

PROJECT REPORT OF 6 WEEKS SUMMER TRAINING

“VOICE-LINGUA”

Submitted By:

Harshdeep Singh

Kirandeep Kaur

Liza Kumari

Under the guidance of

(Ms. Gulbadan Khehra)



Submitted to

CENTRE FOR DEVELOPMENT OF ADVANCED COMPUTING

A-34, PHASE VIII, INDUSTRIAL AREA, MOHALI (PUNJAB), INDIA-160071

in Partial Fulfilment of the Requirements for the Certificate of

6 weeks of summer training in

GENERATIVE AI: FUNDAMENTALS AND TECHNIQUES (10.06.2024-19.07.2024)

DECLARATION

“VOICE-LINGUA”

by Harshdeep Singh, Kirandeep Kaur, and Liza Kumari

Place of work: Centre for Development of Advanced Computing (C-DAC),
Mohali

Submitted to the Centre for Development of Advanced Computing (C-DAC),
Mohali.

July 2024

In Partial Fulfilment of the Requirements for the certificate of 6 weeks of summer training.

Certified by

Dr. Sanjay Madan (Industrial Mentor),

Ms Sonia Dosanjh

INDEX

S.NO	CONTENT	PAGE NO.
Chapter 1	1. Introduction 1.1 Introduction to Organization 1.2 Introduction to Project 1.3 Scope of the Project	7
Chapter 2	2. Technologies and Tools 2.1 The Technologies Used 2.2 The Tools Used	12
Chapter 3	3. Project Details 3.1 Proposed System	15
Chapter 4	4. Limitations	27
Chapter 5	5. Conclusion and Future Works	28
Chapter 6	6. References	30

ABSTRACT

This report presents an overview of our summer training experience at C-DAC, Mohali, where we worked on various Natural Language Processing (NLP) tasks within the domain of Artificial Intelligence, leading to the creation of "*VOICE LINGUA*." It is an advanced AI-powered NLP interface developed using Python and cutting-edge AI technologies, designed to provide a comprehensive set of NLP tasks with high accuracy, scalability, and versatility. The interface offers features such as speech recognition for transcribing audio inputs from any language, translation of the text into 50 languages utilizing the NLLB-200 model, speech generation converting text into speech in the supported languages, video-to-audio generation for extracting audio from video files, and text summarization with customizable lengths from 100 to 250 words. *VOICE LINGUA* is an invaluable tool for various fields, including education, business, content creation, news, and multimedia production, significantly benefiting educators and students by facilitating the transcription, translation, and summarization of study materials.

ACKNOWLEDGEMENT

“Success is a sweet fruit to which everyone strives to taste.”

Any job in this world, however trivial or tough cannot be accomplished without the assistance of others. We would hereby take the opportunity to express our indebtedness to the people who have helped us to accomplish this task.

We would like to take this opportunity to extend our heartfelt gratitude to all those who have been instrumental in making our internship experience at C-DAC, Mohali, a rewarding journey.

We are profoundly grateful to Ms. Gulbadan Khehra our industrial mentor at C-DAC, whose guidance, support, and expertise have been invaluable throughout our training. Her mentorship not only broadened our understanding but also inspired us to strive for excellence in the field of Applied Artificial Intelligence and Analytics.

We would also like to express our gratitude to the entire team at C-DAC, Mohali, for providing us with a conducive learning environment and entrusting us with meaningful projects and responsibilities.

Furthermore, we are grateful to our institution, University Institute of Engineering and Technology, Panjab University, Chandigarh, for providing the necessary resources and support to undertake this training.

Thanks

LIST OF ABBREVIATIONS

S. No.	Abbreviations	Full Form
1.	NLP	Natural Language Processing
2.	PDF	Portable Document Format
3.	C-DAC	Centre for Development of Advanced Computing
4.	R&D	Research and Development
5.	IT	Information Technology
6.	VS	Visual Studio
7.	LLM	Large Language Model

CHAPTER 1

INTRODUCTION TO ORGANIZATION

1.1 About C-DAC

C-DAC Mohali is a premier R&D organization under the Ministry of Electronics and Information Technology, funded by the Government of India. It focuses on advanced computing technologies to drive innovation across various domains, including IT, cybersecurity, e-governance, and healthcare. The organization's mission is to develop cutting-edge solutions for complex challenges and collaborate with industry partners, academia, and government agencies. Its impactful research in AI, cybersecurity, and health informatics often leads to commercial products that contribute to economic growth. It also collaborates nationally and internationally to enhance knowledge exchange and address global technological challenges.

1.2 Products and Services

C-DAC Mohali offers a diverse range of products and services designed to cater to the needs of government, industry, and academia. These offerings include:

1) Software Development:

- **E-Governance Solutions:** Develop applications that streamline administrative processes, increase transparency, and improve citizen services, such as digital portals and online service delivery platforms.
- **Cybersecurity Solutions:** Provides tools and services to protect critical infrastructure and sensitive data from cyber threats, including intrusion detection systems and secure communication platforms.

2) Consultancy Services:

- **IT Consultancy:** Offers services to design, implement, and optimize IT infrastructure, including system integration, network design, and performance tuning.

- **Research and Development:** Engages in R&D projects with academic institutions, industry partners, and government agencies to develop innovative solutions in areas like AI, machine learning, and data analytics.

3) Training and Education:

- **Professional Training Programs:** Conducts training programs and workshops to enhance the skills of IT professionals and students in software development, cybersecurity, data science, and more.
- **Academic Collaborations:** Partners with universities and colleges to offer specialized courses and certification programs in advanced computing technologies.

