

**Live Chat with Virtual Character**

By

Abdelrahman Mostafa (20200827)

Mohamed Hisham (20200483)

Yousef Mohamed (20200669)

Supervised by

Dr. Reda El-Khoribi

T.A. Asmaa Ahmed

Artificial Intelligence Department

Cairo University

2023-2024

Table of Content

[1. Introduction 1](#_Toc170984220)

[1.1. Problem Definition 1](#_Toc170984221)

[1.2. Motivation 1](#_Toc170984222)

[1.3. Applications 1](#_Toc170984223)

[2. Related Work 3](#_Toc170984224)

[2.1. Large Language Models 3](#_Toc170984225)

[2.1.1. Conversational Agents with Embodied Avatars 3](#_Toc170984226)

[2.1.2. Large Language Models in Conversational Agents 4](#_Toc170984227)

[2.1.3. Domain-Specific Conversational Agents 6](#_Toc170984228)

[2.2. Animation Model 6](#_Toc170984229)

[2.3. Text-to-Speech 19](#_Toc170984230)

[5. Methodology 21](#_Toc170984231)

[1. Large Language Models 21](#_Toc170984232)

[2. Animation Model 27](#_Toc170984233)

[6. Deployment 28](#_Toc170984234)

[3.3. Frontend 28](#_Toc170984235)

[3.4. Backend 37](#_Toc170984236)

[4. Results and Analysis 39](#_Toc170984237)

[4.1. Large Language Model 39](#_Toc170984238)

[4.2. Animation Model 40](#_Toc170984239)

[5. References 41](#_Toc170984240)

Abstract

Recently, chatbots have gained attention for their natural language interactions. However, developing chatbots with contextual awareness and varied responses remains challenging.

Our project aims to develop conversation agents that possess unique qualities. These agents will be visually represented by avatars of historical figures such as Napoleon Bonaparte and Albert Einstein. This approach will enhance the interaction and reflect the historical context. We plan to expand beyond text-based interactions and explore incorporating additional ways, such as audio or cameras. This effort aims to create a more comprehensive and natural communication experience.

To achieve this, we will use pre-trained language models such as Microsoft Phi-2 as a foundation and fine-tune them individually for each character. This process will involve tailoring the language models to the specific historical context, personality, and knowledge base of each figure. Our approach is based on using the strengths of LLMs to generate human-like text. We plan to fine-tune individual models for each character to ensure that each response aligns with their unique traits and speech patterns. This helps to create more engaging and immersive conversations.

The applications of AI (Artificial Intelligence) agents simulating historical figures are wide-ranging and offer significant value across various domains. In education, they can serve as personalized virtual tutors, bringing historical figures to life and facilitating interactive learning experiences that enhance comprehension and critical thinking. In the entertainment industry, these agents create immersive and captivating experiences by enabling audiences to engage in interactive storytelling, games, and virtual reality experiences with iconic figures from history. Museums and historical sites can use them to provide visitors with a more interactive and enriching experience by incorporating virtual interactions with historical figures into exhibits.

Beyond these specific domains, the applications of these agents extend to any field where interaction with historical figures could be beneficial, such as healthcare and business.

Overall, these agents have the potential to deepen our understanding of the past, inform our present, and inspire our future.

Abbreviations list.

* CLM: Causal Language Model
* CA: Conversational Agent
* GPT: Generative Pre-trained Transformer
* LORA: LOw-Rank Adaptation
* PEFT: Parameter-Efficient Fine-Tuning
* RAG: Retrieval-Augmented-Generation
* TTS: Text-to-Speech
* STT: Speech to Text
* Conv-LSTM: Convolutional Neural Network with Long Short-Term Memory.
* QA: Question Answering
* GLUE: General Language Understanding Evaluation benchmark
* RAG: Retrieval Augmented Generation
* Frechet Inception Distance (FID): Evaluates the realism of generated frames.
* Cumulative Probability Blur Detection (CPBD): Assesses the sharpness of generated frames.
* Cosine Similarity (CSIM): Measures identity preservation by calculating the cosine similarity of identity embeddings between source images and generated frames using ArcFace.
* LSE-D: Distance score.
* LSE-C: Confidence score.

# Introduction

## Problem Definition

Studying history helps us learn from the past, make better choices for the future, and develop critical thinking skills. It may not be fun to study history, and it may be exhausting to search manually for specific historical characters or events. This can be a daunting task.

## Motivation

Our objective for this project is to use advanced technology to improve the capability of chatbots to understand and respond to historical inquiries with a higher level of contextual accuracy. With this improvement, students will have a more engaging and enjoyable learning experience while studying history with the assistance of chatbots.

## Applications

The applications of AI agents simulating historical figures are wide-ranging and offer significant value across various domains, including:

* **Education**: These agents can serve as personalized virtual tutors, bringing historical figures to life and facilitating interactive learning experiences that enhance comprehension and critical thinking. these tutors can Weave a Narrative, instead of dry facts and searching many resources, they can immerse you in historical events through interactive storytelling in one place. Imagine Julius Caesar guiding you through the battlefields of Gaul, or Marie Curie sharing her scientific discoveries in her own lab. these tutors can Spark Curiosity, they can present you with diverse viewpoints, hidden stories, and unexpected connections, igniting your interest and motivating you to delve deeper. History comes alive, revealing its complexities and intrigue. these tutors can Foster Comprehension, they can tailor their explanations to your understanding, providing additional information or clarifying complex concepts. Think of them as patient and adaptable teachers who adjust their pace based on your needs.
* **Entertainment**: They can create immersive and captivating experiences by enabling audiences to engage in interactive storytelling, games, and virtual reality experiences with iconic figures from history.
* **Museums and historical sites**: They can provide visitors with a more interactive and enriching experience by incorporating virtual interactions with historical figures into exhibits.

Two men in clothing posing for the camera

Description automatically generatedBy leveraging a multi-model approach, dialogue systems can achieve a higher level of characterization, providing users with more personalized and immersive experiences. By adhering to these guidelines, developers can create chatbots that offer a seamless and intuitive user experience, making them indispensable tools for various applications, including customer service, information retrieval, and entertainment.

# Related Work

The field of conversational agents (CAs) is rapidly evolving, with a growing emphasis on enhancing their contextual understanding, response generation, and overall engagement. This project explores a multi-pronged approach towards achieving these goals.

## Large Language Models

### Conversational Agents with Embodied Avatars

* **FurChat**: This project explored a similar approach of combining a large language model (LLM) with an embodied avatar for human interaction. FurChat demonstrated the potential for engaging and informative conversations in a physical setting. However, it focused on general information retrieval, not domain-specific assistance.
* **ERICA**: This project developed an embodied social robot with advanced dialogue capabilities. While not using LLMs, ERICA highlighted the potential of embodied agents for emotional engagement and social interaction.

### Large Language Models in Conversational Agents

* **LaMDA**: This Google AI project demonstrates the power of LLMs for generating human-like dialogue. However, LaMDA focuses on open-ended conversational scenarios and does not address domain-specific expertise.
* **Phi-2**: this offers a readily available foundation for building advanced CAs (e.g., [Radford et al., 2022]). It is a transformer with 2.7 billion parameters. It was trained using the same data sources as Phi-1.5, augmented with a new data source that consists of various NLP synthetic texts and filtered websites (for safety and educational value). When assessed against benchmarks Testing common sense, language understanding, and logical reasoning, Phi-2 highlighted a state-of-the-art performance among models with less than 13 billion parameters. It also employs safety tensors to reduce toxicity and bias in the generated text. It has 24 layers, 32 attention heads, and a hidden size of 40962 with a context length of 2048 tokens. It uses the next-word prediction objective to learn from the training data. In training, it trained on a dataset of size 250B tokens on a 96xA100-80G GPU for 14 days (about 2 weeks), a combination of NLP synthetic data created by AOAI GPT-3.5 and filtered web data from Falcon RefinedWeb and SlimPajama, which was assessed by AOAI GPT-4.
* **Llama-2**: this is a family of pre-trained and fine-tuned large language models (LLMs) released by Meta AI in 2023. These models are made freely available for research and commercial purposes, which has drawn significant interest in the AI community. It refers to a family of second-generation LLMs developed by Meta. These models are designed for various natural language processing tasks, including dialogue generation and text completion. They are available for both research and commercial use. Llama 2 is an auto-regress language-optimized transformer. The tuned versions use supervised fine-tuning (SFT) and reinforcement learning with human feedback (RLHF) to align with human preferences for helpfulness and safety. It has multiple versions, which are 7B, 13B, and 70B with a context length of 4K tokens.
* **Megatron-Turing NLG**: This model from NVIDIA highlights the potential of LLMs for factual language generation. However, it primarily focuses on factual summarization and lacks the interactive dialogue capabilities needed for a conversational agent.
* **BlenderBot**: This Facebook AI project explores LLMs for open-domain dialogue with a focus on factual grounding. While demonstrating progress, it still faces challenges in achieving robust and consistent domain-specific assistance.

