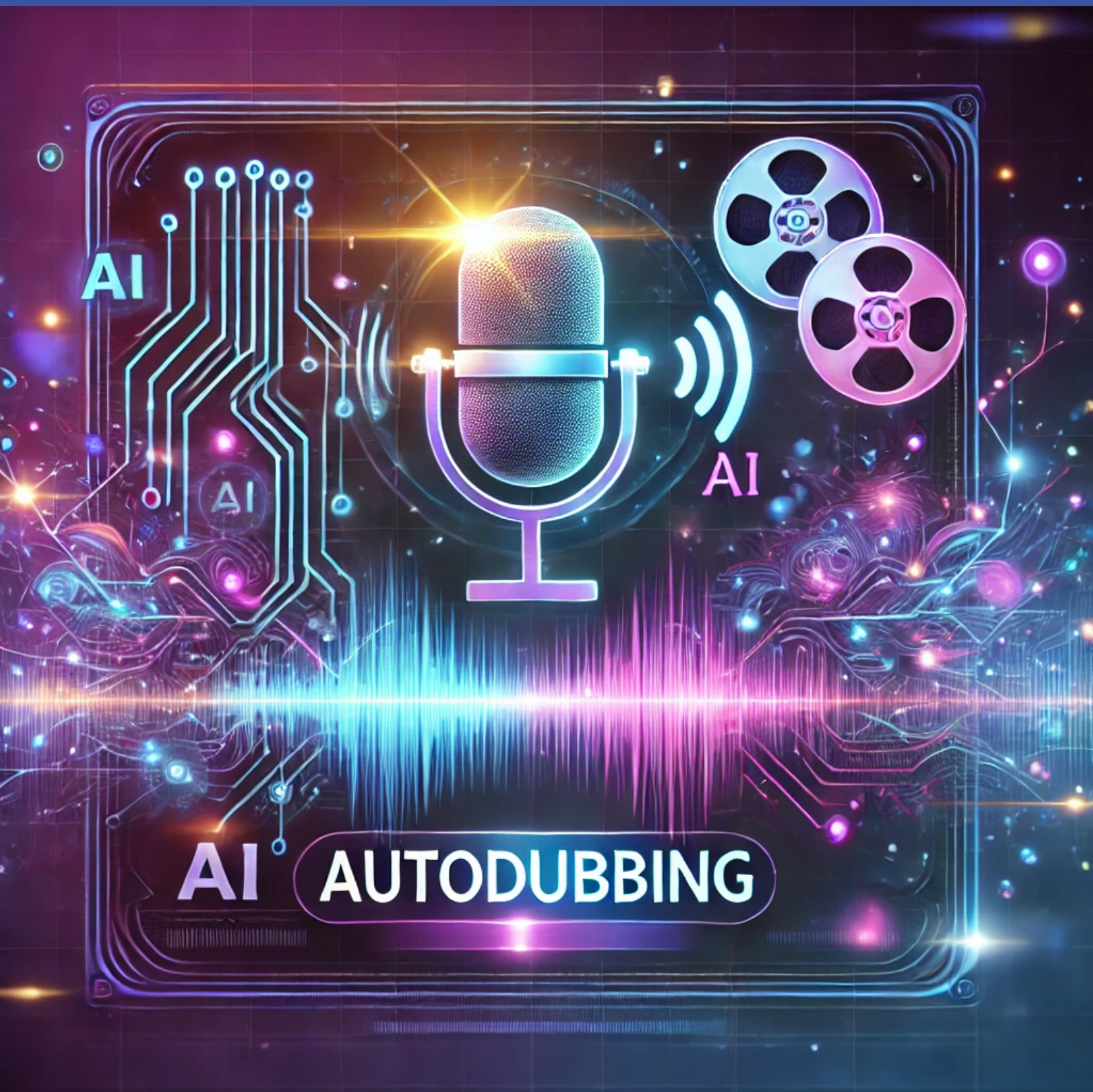
DTU Compute
Department of Mathematics and Computer Science

Bachelor of Artificial Intelligence and Data
Autodubbing

Report



Bachelor of Artificial Intelligence and Data
Autodubbing

Report
05-01-2025

By
Muneer Kayali *(s214642)*

Cover photo: Chat GPT-4

Acknowledgements

[Morten Mørup], [Professor], [Guidance counselor]

Approval

Muneer Kayali *(s214642)* -

Muneer Kayali

.....

Signature

01/05/2025

.....

Date

Contents

Abstract

This project presents an AI-driven autodubbing pipeline designed for the linguistic and cultural localization of English-to-Danish video content. The proposed pipeline integrates state-of-the-art technologies, including Whisper for speech transcription, GPT-4 for speaker diarization, DeepL for machine translation, Google Cloud Neural2 for text-to-speech synthesis, and Seed-VC for voice conversion. The system aims to automate the traditionally labor-intensive dubbing process by preserving speech timing, speaker identity, and emotional expression in the generated audio. Key challenges addressed include achieving accurate speech-to-text alignment, managing voice conversion fidelity, and ensuring synchronization between audio and video. Experimental results demonstrate the pipeline's potential in producing high-quality dubbed content while identifying areas for improvement, such as multi-speaker scenarios and emotional speech representation. The findings suggest the feasibility of a scalable autodubbing system for language pairs with limited resources, contributing to the broader field of AI-driven media localization.