4) Research and Innovation:

- **High-Performance Computing (HPC):** Develops and deploys HPC systems to solve complex computational problems in scientific research and engineering.
- **Multilingual Computing:** Develops language processing tools and technologies to facilitate communication and information access in multiple languages.

5) Health Informatics:

- **Telemedicine Solutions:** Develop platforms to enable remote healthcare services and consultations, improving access to medical services in remote areas.
- **Health Management Systems:** Creates health management systems that integrate hospital management, patient care, and medical data analysis.

C-DAC Mohali's commitment to innovation, quality, and excellence has established it as a leader in advanced computing and information technology by continuously pushing the boundaries of technology.



FIG 1: The main entrance of C-DAC, Mohali

INTRODUCTION TO PROJECT

1.2 Project Overview

The project, *VOICE LINGUA*, is an innovative interface developed using Python and advanced AI technologies to provide a comprehensive suite of Natural Language Processing (NLP) tasks. It is designed to enhance the user experience with high accuracy and versatility in language processing.

The project encompasses five NLP tasks: Speech Recognition, Translation, Speech Generation, Audio Extraction, and Summarization.

1.2.1 SPEECH RECOGNITION:

The project aims to create a robust speech-to-text system capable of accurately transcribing spoken language into written text across multiple languages, 50+ languages supported by the model. By leveraging advanced transformer models, such as the ‘Whisper-large-v3’ model by OPEN AI, and audio pre-processing techniques the system seeks to achieve high levels of transcription accuracy, addressing challenges such as varying accents, dialects, and environmental noise. The goal is to create a system that transcribes pre-recorded audio with high accuracy. Overall, the project endeavors to advance the field of speech recognition by developing innovative solutions that improve accessibility, usability, and reliability in converting spoken language into text across diverse linguistic and environmental conditions.

1.2.2 TRANSLATION:

Powered by the latest ‘Facebook NLLB-200’ model, the speech translation capability allows you to translate effectively between source and target languages. 50+ languages are supported by this model, offering a wide range of linguistic coverage. This advanced system can be useful if you need to make an exact translation while maintaining the overall meaning as well as context. If it is needed for educational purposes for media productions or business meetings, the necessary facility for communication across language barriers is available in developed software. As a result, this software becomes a helpful element to global interactions. Besides, quick interpretation is supported and this improves how people from different parts of the world might engage in national collaborations.

1.2.2 SPEECH GENERATION:

The speech generation feature, leveraging the ‘NLLB-200’ model and the ‘GTTS’ library, converts text into natural-sounding speech with remarkable accuracy and clarity. Supporting 50+ languages, it ensures broad linguistic accessibility and inclusivity. Users can input text in any supported language and receive high-quality speech output that retains the original context. This feature is handy for creating multilingual voiceovers and enhancing language learning applications. Additionally, the speech synthesis provides an authentic auditory experience. Ideal for virtual assistants, accessibility tools, and multimedia production, the speech generation capability significantly enhances user engagement and communication effectiveness across diverse linguistic audiences.

1.2.3 AUDIO EXTRACTION:

The audio extraction feature, leveraging the ‘moviepy’ library, efficiently converts video files into high-quality audio, preserving the original sound. This tool is indispensable for multimedia production, allowing users to reuse video content into podcasts, audio clips, and other audio formats. It supports various video file types, ensuring compatibility and ease of use. The extracted audio maintains synchronization with the original video, making it ideal for further editing and production tasks. Whether for journalism, content creation, or educational purposes, the audio extraction feature streamlines the process of extracting and utilizing audio content from video recordings, enhancing the accessibility of multimedia resources.

1.2.4 SUMMARIZATION:

The summarization feature provides precise and coherent summaries of lengthy texts, reducing the content into customizable lengths which is dependent on the length of the input provided. This tool can be particularly valuable for professionals in journalism, research, and business, where quick and accurate content summarization is essential. Users can input articles, reports, or any textual content and receive concise summaries that retain key information and context. The summarization algorithm, which uses the ‘sshleifer/distilbart-cnn-12-6 model’, ensures that the essence of the original text is preserved, making it easier to carry out the main points and essential details.

1.3 SCOPE OF THE PROJECT

VOICE LINGUA is designed to be a versatile and powerful tool for a wide range of Natural Language Processing (NLP) tasks.

USE CASE:

VOICE LINGUA is applicable in various fields, including:

- **Medical:** Improving healthcare communication by transcribing doctor-patient interactions, translating medical documents, summarizing patient notes, and generating audio content for patient.
- **Education:** Facilitating transcription, translation, and summarization of study materials for students and educators.
- **Business:** Enhancing productivity with accurate meeting transcriptions, multilingual communication, and content summarization.
- **Content Creation:** Assisting creators with generating voiceovers, translating content for a global audience, and extracting audio from videos for podcasts.
- **News and Multimedia:** Supporting journalists and producers in transcribing interviews, translating reports, generating multilingual news segments, and repurposing video content.

By integrating these features, *VOICE LINGUA* aims to streamline and enhance the process of handling and processing language data, making it an invaluable tool for a diverse range of professional and creative applications.

CHAPTER-2

2.1 THE TECHNOLOGIES USED

The project, *VOICE LINGUA*, is an innovative interface developed using Python and advanced AI technologies to provide a comprehensive suite of Natural Language Processing (NLP) tasks.

2.1.1 PYTHON:

Python is an interpreted, object-oriented, high-level programming language with dynamic semantics. Its high-level built-in data structures, combined with dynamic typing and dynamic binding, make it very attractive for Rapid Application Development, as well as for use as a scripting or glue language to connect existing components. Python's simple, easy-to-learn syntax emphasizes readability and therefore reduces the cost of program maintenance. Python supports modules and packages, which encourages program modularity and code reuse. The Python interpreter and the extensive standard library are available in source or binary form without charge for all major platforms and can be freely distributed.

