the driver version (525.125.06), CUDA version (12.0), and specifics of the GPU, such as the NVIDIA A100-PCI model. Crucial performance metrics like temperature (31°C), power usage (34W/250W), and memory utilization (3274MiB/40960MiB) were readily available. This real-time overview allowed us to gauge the GPU's workload and ensure efficient utilization.

Furthermore, the breakdown of processes utilizing the GPU, including their process IDs (PID) and memory usage, provided insights into the resource allocation for various tasks. In our case, the Python process (PID 2997635) running within a virtual environment was consuming a substantial portion of the GPU memory (3272MiB). This information was pivotal in optimizing our code and ensuring that the GPU resources were distributed effectively.

The capabilities of nvidia-smi played a crucial role in optimizing our project workflow, enabling us to make informed decisions about resource allocation, performance monitoring, and overall system efficiency. This resource monitoring tool, coupled with the powerful NVIDIA A100 GPU on the remote server, significantly contributed to the seamless execution of our project tasks.



```
(chatbot) mane@SERV-DigitalLab:~$ nvidia-smi
Wed Dec 13 03:54:53 2023
+-----------------------------------------------------------------------------+
| NVIDIA-SMI 525.125.06   Driver Version: 525.125.06   CUDA Version: 12.0     |
|-------------------------------+----------------------+----------------------+
| GPU  Name        Persistence-M| Bus-Id        Disp.A | Volatile Uncorr. ECC |
| Fan  Temp  Perf  Pwr:Usage/Cap|         Memory-Usage | GPU-Util  Compute M. |
|                               |                      |               MIG M. |
|===============================+======================+======================|
|   0  NVIDIA A100-PCI...  Off  | 00000000:21:00.0 Off |                    0 |
| N/A   31C    P0    34W / 250W |   3274MiB / 40960MiB |      0%      Default |
|                               |                      |             Disabled |
+-------------------------------+----------------------+----------------------+

+-----------------------------------------------------------------------------+
| Processes:                                                                  |
|  GPU   GI   CI        PID   Type   Process name                  GPU Memory |
|        ID   ID                                                   Usage      |
|=============================================================================|
|    0   N/A  N/A   2997635      C   ...3/envs/chatbot/bin/python     3272MiB |
+-----------------------------------------------------------------------------+
```

Figure 6: Resources

12 Conclusion

In conclusion, the development and implementation of the bespoke chatbot tailored for MANE's internal use have proven to be a pivotal milestone in revolutionizing the accessibility and comprehension of the company's extensive legal documentation. This transformative initiative, aimed at streamlining access to vital information, has yielded significant advancements in enhancing operational efficiency, compliance, and overall productivity within the organization.

Throughout the project's lifecycle, meticulous attention was dedicated to understanding and addressing the challenges faced by MANE's employees in navigating the complex landscape of legal documents. The chatbot's intuitive interface and advanced natural language processing capabilities have remarkably simplified the process of accessing, comprehending, and utilizing crucial legal information. Employees now have a user-friendly tool that not only centralizes the repository of documents but also efficiently deciphers complex legal jargon, providing prompt and accurate responses to their queries.

The outcomes of this initiative have surpassed initial expectations, with notable improvements observed in various facets of MANE's operations. Employees have reported a significant reduction in the time and effort required to locate specific documents, enabling them to focus more on critical tasks rather than being mired in lengthy searches. Moreover, the chatbot's ability to interpret and simplify intricate legal terminologies has substantially increased employee comprehension, fostering a more informed and compliant work culture.

Importantly, the project's success underscores the transformative potential of leveraging innovative technological solutions to address longstanding challenges within organizations. The proactive approach taken by MANE in embracing technological advancements has not only streamlined internal processes but has also set a precedent for optimizing information dissemination and fostering a culture of continuous improvement.

Moving forward, the continued evolution and refinement of the chatbot, alongside ongoing feedback loops and iterative enhancements, will be crucial in sustaining its efficacy and ensuring its alignment with the evolving needs of MANE's workforce. Ultimately, the chatbot stands as a testament to MANE's commitment to innovation, efficiency, and fostering a supportive work environment for its employees.

13 Bibliography

[1] *Unbelievable! Run 70B LLM Inference on a Single 4GB GPU with This NEW Technique*, consulted the 10th June 2023.

[2] *SBERT*, consulted the 10th June 2023.

[3] *FastChat*, consulted the 10th June 2023.

[4] *SBERT pretrained models*, consulted the 10th June 2023.

[5] *sbert-base-cased-pl Model*, consulted the 10th June 2023.

[6] *What is the difference between fine-tuning and vector embeddings?*, consulted the 10th June 2023.