1 Introduction

Globalization and advancements in technology have increasingly bridged language barriers through state-of-the-art translation and artificial intelligence (AI). Traditionally, subtitles and manual dubbing have been used to make international films, TV shows, and other video content accessible to audiences speaking different languages. However, these manual processes are time-consuming, expensive, and heavily reliant on human expertise. As the demand for localized content continues to grow, there is a pressing need for automated solutions to reduce the cost and time associated with dubbing.

This project focuses on developing an AI-driven autodubbing pipeline designed for localization (linguistic and cultural adaptation of media content) of specifically English-to-Danish media. English, being one of the most widely spoken languages globally, contrasts significantly with Danish, which is spoken by a much smaller population. Danish speakers often rely on subtitles or expensive manual dubbing to enjoy international content. By automating this process for this specific language pair, the project aims to improve accessibility to foreign media for Danish-speaking audiences while demonstrating the feasibility of scaling this technology to other languages.

The choice of Danish as the target language provides a unique set of linguistic and cultural challenges, making it an ideal testing ground for a robust and adaptable autodubbing pipeline. Unlike larger languages like Spanish or French, Danish offers fewer resources and tools for AI-driven media localization, emphasizing the importance of creating a scalable and efficient system.

This project also serves as a prototype for broader applications. If successful, the methodologies and technologies developed here could be generalized and adapted to other language pairs, particularly for underrepresented languages. Such a system has the potential to revolutionize media localization, enabling fully automated dubbing across a wide range of languages, and significantly improving global accessibility to video content.

1.1 State of the Art

1.1.1 Autodubbing Overview

Autodubbing, the process of automatically translating and dubbing speech in video content, is an emerging field with the potential to significantly reduce the cost and time associated with traditional dubbing practices. While traditional dubbing requires human translators, voice actors, and careful synchronization, autodubbing aims to automate this process using advances in artificial intelligence (AI). The field, however, is still in it's early stages, with only a few companies having published papers on the matter. For example, Amazon published a paper (1) in 2020, where they suggest improving on the Speech-to-Speech pipeline (transcription-translation-synthesis) such that it can be used for dubbing. In said paper, the focus is heavily on alignment of speech by use of methods such as controlling the translated text's length and using prosodic alignment strategies such as syllable-level synchronization. Another example of this is a paper by Microsoft (11), where the focus and attention is heavily on length control to allow for better speech alignment. However, none of these papers try to focus on preserving the voice and emotion of the source speaker, which this project will explore. Papercup and Google have experimented with autodubbing, but no research papers are published on this. Google has been contributing to speech recognition and translation technologies, particularly through models like Google ASR

and DeepMind's WaveNet for text-to-speech (17). While Google hasn't yet commercialized a full autodubbing system, its work on tools like Google Translate and speech synthesis models forms the foundation for potential AI dubbing systems. These tools are vital for the development of end-to-end autodubbing pipelines.

1.1.2 Speech-to-Text

One of the leading systems in this space is Whisper, developed by OpenAI. Whisper is trained on 680,000 hours of multilingual and multitask speech data using a large-scale weak supervision approach (also uses zero-shot transfer), allowing it to generalize well to new datasets without the need for fine-tuning. This means that Whisper does well in transcription tasks where diverse accents and audio environments are encountered. It has demonstrated robustness in transcribing English speech, even in noisy environments, which is essential for video content where audio quality may vary. (12)

1.1.3 Machine Translation

Machine translation (MT) has evolved dramatically from rule-based systems to the current generation of neural machine translation (NMT) models, which leverage deep learning techniques to provide highly accurate translations between languages. NMT systems, such as Google Translate, OpenNMT and DeepL are state-of-the-art models designed to translate more accurately through contextual understanding. What makes these systems state-of-the-art is their use of transformer architectures. Transformers are a big topic in this project, and will be explained later in the discussion section. Moreover, their training on large, multilingual datasets enables them to generalize better across languages, including smaller ones like Danish. (4)

1.1.4 Text-to-Speech

Text-to-speech systems have similarly undergone significant transformation with the introduction of neural networks. These systems aim to convert written text into spoken language, with a focus on producing natural, human-like speech. An example of a state-of-the-art text-to-speech system today is WaveNet by DeepMind. WaveNet generates audio waveforms from scratch using deep generative models. This allows it to capture differences in tone, pitch and rhythm Unlike earlier TTS systems that relied on concatenative synthesis (piecing together pre-recorded sound samples), WaveNet can produce more natural-sounding voices by modeling raw audio waveforms at a granular level. (3)

This system is considered state-of-the-art because of their ability to synthesize speech that sounds natural and expressive, retaining the emotional tone of the original text. They also provide options for customization and fine-tuning, allowing users to adjust voice pitch, speed, and emotion to better suit specific contexts like dubbing for media content.

1.1.5 Voice Conversion

Voice conversion (VC) focuses on modifying an input speech signal to match the voice characteristics of a target speaker while retaining the linguistic content. This technology plays a critical role in applications such as dubbing, enabling the system to reproduce voices that are consistent with the original speaker's tone, pitch, and style.

State-of-the-art voice conversion systems include SEED-VC and VQ-VAE-based methods. SEED-VC, used in this project, leverages advanced diffusion models and neural vocoders to achieve highly realistic and expressive voice conversion. By using pre-trained embeddings and fine-tuned diffusion networks, SEED-VC achieves smooth transitions and natural-sounding voices. (14)

These systems are state-of-the-art because of their ability to retain linguistic and prosodic details of the original content, match the vocal timbre of the target speaker with high accuracy and adapt dynamically to new speakers using pre-trained models without extensive fine-tuning. Voice conversion bridges the gap between raw translated text and the desired natural-sounding target audio, making it a crucial component in the autodubbing pipeline.