2.1.2 DEEP LEARNING:

Deep learning is a subset of machine learning that uses multilayered neural networks, called deep neural networks, to simulate the complex decision-making power of the human brain. It has revolutionized fields such as image and speech recognition and natural language processing. The architecture of deep learning models, particularly convolutional neural networks (CNNs) and recurrent neural networks (RNNs), allows them to automatically extract and learn hierarchical representations of data. This capability enables deep learning systems to achieve high levels of accuracy, often surpassing traditional machine learning techniques, especially when large amounts of labeled data are available. Deep learning has been integral to advancements in artificial intelligence, making it possible to develop sophisticated applications.

2.1.3 GENERATIVE ARTIFICIAL INTELLIGENCE:

Generative AI, sometimes called *gen AI*, is artificial intelligence (AI) that can create original content—such as text, images, video, audio, or software code—in response to a user's prompt or request. Generative AI relies on sophisticated machine learning models called deep

learning models—algorithms that simulate the learning and decision-making processes of the human brain. These models work by identifying and encoding the patterns and relationships in huge amounts of data and then using that information to understand users' natural language requests or questions and respond with relevant new content. Generative AI offers enormous productivity benefits for individuals and organizations.

2.2 TOOLS USED FOR THE PROJECT

2.2.1 GOOGLE COLAB

Google Colab, short for Google Collaboratory, is a free, cloud-based Jupyter notebook environment that allows users to write and execute Python code through their browser which allows us to train our machine learning and deep learning models on CPUs, GPUs, and TPUs. It is a versatile and user-friendly tool. Users can start coding immediately without the need for any installation or configuration, as everything runs on Google's cloud infrastructure.



FIG 2:COLAB

2.2.2 HUGGING FACE

Hugging Face is a machine learning (ML) and data science platform and community that helps users build, deploy, and train machine learning models. It provides the infrastructure to demo, run, and deploy artificial intelligence (AI) in live applications. Users can also browse through models and data sets that other people have uploaded. Hugging Face is often called the GitHub of machine learning because it lets developers share and test their work openly. Hugging Face is known for its Transformers Python library, which simplifies the process of downloading and training ML models. The library gives developers an efficient way to include one of the ML models hosted on Hugging Face in their workflow and create ML pipelines.



FIG 3:HUGGING FACE

2.2.3 STREAMLIT

Streamlit is an open-source Python library designed for creating and sharing custom web applications for machine learning and data science projects. It simplifies the process of building interactive and visually appealing web apps. With Streamlit, users can create applications by writing straightforward Python scripts, using its flexible API to add interactive widgets like sliders, buttons, and charts. It supports real-time updates, making it easy to visualize and explore data, test models, and results.

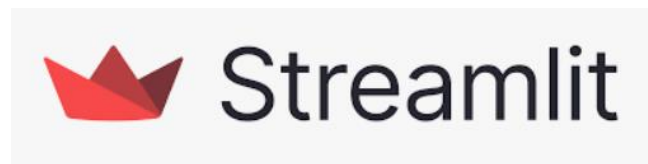


FIG 4:STREAMLIT

CHAPTER-3

PROJECT DETAILS

The proposed system, *VOICE-LINGUA*, encompasses five NLP tasks: Speech Recognition, Translation, Speech Generation, Audio Extraction, and Summarization.

We have implemented these functionalities by using Streamlit, an open-source Python library to create this web page.

3.1 PROPOSED SYSTEM:

VOICE LINGUA aims to streamline and enhance the process of handling and processing language data, making it a useful tool for a diverse range of professional and creative applications.

3.1.1 SPEECH RECOGNITION :

The project aims to create a robust speech-to-text system capable of accurately transcribing spoken language into written text across multiple languages, 50+ languages supported by the model.

MODEL USED: **openai/whisper-large-v3**

Whisper is a pre-trained model for automatic speech recognition (ASR) and speech translation. Trained on 680k hours of labelled data, Whisper models demonstrate a strong ability to generalize to many datasets and domains without the need for fine-tuning. Whisper was proposed in the paper ‘**Robust Speech Recognition via Large-Scale Weak Supervision**’ by Alec Radford et al. from OpenAI. The dataset includes a diverse range of audio from different environments, recording setups, speakers, and languages. It encompasses 117,000 hours covering 96 languages and 125,000 hours of translation data. Whisper is a transformer-based encoder-decoder model, also referred to as a sequence-to-sequence model. The multilingual models were trained on both speech recognition and speech translation. For speech recognition, the model predicts transcriptions in the same language as the audio. For speech translation, the model predicts transcriptions to a different language to the audio. Whisper checkpoints come in five configurations of varying model sizes. The smallest four are trained on either English-only or multilingual data. The largest checkpoints are

multilingual only. All ten of the pre-trained checkpoints are available on the Hugging Face Hub. The checkpoints are summarised in the following table with links to the models on the Hub:

Size	Parameters	English-only	Multilingual
tiny	39 M	✓	✓
base	74 M	✓	✓
small	244 M	✓	✓
medium	769 M	✓	✓
large	1550 M	×	✓
large-v2	1550 M	×	✓
large-v3	1550 M	×	✓

FIG 5:Whisper Models

ACCURACY:

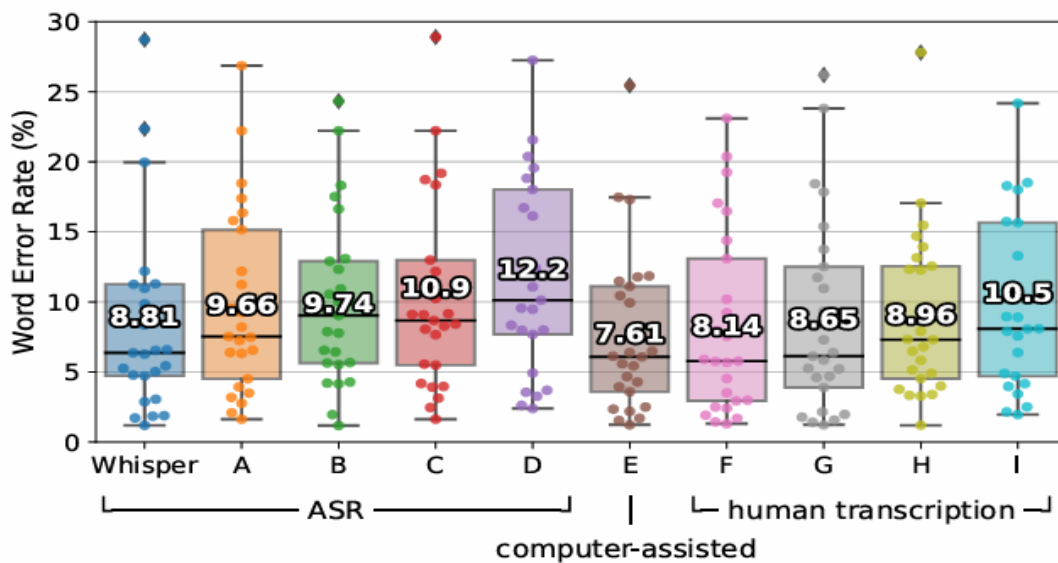


FIG 6:Accuracy of Whisper Transcription with Human transcription

Whisper’s performance is close to that of professional human transcribers. This plot shows the WER distributions of 25 recordings from the Kincaid46 dataset transcribed by Whisper, the same 4 commercial ASR systems from Figure 6 (A-D), one computer-assisted human transcription service (E) and 4 human transcription services (F-I). The box plot is superimposed with dots indicating the WERs on individual recordings, and the aggregate WER over the 25 recordings are annotated on each box.

WORKFLOW:

1. **Audio Preprocessing:** Convert audio signals into Mel spectrogram frames.
2. **Feature Extraction:** Extract important features from the audio frames.
3. **Positional Encoding:** Add positional information to the frames.
4. **Transformer Encoder:** Process frames to capture long-range dependencies.
5. **Transformer Decoder:** Decode frames to generate the text token-by-token.
6. **Output Generation:** Produce the final textual output .

INPUT UI:

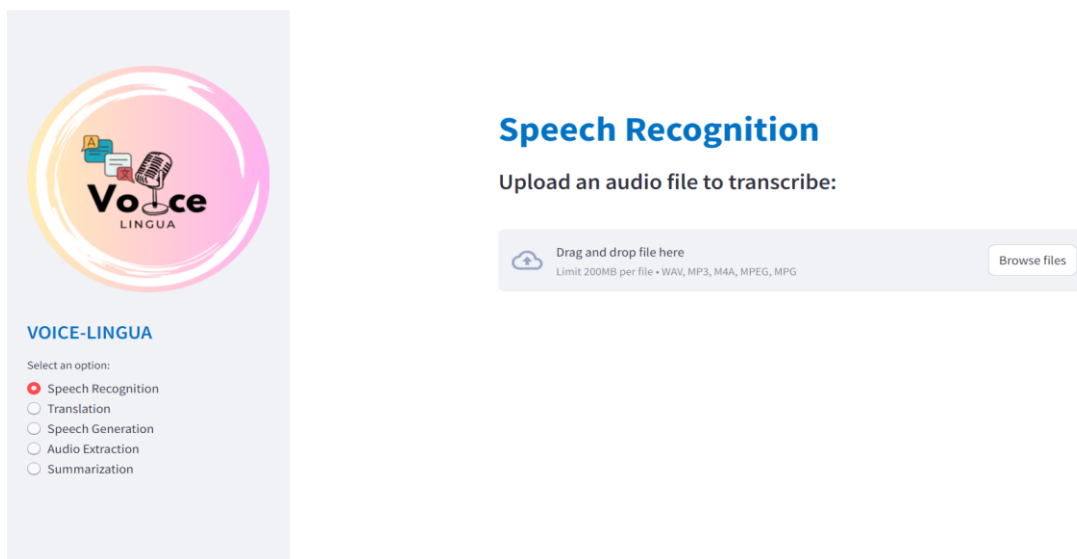


FIG 7:Input UI For Speech Recognition

OUTPUT UI:

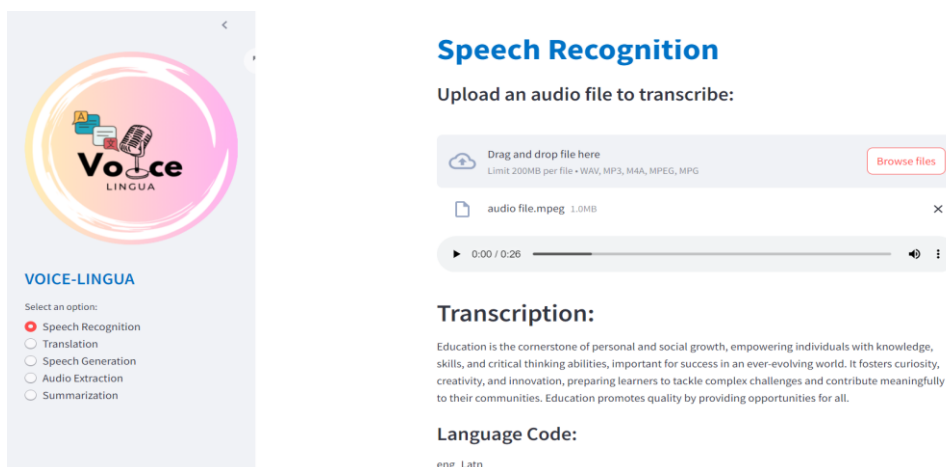


FIG 8:Output UI For Speech Recognition

3.1.2 TEXT TRANSLATION:

Powered by the latest NLLB-200 model, the speech translation capability allows you to translate effectively between source and target languages. 50 languages are supported by this sophisticated model, offering a wide range of linguistic coverage. This advanced system can be useful if you need to make an exact translation while maintaining the overall meaning as well as context.