### Domain-Specific Conversational Agents

Several industries have developed chatbots for specific tasks like customer service or technical support. However, these often lack the embodiment and multi-modal capabilities of your project.

* **Mitsuku**: This chatbot exhibits impressive performance in open-domain conversation tasks. However, it lacks the embodiment and domain-specific focus of your project.
* **Cleverbot**: This online chatbot learns through user interactions but does not leverage domain-specific knowledge or LLMs.

## Animation Model

The creation of talking head videos from a single-face image and speech audio is fraught with challenges, such as unnatural head movements, distorted facial expressions, and modifications to the subject's identity. These issues are attributed to the reliance on learning from coupled 2D motion fields, which can lead to unnatural and incoherent results. Moreover, the use of explicit 3D information has been found to introduce its own set of problems, such as stiff expressions and videos that lack coherence.

**Animation models** bring the interaction to life by creating real-time animations of characters that deliver the responses. These models use advanced rendering techniques to produce high-quality animations that reflect the emotional tone and context of the conversation.

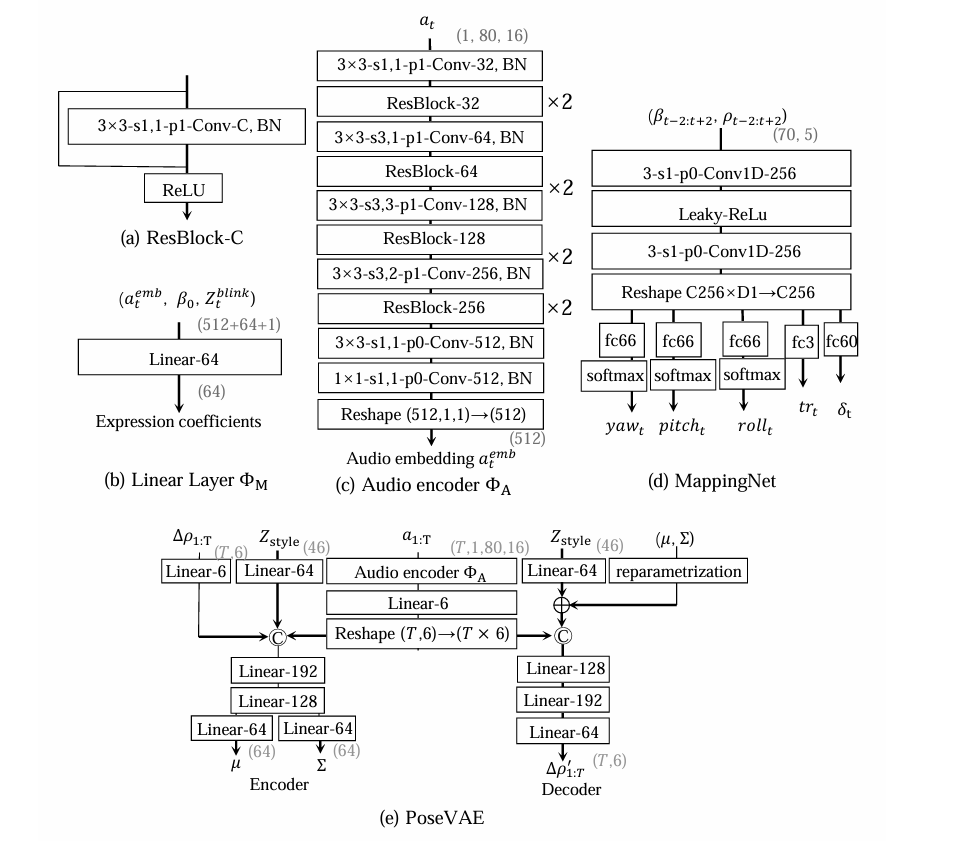
Generating talking head videos from a face image and a piece of speech audio remains a complex challenge, often resulting in unnatural head movements, distorted expressions, and identity modifications. These issues primarily stem from learning based on coupled 2D motion fields. Alternatively, using explicit 3D information can lead to problems like stiff expressions and incoherent video sequences.

There are Many Models like:

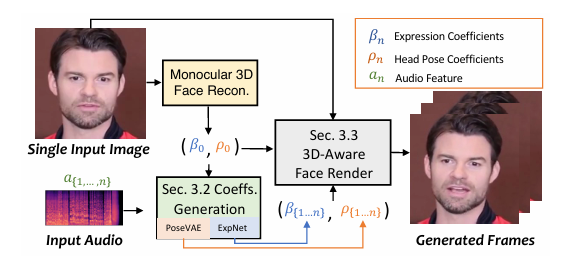
1. Talking Character by Google:
   1. Google’s Partner Innovation team has developed a customizable 3D avatar builder called Talking Character.
   2. Developers and users can configure the avatar’s personality, backstory, and knowledge base.
   3. The avatar interacts with users through both text and verbal conversation.
   4. It highlights the power of Large Language Models (LLMs) combined with Google APIs and technologies.
2. Synthesia:
   1. Synthesia offers an AI video generator that creates talking avatars.
   2. You can choose a template, input your script, and generate a talking avatar video.
   3. It is a quick and efficient way to bring characters to life with speech2.
3. General3D:
   1. The General 3D project focuses on talking face generation.
   2. It includes a coarse-to-fine generic (CTFG) talking face model, static-dynamic hybrid (SD-Hybrid) adaptation pipeline, and in-context stylized audio-to-motion (ICS-A2M) model3.
4. Sora by OpenAI:
   1. Sora is a language model that understands prompts and generates expressive characters with vibrant emotions.
   2. It can create multiple shots within a single video, maintaining character consistency and visual style4.
5. GeneFace++:
   1. It achieves high lip-sync, high video realism, and high system efficiency in generating 3D talking faces.
   2. The model focuses on three key goals: generalized audio-lip synchronization, good video quality, and efficient performance1.
   3. It represents a significant advancement from its predecessor, GeneFace.