1.2 Research Questions

How effectively can a pipeline integrating Whisper (speech-to-text), Chat GPT-4 (text-based diarization) DeepL (machine translation), Google Cloud text-to-speech and Seed-VC (voice conversion) be constructed to facilitate autodubbing?

This question investigates the practical feasibility and performance of combining state-of-the-art AI models into a unified pipeline to produce high-quality autodubbed videos. It examines the fluidity of data transitions between pipeline components, evaluating whether significant preprocessing or structural modifications are required for seamless communication. Additionally, it explores computational efficiency, identifying potential bottlenecks that may hinder real-time or scalable performance. Finally, it assesses the overall quality of the output, determining whether the generated autodubbed videos meet practical and aesthetic standards or reveal critical shortcomings in transcription, translation, diarization, or voice synthesis.

How can speech input be optimally fragmented to ensure accurate video-to-audio synchronization and speaker identification?

This question examines the methodologies and strategies required to segment input video effectively, ensuring the resulting autodubbed audio aligns closely with the visual content and maintains the integrity of speaker transitions. It emphasizes achieving realistic synchronization between lip movements and dubbed speech, ensuring the voices accurately represent the respective speakers and closely mimic their original tonal and emotional qualities. Additionally, this question evaluates various segmentation and diarization techniques, focusing on identifying approaches that are not only highly accurate but also seamlessly integrable into the pipeline, minimizing computational overhead and preserving the overall quality of the output.

How can voice generation methods distinguish between different voice types (e.g., gender, age) and capture tonal variations to simulate emotions?

This question focuses on the capabilities and limitations of voice generation technologies, aiming to produce output audio that closely resembles natural human voices and mimics the specific characteristics of individual speakers. It explores which aspects of voice—such as gender, age, emotional tone, and overall likeness—are most critical to achieving realism and listener immersion. Additionally, this question evaluates the feasibility of incorporating these characteristics into the pipeline while maintaining computational efficiency and ensuring seamless integration. The emphasis is placed on balancing technical feasibility with output quality, assessing whether a focus on broader attributes (e.g., gender and basic tone) or more nuanced details (e.g., emotional shifts, speech idiosyncrasies) yields the most practical and effective results.

2 Methods & Theory

2.1 The Pipeline

The pipeline of this projects consists of multiple different AI models, where the technical specifics of the models are explored below in this chapter. Specifically, the pipeline follows this cascaded structure:

$$\texttt{Transcription} -> \texttt{Textual Diarization} -> \texttt{Machine Translation} ->$$

$$\texttt{Voice Synthesis} -> \texttt{Voice Conversion}$$

Here, Whisper is used for transcription, Chat GPT-4 for diarization, DeepL for translation, Google Cloud Neural2 voices for speech synthesis, and Seed-VC for voice conversion. For a more detailed overview of how the pipeline works, refer to Figure 2.1.

2.2 ”Attention is All You Need”

Before we dive deeper into the individual models, some concepts that are key in problem formulations that involve text or audio will first be explained. As it turns out, for most of these problems, Attention is all you need. This section will focus on the theoritical aspect of Attention in Transformers. The sources of this theory is mostly based on the corresponding famous paper by Google. (7)

2.2.1 Sequence-to-Sequence

Most of the technologies used in this pipeline follow sequence-to-sequence methods, a deep learning framework designed for tasks where both the input and output consist of sequences of varying lengths. In the context of for example STT, the input is a sequence of audio features extracted from speech, while the output is a sequence of text tokens representing the transcribed speech. Seq2Seq models are particularly suited for tasks like transcription, translation and voice synthesis, because they can handle the temporal nature of audio data and the variable length of both the input and the output. In Seq2Seq models, the goal is to learn the function:

$$P(Y|X) = \prod_{t=}^{T} P(y_t|y_{1..t-1}, X)$$

Where X is the input sequence, Y is the output sequence, and $P(y_t|y_{1..t-1}, X)$ is the probability of predicting token y at timestep t based on the input and previous output tokens.

2.2.2 Encoder-Decoder structure

A fundamental component of sequence-to-sequence models is the encoder-decoder architecture, which was popularized in neural machine translation (NMT) and has since been adapted for tasks like speech-to-text. This architecture is split into two parts, each serving a distinct role in the transcription process:

The Encoder: The encoder receives the input audio data, typically in the form of mel spectrograms (a time-frequency representation of audio signals). It processes this data and transforms it into a continuous latent representation, often called a context vector or hidden state. The encoder learns to capture key features from the audio signal, such as phonetic patterns, pitch, and timing, while reducing the dimensionality of the input.
The Decoder: The decoder takes the context vector produced by the encoder and generates text tokens step by step. At each step, the decoder uses both the context vector and previously