MODEL USED: "facebook/nllb-200-distilled-600M"

Description: A distilled version of the NLLB-200 model, which is a multilingual neural machine translation model trained on 200 languages.

Model Size: 600M parameters

Trained by: Facebook AI Research (FAIR)

Part of: No Language Left Behind (NLLB) project

This is the model card of NLLB-200's distilled 600M variant. NLLB-200 is a machine translation model primarily intended for research in machine translation, - especially for low-resource languages. It allows for single-sentence translation among 200 languages.

Information on how to - use the model can be found in the Fairseq code repository along with the training code and references to evaluation and training data. NLLB-200 is trained on general domain text data and is not intended to be used with domain-specific texts, such as medical domain or legal domain. The model is not intended to be used for document translation. The model was trained with input lengths not exceeding 512 tokens, therefore translating longer sequences usage is less favourable for this application.

ACCURACY:

eng-xx		xx-eng		eng-xx		xx-eng	
Published	NLLB-200	Published	NLLB-200	Published	NLLB-200	Published	NLLB-200
<u>khm</u> (b)5.9/-	0.4/27.4	(b)10.7/-	16.8 /36.5	<u>hin</u> (l)22.1/-	27.2 /51.5	(l)32.9/-	37.4 /61.9
<u>npi</u> (c)7.4/-	10.4 /39.0	(c)14.5/-	29.3 /54.8	<u>khm</u> (l)43.9/-	45.8 /42.3	(l)27.5/-	39.1 /61.1
<u>pbt</u> (b)9.3/-	10.5 /34.3	(b)15.7/-	22.0 /46.8	<u>mya</u> (c) 39.2 /-	23.5/31.5	(c) 34.9 /-	32.7/57.9
<u>sin</u> (c)3.3/-	11.6 /40.9	(c)13.7/-	23.7 /49.8				

FIG 9:Accuracy Scores for NLLB model

NLLB-200 model is evaluated using BLEU, spBLEU, and chrF++ metrics widely adopted by the machine translation community. Additionally, human evaluation has been done with the XSTS protocol.

WORKFLOW:

1. **Tokenization:** Convert input text into tokens.
2. **Positional Encoding:** Add positional information to the tokens.
3. **Transformer Encoder:** Process tokens to capture contextual relationships.
4. **Transformer Decoder:** Decode tokens sequentially to generate the translation.
5. **Output Generation:** Detokenize the generated tokens to produce the final translated text.

INPUT UI:

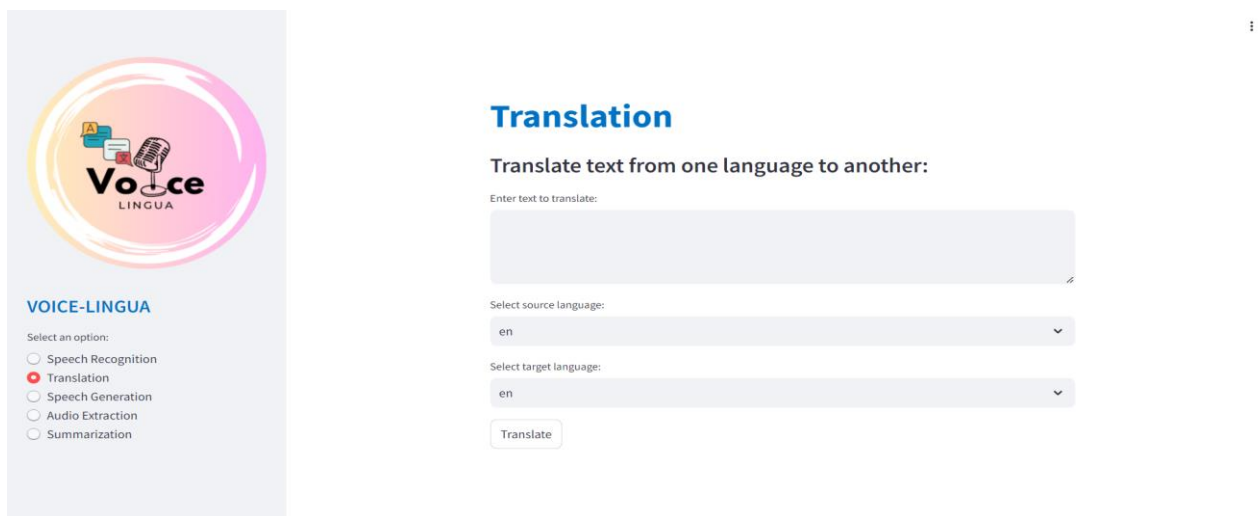


FIG 10:Input UI For Text Translation

OUTUT UI:

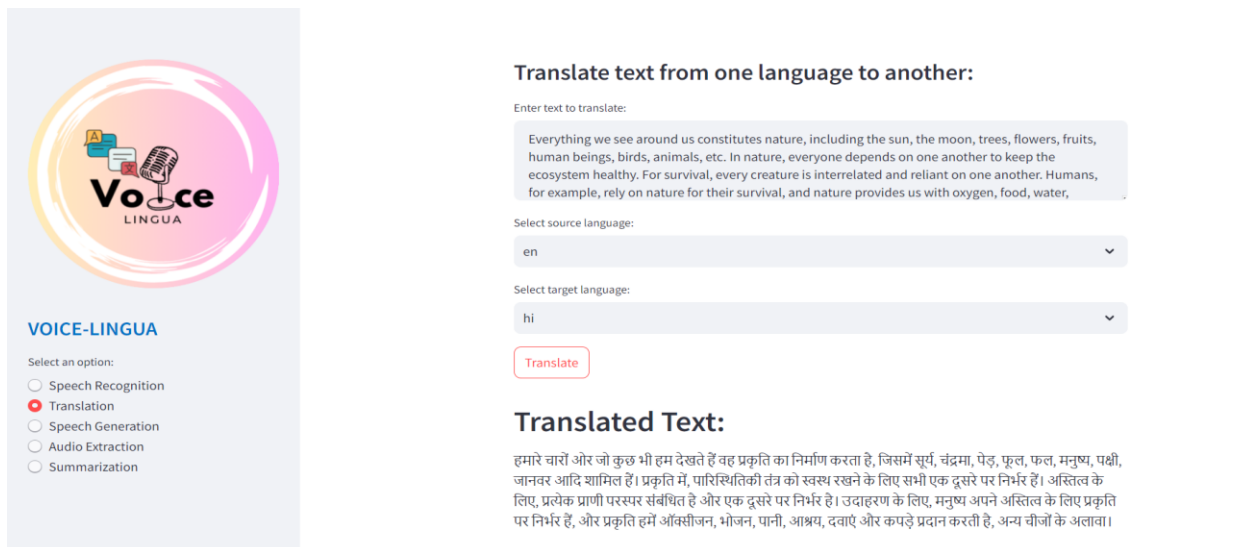


FIG 11:Output UI For Text Translation

3.1.3:SPEECH GENERATION:

The speech generation feature, leveraging the NLLB-200 model, converts text into natural-sounding speech with remarkable accuracy and clarity. Supporting 50 languages, it ensures broad linguistic accessibility and inclusivity. Users can input text in any supported language and receive high-quality speech output that retains the original context. This feature is handy for creating multilingual voiceovers and enhancing language learning applications.

LIBRARY USED: gtts (Google Text-to-Speech)

GTTS (Google Text-to-Speech) is a Python library and command-line tool for interfacing with Google Translate's text-to-speech API. This tool allows users to convert text into speech using the power of Google's TTS engine, which supports multiple languages and accents.

The gTTS library leverages the undocumented Google Translate text-to-speech functionality, which means the accuracy of the generated speech is dependent on the quality of the underlying Google Translate TTS system.

WORKFLOW:

1. Text Translation: The code uses the GoogleTranslator from the googletrans library to translate the input text.

The translate method is called on the translator object, passing the input text as the argument. The translated text is stored in the translated text variable.

The input text is translated into the target language using the Google Translator API. The source language is automatically detected, and the target language is specified by the user.

2. Audio Generation: The translated text is then used to generate an audio file using the Google Text-to-Speech (gTTS) API. The target language is specified to ensure the correct pronunciation and accent.

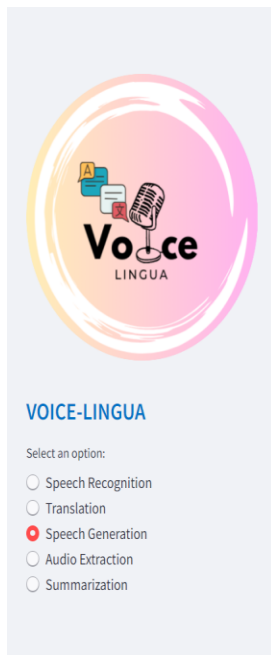
3. Audio File Saving: The generated audio file is saved with a specified filename.

4. Error Handling: If the target language is not valid, an error message is displayed.

5.Function Call: The `translate and generate audio` function is called with the input text, target language, and filename as arguments.

6.Output: The generated audio file is saved and can be played back.

INPUT UI:



Speech Generation

Translate text and generate audio in the target language:

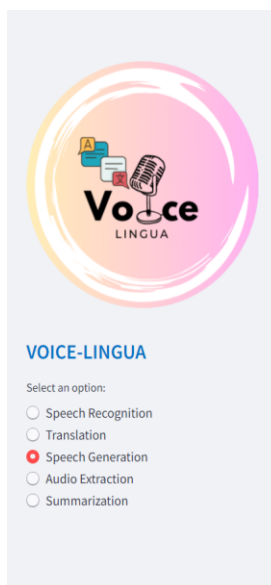
Enter text to translate:

Select target language:

Speech Generation

FIG 12:Input UI For Speech Generation

OUTUT UI:



Speech Generation

Translate text and generate audio in the target language:

Enter text to translate:

Everything we see around us constitutes nature, including the sun, the moon, trees, flowers, fruits, human beings, birds, animals, etc. In nature, everyone depends on one another to keep the ecosystem healthy. For survival, every creature is interrelated and reliant on one another. Humans, for example, rely on nature for their survival, and nature provides us with oxygen, food, water,

Select target language:

Speech Generation

Audio file generated successfully!