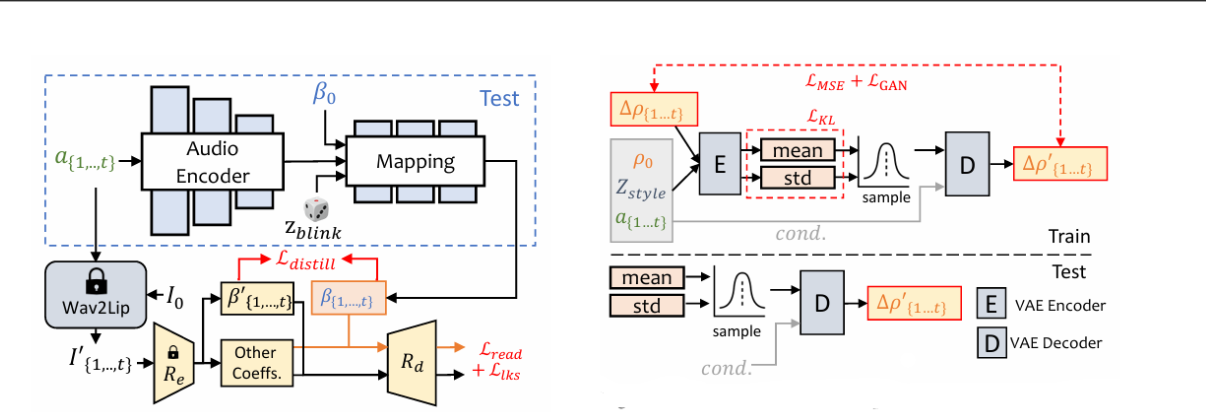
One of the most powerful models is SadTalker. To address these challenges, SadTalker has been developed. This system generates 3D motion coefficients, including head pose and facial expressions, from audio using a 3D Morphable Model (3DMM) and modulates a novel 3D-aware face render to create talking head videos. SadTalker stands out by explicitly modeling the connections between audio and several types of motion coefficients individually, which helps in achieving more accurate facial expressions and head movements. To learn the realistic motion coefficients, we explicitly model the connections between audio and several types of motion coefficients individually. Precisely, we present ExpNet to learn the accurate facial expression from audio by distilling both coefficients and 3D-rendered faces. The SadTalker model, as described in the paper, employs a sophisticated architecture to generate realistic talking head videos by leveraging 3D motion coefficients and an intermediate representation based on 3D Morphable Models (3DMM). Here is an expanded explanation based on the details provided and the paper:

* Components:
  + **3D Motion Coefficients**: These are derived from the original image and serve as the core data used to animate the talking head.
  + **Intermediate Representation:** Utilizes 3DMM, which is a widely used representation for modeling 3D facial structures.
  + **ExpNet**: This component is designed to learn accurate facial expressions directly from audio. It does so by distilling information from both the motion coefficients and 3D-rendered faces, which helps in capturing the nuances of facial expressions that are synchronized with the audio.
  + **PoseVAE**: it is a conditional Variational Autoencoder that synthesizes head motion in several styles. This allows for the generation of head movements that are natural and match the speech style, contributing to the video's overall realism.



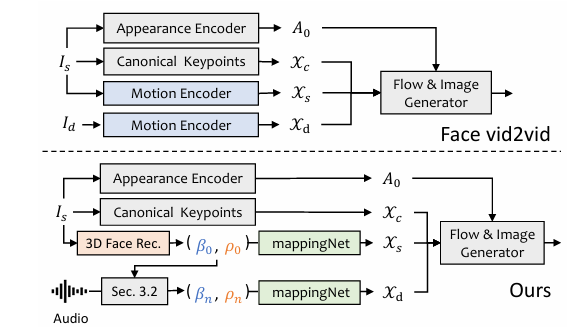
* + **3D Key Points Mapping:** The 3D motion coefficients generated by SadTalker are mapped onto an unsupervised 3D key points space of the proposed face render. This mapping is crucial for synthesizing the final video, ensuring that the motion is accurately reflected in the visual output.





* Benefits:
  + **Individual Modeling**: Explicitly modeling audio-to-motion connections for expression and pose leads to improved realism.
  + **Distillation-based Learning:** ExpNet's learning from both coefficients and rendered faces enhances expression accuracy.
  + **Style Control**: PoseVAE enables generating head motion with several styles based on audio.
  + **Unsupervised 3D Key Points**: Mapping to this space leverages 3D information without introducing stiffness or incoherence.
* Process Flow:

1. Initial Frame Analysis:
   * Extract 3D motion coefficients from the original image using deep 3D reconstruction methods.
   * Initialize the system with the average shape, identity basis, and expression basis.
2. Audio-Driven Motion Generation:
   * Input audio is processed to extract relevant features.
   * ExpNet generates expression coefficients corresponding to the lip movements and other facial expressions.
   * PoseVAE separately generates head pose coefficients to ensure natural head movements.
3. 3D-Aware Rendering:



* Combine the generated motion coefficients (expression and head pose) to render the final talking head.
* Utilize the 3DMM space to maintain high realism and accurate facial features.

1. Final Video Synthesis:

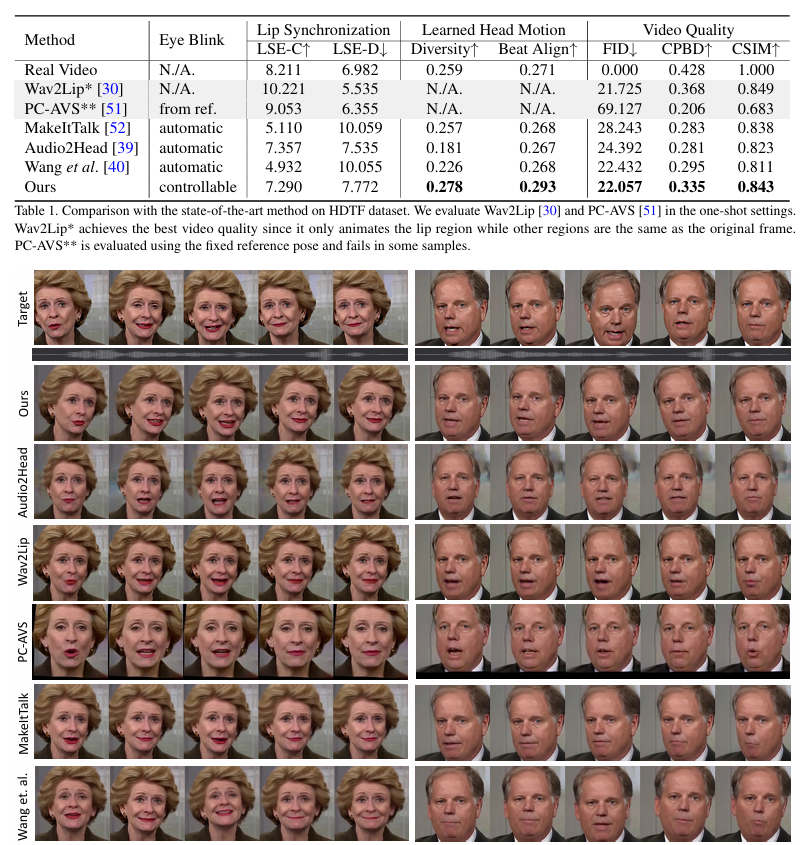
* The combined coefficients are used to create a 3D-aware face render.
* The system synthesizes the final talking head video, ensuring both the lip sync and facial expressions are natural and realistic.

The key innovation in the SadTalker model lies in its ability to effectively decouple and manage the several aspects of facial motion (lip movement and head pose) through specialized networks, leading to highly realistic and synchronized talking head videos. The use of 3DMM as an intermediate representation plays a crucial role in achieving this realism by accurately modeling the 3D structure and motions of the face.

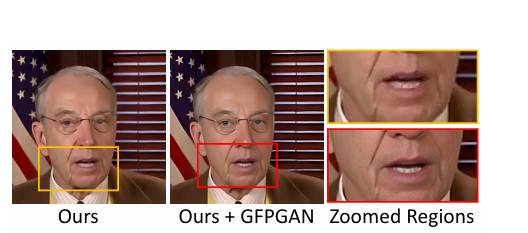
* Datasets
* Training Dataset:
  + VoxCeleb: Contains over 100k videos of 1251 subjects.
  + Preprocessing: Videos are cropped and resized to 256x256 pixels following previous image animation methods.
  + Data Selection: Due to some misalignments between video and audio in VoxCeleb, 1890 aligned videos and audios of 46 subjects were selected for training PoseVAE and ExpNet.
  + Audio Processing: Input audios are down-sampled to 16 kHz and transformed into mel-spectrograms with settings like Wav2Lip.
* Testing Dataset:
  + HDTF: Contains high-resolution, in-the-wild talking head videos.
  + Preprocessing: First 8-second video clips (approximately 70k frames in total) from 346 videos are used. Videos are cropped and resized to 256x256 pixels for evaluation. The first frame of each video is used as the reference image for video generation.
* Training:
  + ExpNet, PoseVAE, and FaceRender are trained separately.
  + Optimizer: Adam optimizer is used for all experiments.
  + Training Environment: Experiments are conducted on 8 A100 GPUs.
  + Temporal Consideration:
    - ExpNet: Learns from continuous 5 frames.
    - PoseVAE: Learned via continuous 32 frames.
    - FaceRender: Generates frames frame-by-frame using coefficients of 5 continuous frames for stability.