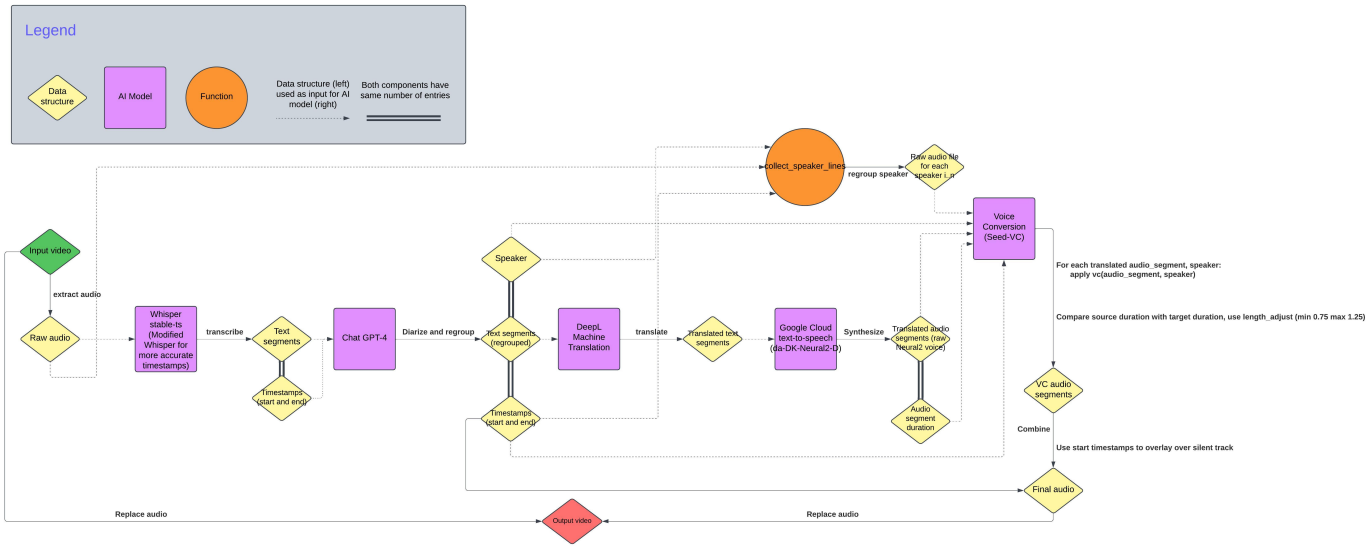


Figure 2.1: The Autodubbing Pipeline Flowchart

generated tokens to predict the next token in the sequence. This autoregressive nature means the decoder generates each word or subword one at a time, progressively building the final transcription. Together, the encoder and decoder create a powerful framework for transforming complex, high-dimensional audio data into structured textual output.

2.2.3 Attention

A major advancement in sequence-to-sequence models was the introduction of attention mechanisms, which revolutionized the handling of long sequences. Traditional encoder-decoder models relied on a fixed-length context vector to summarize the entire input sequence, which often resulted in information bottlenecks, especially for longer sequences. Attention mechanisms addressed this limitation by allowing the decoder to dynamically focus on different parts of the input sequence during the decoding process.

The attention mechanism works by computing a weighted sum of the encoder outputs, where the weights are determined by the relevance of each input token to the current decoding step. Formally, for each decoder time step t:

$$c_t = \sum \alpha_{t,i} h_i$$

(5)

Where c_t is the context vector used in the decoding step, h_i is the hidden state of the encoder (meaning the information encoded from the input), and $\alpha_{t,i}$ is the attention weight, meaning the weight that determines the importance of the ith index for the current decoder timestep.

The attention weight is calculated as such:

$$\alpha_{t,i} = \frac{\texttt{exp}(e_{i,t})}{\sum_j \texttt{exp}(e_{i,t})}$$

Where $e_{i,t}$ is the alignment score, where the most common are Dot-product and Additive alignment score between encoder and decoder hidden states.

This introduction of attention ensures context-awareness without the need of a fixed-length context vector, and at the same time, can encompass more context with less computational costs than context vectors.

2.2.4 Multi-head attention

A further improvement introduced by the Transformer architecture is multi-head attention (MHA), which enhances the model's ability to focus on different parts of the input sequence simultaneously. Instead of a single attention mechanism, multiple attention heads operate in parallel, each learning different representations of the input. The attention mechanism computes a weighted sum of the values V based on the compatibility between the query Q and each key K. This compatibility is measured by the attention score, which determines how much focus should be placed on each input token when generating the next output token.

Let's look at an example of multihead attention based on Scaled-Dot-Product Attention used in the paper. For a query matrix Q, key matrix K, and value matrix V:

$$Attention(Q, K, V) = softmax(\frac{QK^T}{\sqrt{d_k}})V$$

Then, to achieve Multi-head Attention, instead of performing a single Attention computation, the Q, K and V vectors are linearly projected using learned projections to d_q, d_k and d_v dimensions respectively. Then, for each projection the Attention function is computed on these values instead of the standard Q, K and V. This is done h times (so $h \cdot 3$ learned projections W in total) and finally the output values are concatenated and projected once again to the desired model dimension. Formally, this can be written as such:

$$Multihead(Q, K, V) = Concat(head_1 ... head_h)W^O$$

Where

$$head_i = Attention(QW_i^Q, KW_i^K, VW_i^V)$$

2.2.5 Positional Encoders

Since no recurrence or convolution is used in Transformers, then, in order for the model to make use of the order of the sequence, there must be some information about the relative or absolute position of the tokens in the sequence. To this end, "positional encodings" are added to the input embeddings at the bottoms of the encoder and decoder stacks. The positional encodings have the same dimension dmodel as the embeddings, so that the two can be summed. There are many choices of positional encodings, learned and fixed.