0:00 / 0:42

FIG 13:Output UI For Speech Generation

3.1.4 :AUDIO EXTRACTION:

The audio extraction feature efficiently converts video files into high-quality audio, preserving the original sound. This tool is indispensable for multimedia production, allowing users to reuse video content into podcasts, audio clips, and other audio formats. It supports various video file types, ensuring compatibility and ease of use.

LIBRARY USED: moviepy

MoviePy is a Python library for video editing: cutting, concatenations, title insertions, video compositing (a.k.a. non-linear editing), video processing, and creation of custom effects.

MoviePy is an open source software originally written by [Zulko](#) and released under the MIT licence. It works on Windows, Mac, and Linux, with Python 2 or Python 3.

The VideoFileClip class within MoviePy serves as a fundamental component for loading, editing, and extracting audio from video files. It provides an intuitive interface to handle various video formats seamlessly, including popular formats like MP4, AVI, and more. By leveraging VideoFileClip, users can easily load video files, manipulate their contents such as extracting audio tracks or editing segments, and export the processed content into desired formats.

WORKFLOW:

1.Check File Existence:

The code first checks if the video file path provided is a valid file using the `os.path.isfile()` function. If the path is not a file, an error message is displayed.

2.Load the Video File:

If the path is a valid file, the code loads the video file using the VideoFileClip class from the `moviepy.editor` module.

The video object is stored in the video variable.

3.Extract the Audio:

The code extracts the audio from the video using the audio attribute of the video object.

The extracted audio is stored in the audio variable.

4.Write the Audio to a File:

The code writes the extracted audio to a file with the filename "output_audio.mp3" using the `write_audiofile()` method of the audio object.

INPUT UI:

Audio Extraction

Extract audio from a video file:

Enter the path to the video file:

Extract Audio

FIG 14:Input UI For Audio Extraction

OUTUT UI:

Audio Extraction

Extract audio from a video file:

Enter the path to the video file:

Extract Audio

Audio extracted successfully!

Download the extracted audio file

Audio file not downloaded.

FIG 15:Output UI For Audio Extraction

3.1.4:SUMMARIZATION:

The summarization feature provides precise and coherent summaries of lengthy texts, reducing the content into customizable lengths according to the length of the input. This tool can be particularly valuable for professionals in journalism, research, and business, where quick and accurate content summarization is essential.

MODEL USED: sshleifer/distilbart-cnn-12-6

DistilBART is a compressed version of the BART (Bidirectional and Auto-Regressive Transformers) model, specifically fine-tuned for the CNN/DailyMail dataset. The model has been designed to perform abstractive summarization tasks efficiently while maintaining a balance between performance and computational efficiency. For specific tasks like summarization, BART is fine-tuned on datasets such as CNN/DailyMail. The facebook/bart-large-cnn model is particularly fine-tuned for summarization tasks.

Model	SQuAD 1.1 F1	MNLI Acc	ELI5 PPL	XSum PPL	ConvAI2 PPL	CNN/DM PPL
BERT Base (Devlin et al., 2019)	88.5	84.3	-	-	-	-
Masked Language Model	90.0	83.5	24.77	7.87	12.59	7.06
Masked Seq2seq	87.0	82.1	23.40	6.80	11.43	6.19
Language Model	76.7	80.1	21.40	7.00	11.51	6.56
Permuted Language Model	89.1	83.7	24.03	7.69	12.23	6.96
Multitask Masked Language Model	89.2	82.4	23.73	7.50	12.39	6.74

FIG 16:Score of Bart Model fine tuned on CNN/DailyMail datasets

For summarization tasks (XSum and CNN/DM), the Masked Seq2seq model shows the best performance with the lowest perplexity scores (6.60 for XSum and 6.19 for CNN/DM).

WORKFLOW:

1. **Input Encoding:** The input text is first tokenized and fed into the encoder. The encoder generates contextualized representations of the input tokens.
2. **Denoising Objective:** During training, BART is trained with a denoising objective where parts of the input are masked or corrupted, and the model learns to reconstruct the original text. This helps the model learn robust representations.

3. **Decoding:** In the summarization task, the decoder generates the summary by predicting the next token based on the contextualized representations from the encoder and previously generated tokens

METRIC BASED EVALUATION:

Model	Conciseness (Compression Ratio)	Relevance (ROUGE)	Coherence (LSA)	Readability (Flesch-Kincaid)
DISTILBART	0.19	0.36	0.57	0.45
BERT	0.17	0.25	0.56	0.42
PROPHETNET	0.05	0.08	0.29	0.38
T5	0.15	0.29	0.59	0.43
BART	0.16	0.33	0.57	0.40
PEGASUS	0.13	0.28	0.49	0.38

FIG 17: Evaluation scores

The metrics-based scores for each LLM are shown in Figure 17. These scores are averages computed from 30 summaries and have been normalized to a range from 0 to 1, where 0 indicates the lowest performance and 1 the highest. Notably, for the compression ratio, we adjusted the scores by subtracting them from 1 to better represent the conciseness as a desirable attribute (lower compression ratio indicating higher conciseness).

INPUT UI:

VOICE-LINGUA

Select an option:

- ☐ Speech Recognition
- ☐ Translation
- ☐ Speech Generation
- ☐ Audio Extraction
- ☒ Summarization

Text Summarizer

Enter the text to summarize:

Nature is beautiful, yet it is difficult to put into words. Nature is honoured with a variety of religious traditions. The primary source of life on Earth is the components that exist naturally. All of the elements are linked. Natural ingredients can never be substituted. Humans process natural materials for use in today's ever-changing world and destroy their rawness and individuality. Human activities have an impact on nature, and as a result, the quality of the environment is deteriorating. Deterioration of nature is primarily caused by pollution of the air and water. The oxygen produced by a single fully-grown tree is enough to supply ten people, and the amount of oxygen released by a forest is unquestionably adequate for a metropolis or town. Nature is a healer, and it is the foundation for many industries. Nature, on the other hand, is both a giver and a taker, according to...