#### Evaluation Metrics

* Image Quality:
  + Frechet Inception Distance (FID): Evaluates the realism of generated frames.
  + Cumulative Probability Blur Detection (CPBD): Assesses the sharpness of generated frames.
* Identity Preservation:
  + Cosine Similarity (CSIM): Measures identity preservation by calculating the cosine similarity of identity embeddings between source images and generated frames using ArcFace.
* Lip Synchronization and Mouth Shape:
  + Perceptual Differences: Evaluated using metrics from Wav2Lip:
    - LSE-D: Distance score.
    - LSE-C: Confidence score.
* Head Motion:
  + Diversity: Standard deviation of head motion feature embeddings extracted from generated frames using Hopenet.
  + Audio and Head Motion Alignment: BeatAlign Score as used in Bailando.
* **Experiments**: The SadTalker model effectively combines advanced techniques in 3D face modeling, audio-driven motion generation, and deep learning to create realistic talking head videos. The model is rigorously evaluated on multiple metrics to ensure high-quality, identity-preserving, and synchronized animations. The use of 3D Morphable Models (3DMM) and separate training of motion-related components (ExpNet and PoseVAE) contribute to the system's robust performance. Extensive experiments have been conducted to validate the effectiveness of SadTalker. These studies show the method's superiority in motion realism and video quality, outperforming existing approaches in the field. Extensive experiments demonstrate SadTalker's superiority in terms of:
  + **Motion Naturalness**: More realistic and varied head motions compared to other methods.
  + **Expression Quality**: Accurate and nuanced facial expressions driven by audio.
  + **Video Quality**: Overall higher perceived quality and identity preservation.
* **Comparison**



* **Limitation**



SadTalker model method generates realistic video from a single image and audio, but there are still some limitations in our system. Since 3DMMs do not model the variation of eyes and teeth, the mappingNet in our Face Render will also struggle to synthesize the realistic teeth in some cases. This limitation can be improved via the blind face restoration networks.

Another limitation of our work is that we only concern the lip motion and eye blinking other than the other facial expressions, e.g., emotion and gaze direction. Thus, the generated video has a fixed emotion, which also reduces the realism of generated content. We Consider it as future work.

## Text-to-Speech

TTS models convert textual information into spoken words, enabling machines to communicate with users verbally. Advanced TTS models can generate natural-sounding speech that reflects various emotions, enhancing the user experience. Key features include support for multiple languages, customization options, and high-quality audio output.

**TTS technology** has advanced significantly, with modern models capable of producing natural-sounding speech that closely mimics human voices. These models use deep learning techniques to generate speech that is clear, fluent, and emotionally expressive.

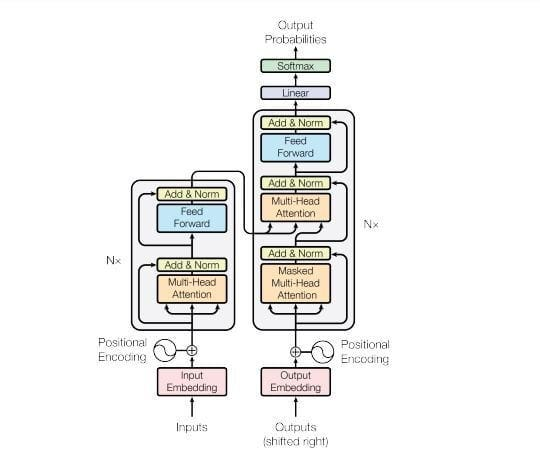
**The TTS model** must convert text into speech that reflects the appropriate emotional tone. This involves adjusting various parameters, such as pitch, tone, and pace, to match the detected emotion. For example, a calm and soothing voice might be used for a supportive response.

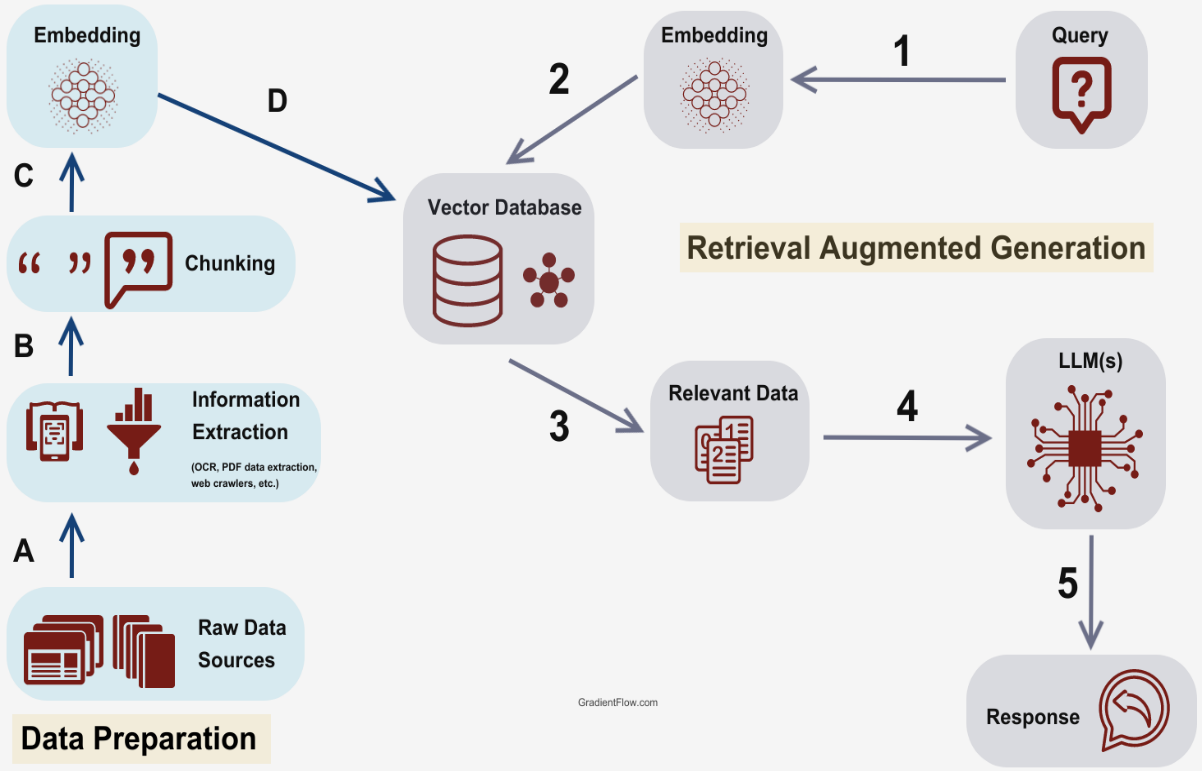
Bark is a transformer-based text-to-audio model created by [Suno](https://www.suno.ai/). Bark can generate highly realistic, multilingual speech as well as other audio - including music, background noise and simple sound effects. The model can also produce nonverbal communications like laughing, sighing, and crying. To support the research community, we are providing access to pretrained model checkpoints ready for inference.

Bark is an innovative text-to-audio model developed by Suno that goes beyond just realistic speech generation. Unlike conventional systems, Bark utilizes a transformer-based architecture to directly convert text prompts into audio, enabling it to produce a wider range of outputs. This means Bark can not only create natural-sounding speech in multiple languages, but also generate background noise, sound effects, and even music. It can even capture non-verbal cues like laughter, sighs, and cries, adding another layer of realism to the audio. Suno offers pre-trained versions of the model for research and development purposes, making it a valuable tool for researchers and developers in the field of artificial intelligence.