All in all, in the context of this autodubbing pipeline, attention mechanisms play a pivotal role in ensuring accurate transcription, translation, and voice synthesis by allowing models like Whisper and GPT-4 to handle long and complex audio sequences effectively. Their integration into the Transformer architecture underpins the success of contemporary speech and language models, making attention mechanisms a foundational concept for the technologies used in this project.

2.3 Whisper by OpenAI

Whisper is an advanced speech-to-text model developed by OpenAI, designed to tackle complex transcription and translation tasks with remarkable robustness. At its core, Whisper leverages a sequence-to-sequence learning framework based on the Transformer architecture discussed earlier, enabling it to handle diverse audio inputs and produce high-quality, context-aware textual outputs. The model operates by converting audio signals into log-Mel spectrograms, a time-frequency representation that captures essential acoustic features while discarding extraneous information. This spectrogram serves as the input to a series of Transformer

encoder blocks, which process the data using multi-head self-attention mechanisms and feed-forward layers. These encoders extract latent audio features, which are then passed to the Transformer decoder blocks. The decoders, equipped with cross-attention layers, generate the textual output token-by-token, guided by learned positional encodings. A more detailed overview of the model architecture can be seen in Figure 2.2.

What sets Whisper apart is its multitask training approach, incorporating a vast dataset of 680,000 hours of audio. This dataset includes English and non-English transcription, any-to-English speech translation, and even non-speech data, such as background noise. By incorporating such diverse data, Whisper learns to perform a range of tasks beyond transcription, including language identification, voice activity detection (VAD), and zero-shot transfer to unseen languages or scenarios.

Whisper's training uses a multitask training format, where input audio is paired with tokens that specify tasks (e.g., transcription or translation) and metadata (e.g., language or timestamps). This allows the model to flexibly adapt its outputs based on the context. For instance, Whisper can produce time-aligned transcriptions with precise timestamps or text-only outputs for broader use cases.

One of Whisper's standout features is its resilience to noisy data and challenging environments, such as diverse accents, poor audio quality, and overlapping speech. These qualities are particularly advantageous for applications like dubbing pipelines, where audio conditions may vary significantly (12).
In this project, a modified version of Whisper, stable-ts Whisper, is utilized. This variant introduces optimizations for generating precise timestamps, a critical requirement for maintaining alignment between audio segments and the video content in the autodubbing pipeline (9). These enhancements ensure seamless synchronization, forming a foundation for subsequent stages in the pipeline, including translation, synthesis, and voice conversion. The large Whisper model is used, and the parameters chosen for the stable-ts transcribing are:

- `vad = True`
- `min_word_dur = 0.3`
- `nonspeech_error = 0.25`

2.4 GPT-4

GPT-4, developed by OpenAI, represents a significant advancement in the field of natural language processing (NLP). As a large-scale multimodal model, it is capable of processing both text and image inputs to generate coherent and contextually relevant text outputs. This versatility enables GPT-4 to perform a wide array of tasks, including language translation, content generation, and complex reasoning.

At its core, GPT-4 utilizes the Transformer architecture, which has become a foundational model in NLP due to its efficiency in handling sequential data. The Transformer architecture, as discussed earlier, usually comprises an encoder-decoder structure; however, GPT-4 employs a decoder-only configuration optimized for generative tasks. This architecture facilitates the model's ability to generate human-like text by predicting subsequent words in a sequence based on the preceding context. A notable feature of GPT-4 is its extensive training on a diverse dataset encompassing a wide range of internet text and images. This comprehensive training enables the model to acquire a vast repository of knowledge, allowing it to understand and generate text across various domains and styles. The training process involves adjusting

