SMART COUNSELING BOT

Final Year Project Report

By

Qandeel Fatima

Aleena Zahid

Muhammad Abdullah

Muhammad Safi Ullah

In Partial Fulfillment

Of the Requirements for the degree

Bachelor of Engineering in Software Engineering (BESE)

Military College Of Signals

National University of Sciences and Technology

Islamabad, Pakistan (2023)

In the name of ALLAH, the Most benevolent, the Most courteous.

CERTIFICATE OF CORRECTNESS AND APPROVAL

This is to officially state that the thesis work contained in this report for Final Year Project "Smart Counselling Bot" is carried out by Qandeel Fatima, Aleena Zahid, Muhammad Abdullah and Safiullah under my supervision and that in my judgement, is fully ample, in scope and excellence, for the degree of Bachelor of Software Engineering and Military College of Signals, National University of Sciences and Technology(NUST), Islamabad.

Approved by Supervisor

Signature:_____

Name of Supervisor: Prof. Dr. Hammad Afzal

Date:_____.

DECLARATION OF ORIGINALITY

We hereby declare that no portion of work presented in this thesis has been submitted in support of another award or qualification in either this institute or anywhere else.

DEDICATION

To Allah belongs the dominion of the heavens and the earth; He creates what he wills.
He gives to whom He wills female [children], and He gives to whom He wills males.
Or He makes them [both] males and females, and He renders whom He wills barren.
Indeed, He is Knowing and Competent.

(Chapter 25: Surah Ash-Shura: Ayat 49-50)

Dedicated to our beloved families and our country Pakistan.

ACKNOWLEDGEMENTS

We are grateful to Allah Almighty for giving us strength to keep going on with this project, irrespective of many challenges and troubles.

Next, we are grateful to all our families. Without their consistent support and prayers, a work of this magnitude wouldn't have been possible.

We are very grateful to our Project Supervisor Dr. Hammad Afzal who supervised the project in a very encouraging and helpful manner. As a supervisor, her support and supervision has always been a valuable resource for our project.

Last but not the least special acknowledgement to all the members of this group who tolerated each other throughout the whole year.

Plagiarism Certificate (Turnitin Report)

This thesis has 17 similarity index. Turnitin report endorsed by supervisor is attached.

Name and signature of Supervisor

_____
Prof. Dr. Hammad Afzal
(Dept of CS)

Names and signature of Students

_____
Qandeel Fatima

_____
Aleena Zahid

_____
Muhammad Abdullah

_____
Safiullah

TABLE OF CONTENT

List of Figures

List of Tables

ABSTRACT

Problem Statement:

The problem is to create a conversational bot with a synthetic face that can interact with humans in a natural and engaging manner. This involves developing both the natural language processing (NLP) capabilities of the bot to understand and respond to human queries and developing a synthetic face that can convey emotions and expressions to enhance the user experience.

Challenges:

There are several challenges in creating a conversational bot with a synthetic face. Firstly, the NLP capabilities of the bot need to be robust and accurate enough to understand and respond to a wide range of user queries. Secondly, the synthetic face needs to be realistic and expressive enough to convey emotions and engage the user. Finally, integrating the NLP and synthetic face components into a seamless user experience poses a significant technical challenge.

Scope:

The scope of the conversational bot with a synthetic face is broad and can be used in a variety of applications. The bot can be used in customer service, education, entertainment, and other industries where human-like interaction is desirable.

Comparison with Existing Systems/Evaluation Methods:

Existing conversational AI systems typically rely on text-based interactions and do not include a synthetic face component. Therefore, the conversational bot with a synthetic face offers a more engaging and human-like user experience. Evaluation of the system can be done through user testing and feedback to assess the effectiveness of the NLP and synthetic face components.

Conclusion and Future Directions:

Creating a conversational bot with a synthetic face is a challenging task. However in the future, the scope of the system can be expanded to include multi-lingual support and integration with other technologies.

Chapter 1

INTRODUCTION

The software being described is a smart counselling bot, which is an interactive, visual interface for users to communicate with a bot. The bot can understand and respond to audio-based input from users, as well as speak with the user and show basic human emotions. It can maintain a conversation history with each user and provide personalized responses based on the user's past interactions.

The scope of the smart counselling bot software includes the following features and functionality:

1. An interactive, visual interface for users to communicate with the chatbot.
2. The ability to understand and respond to audio-based input from users.
3. The ability to talk back and show emotions in response to user input or as part of a pre-defined conversation flow.
4. The ability to maintain a conversation history with each user and provide personalized responses based on the user's past interactions.

The goals and objectives of the smart counselling bot project are as follows:

1. To create an interactive and engaging interface for users to communicate with a chatbot.
2. To improve the user experience of interacting with a chatbot by adding a visual element(emotions).
3. To provide personalized responses to users based on their past interactions with the bot.

The benefits of the smart counselling bot project include:

1. Improved user engagement and satisfaction with the chatbot experience.
2. The ability to provide more personalized and relevant responses to users.
3. The potential for increased efficiency and effectiveness in communicating with

the chatbot.

4. To eradicate the hesitation element in the communication.

1.1 STYLES

1.1.1 Typeface

Text should be justified, spaced at 1.5, and in the Times New Roman (TNR) font. A single line space and an indentation on the first line of each paragraph are required.

1.1.2 Margins

Left 1.5 inches Right, Top, Lower 1.2 inches.

1.1.3 Headings

The initial letter of Chapter Number 20 TNR (Italics, Bold, Justified to the Right, capital "C") is capitalized. Chapter Heading 20 TNR (Bold, center-adjusted, all-caps) All of the following headings should be aligned to the left, and indented text should start on the following line. Heading at the top 16 TNR (bold, all caps) Second Level Heading 14 TNR (Bold, Main Words' First Letters in Capital Letters).

1.2 TABLES AND FIGURES

The title and table number for each table must be written on the top of the table (Table 1: ABC). Under each figure, include the figure number and the figure's title (Figure 1: Xyz).

Chapter 2

LITERATURE REVIEW

2.1 RAD-NeRF

The task of audio-driven talking portrait synthesis and its multiple uses, including digital human creation, virtual video conferencing, and filmmaking, are covered in the essay. The authors discuss the difficulties that earlier approaches had in producing accurate findings in real-time and provide a brand-new framework called Realtime Audio-spatial Decomposed NeRF (RAD-NeRF) that solves these issues.

By utilising a modern grid-based NeRF representation, RAD-NeRF can efficiently model static scenes and adapt that capability to dynamic audio-driven portrait modelling. The framework divides the three low-dimensional trainable feature grids from the naturally high-dimensional audio-guided portrait representation. The authors specifically suggest a Decomposed Audio-spatial Encoding Module, which divides audio and spatial representations into two grids, allowing dynamic head modelling. The authors demonstrate how the cost of interpolation can be further decreased by splitting the audio and spatial coordinates into two different lower-dimensional feature grids.

The writers discuss the difficulty of modelling the torso, which is less difficult but just as important for creating realistic pictures, in addition to the head. The authors suggest a lightweight Pseudo-3D Deformable Module to model the torso with a 2D feature grid in light of the discovery that topological alterations are less involved in torso movements.