# Methodology

## Large Language Models

In interactive systems that involve conversational agents, each character can be characterized by their distinctive way of communicating. This can include their word choices, sentence structure, tone, and other linguistic features that help establish their personality and identity. Employing a multi-model approach can enhance the system's ability to generate diverse and character-specific responses. To design and develop effective chatbots that can engage users in natural and personalized conversations, The following guidelines should be considered:

1. **Collecting and Processing Data**: For each character, gather a substantial dataset of text that reflects their unique speech patterns and personalities. This data can include dialogue, monologue, and other relevant text sources.
   1. **Collecting:** Using web scraping, we will collect about characters, converting it into a Q&A format.
   2. **Processing:** Using Data Serialization can convert complex data structures or objects into a format that can be stored or transmitted and then reconstructed later. It allows data to be saved to files, sent over networks, or stored in databases in a way that preserves its original structure and meaning.
      1. **Some Popular Benefits of Data Serialization**:
         1. Serialized data is typically stored in a binary or text-based format like XML, JSON, or Protocol Buffers.
         2. The reverse process, deserializing, reconstructs the original data structure from the serialized format.
         3. Serialization is commonly used for persisting data to files, storing data in databases, transferring data over networks, and enabling remote method invocation.
         4. Choosing a serialization format depends on factors like data complexity, human readability, performance, and storage space requirements.
      2. **Some Popular Serialization Formats and their Benefits**:
         1. JSON (JavaScript Object Notation): Text-based, human-readable, language-independent, good for simple data structures.
         2. XML (Extensible Markup Language): Text-based, human-readable, language-independent, supports hierarchical data.
         3. Protocol Buffers: Binary, language-independent, optimized for performance, used by Google.
         4. Pickle: Python-specific, binary, allows serializing Python objects with methods.
         5. MessagePack: Binary, compact, language-independent, like JSON
2. **Exploring pre-trained models to leverage existing advancements**: Choose appropriate pre-trained large language models, as the foundation for each character's model to save time and resources. list of suggested LLMs:
   1. **Phi-2/3**: they offer a readily available foundation for building advanced. This offers a readily available foundation for building advanced CAs with low hardware requirements. Phi-3, developed by Microsoft, is a meaningful change in the world of small language models (SLMs). This family of models offers capabilities that rival much larger models, all while remaining lightweight and efficient. The secret sauce lies in Phi-3's training data. It incorporates not only filtered, high-quality web data, but also synthetically generated information. This focus on well-structured and reasoning-intensive content allows Phi-3 to excel at tasks that often trip up traditional SLMs. Furthermore, Phi-3 undergoes rigorous safety measures during development. Through techniques like supervised fine-tuning and preference optimization, the models are aligned with human values and safety principles. This ensures Phi-3 is not only powerful but also responsible. Available in diverse sizes, from the compact Phi-3 Mini to larger options, Phi-3 caters to a range of needs. The Phi-3 family is poised to revolutionize how we interact with AI, bringing powerful language processing capabilities to even resource-constrained environments.
   2. **Llama-2/3**: they are family of pre-trained and fine-tuned large language models (LLMs) released by Meta AI in 2023. These models are made freely available for research and commercial purposes, which has drawn significant interest in the AI community. Llama3 is a powerhouse large language model (LLM) created by Meta AI. It stands out for being one of the most capable LLMs publicly available. This means researchers, businesses, and even individuals can experiment with its capabilities. Compared to its predecessor, Llama2, Llama3 boasts several improvements. Firstly, it utilizes a more efficient tokenizer, a translator for the model, allowing it to process information with a smaller vocabulary while maintaining performance. Additionally, Llama3 comes in two sizes: 8 billion and 70 billion parameters. These parameters are like the building blocks of the model's knowledge, with a higher number indicating a greater capacity for complex tasks. Finally, to enhance its processing speed, Llama3 incorporates a technique called grouped query attention. This allows the model to focus on relevant parts of the data more efficiently, leading to faster results. Overall, Llama3's advancements in efficiency, size options, and processing speed make it a valuable tool for anyone looking to push the boundaries of what is possible with LLMs.
   3. **Mistral**: Mistral AI is a rising star in the large language model (LLM) landscape, known for its commitment to open-source innovation. Founded in 2023 by AI veterans, Mistral has made a name for itself by developing powerful LLMs with a focus on accessibility. They offer a range of models, including the dense Mistral 7B, known for its efficiency, and the Mixtral series, which utilizes a sparse mixture-of-experts approach. The most recent addition, Mixtral 8x22B, stands out as the strongest open-source LLM currently available, boasting impressive performance with a smaller parameter size compared to competitors. This focus on efficiency translates to faster inference times and lower costs. Mistral prioritizes open-source development, offering several models with freely downloadable weights and an Apache 2.0 license, allowing for unrestricted use, and fostering collaboration within the AI community. They also provide access to additional models through their developer platform, making Mistral's innovative capabilities readily available to a wide range of users.
3. **Strategies of Adjusting pre-trained models:**
   1. **Fine-tuning individual models**: Create chatbots tailored to the specific characteristics and personalities of each historical figure. Fine-tune each pre-trained model on its respective character-specific dataset. This involves adjusting the model's parameters to optimize its performance on the task of generating text in the style of that character.
   2. **Retrieval Augmented Generation**: Retrieval-Augmented Generation (RAG) is an AI framework that combines large language models (LLMs) with external knowledge bases to improve the accuracy, relevance, and reliability of generated text. It works by retrieving relevant information from a database using a query generated by the LLM, then integrating that information into the LLM's input to generate more accurate and contextually relevant responses. 
      1. **Key benefits of RAG include:**

* Providing LLMs access to up-to-date, authoritative information beyond their static training data
* Improving factual accuracy by grounding LLM outputs in curated knowledge bases
* Enhancing coherence and relevance by retrieving context-specific information
* Promoting consistency by conditioning generation on retrieved facts
* Enabling efficient retrieval using vector databases
* Reducing computational costs by avoiding retraining the LLM for latest information
  + 1. **RAG operates in two main phases:**

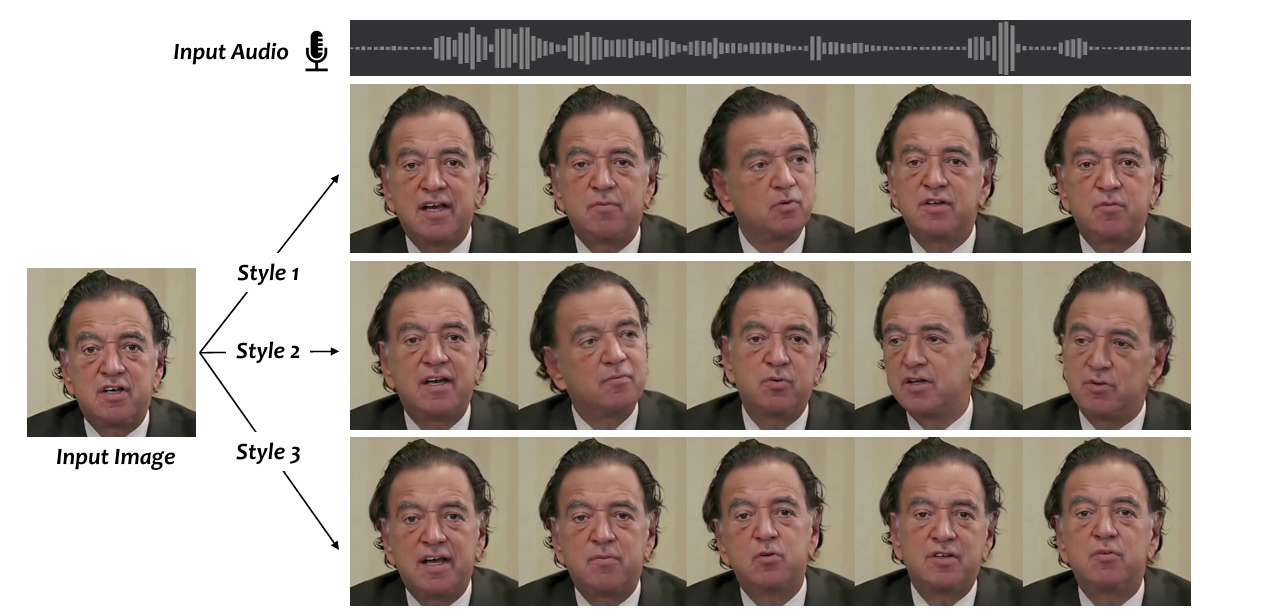
1. **Retrieval**: Algorithms search for and retrieve relevant information snippets based on the user's prompt or question. In open-domain settings, this can come from indexed web documents, while in enterprise settings, a narrower set of trusted sources is used.
2. **Generation**: The retrieved information is appended to the user's prompt and passed to the LLM, which generates an answer tailored to that specific context by drawing from the augmented prompt and its internal training data representation.