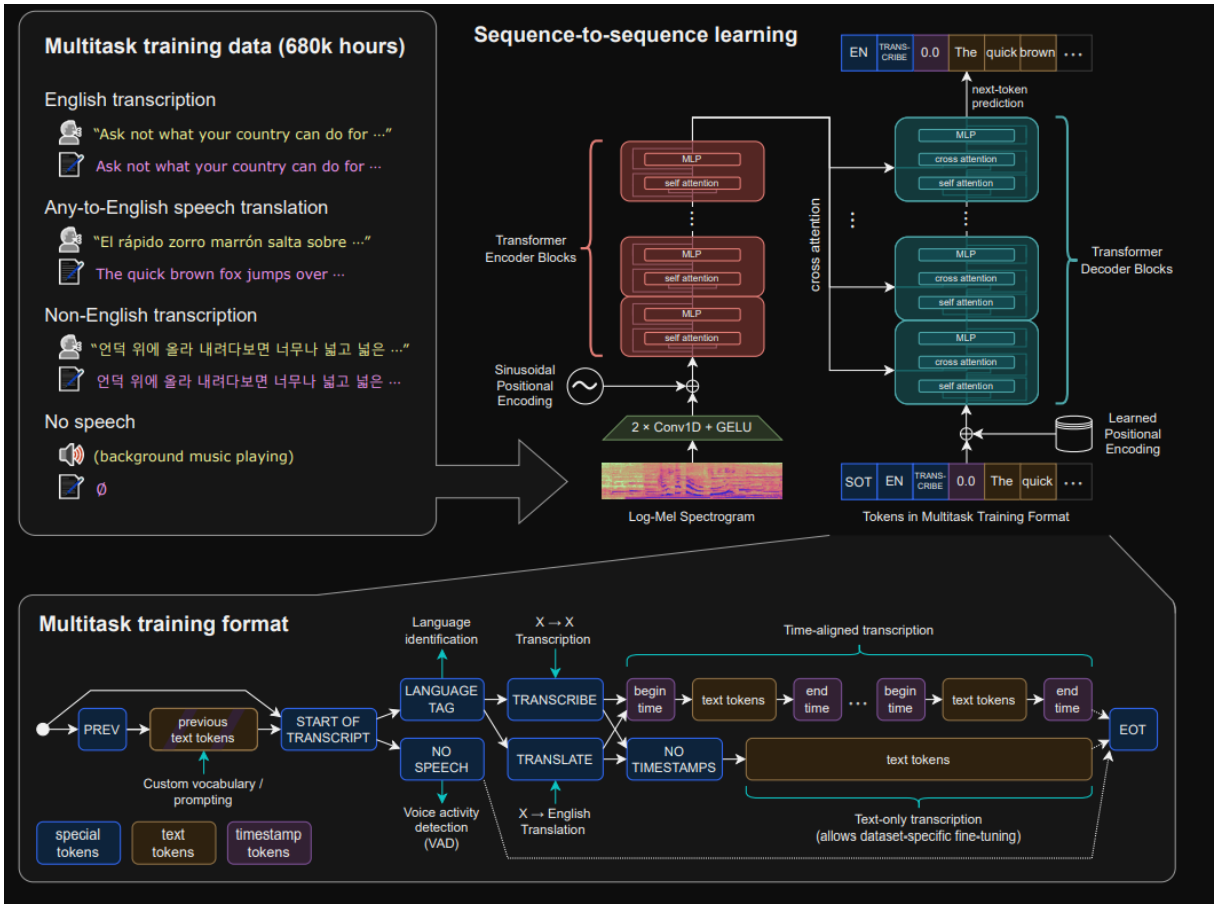


Figure 2.2: Source: Whisper Model Architecture, OpenAI

billions of parameters to minimize prediction errors, thereby enhancing the model's accuracy and fluency in text generation.

In the context of speaker diarization and segment grouping, GPT-4's advanced language understanding capabilities are particularly beneficial (13). Traditional speaker diarization methods often rely on acoustic features to distinguish between speakers. In contrast, GPT-4 can analyze textual content to identify subtle cues indicative of different speakers, such as variations in vocabulary, syntax, and stylistic nuances. This text-based approach allows for effective speaker identification even in scenarios where audio quality is compromised or when speakers have similar vocal characteristics. Furthermore, GPT-4's proficiency in contextual analysis enables it to group adjacent segments spoken by the same speaker within a short time frame. By evaluating the coherence and continuity of the text, the model can accurately merge segments, ensuring that the transcribed dialogue reflects the natural flow of conversation. This capability is essential for applications requiring precise transcription and speaker attribution, such as automated subtitling and meeting transcription services. The prompt used to generate diarized output is:

"You are an expert at analyzing conversational text to determine speakers in a dialogue. The text segments provided are from a conversation involving two speakers or more speakers: 'Speaker 1', 'Speaker 2', ..., 'Speaker n'. Your task is to classify each segment into one of these speakers based on their style, content, or clues. If a segment clearly belongs to one speaker, respond with 'Speaker i' for i in all speakers. If the speaker cannot be confidently determined, force yourself to make a decision. Furthermore, you should group the segments if two or more ad-

jacent segments are spoken by the same speaker, however, this should only be done if the sentences are spoken right away (if there is silence of 2 seconds or more then dont group), Also, the segments should not be long, so if regrouping gives you a long segment of multiple sentences (2-3), then this should not be done.Please provide your output after classifying and regrouping in the following format:
Speaker: speaker_num, Text: text, Start: start, End: end
Avoid formatting the text with bold or italics. Use the exact format as shown above.
Here are the segments: ”

2.5 DeepL: Advanced Neural Machine Translation

DeepL is a prominent neural machine translation (NMT) service that has garnered attention for its high-quality translations and innovative use of deep learning technologies. Launched in 2017, DeepL has distinguished itself through its proprietary neural network architecture and extensive training methodologies. (4)

DeepL's translation system is built upon a sophisticated neural network framework that incorporates elements of the Transformer architecture, such as attention mechanisms. However, DeepL has introduced significant modifications to this architecture, resulting in a unique topology that enhances translation quality beyond the current state of the art in public research.

The effectiveness of DeepL's translations is partly attributed to its training on large-scale datasets comprising billions of parallel sentences. This extensive training enables the model to capture intricate linguistic nuances and produce more natural-sounding translations.

Independent evaluations have demonstrated that DeepL's translations often surpass those of competing services in terms of accuracy and fluency. Studies have shown that DeepL consistently achieves higher BLEU scores—a metric for evaluating the quality of machine-translated text—indicating its superior performance in maintaining the meaning and tone of the original content. (10)
In the context of this project, DeepL is used with it's default parameters to translate each segment from English to Danish.

2.6 Neural2 Voices: Advanced Text-to-Speech Technology

The Neural2 Voices, developed by Google, represent the cutting edge in text-to-speech (TTS) technology, offering unparalleled naturalness and expressiveness in speech synthesis. These voices are specifically designed to generate lifelike audio, making them ideal for applications such as dubbing, virtual assistants, and automated announcements. While the precise methodologies behind Neural2 Voices remain proprietary, it is widely accepted that this technology builds upon foundational advancements like WaveNet, which revolutionized TTS systems and set a benchmark for neural speech synthesis.