The authors show that RAD-NeRF can operate 500 times faster than earlier efforts while generating improved rendering quality and real-time inference performance with a contemporary GPU. Along with implicit controls for the talking portrait's head posture, eye blink, and backdrop picture, the framework also allows a number of explicit controls.

The article also offers a thorough analysis of previous work on audio-driven talking portrait synthesis, including traditional methods that use stitching-based methods to alter mouth shapes and define a set of phoneme-mouth correspondence rules, image-based methods that create images that correspond to audio inputs, and model-based methods that make use of explicit facial structural priors like landmarks and meshes.

Overall, the difficulty of real-time audio-driven talking portrait synthesis with excellent rendering quality and explicit controls is addressed by RAD-NeRF, which offers a viable solution.

2.2 Add Sympathy To ChatBot

The article discusses five ways to add empathy to a digital service, which can improve the user experience and make customers feel more understood and valued. The five ways are:

Use inclusive language: Use language that is welcoming and inclusive of all users, regardless of their background or identity.

Offer personalized recommendations: Use data and user behavior to offer personalized recommendations that cater to the individual needs and preferences of users.

Use human-like interactions: Use chatbots and other digital tools to simulate human-like interactions and make users feel like they are talking to a real person.

Provide emotional support: Offer emotional support to users who are going through a difficult time or experiencing a problem. This can be done through chatbots or other tools that offer resources and advice.

Solicit feedback: Ask users for feedback and listen to their suggestions and concerns. Use this feedback to improve the service and show users that their opinions are valued.

Overall, incorporating empathy into digital services can help build stronger relationships with users and improve their overall experience.

2.3 Teach Your Chatbot With Training Data

The article describes using training data to teach a chatbot. The author suggests that training data is essential for chatbot development because it helps the chatbot understand the user's intent and respond appropriately. The article outlines several steps for teaching a chatbot using training data:

Identify the chatbot's purpose: Determine the chatbot's purpose and the types of questions and tasks it will handle.

Gather training data: Collect a dataset of questions and responses that the chatbot will encounter.

Clean and preprocess the data: Clean the training data by removing any irrelevant or redundant information and preprocess the data by converting it into a usable format.

Train the chatbot: Use machine learning algorithms to train the chatbot using the cleaned and preprocessed training data.

Test and refine the chatbot: Test the chatbot with new data and refine its responses based on user feedback.

The author emphasizes the importance of ongoing refinement and improvement, as chatbots need to be continuously trained and updated to improve their accuracy and effectiveness.

2.4 Blender Vs Rasa

The article compares two open-source chatbot development platforms, Blender and Rasa. The author provides an overview of each platform's features and capabilities and compares them in several key areas, including natural language processing, machine learning, and conversation flow management.

Blender is a chatbot development platform that uses a neural network-based approach for natural language processing and generation. It is designed to handle complex conversations and is particularly well-suited for chatbots that require a high degree of personalization. Blender also has a robust conversation management system and a visual interface for building conversation flows.

Rasa, on the other hand, is a chatbot development platform that is focused on natural language understanding and machine learning. It uses machine learning algorithms to understand and interpret user input and generate appropriate responses. Rasa also has a flexible conversation flow management system that allows developers to create complex chatbot interactions.

The article concludes that both Blender and Rasa are powerful and capable chatbot development platforms, but they have different strengths and weaknesses. Developers should carefully evaluate their needs and goals when choosing between the two platforms. Blender is better suited for complex, personalized chatbots, while Rasa is ideal for chatbots that require a high degree of natural language understanding and machine learning capabilities.

2.5 Recreating Myself from WhatsApp Chats

The article describes how the author used their WhatsApp chat history to create a chatbot that emulates their conversational style. The author used Python and several libraries, including spaCy and NLTK, to preprocess the chat data, extract features, and train a machine learning model to generate responses.

The author outlines the steps they took to create the chatbot, including cleaning and preprocessing the data, tokenizing and lemmatizing the text, and using spaCy to extract named entities and other features. They also used a combination of rule-based and machine learning approaches to generate responses that were similar in tone and style to their own.

The article includes several examples of the chatbot in action, demonstrating its ability to respond to a wide range of conversational topics and provide personalized and engaging responses. The author concludes by highlighting the potential applications of this approach, including creating personalized chatbots for customer service or other applications.

2.6 ParlAI

ParlAI is an open-source platform developed by Facebook AI Research (FAIR) for training and testing AI models in a variety of conversational tasks, including dialogue generation, question-answering, and language modeling. It provides a unified framework for building and evaluating chatbots and other conversational agents using a variety of models and training data.

The platform includes a wide range of pre-built datasets and models, as well as tools for creating custom datasets and models. It also supports a variety of popular deep learning frameworks, including PyTorch and TensorFlow.

ParlAI provides a flexible and extensible platform for developing and testing AI models in a wide range of conversational tasks. It is actively maintained by FAIR and has a large community of contributors who contribute new datasets, models, and tools to the platform.

Chapter 3

PROBLEM DEFINITION

3.1 REAL-TIME CONVERSATION

Real-time conversation refers to the ability to communicate with someone in real-time, such as through a phone call, video chat, or messaging application. The problem that arises in real-time conversation is the need to process and understand the spoken language of the participants in real-time. This requires the use of natural language processing (NLP) techniques, such as speech recognition and language translation, to accurately capture and interpret the meaning of the spoken words.

3.2 FACIAL SYNTHESIS

Facial synthesis refers to the process of generating realistic facial expressions, movements, and features in a digital format, such as through a computer-generated image or video. The problem with facial synthesis is the need to accurately capture the nuances of human facial expressions and movements, which are complex and multifaceted. This requires the use of advanced computer vision techniques, such as facial landmark detection and deep learning algorithms, to accurately synthesize realistic facial expressions and movements in real-time.

Chapter 4

METHODOLOGY

4.1 RAD-NeRF

Figure 1 of the Real-time Audio-spatial Decomposed NeRF (RAD-NeRF) framework is presented in this section. We begin by going over NeRF's foundations and the issue formulation for audio-driven portrait synthesis (Section 3.1). Then, for modelling the head and torso, we build the Decomposed Audiospatial Encoding Module (Section 3.2) and the Pseudo-3D Deformable Module (Section 3.3). Finally, in Section 3.4, we go through the key training components that set this approach apart from earlier ones. 3.1. Neural Radiance Fields at the Initial Stage. With a 5D plenoptic function F: x, d,, c, where x = (x, y, z) is the 3D coordinate, d = (,) is the viewing direction, is the volume density, and c = (r, g, b) is the emitted light, NeRF [36] is proposed to represent a static 3D volumetric picture. We ask F for information at positions xi = o + tid progressively sampled along a ray r with direction d that originates from o in order to calculate densities i and colours ci. By using numerical quadrature, the colour of the pixel corresponding to the ray is determined: Ti is the transmittance, i = 1 exp(ii) is the opacity, and i = ti+1 ti is the step size, with C(r) = X i Tiici; Ti = Y ji (1 j), (1). NeRF can learn 3D scenes with supervision from just 2D photos because to our fully differentiable volume rendering technique. NeRF in motion. An additional requirement (i.e., the present time t) is necessary for the innovative perspective synthesis of dynamic situations. Dynamic scene modelling is typically carried out using two strategies in previous methods: 1) Deformation-based methods [38, 44] learn a deformation Δx at each position and time step: $G : x, t \rightarrow \Delta x$, which is subsequently added to the original position x. 2) Modulation-based methods [19, 22] directly condition the plenoptic function on time: $F : x, d, t \rightarrow \sigma, c.$

