SELF INTRO

Hi, I'm Thoufeeq M. from Bangalore. I have 2 years of experience in data science and currently work at AIML LABS Pvt Ltd in Bangalore. Throughout my career, I've developed a skill set in data science and machine learning, NLP and GENAI which I'd like to elaborate on.

I am proficient in Python with the libraries of pandas, NumPy, seaborn and matplotlib, spacy for NLP, SciPy for scientific computing, and scikit-learn for machine learning, and Streamlit.

In terms of frameworks, I have hands-on experience with TensorFlow, Keras, Pytorch, for the machine learning, and Flask for deploying models as web applications, spacy, NLTK and transformer for the nlp, and the Huggingfaces transformer,GPT-4API, DALL-E for the genai

My expertise in machine learning spans various algorithms, including linear regression, logistic regression, decision trees, random forests, Naïve Bayes, gradient boosting, AdaBoost, KNN classification, K-means clustering, and support vector machines (SVM). Additionally, I have a solid background in deep learning, having worked with artificial neural networks (ANN), convolutional neural networks (CNN), and recurrent neural networks (RNN).

My responsibilities encompass the entire data science workflow: data preprocessing, exploratory data analysis (EDA), feature engineering, model building, evaluation, and deployment., conduct thorough EDA to uncover insights, create and select relevant features to enhance model performance, and deploy models into production environments using Flask to create APIs.

I thrive in both individual and team settings, collaborating with cross-functional teams including data engineers, software developers, and business analysts to deliver comprehensive data science solutions.

Iam work with the 3 project like

I) **E-commerce Recommendation System-ML**
II) **Text Summarization for Document Management-NLP**
III) **Personalized Recommender System(Restaurant Recommendation - GenAi)**

E-commerce Recommendation System-Client(H&M)

Project Overview: I worked on developing an E-commerce Recommendation System to enhance user engagement and retention by offering personalized product suggestions on an e-commerce platform.

Project Objective: The goal was to create a system that provides tailored product recommendations to users, improving their shopping experience and increasing the chances of making a purchase.

Key Responsibilities:

1. **Data Collection and Integration:**
   - Gathered data from various sources, including user interactions, purchase history, and product details.
   - Integrated data from multiple databases and APIs to create a comprehensive dataset.
2. **Data Preprocessing and Cleaning:**
   - Cleaned and preprocessed the data to handle missing values, outliers, and inconsistencies, ensuring data quality and reliability.
3. **Feature Engineering:**
   - Extracted relevant features like user behavior patterns and product attributes.
   - Used techniques such as correlation-based feature selection and dimensionality reduction to optimize the features.
4. **Model Selection and Training:**
   - Evaluated various machine learning algorithms, including collaborative filtering (using matrix factorization and k-NN), content-based filtering, and hybrid models.
   - Trained models to learn user preferences and product features.
5. **Evaluation and Optimization:**
   - Assessed model performance using metrics like precision, recall, and F1score.
   - Used cross-validation to ensure model robustness and continuously improved the models.
6. **Deployment and Monitoring:**
   - Deployed the final models using Flask to create a RESTful API for integration with the e-commerce platform.
   - Monitored system performance post-deployment and gathered user feedback for improvements.

Results and Impact:

- The recommendation system enhanced user engagement, resulting in a higher click-through rate (CTR) and increased average time spent on the platform.
- It also boosted sales, as users found products of interest more easily.

Summary: This project demonstrated my ability to build and deploy a complete machine learning solution, from data collection to production deployment. It showcased my skills in data preprocessing, feature engineering, model development, and system integration, leading to improved user engagement and increased sales.

Project Explanation: Text Summarization for Document Classification – (Dollar Tree)

Problem Statement:
The goal of this project was to develop a system that could automatically summarize documents and classify them into specific categories based on their content. This would help in automating document management, making information retrieval faster, and improving understanding of the content.

Key Steps in the Project:

1. **Data Collection:**
   We gathered documents from various sources, such as web scraping and APIs. Along with the documents, we also collected relevant metadata and category labels to use in training the model.
2. **Data Exploration:**
   We loaded the data into a structured format to understand its composition. This helped us identify any missing information, outliers, or inconsistencies early in the process.
3. **Text Quality Assessment and Cleaning:**
   We cleaned the text data to improve its quality. This included correcting spelling errors, expanding abbreviations, removing HTML tags, and handling special characters. The goal was to ensure that the text was clean and ready for analysis.
4. **Text Preprocessing:**
   We standardized the text by converting it to lowercase, removing punctuation and stopwords (like "and" or "the"), and reducing words to their base forms using techniques like stemming and lemmatization. This helped in reducing noise and focusing on meaningful content.
5. **Tokenization:**
   We broke down the text into smaller units, called tokens (words or phrases), which made it easier to analyze the text and extract features for the model.
6. **Data Labeling:**
   Each document was labeled with a category based on its content. This step was crucial for training the model, as it learned to associate certain types of content with specific categories.
7. **Dataset Splitting:**
   We split the data into training and testing sets. This allowed us to train the model on one part of the data and evaluate its performance on another, ensuring that it could generalize well to new, unseen documents.
8. **Feature Extraction:**
   We converted the text into numerical representations using methods like TF-IDF and word embeddings. This transformation was essential for the machine learning model to process and learn from the text data effectively.
9. **Model Selection and Training:**
   We evaluated different machine learning models, including Naive Bayes, SVM, and Random Forest, to find the best fit for document classification. The models were trained using the processed data to learn patterns and relationships.
10. **Model Evaluation:**
    We tested the model's performance using metrics like accuracy, precision, recall, and F1-score. This helped us understand how well the model was classifying the documents and where improvements could be made.