By combining the strengths of traditional information retrieval with the generative capabilities of LLMs, RAG enables AI systems to provide more accurate, up-to-date, and relevant responses, while also giving users insight into the model's reasoning process

## Animation Model

To integrate the computer vision system into our live chat system, we need to add the Talking Face Animator system. Generating realistic talking head videos using a single face image and audio presents several challenges, including unnatural head movements, distorted expressions, and loss of identity. This paper argues that these issues stem from learning from coupled 2D motion fields. While directly using 3D information can improve realism, it can also lead to stiff expressions and incoherent videos.

**SadTalker**: This work proposes a novel approach, SadTalker, that leverages 3D motion coefficients (head pose, expression) and implicitly modulates a 3D-aware face renderer for generating talking heads.



# Deployment

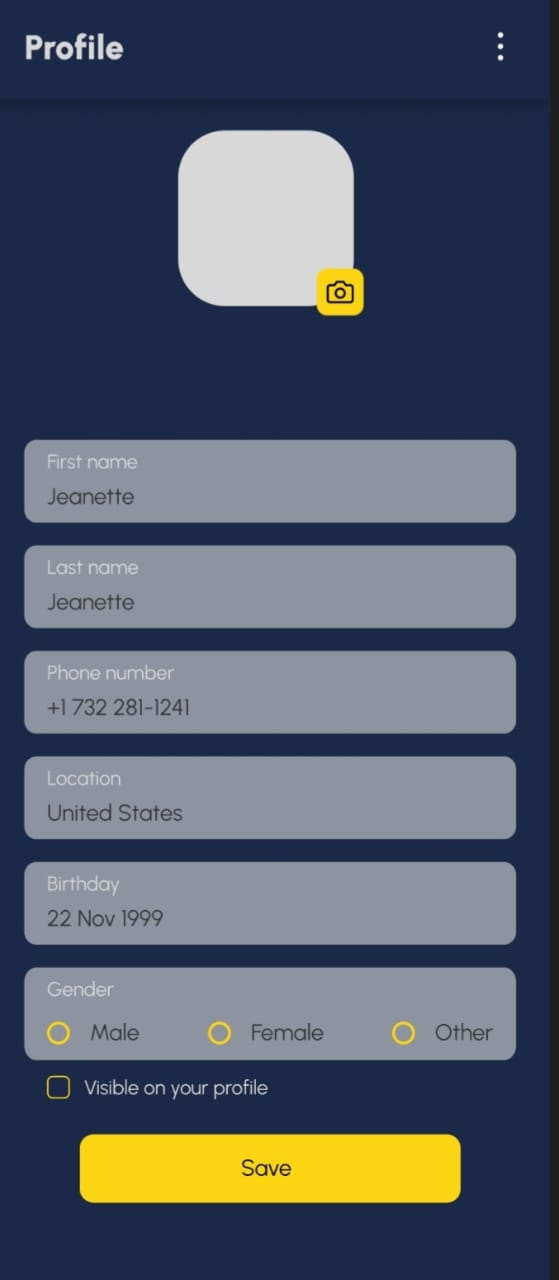
## Frontend

Flutter has emerged as a meaningful change in mobile app development due to its ability to bridge the gap between cross-platform development and native app performance. Traditionally, developers had to choose between building separate apps for iOS and Android or sacrificing performance with a single codebase solution. Flutter breaks this mold by using its own high-performance rendering engine to draw UI elements. This allows developers to write a single codebase in Dart, a modern and readable language, which compiles into native code for both Android and iOS. This translates to significant advantages:

Firstly, Flutter fosters faster development cycles. By eliminating the need to maintain separate codebases, developers can focus on building features and delivering a consistent user experience across platforms. This not only reduces development time but also streamlines maintenance and updates.

Secondly, Flutter apps boast exceptional performance and a smooth user experience. The custom rendering engine renders UI elements directly, bypassing the need for WebView technologies often used in cross-platform development. This results in apps that feel as responsive and native as those built with platform-specific tools.

Finally, Flutter empowers developers to create visually stunning and unique app experiences. Its rich widget library provides a vast array of pre-built UI components and the flexibility to craft custom widgets. This allows developers to design apps with a distinct brand identity that adheres to platform design guidelines, ensuring a familiar yet unique feel for users on both Android and iOS.

Let us talk about our profile creation page, it allows users to enter their first and last name, phone number, location, birthday, and gender. Users can also choose to make their gender visible on their profile.

Here is a more expanded description of what the profile creation app might offer:

* User Authentication: Users would be required to create an account or log in before using the app. This would allow them to save their profile information and access it from any device.
* Profile Picture: Users can add a profile picture to personalize their profile.
* Bio: A section where users can write a short bio about themselves.
* Contact Information: In addition to their phone number, users can include their email address and social media handles.
* Privacy Settings: Users can control what information is visible on their profile and who can see it.

A screenshot of a login screen

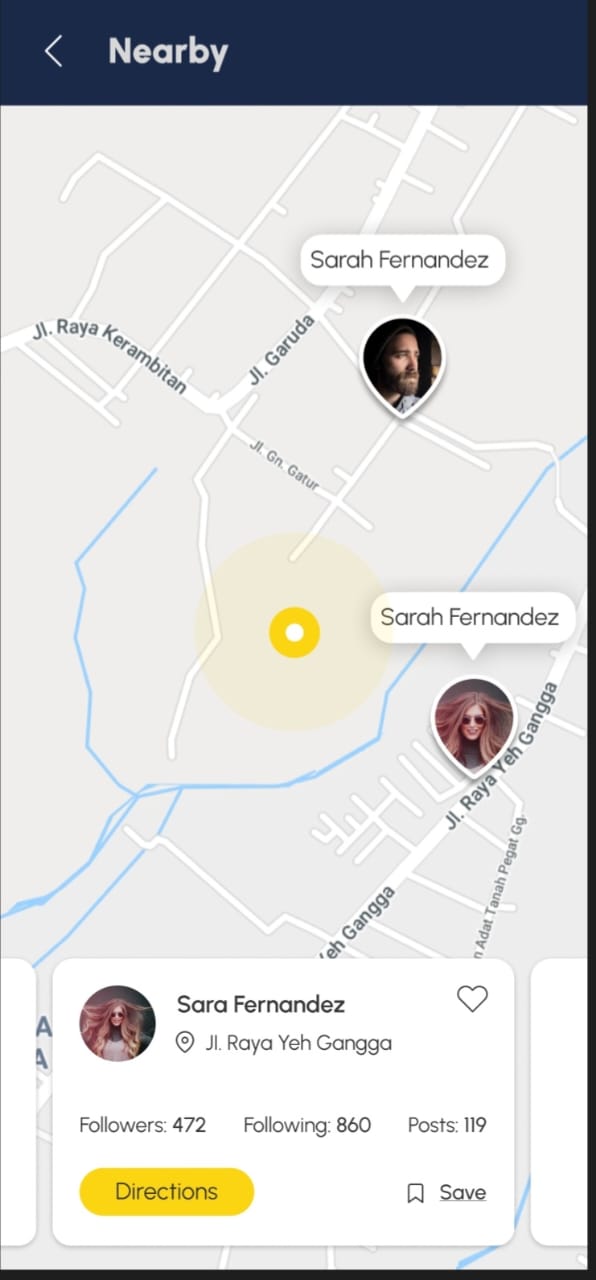
Description automatically generatedLet us talk about our sign-up page, Users can create an account by entering their username, email address, first name, and last name. There is also a yellow button labeled "Sign up" at the bottom of the form.