We choose the modulation-based strategy to model the head part and the deformation-based strategy to model the torso part with simpler motion patterns because the deformation-based methods are not good at modelling topological changes (e.g., mouth openings and closings) because of the intrinsic continuity of the deformation field [39]. Neural Radiance Talking Portrait with audio input. We provide a brief overview of the standard audio-driven neural talking portrait generation pipeline [22, 32, 46]. The training data is often a static camera recording of a 3-5 minute scene-specific video with synchronized audio track. Each image frame goes through three basic preprocessing steps: (3) Face-tracking [54] to estimate the head pose parameters. (1) Semantic parsing [26] of the head, neck, torso, and backdrop part. (2) Extraction of 2D facial landmarks [8], including the eyes and lips. Please take note that just the training method requires these steps. An Automatic Speech Recognition (ASR) model is used to extract audio features from the audio track during audio processing [1, 2]. A NeRF can be used to learn to synthesise the head part based on the head positions and audio circumstances. The torso section requires additional modelling, such as by another full NeRF [22], because it is not in the same coordinate system as the head partNeRF based on a grid. Recent grid-based NeRF [10,37,49] encode 3D spatial data of static scenes using a 3D feature grid encoder, where f is the encoded spatial features and x R 3 is the spatial coordinate. Such feature grid encoders considerably increase the efficiency of both training and inference by substituting less expensive linear interpolation for MLP forwarding when requesting spatial information. Real-time rendering speed for static 3D scenes can now be achieved [37]. We expand on this idea to encapsulate the high-dimensional audio-spatial data needed for dynamic talking portrait creation.

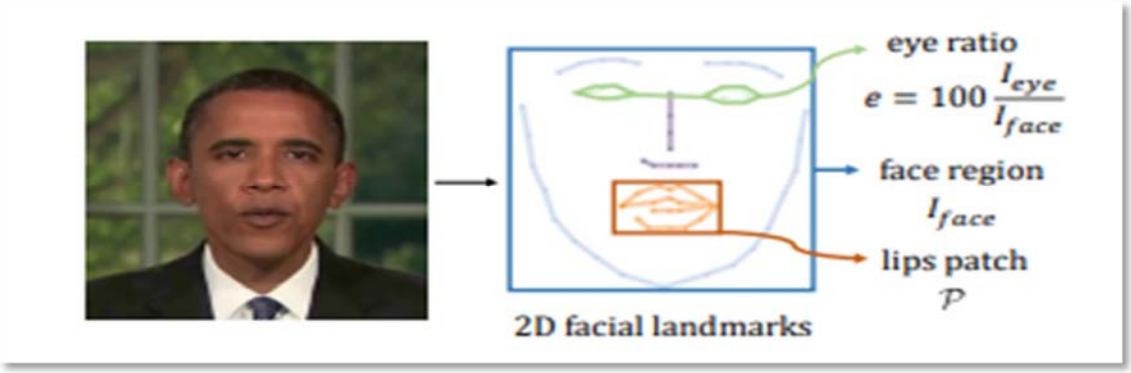
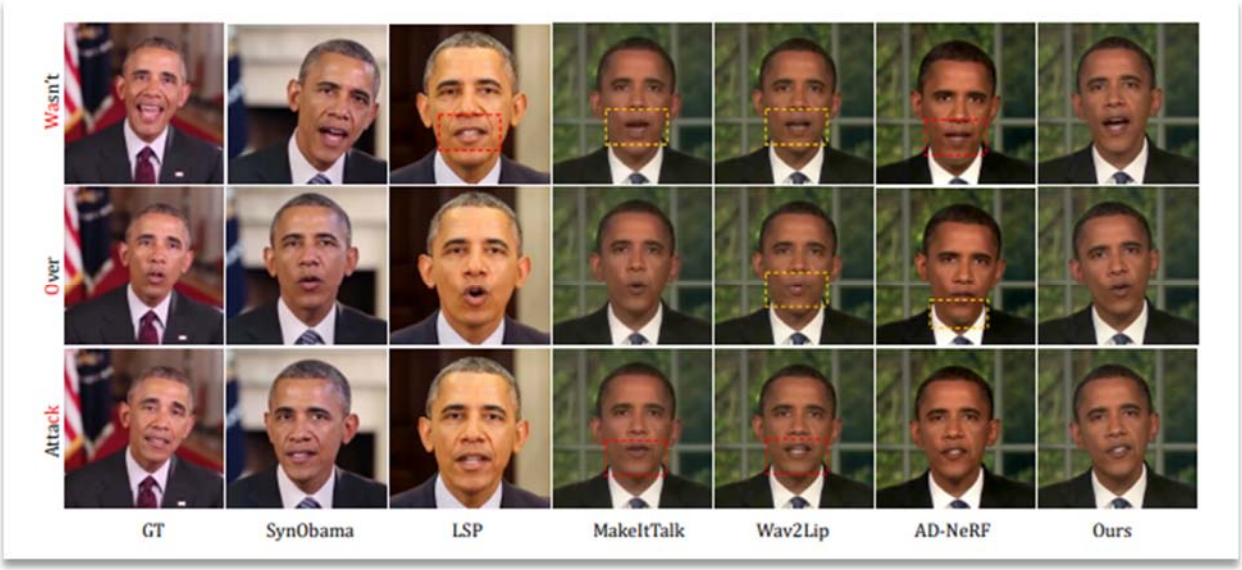


Figure 1 An illustration of landmark details. The face area Iface for dynamic regularization, the eye ratio e for eye control, and the lips patch P for lips fine-tuning are three properties that we extract from the projected 2D facial landmarks to help with training.

4.2 IMPLEMENTATION DETAILS

On the datasets compiled by prior works [22, 46], we conduct experiments. We train the head part for 20,000 steps for each person-specific dataset then fine-tune the lips for 5,000 steps, with each step including 2562 rays. Per ray, the maximum occupancy grid prunes the sampling to a maximum of 16 points. Lentropy loss scale is set to 0.001, Ldynamic loss scale to 0.1, and LPIPS loss scale to 0.01. It uses the Adam optimizer [25] with a 0.0005 starting learning rate for the network and a 0.005 initial learning rate for the feature grid. Exponentially decaying to 0.1 of the beginning values, the final learning rates. To make training more stable, we also employ an exponential moving average (EMA) of 0.95. Following convergence of the head part, the torso part is trained for 20,000 steps using the same learning rate technique. The head and torso exercises last around five and two hours, respectively. One NVIDIA V100 GPU serves as the platform for all the studies. For more information, please see the supplemental materials



.**Figure 2: cross-driven evaluation of quality. We display cross-driven visualizations of representative methodologies. Red boxes indicate incorrect lips, while yellow boxes indicate poor image quality. Our techniques provide images with high image clarity and precise lips movement. For further information, we suggest watching the supplemental video.**

Table 1. Comparative analysis in a self-driven environment.We perform self-driven synthesis on the same identity's testset and compare the face reconstruction quality. MakeItTalk [67] cannot generate the same head poses as the ground truth video, so the PSNR and LPIPS are not reported. The training time is only reported for person-specific methods.