WaveNet, introduced by DeepMind in 2016, represents a groundbreaking approach in the field of text-to-speech (TTS) systems and audio generation. Unlike traditional concatenative or parametric synthesis methods, WaveNet directly generates raw audio waveforms, resulting in a significant leap in audio quality and naturalness. Its architecture and probabilistic framework have become a foundation for many modern text-to-speech systems, including Google's Neural2 Voices. WaveNet is built on a convolutional neural network (CNN) architecture with two defining features: causality and dilation.

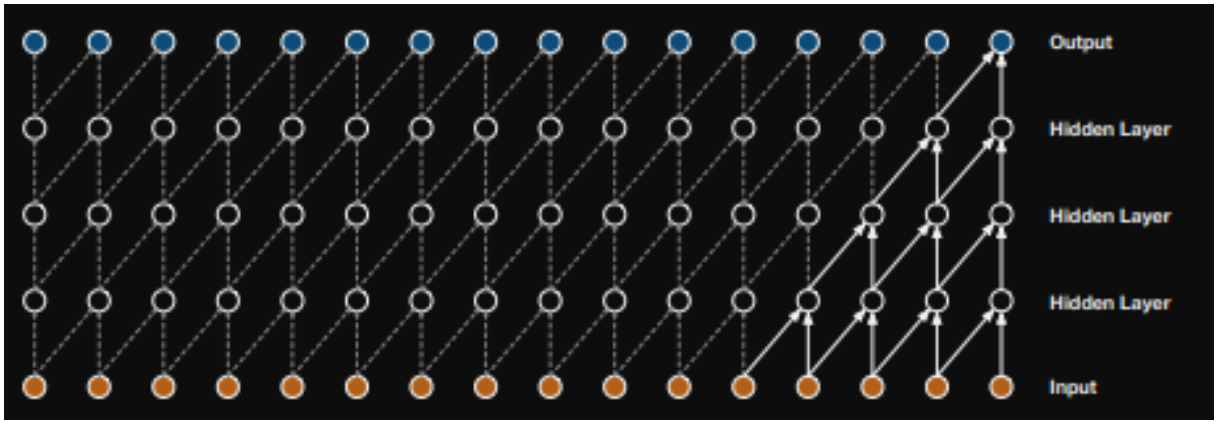


Figure 2.3: Stack of causal convolution layers. From WaveNet paper. (17)

2.6.1 Causal Convolutions

Causal convolutions ensure that the generation of each audio sample is conditioned only on past samples. This prevents information leakage from future samples, preserving the sequential nature of audio data. Mathematically, for a 1D convolution over an input sequence x with a filter f, the output at time step t is:

$$y_t = \sum_{k=0}^{K-1} f_k \cdot x_{t-k}$$

where K is the filter size, and x_{t-k} represents past inputs only, ensuring causality. A visual representation can be seen in Figure 2.3

2.6.2 Dilated Convolutions

Dilated convolutions extend causal convolutions by introducing gaps between filter elements, effectively increasing the receptive field without requiring deeper networks. A dilation rate d defines the spacing between filter elements. The output is calculated as: The output of a dilated convolution can be expressed as:

$$y[t] = \sum_{k=0}^{K-1} w[k] \cdot x[t + d \cdot k]$$

where:

- $y[t]$ is the output at time step t,
- $K$is the filter size,
- $w[k]$ is the filter weight at position k,
- x is the input signal,
- d is the dilation rate defining the step size between filter elements.

By stacking layers with exponentially increasing dilation rates (e.g., $d = 1, 2, 4, 8$), the receptive field grows exponentially, allowing the network to capture long-term dependencies while keeping the model computationally efficient (17).

WaveNet uses a combination of these two concepts, where causal convolutions ensure that output only depends on previous inputs, and dilated convolution ensures long-term dependencies without sacrificing computational efficiency. A visual representation of this combination can be seen in Figure 2.4.

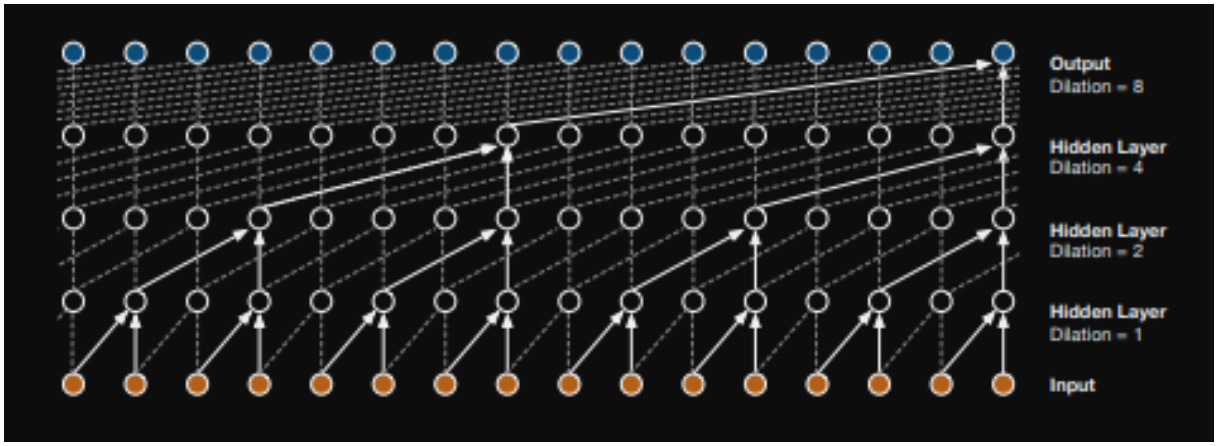


Figure 2.4: Visualization of a stack of dilated causal convolutional layers. (17)

While the exact details of Neural2's architecture remain undisclosed, it is reasonable to infer that it incorporates advancements in prosody modeling and context-aware generation (3). Furthermore, Neural2 is optimized for scalability, enabling real-time or near-real-time synthesis without compromising on quality. These innovations make Neural2 Voices suitable for large-scale commercial applications, where both efficiency and quality are paramount. In this project, Neural2 Voices were employed to synthesize Danish speech from translated text segments. The specific voice utilized, "da-DK-Neural2-D", was chosen as it was one of the only semi-natural sounding danish voices.