11. **Model Interpretation and Visualization:**
    We analyzed the model to understand which features (words or phrases) were most important in making predictions. Visualizations, like word clouds, provided insights into the key themes identified by the model.
12. **Deployment and Continuous Improvement:**
    We deployed the final model into a production environment, making it accessible via a web application or API. Post-deployment, we monitored its performance, collected feedback, and made updates to improve its accuracy and efficiency.

Results and Impact:

- The system successfully automated the summarization and classification of documents, making it easier to manage large volumes of text.
- It improved the speed and accuracy of information retrieval, enabling users to find relevant content more efficiently.
- This project demonstrated my ability to handle complex tasks, from data collection and preprocessing to model training and deployment, highlighting my skills in natural language processing, machine learning, and system integration.

In summary, this project showcases my end-to-end experience in developing a robust, real-world solution that enhances document management and information retrieval through advanced machine learning techniques.

Virtual Assistant Chatbot Project-Genai- 7-Eleven.

I'm currently working on a generative AI project aimed at developing a virtual assistant chatbot for weather updates and restaurant recommendations for our client, 7-Eleven. The chatbot's purpose is to provide personalized restaurant recommendations and answer user inquiries. These inquiries may involve questions about top-rated restaurants, cuisine types, locations, reviews, opening hours, and how to make reservations. Previously, we had a basic chatbot, but now our client is seeking a generative AI chatbot with advanced recommendation features tailored for food enthusiasts.

Key Responsibilities

1. **Package Installation and Library Importation**:
    - I installed and set up all the necessary packages for this project, including LangChain, PyPDF, OpenAI, and tiktoken. I also imported essential libraries such as PyPDFLoader, HuggingFaceEmbeddings, and Docx2txtLoader.
2. **Data Handling and Preprocessing**:
    - The project involved loading unstructured data from various documents and extracting text from them. I split these documents into manageable chunks to facilitate efficient processing and analysis.
3. **Embedding and Database Setup**:
    - I downloaded embeddings from Hugging Face, utilizing the Sentence Transformer embeddings. For storage, I set up FAISS (Facebook AI Similarity Search) as our vector database, ensuring efficient similarity search and retrieval operations.

4. **Model Download and Setup**:
    - After logging into my Hugging Face account, I downloaded the Llama 2 7Billion Chat Model and created a Hugging Face pipeline. This model is crucial for generating context-aware responses.
5. **Memory and Retrieval Chain Creation**:
    - I created a memory object to track inputs and outputs, maintaining the conversation's context. Additionally, I developed a Conversation Retrieval QA Chain to manage the chatbot's interactions and responses effectively. Key parameters such as top k, top p, model size, token size, temperature, context window, stop sequence, and frequency were configured to optimize the chatbot's performance.

Technologies Used

- **LLM**: Employed large language models like GPT-3.5 for generating responses.
- **LangChain**: Utilized for managing the conversational logic and flow.
- **Hugging Face**: Leveraged Hugging Face's resources for model and embedding downloads.
- **FAISS**: Used for efficient vector similarity search and storage.

Client: 7-Eleven

- **Founder**: Joe C. Thompson Sr. founded 7-Eleven.
- **Sales Rate**: Among the top convenience store chains globally in terms of sales.
- **Stock**: Publicly traded on the New York Stock Exchange (NYSE) under the ticker symbol "SEVN".
- **Founded Date**: Established in 1927.
- **Global Presence**: Stores worldwide, making it accessible in many countries.
- **Product Range**: Offers a wide variety of products, including snacks, beverages, and groceries.
- **Franchise Model**: Many stores operated by individual franchisees.
- **Innovation**: Known for adopting new retail technologies for better customer experiences.
- **Accessibility**: Famous for its 24/7 operation, providing convenience to customers.
- **Iconic Products**: Notable for its proprietary items like the Slurpee and Big Gulp.

Project Impact

This project is set to significantly enhance the user experience for 7-Eleven customers by providing timely and personalized restaurant recommendations and weather updates. The advanced features and capabilities of the chatbot are expected to improve customer satisfaction and engagement, thereby contributing to the client's business objectives.