Methods	PSNR↑	LPIPS↓	LMD↓	Sync↑	AUE↓	Training Time↓	Inference FPS↑
Ground Truth	∞	0	0	8.448	0	-	-
MakeItTalk [67]	-	-	4.484	5.440	1.327	-	12
Wav2Lip [43]	30.94	0.0639	3.318	**7.756**	1.120	-	15
AD-NeRF [22]	25.53	0.0901	2.858	5.428	1.043	36h	0.08
Ours	**34.00**	**0.0387**	**2.696**	6.664	**0.882**	**7h**	**40**

Chapter 5

DETAILED DESIGN AND ARCHITECTURE

5.1 SYSTEM ARCHITECTURAL

5.1.1 Architectural Design

The modular structure of the smart counselling bot system is as follows:

Frontend:

Chat window: Displays the video-call conversation between the user and the chatbot and allows the user to input text and send it to the chatbot for processing.

Facial display**:** Displays emotions and other visual content in the window.

Backend:

Chatbot engine**:** Processes user input and generates responses. Uses natural language processing (NLP) techniques to understand user input and accesses the conversation history database to provide personalized responses.

Conversation history database:

Stores information about past interactions with users, including the conversation history and any personalization data.

External dependencies:

The chatbot may need to integrate with external APIs or services to access additional data or functionality.

Here is a diagram showing the major subsystems and their interconnections of Smart Counselling Bot:

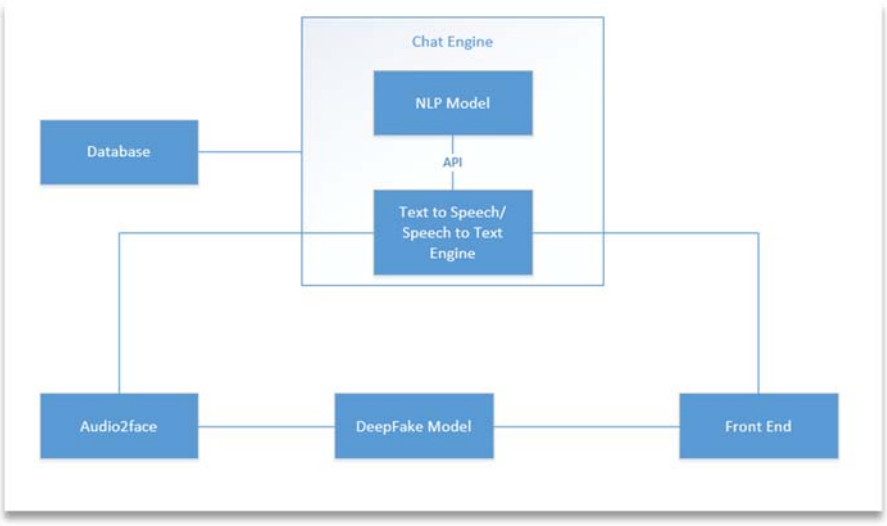


Figure 3: Subsystems Diagram

To achieve the complete functionality of the system, the frontend and backend subsystems work together as follows:

The user inputs audio into the microphone on the frontend.

The audio is sent to the chatbot engine on the backend for processing.

The chatbot engine uses NLP techniques to understand the user's input and accesses the conversation history database to retrieve any relevant personalization data.

The chatbot engine generates a response and sends it back to the Audio2face interface.

The response is displayed in the call window, along with any appropriate emotions.

5.2 DETAILED SYSTEM DESIGN

5.2.1 Decomposition Description

Here is a top-level data flow diagram (FLOW CHART) for the smart counselling bot system:

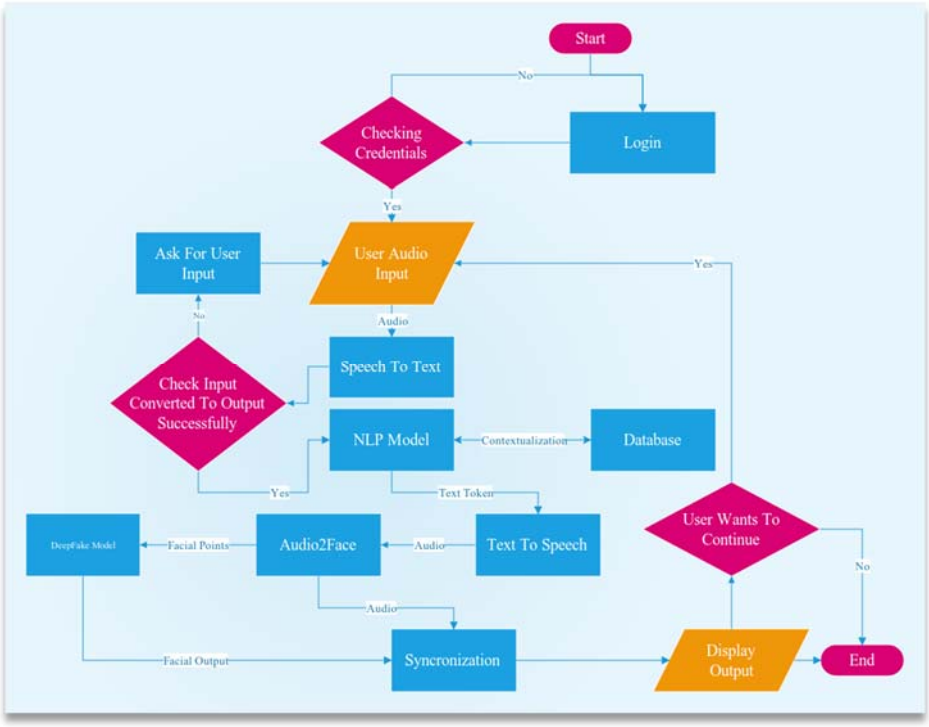


Figure 4: Flow chart Diagram

This FLOW CHART shows the flow of data between the frontend, backend, and conversation history database of the smart counselling bot system.