**Key Features:**

* **Account Creation:** The app allows users to sign up for an account by providing their username, email address, first name, and last name. This account creation process is likely the first step users take to access the core features of your app.
* **Username:** A unique identifier a user creates to be recognized on the app.
* **Email Address:** Users provide their email address during sign-up. This can be used to send them valuable information from the app, such as password reset instructions or notifications.
* **First and Last Name:** Users enter their first and last name during sign-up. This information may be displayed on their profile or used to personalize their experience within the app.
* **Password:** In addition to the information requested in the image, users may also be required to create a password during sign-up. This password would be used to authenticate users when they log in to the app.

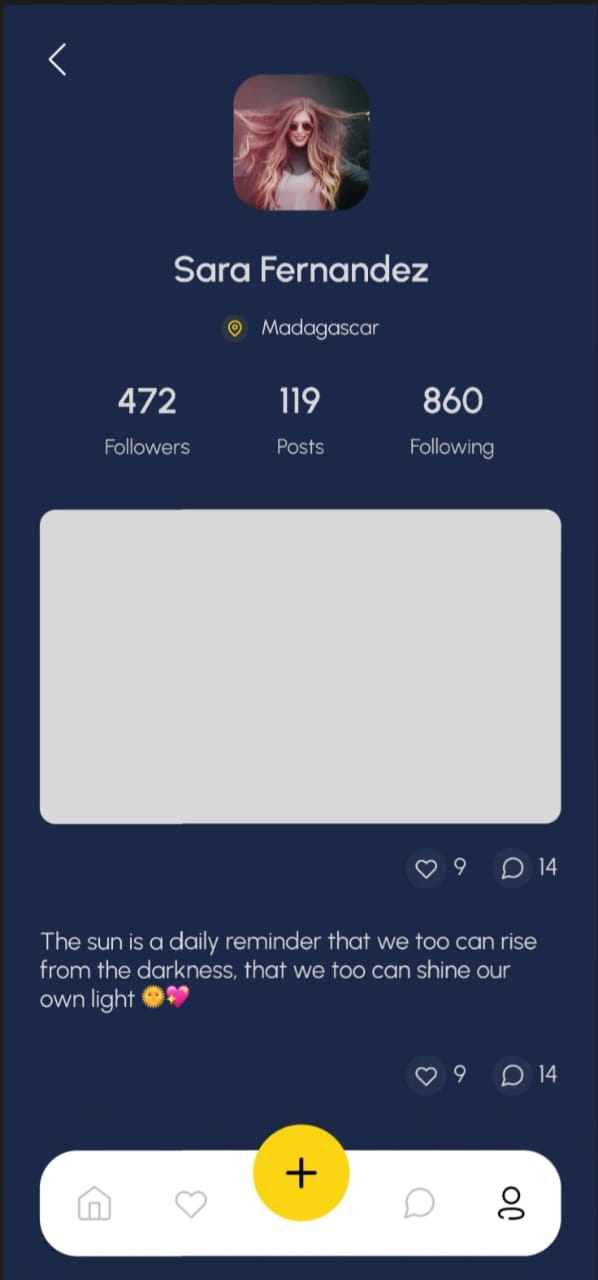
**Possible Additional Features:**

* **Confirm Password:** As an extra security measure, you can include a field where users re-type their password during sign-up. This helps to ensure that the password was entered correctly the first time.
* **Social Media Login:** Besides creating an account with username and email, users can sign up using their existing social media accounts.
* **Terms and Conditions:** You may want to include a link to your app's terms and conditions that users must agree to before signing up. This legal document outlines the rules and regulations that users must abide by when using your app.

Let us talk about our Nearby page, our Flutter app allows users to see the locations of their friends who are nearby and chat with them in real-time. This can be useful for coordinating meetups, staying safe when going out at night, or simply tracking your friends' whereabouts.

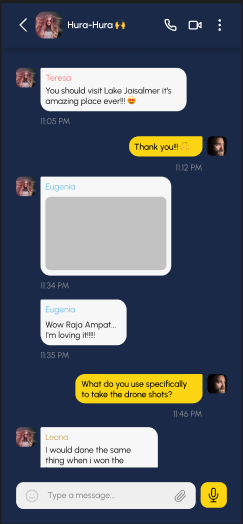
Key Features:

* Real-time Location Sharing: Leveraging location services on users' devices, the app can display their friends' locations on a map. This could be a static snapshot or update dynamically as users move around.
* Privacy Controls: Users should be able to control who can see their location and for how long. This can be implemented through privacy settings where users can choose to share their location with all friends, selected friends, or no one. There could also be an option to share location for a specific duration.
* Chat Functionality: The app should include a chat functionality that allows users to send messages to their friends directly. This could be a basic text chat or include features like sending photos, videos, or voice messages.
* Map Integration: To visualize user locations, you can integrate a mapping service like Google Maps or Mapbox into your app. This will allow users to see their friends' locations on a familiar map interface.

Let us talk about our profile page, our Flutter app is a social media profile page. The profile page displays a user’s profile picture, location, follower count, post count, following count, bio, and a button to follow the user.

**Key Features:**

* **Profile Information:** The profile page displays basic user information including their name, location, follower count, post count, following count, and bio.
* **Bio:** A section where users can write a brief introduction about themselves.
* **Follow Functionality:** Users can follow other users to see their posts and updates in a feed. There is also a follower count that shows how many people are following the user.
* **Posts:** While not explicitly shown in the image, it is likely that users can also create and share posts on the app. This could include text posts, photos, or videos. There is also a post count that shows how many posts a user has shared.

Let us talk about our chat page, our Flutter app is a chat interface where users can converse with a Large Language Model (LLM) character. This LLM character is powered by Retrieval Augmented Generation (RAG), which allows it to access and process information from the internet to respond to user queries in an informative way.

Key Features:

* Chat Interface: The app provides a user-friendly chat interface where users can type in their questions or prompts and receive responses from the LLM character.
* Large Language Model (LLM): The app integrates an LLM that is trained on a massive dataset of text and code. This allows the LLM to communicate and generate human-like text in response to a wide range of prompts and questions.
* Retrieval Augmented Generation (RAG): The LLM leverages RAG technology to access and process information from the internet in real-time. This enables the LLM to provide more comprehensive and informative responses to user queries. For instance, if a user asks the LLM “What is the capital of France?”, the LLM can not only respond with “Paris” but also provide additional information about Paris, such as its history, landmarks, or cultural significance.

## Backend

The development and deployment process for modern applications can be complex, especially when dealing with multiple components and dependencies. This is where come in, forming a powerful trio that streamlines the entire workflow. We use:

* **Docker:** It is popular because it simplifies the process of developing, deploying, and running applications. It packages your application and its dependencies into a self-contained unit called a container. This container isolates your application from the underlying system, ensuring it runs consistently regardless of the environment. This makes Docker ideal for developers who can test and share code without worrying about conflicts, and for system administrators who can deploy applications quickly and efficiently.
* **Docker Compose:** It is a valuable tool because it simplifies managing applications that rely on multiple containers. Instead of individually starting and configuring each container, you define everything in a single YAML file. This file specifies the containers needed, their configurations, and how they interact. With a single command, you can launch and manage all the containers at once, streamlining development, testing, and deployment workflows.
* **FastAPI:** It has become a popular choice for building APIs due to its combination of speed, ease of use, and modern features. It boasts high performance rivaling frameworks like Node.js and Go, thanks in part to its asynchronous nature. This means your code can handle multiple requests efficiently. FastAPI is also designed to be developer-friendly, with features like automatic documentation reducing development time. Python developers familiar with type hints will find the syntax intuitive, and built-in features like data validation help catch errors early on.