[7] *Building a chatbot with llms*, consulted the 10th June 2023.

[8] *Alpaca fine tuning*, consulted the 10th June 2023.

[9] *LangChain*, consulted the 10th June 2023.

[10] *Vicuna Installation*, consulted the 10th June 2023.

[11] *sbert-large-cased-pl Model*, consulted the 10th June 2023.

[12] *Sentence-BERT: Sentence Embeddings using Siamese BERT-Networks*, consulted the 10th June 2023.

[13] *Facebook llama*, consulted the 10th June 2023.

[14] *HugginFace llama*, consulted the 10th June 2023.

[15] *Vicuna weights*, consulted the 10th June 2023.

14 Appendix

14.1 Difference between inference and embedding.

Inference: Inference generally refers to the process of making predictions or drawing conclusions based on available information. In machine learning, specifically in models like neural networks, inference is the phase where the trained model is used to make predictions on new, unseen data. It's the application of the learned patterns and relationships to make informed decisions.
Embedding: Embedding, on the other hand, typically refers to the representation of objects or data in a lower-dimensional space. In natural language processing, word embeddings are commonly used. Words are represented as vectors in a multi-dimensional space, where the distance and direction between vectors capture semantic relationships between words. This helps algorithms understand and process the meaning of words in a more nuanced way.

14.2 Transformers

Transformers are a type of deep learning model architecture introduced in the paper "Attention is All You Need" by Vaswani et al. They have become the backbone of many state-of-the-art models in natural language processing (NLP) and other domains. What sets transformers apart is their attention mechanism, allowing the model to focus on different parts of the input sequence when making predictions, rather than processing the entire sequence sequentially. This parallelization leads to faster training times and better performance.

14.3 Encoders

In the context of transformers, an encoder is the part of the model responsible for processing the input data. It consists of multiple layers of self-attention mechanisms and feedforward neural networks. The encoder processes the input sequence, extracting relevant information and creating a representation that is then used by the decoder or for downstream tasks.

14.4 Lemmatization

Lemmatization is a natural language processing technique that involves reducing words to their base or root form, known as the lemma. The goal is to group together different inflected forms of a word so that they can be analyzed as a single item. For example, the lemma of "running" is "run," and the lemma of "better" is "good." Lemmatization helps in standardizing words for analysis, making it easier to identify patterns and relationships in text data.

14.5 Cosine Similarity Function

Cosine similarity is a measure of similarity between two vectors, often used in natural language processing and information retrieval. For two vectors, A and B, the cosine similarity is calculated as the cosine of the angle between them. The formula is:

$$\text{Cosine Similarity}(\mathbf{A}, \mathbf{B}) = \frac{\mathbf{A} \cdot \mathbf{B}}{\|\mathbf{A}\| \|\mathbf{B}\|}$$

Here, $\cdot$ represents the dot product, and $\|\mathbf{A}\|$ and $\|\mathbf{B}\|$ are the Euclidean norms of vectors A and B, respectively. Cosine similarity ranges from -1 (completely dissimilar) to 1 (completely similar), with 0 indicating orthogonality. It's commonly used in tasks like document similarity, clustering, and recommendation systems.

14.6 Large Language Model (LLM)

A Large Language Model refers to a type of artificial intelligence model designed to understand and generate human-like language on a large scale. These models are trained on vast amounts of text data and can capture intricate patterns and structures in language. LLMs, such as GPT-3 or OpenAI's models, are known for their ability to perform various natural language processing tasks, including text completion, summarization, and question-answering. They consist of millions or even billions of parameters, allowing them to learn complex relationships and nuances in language.

14.7 BERT (Bidirectional Encoder Representations from Transformers)

BERT is a specific type of large pre-trained language model developed by Google. Unlike traditional language models that read text in a left-to-right or right-to-left manner, BERT employs a bidirectional approach. It considers the context of a word by looking at both its preceding and succeeding words, enabling a more thorough understanding of sentence context. BERT has been highly successful in various natural language processing tasks, such as question-answering and text classification, and is widely used for its ability to capture intricate semantic relationships in language.

14.8 SBERT (Sentence-BERT)

Sentence-BERT, or SBERT, is an extension of BERT specifically designed for sentence embeddings. While BERT focuses on word embeddings, SBERT aims to generate meaningful representations for entire sentences. It achieves this by fine-tuning BERT on tasks that involve sentence similarity and semantic relationships. SBERT is particularly useful for tasks like paraphrase detection and information retrieval, where understanding the similarity between sentences is crucial. It enhances the capabilities of BERT by enabling it to provide more contextually relevant sentence representations.