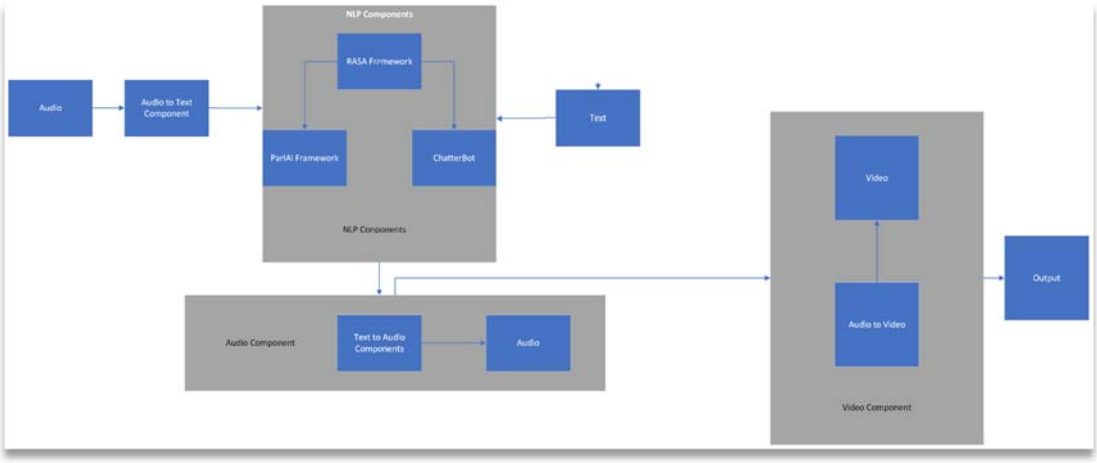
These interface specifications define the methods and properties of the objects in the smart counselling bot system. The call window have methods for displaying virtual face and visual content, while the chatbot engine object has methods for processing user input and generating responses. The conversation history database object has methods

for storing and retrieving information about past interactions with users.

5.2.2 Object-Oriented Description:

Here is a subsystem model for the smart counselling bot system:

Figure 5: Object Oriented Diagram



This subsystem model illustrates the main components of the smart counselling bot system and their relationships. The frontend subsystem consists of the chat window and image display modules, which handle the display of the text-based conversation and any visual content. The backend subsystem consists of the chatbot engine module, which processes user input and generates responses. The conversation history database stores information about past interactions with users.

Here are some **object diagrams** for the smart counselling bot:

These object diagrams show the relationships between the objects in the smart counselling bot system. The objects in the frontend subsystem include the call window, microphone and mute button, which are used to display the call-based conversation. The objects in the backend subsystem include the chatbot engine, which processes user input and generates responses. The conversation history database stores information about past interactions with users.

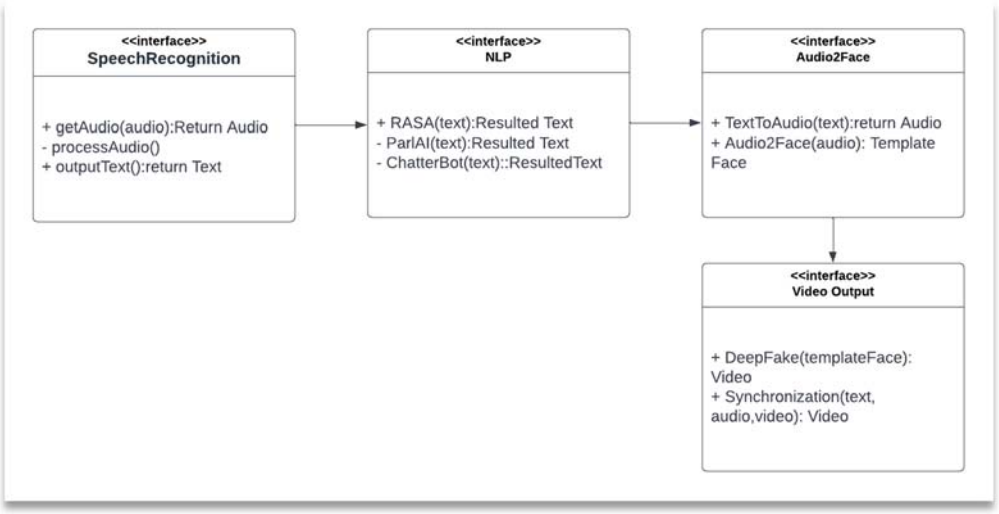
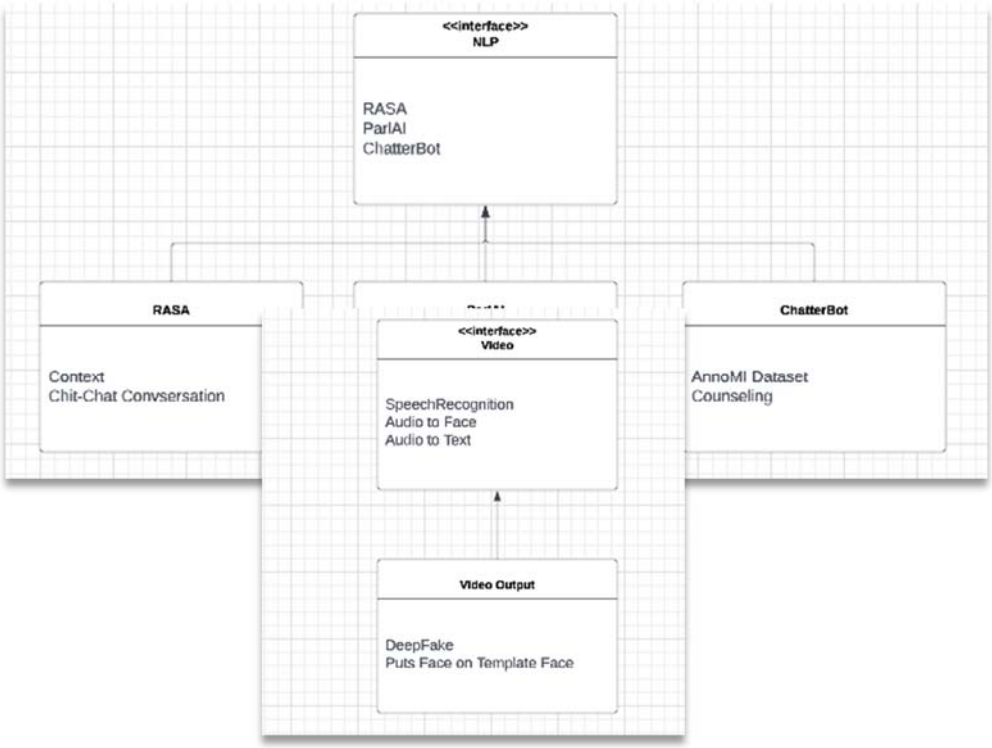


Figure 6: Object Diagram

Here is a generalization hierarchy diagram for the smart counselling bot system:



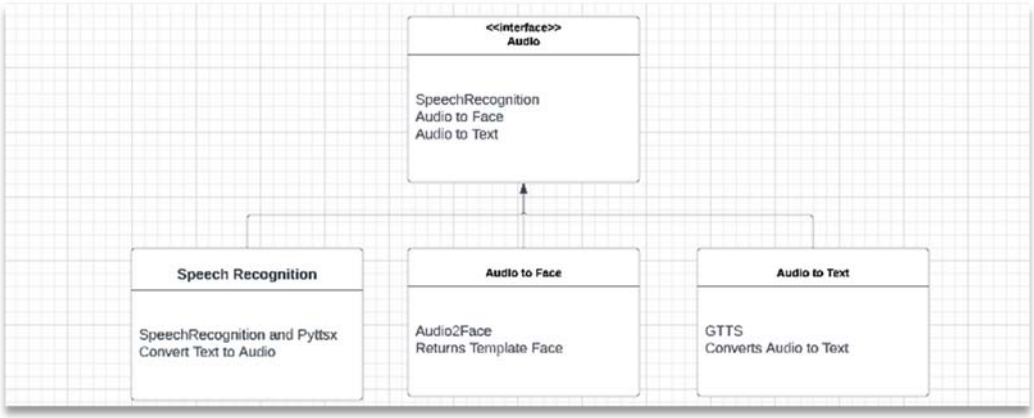


Figure 7: Generalization Hierarchy Diagram

5.2.3 Resources

An explanation of all resources that this entity manages, influences, or requires.