# Results and Analysis

## Large Language Model

**Our initial attempt involved fine-tuning a large language model (LLM), but limitations like our restricted computing resources and lack of sufficient training data hampered our efforts. We have since transitioned to using LLAMA-3 with a Retrieval-Augmented Generation (RAG) approach. RAG offers several advantages that make it preferable to fine-tuning in our scenario.**

* **Freshness and Relevance:** RAG excels at keeping responses current by directly retrieving and incorporating information from continuously updated external knowledge sources. This contrasts with fine-tuning, which relies on static training data that can become outdated quickly, especially in rapidly evolving fields. For tasks demanding up-to-the-minute information, like summarizing current events or offering dynamic product details, RAG is the perfect choice.
* **Factual Accuracy:** RAG boasts a significant edge in factual accuracy. By grounding its responses in verified external data, RAG minimizes the risk of generating nonsensical or misleading information, often referred to as "hallucinations" in the context of LLMs. Fine-tuning, while helpful in improving domain-specific knowledge, can still be susceptible to these hallucinations if the training data itself is inaccurate or incomplete.
* **Cost-Effectiveness:** Cost is another area where RAG shines. Fine-tuning often requires a substantial investment in both data and computational resources. Acquiring and labeling enormous amounts of domain-specific data can be expensive, and training these models often necessitates powerful hardware like GPUs. RAG, on the other hand, focuses on managing and updating its external knowledge sources, a less resource-intensive process.

## Animation Model

Breathing life into historical figures took a leap forward thanks to our recent project. By leveraging SadTalker within a Docker container, we successfully transformed a static portrait of Napoleon Bonaparte into a speaking visage. Docker's isolated environment provided a stable and efficient platform for SadTalker's magic to work. SadTalker itself, with its proficiency in generating realistic facial expressions driven by audio, allowed us to meticulously craft the nuances of Napoleon's speech onto his virtual face.

A qr code on a white background

Description automatically generated

# References

1. Zhang, W., Cun, X., Wang, X., Zhang, Y., Shen, X., Guo, Y., Shan, Y., & Wang, F. (2023). SadTalker: Learning Realistic 3D Motion Coefficients for Stylized Audio-Driven Single Image Talking Face Animation. Computer Vision and Pattern Recognition (CVPR).
2. <https://huggingface.co/microsoft/phi-2>
3. [A Comprehensive Guide to Fine-Tuning the Microsoft Phi-2 Model (Free Notebook) | by Mohamed Ahmed Krichen | Dec 2023 | Medium](https://medium.com/@mohamedahmedkrichen/a-comprehensive-guide-to-fine-tuning-the-microsoft-phi-2-model-free-notebook-52a4b5e486aa)
4. Hu, Z., Wang, L., Lan, Y., Xu, W., Lim, E., Bing, L., Xu, X., Poria, S., & Lee, R. K.-W. (2023). LLM-Adapters: An Adapter Family for Parameter-Efficient Fine-Tuning of Large Language Models.
5. Dosovitskiy, A., Beyer, L., Kolesnikov, A., Weissenborn, D., Zhai, X., Unterthiner, T., Dehghani, M., Minderer, M., Heigold, G., Gelly, S., Uszkoreit, J., & Houlsby, N. (2020). An Image is Worth 16x16 Words: Transformers for Image Recognition at Scale
6. Vaswani, A., Shazeer, N., Parmar, N., Uszkoreit, J., Jones, L., Gomez, A. N., Kaiser, L., & Polosukhin, I. (2017). Attention Is All You Need.
7. [https://sadtalker.github.io](https://sadtalker.github.io/)
8. <https://huggingface.co/spaces/vinthony/SadTalker>
9. Zhang, Wenxuan & Cun, Xiaodong & Wang, Xuan & Zhang, Yong & Shen, Xi & Guo, Yu & Shan, Ying & Wang, Fei. (2022). SadTalker: Learning Realistic 3D Motion Coefficients for Stylized Audio-Driven Single Image Talking Face Animation. 10.48550/arXiv.2211.12194.
10. Zhang, W., Cun, X., Wang, X., Zhang, Y., Shen, X., Guo, Y., Shan, Y., & Wang, F. (2022). SadTalker: Learning Realistic 3D Motion Coefficients for Stylized Audio-Driven Single Image Talking Face Animation. ArXiv. /abs/2211.12194
11. Zhang, W., Cun, X., Wang, X., Zhang, Y., Shen, X., Guo, Y., Shan, Y., & Wang, F. (2022). SadTalker: Learning Realistic 3D Motion Coefficients for Stylized Audio-Driven Single Image Talking Face Animation. *ArXiv*. /abs/2211.12194
12. <https://techxplore.com/news/2023-09-embodied-conversational-agent-merges-large.html>
13. <https://www.sciencedirect.com/science/article/pii/S2666920X21000278>
14. Cherakara, S., Wijeratne, H., & Papadopoullos, A. (2023). An embodied conversational agent that merges large language models and domain-specific assistance. arXiv preprint arXiv:2309.02684.
15. Ishiguro, H., Ono, T., Kobayashi, M., & Ikeda, H. (2006). ERICA: A life-sized human-like android. IEEE Intelligent Systems, 21(4), 12-21.
16. Breazeal, C. (2006). Designing sociable robots. MIT Press.
17. A. Barua, M. U. Ahmed, and S. Begum, "A Systematic Literature Review on Multimodal Machine Learning: Applications, Challenges, Gaps and Future Directions," in IEEE Access, vol. 11, pp. 14804-14831, 2023, doi: 10.1109/ACCESS.2023.3243854.
18. Zhang, W., Cun, X., Guo, Y., Wang, X., Shan, Y., Zhang, Y., & Wang, F. (2023). SadTalker: Learning realistic 3D motion coefficients for stylized audio-driven single image talking face animation. ArXiv Preprint, arXiv:2211.12194v2 [cs.CV]. Retrieved from https://arxiv.org/abs/2211.12194v2.
19. Duan, H.-B., Wang, M., Shi, J.-C., Chen, X.-C., & Cao, Y.-P. (n.d.). BakedAvatar: Baking neural fields for real-time head avatar synthesis. State Key Laboratory of Virtual Reality Technology and Systems, Beihang University; Zhongguancun Laboratory; ARC Lab, Tencent PCG.
20. Volker Blanz and Thomas Vetter. A morphable model for the synthesis of 3d faces. In ACM SIGGRAPH, 1999.
21. Kun Cheng, Xiaodong Cun, Yong Zhang, Menghan Xia, Fei Yin, Mingrui Zhu, Xuan Wang, Jue Wang, and Nannan Wang. Videoretalking: Audio-based lip synchronization for talking head video editing in the wild. In SIGGRAPH Asia 2022 Conference Papers, pages 1–9, 2022.
22. Diederik P Kingma and Jimmy Ba. A method for stochastic optimization. arXiv preprint arXiv:1412.6980, 2014. 5 [21] Diederik P Kingma and Max Welling. Auto-encoding varia tional bayes. CoRR, abs/1312.6114, 2014
23. Niranjan D. Narvekar and Lina J. Karam. A no-reference image blur metric based on the cumulative probability of blur detection (cpbd). TIP, 2011.
24. Youxin Pang, Yong Zhang, Weize Quan, Yanbo Fan, Xi aodong Cun, Ying Shan, and Dong-ming Yan. Dpe: Disen tanglement of pose and expression for general video portrait editing. arXiv preprint arXiv:2301.06281, 2023.
25. Aditya Ramesh, Prafulla Dhariwal, Alex Nichol, Casey Chu, and Mark Chen. Hierarchical text-conditional image genera tion with clip latents. arXiv preprint arXiv:2204.06125, 2022.
26. Yurui Ren, Ge Li, Yuanqi Chen, Thomas H Li, and Shan Liu. Pirenderer: Controllable portrait image generation via semantic neural rendering. In ICCV, 2021.
27. <https://huggingface.co/suno/bark>