Number of words in the input text:

234

Choose the minimum summary length

78 103 117

Choose the maximum summary length

117 234 234

Summarize

FIG 18: Input UI for Text Summarization

OUTPUT UI:



Number of words in the input text:

234

Choose the minimum summary length

78

183

117

Choose the maximum summary length

117

234

234

Summarize

Summary:

Human activities have an impact on nature, and as a result, the quality of the environment is deteriorating . The oxygen produced by a single fully-grown tree is enough to supply ten people . The benefits of sunlight and fresh air to our health cannot be overstated . Earthquakes, volcanic eruptions, floods, and cyclones are examples of natural calamities . People prefer to raise their children in the countryside in the U.S. Their primary goal is to introduce their children to the benefits of natural beauty .

FIG 18: Output UI for Text Summarization

CHAPTER 4

4.1 LIMITATIONS

For a project like VOICE LINGUA, which involves advanced AI and NLP technologies, there can be several potential limitations:

1. **Data Quality and Availability:**

High-Quality Data: Access to large datasets of high-quality, annotated data is crucial for training effective NLP models. If training data contains biases, the AI system might perpetuate those biases in its outputs.

2. **Computational Resources:**

Hardware Requirements: Advanced AI models often require significant computational power, which can be costly and resource-intensive.

3. **Technical Challenges:**

Integration and Compatibility: Ensuring seamless integration with various platforms and maintaining compatibility with different languages.

4. **Performance and Accuracy:**

Speech Recognition: Accurately transcribing speech, especially in noisy environments or with accents, can be difficult.

Speech Generation: Creating natural and human-like speech, especially with the correct emotional tone and intonation, can be challenging.

6. **User Experience:**

Usability: Designing a user-friendly interface that caters to a wide range of users, including those with limited technical skills.

Understanding these limitations can help in planning strategies to mitigate them and improve the overall effectiveness and reliability of *VOICE LINGUA*.

CHAPTER 5

5.1 CONCLUSIONS AND FUTURE WORK

CONCLUSIONS:

The Voice Lingua project is a successful example of how several advanced language processing capabilities can be combined to create a coherent system. Voice Lingua provides a complete multilingual communication and information processing solution by tackling the problems related to speech recognition, translation, speech generation, audio extraction, and summarization. This project's major accomplishments include:

1. **Speech Generation:** Capable of generating natural-sounding speech in multiple languages, enhancing accessibility and user engagement.
2. **Translation:** Accurate and context-aware translation between various languages has been achieved, facilitating cross-linguistic communication.
3. **Speech Recognition:** High accuracy in transcribing spoken language into text has been achieved.
4. **Audio Extraction:** Efficient extraction of audio segments from longer recordings, enabling targeted analysis and processing.
5. **Summarization:** Effective summarization of textual content, allowing users to quickly grasp the essence of long texts.

FUTURE WORK:

Even with the significant advancements, there are still several areas that might use further research and development to expand Voice Lingua's functionality and applications:

1. **Improve Naturalness and Accuracy:**

- a. **Speech Generation:** Improve the produced speech's naturalness by adjusting its grammar and emotional tone.

b. **Speech Recognition:** Improve speech recognition accuracy for a variety of accents and dialects, as well as in loud settings.

2.**Expansion of Language Support:** Increase the number of supported languages and dialects, focusing on underrepresented languages to make the system more inclusive.

3.**Real-time Processing:** Optimize the system for real-time processing, reducing latency in speech recognition, translation, and generation to support live interactions.

4.**Security and Privacy:** To safeguard user data, particularly in apps that involve sensitive data, it is important to enhance security and privacy safeguards.

Voice Lingua can further establish itself as a top solution in the multilingual speech and language processing space and offer even more benefits to its consumers by tackling these next work topics.

CHAPTER 6

6.1 REFERENCES

1. openai/whisper-large-v3

Robust Speech Recognition via Large-Scale Weak Supervision:

<https://arxiv.org/pdf/2212.04356>

Hugging face model: [openai/whisper-large-v3](#)

Hugging face link: <https://huggingface.co/openai/whisper-large-v3>

2.facebook/nllb-200-distilled-600M

No Language Left Behind: Scaling Human-Centered Machine Translation:

<https://arxiv.org/pdf/2207.04672>

Hugging face model: [facebook/nllb-200-distilled-600M](#)

Hugging face link: <https://huggingface.co/facebook/nllb-200-distilled-600M>

3. facebook/bart-large

BART: Denoising Sequence-to-Sequence Pre-training for Natural Language Generation, Translation, and Comprehension: <https://arxiv.org/pdf/1910.13461>

Hugging face model: [facebook/bart-large](#)

Hugging face link: <https://huggingface.co/facebook/bart-large>

4. sshleifer/distilbart-cnn-12-6

DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter:

<https://arxiv.org/pdf/1910.01108>.

EVALUATING TEXT SUMMARIES GENERATED BY LARGE LANGUAGE MODELS
USING OPENAI'S GPT:

<https://arxiv.org/pdf/2405.04053>

Hugging face model: [sshleifer/distilbart-cnn-12-6](https://huggingface.co/sshleifer/distilbart-cnn-12-6)

Hugging face link: <https://huggingface.co/sshleifer/distilbart-cnn-12-6>

5. moviepy

Associated documentation: <https://pypi.org/project/moviepy/>

6.gTTS

Associated documentation: <https://gtts.readthedocs.io/en/latest/>