Resources include things like memory, CPUs, printers, databases, or a software library that are not part of the design. This should cover any potential race conditions and/or stalemate circumstances, as well as potential solutions.

5.2.4 Processing

We might think of our approach as a deformation-based dynamic NeRF in two dimensions. We just need to sample one pixel coordinate xt R 2 from the image space, as opposed to sampling several points along each camera ray. The torso motion and head motion are synchronised thanks to the deformation, which is dependent on the head posture p. To forecast the deformation, we use an MLP: x = MLP(xt, p). To obtain the torso feature, the distorted coordinate is given to a 2D feature grid encoder as follows: ft = E2 torso(xt + x). For the torso RGB colour and alpha values, a different MLP is used: ct, t = MLP(ft, it) (3), where it is a latent appearance embedding to add extra model capacity. We demonstrate how this deformation-based module can effectively represent torso motions and create realistic-looking torso images that match the head. More crucially, the 2D feature grid-based pseudo-3D representation is incredibly compact and effective. To create the final portrait image, the individually rendered head and torso images can be alpha composited with any background image that is made available.

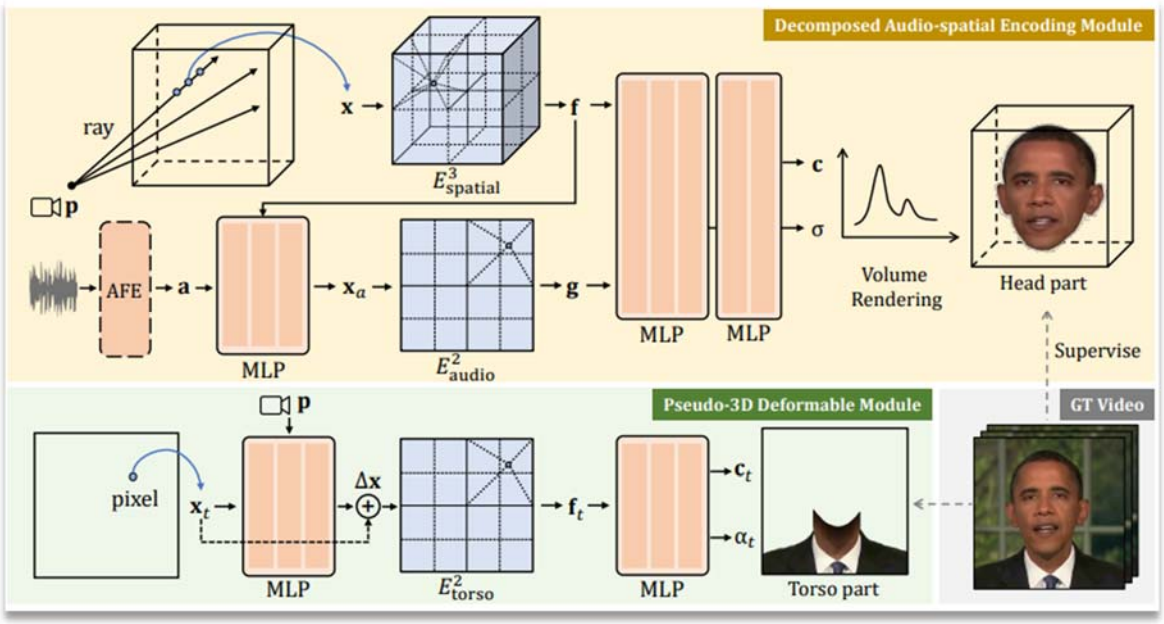


Figure 8: 2D-Version

5.3 CLASS DIAGRAM

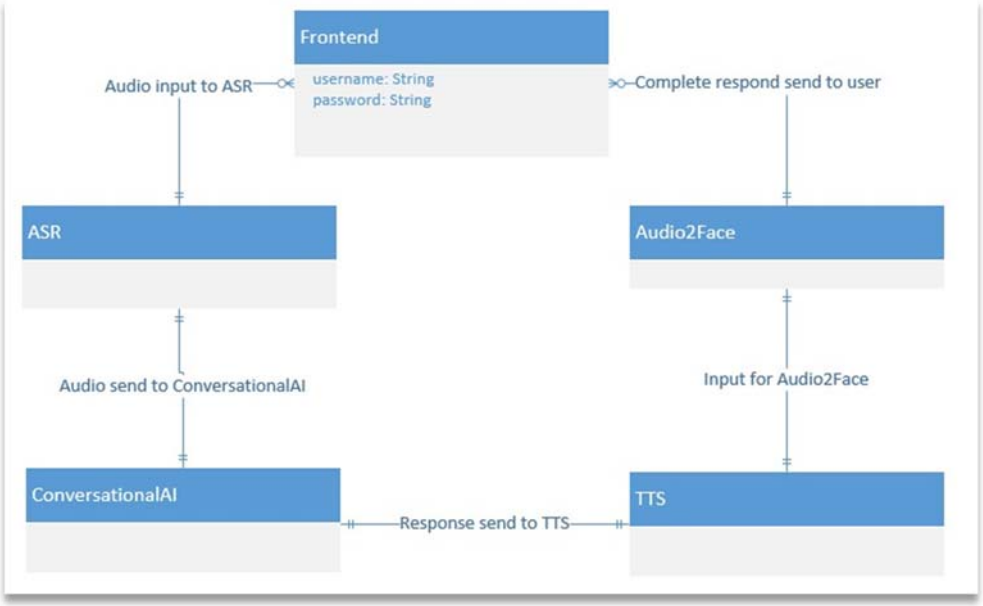


Figure 9: Class Diagram

5.4 ER DIAGRAM

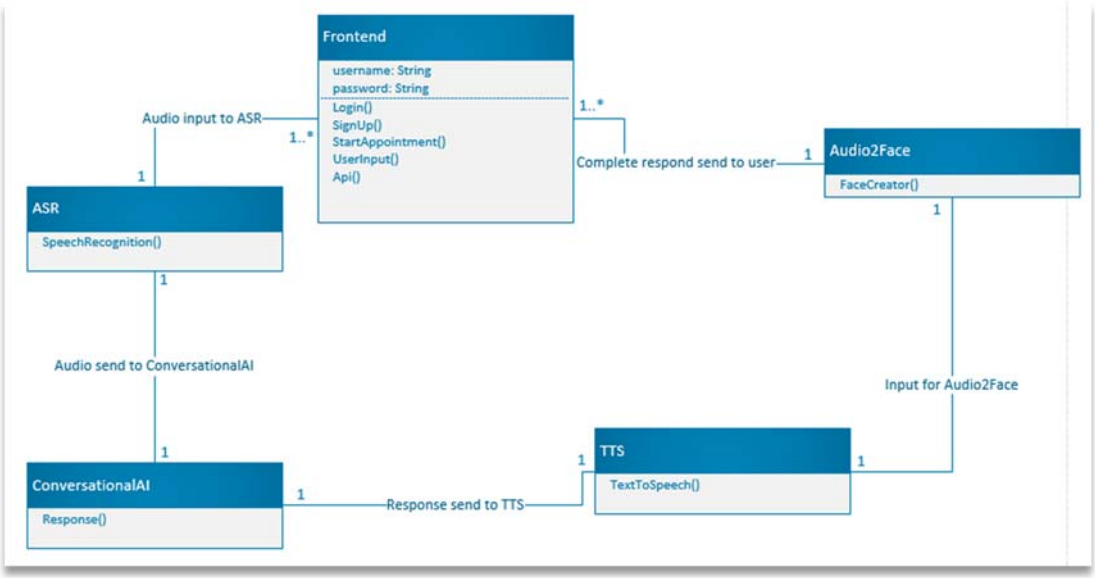


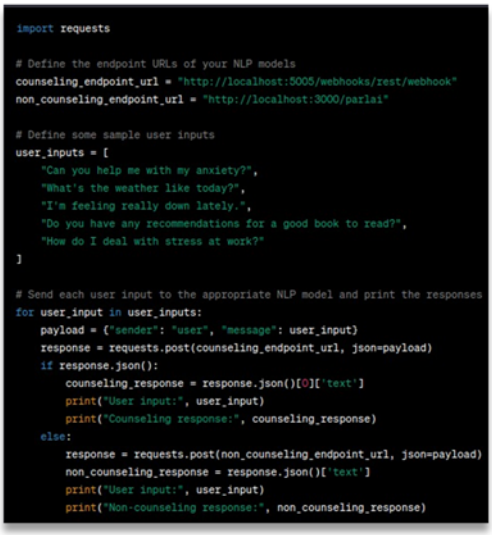
Figure 10 : ER

Chapter 6

IMPLEMENTATION AND TESTING

The project is simply implemented on a pipeline model taking the input in from one side, i.e. the website interface and then converting it to output video feed after being processed at multiple steps.

First the input in form of audio is taken by the microphone and is transferred to RIVA TTS to be converted to textual tokens, those tokens are taken in by the NLP model to derive a contextual response for the statement, given as input by the user. The response again passes through the RIVA TTS to be converted to synthetic speech and the synthetic audio is the input for the RAD-NeRF model that creates a synthetic face for the audio. The audio and video created by the RAD-NeRF are displayed to the user.Currently the models is running on a local machine supported by an 50 Gbs SSD, Intel i7 11th Gen processor, and RTX 3060 GPU. For the prototype the following specs are enough to support 1 concurrent demo. To increase service side a Docker image could be made and uploaded to the cloud with hardware equivalent to the expected usage on the system.



```
import requests

# Define the endpoint URLs of your NLP models
counseling_endpoint_url = "http://localhost:5005/webhooks/rest/webhook"
non_counseling_endpoint_url = "http://localhost:3000/parlai"

# Define some sample user inputs
user_inputs = [
    "Can you help me with my anxiety?",
    "What's the weather like today?",
    "I'm feeling really down lately.",
    "Do you have any recommendations for a good book to read?",
    "How do I deal with stress at work?"
]

# Send each user input to the appropriate NLP model and print the responses
for user_input in user_inputs:
    payload = {"sender": "user", "message": user_input}
    response = requests.post(counseling_endpoint_url, json=payload)
    if response.json():
        counseling_response = response.json()[0]['text']
        print("User input:", user_input)
        print("Counseling response:", counseling_response)
    else:
        response = requests.post(non_counseling_endpoint_url, json=payload)
        non_counseling_response = response.json()['text']
        print("User input:", user_input)
        print("Non-counseling response:", non_counseling_response)
```

Figure 11: Testing

Chapter 7

RESULTS AND DISCUSSION

Results of the self-driven evaluation are displayed in Table. We contrast our results with more modern techniques that can provide full-resolution video as the actual data. With a real-time inference FPS, our method produces better quality results in the majority of measures. In particular, our technique converges roughly 5 times faster and infers data about 500 times faster than the industry standard AD-NeRF. The table below lists the cross-driven outcomes. We attain performance that is comparable to that of recent representative approaches, proving the correctness of our synthetic lips.

Table 2: Comparison of methods generation full-resolution video

	Testset A [50]		Testset B [53]	
Methods	Sync↑	AUE↓	Sync↑	AUE↓
Ground Truth	7.999	0	6.913	0
MakeItTalk [67]	5.287	1.722	5.144	1.359
Wav2Lip [43]	9.257	1.904	8.565	1.493
SynObama [50]	7.297	1.949	–	–
NVP [53]	–	–	3.476	1.110
LSP [33]	5.623	1.895	5.732	1.363
AD-NeRF [22]	4.455	2.133	4.614	1.801
Ours	6.218	1.854	5.500	1.584

Chapter 8

CONCLUSION AND FUTURE WORK

So far the project serves as a proof of concept for the conversational AI and as a starting point for further development in the field as it is a prototype for a concept, it is nowhere good enough to be used as a finished product. There are a few redundancies in the system like, it each modules takes in a specific kind on input, audio/text. Which calls for implementing data conversion at every other module, that casts extra processing power and extra time being consumed.

The future work would remove the extra steps and making the model more flexible on the input to save time and reduce the pipeline length. Other then that making an automated system for anyone to easily add faces of their choice without being involved in the technical details of the system.

Commercialize the product as an end product is also an essential for-sight, offering higher quality services to the people that are willing to pay for that.

Chapter 9

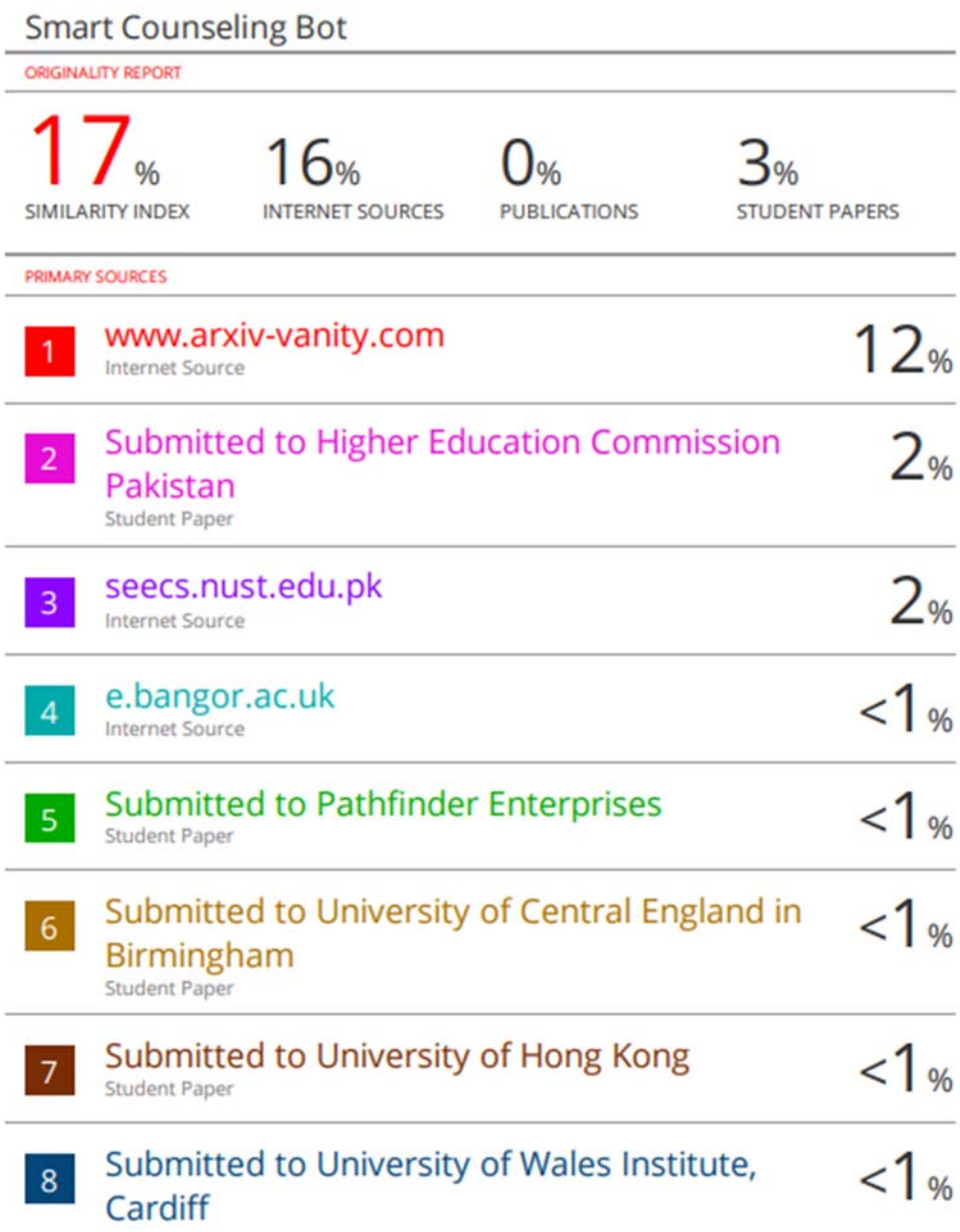
REFERENCES

- Adrian Bulat and Georgios Tzimiropoulos. How far are we from solving the 2d & 3d face alignment problem? (and a dataset of 230,000 3d facial landmarks). In ICCV, 2017. 4
- Alexei Baevski, Yuhao Zhou, Abdelrahman Mohamed, and Michael Auli. wav2vec 2.0: A framework for self-supervised learning of speech representations. NeurIPS, 33:12449– 12460, 2020. 4, 8, 11
- Anpei Chen, Zexiang Xu, Andreas Geiger, Jingyi Yu, and Hao Su. Tensorf: Tensorial radiance fields. arXiv preprint arXiv:2203.09517, 2022. 1, 2, 4, 5
- chatbotslife.com/5-ways-to-add-empathy-to-a-digital-service-952e46cc3c6c
- Christoph Bregler, Michele Covell, and Malcolm Slaney. Video rewrite: Driving visual speech with audio. In Proceedings of the 24th annual conference on Computer graphics and interactive techniques, pages 353–360, 1997. 2
- Dario Amodei, Sundaram Ananthanarayanan, Rishita Anubhai, Jingliang Bai, Eric Battenberg, Carl Case, Jared Casper, Bryan Catanzaro, Qiang Cheng, Guoliang Chen, et al. Deep speech 2: End-to-end speech recognition in english and mandarin. In ICML, pages 173–182. PMLR, 2016. 4, 8, 11
- Eric R Chan, Connor Z Lin, Matthew A Chan, Koki Nagano, Boxiao Pan, Shalini De Mello, Orazio Gallo, Leonidas Guibas, Jonathan Tremblay, Sameh Khamis, et al. Efficient geometry-aware 3d generative adversarial networks. arXiv preprint arXiv:2112.07945, 2021. 1
- https://chatbotslife.com/blender-vs-rasa-open-source-chatbots-efae383b9d33
- https://chatbotslife.com/how-to-teach-your-chatbot-with-training-data-46c58b873c31
- https://github.com/facebookresearch/ParlAI
- https://towardsdatascience.com/recreating-myself-from-whatsapp-chats-

6dadfaff0d2b

- Jonathan T Barron, Ben Mildenhall, Dor Verbin, Pratul P Srinivasan, and Peter Hedman. Mip-nerf 360: Unbounded anti-aliased neural radiance fields. arXiv preprint arXiv:2111.12077, 2021. 2
- Jonathan T. Barron, Ben Mildenhall, Matthew Tancik, Peter Hedman, Ricardo Martin-Brualla, and Pratul P. Srinivasan. Mip-nerf: A multiscale representation for anti-aliasing neural radiance fields, 2021. 2
- Matthew Brand. Voice puppetry. In Proceedings of the 26th annual conference on Computer graphics and interactive techniques, pages 21–28, 1999. 2
- Tadas Baltrusaitis, Amir Zadeh, Yao Chong Lim, and LouisPhilippe Morency. Openface 2.0: Facial behavior analysis toolkit. In FG, pages 59–66. IEEE, 2018. 6

TURNITIN REPORT:



Smart Counseling Bot

ORIGINALITY REPORT

17%	16%	0%	3%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	www.arxiv-vanity.com Internet Source	12%
2	Submitted to Higher Education Commission Pakistan Student Paper	2%
3	seecs.nust.edu.pk Internet Source	2%
4	e.bangor.ac.uk Internet Source	<1%
5	Submitted to Pathfinder Enterprises Student Paper	<1%
6	Submitted to University of Central England in Birmingham Student Paper	<1%
7	Submitted to University of Hong Kong Student Paper	<1%
8	Submitted to University of Wales Institute, Cardiff	<1%

9	researchspace.ukzn.ac.za Internet Source	<1%
10	Submitted to University of Portsmouth Student Paper	<1%
11	botpress.com Internet Source	<1%
12	arxiv-export-lb.library.cornell.edu Internet Source	<1%
13	i.blackhat.com Internet Source	<1%
14	scholar.sun.ac.za Internet Source	<1%

Exclude quotes	On	Exclude matches	Off
Exclude bibliography	On		