2.7 Voice Conversion with Seed-VC

Seed-VC is an advanced voice conversion framework that employs diffusion probabilistic models, pre-trained feature extractors, and robust alignment mechanisms to achieve high-fidelity, zero-shot speech transformation. The system is designed to modify speech to match a target speaker's vocal characteristics while preserving linguistic content, prosody, and emotional nuances.

2.7.1 Feature Extraction

At the core of Seed-VC are pre-trained models such as Whisper, HuBERT, and Wav2Vec2, each serving a critical role in extracting semantic and prosodic features from speech signals. These models generate content embeddings that encode linguistic and prosodic information, ensuring high semantic fidelity during voice conversion. Their pre-training on diverse datasets ensures robust generalization even in zero-shot settings. While there is no paper on Seed-VC, it is not possible to delve deep into the structure of the program, however some clues of it's architecture can be found in the Github repository. (14)

2.7.2 Diffusion Probabilistic Models

The generative backbone of Seed-VC is a diffusion probabilistic model, a class of generative models inspired by thermodynamic processes. The diffusion model learns to reverse a gradual noise corruption process applied to data, eventually reconstructing the original input distribution.

Mathematically, a diffusion model defines a forward process that incrementally adds Gaussian noise to an input x_0 over T steps:

$$q(x_t|x_{t-1}) = \mathcal{N}(x_t; \sqrt{1-\beta_t}x_{t-1}, \beta_t I)$$

where x_t is the noisy version of the input at time step t and β_t is a noise schedule parameter. is the noisy version of the input at time step t and β_t is a noise schedule parameter. The reverse

process attempts to remove noise step by step using a score network s_θ trained to estimate the gradient of the data distribution:

$$p_\theta(x_{t-1}|x_t) = \mathcal{N}(x_{t-1}; \mu_\theta(x_t, t), \sigma_t^2 I)$$

This iterative denoising allows the generation of high-quality mel-spectrograms from a latent representation. (16)

2.7.3 Temporal Alignment and Length Regulation

Temporal consistency is critical in voice conversion to ensure the transformed speech matches the rhythm and timing of the source audio. Seed-VC employs a length regulator, which adjusts the duration of phonetic units in the generated mel-spectrogram. This ensures that speech timing is preserved, enhancing the naturalness and synchrony of the output. This is a very important aspect for this project, as this feature is used to shift the synthesized audio segment's duration to more closely match the original audio. Here, to further ensure natural voice, the minimum factor of `length_adjust` is set to 0.75, and the maximum is set to 1.25. This ensures that if the generated audio for any reason is quite shorter/longer than the original, then the audio length is adjusted to minimize as much error as possible. Errors in this area do not propagate, considering the different audio segments are overlayed on a silent track based on the starting timestamps, which do not change. In worst case, if a sentence takes too long, the next sentence will be uttered in overlap, and in that way the error does not propagate.

2.7.4 Mel-Spectrogram Generation

A **mel-spectrogram** is a time-frequency representation where frequency bins are mapped to the mel scale, reflecting human auditory perception. It is computed by applying a Short-Time Fourier Transform (STFT) followed by a mel filterbank transformation, producing a compressed frequency representation suitable for speech synthesis. (15)

2.7.5 Vocoder and Waveform Reconstruction

To convert mel-spectrograms into audio waveforms, Seed-VC integrates **BigVGAN**. BigVGAN is an advanced neural vocoder designed to synthesize high-fidelity audio waveforms from input features like mel spectrograms. It builds upon the foundational Generative Adversarial Network (GAN) framework that uses the well known Discriminator-Generator structure (8), but introduces several key innovations to enhance audio quality and generalization capabilities.

BigVGAN introduces periodic activation functions designed to better model periodic waveform structures in speech data. Instead of standard ReLU or LeakyReLU, BigVGAN uses a sinusoidal nonlinearity inspired by the periodic nature of audio signals, called the *Snake* function:

$$f_\alpha(x) = x + \frac{1}{\alpha}\sin^2(\alpha x)$$

where α is a learnable parameter.. This periodic activation helps the network represent the inherent periodic structure of audio waveforms more effectively.

Furthermore, BigVGAN introduces anti-aliasing. A quote from the paper states "The Snake activations provide the required periodic inductive bias for modeling raw waveform, but it can produce arbitrary high frequency details for continuous-time signals that can not be represented by the discrete-time output of the network, 2 which can lead to aliasing artifacts. This side effect can be suppressed by applying a low-pass filter. The anti-aliased nonlinearity operates by up-sampling the signal 2× along time dimension, applying the Snake activation, then downsampling the signal by 2×" (6)

2.7.6 Zero-Shot Voice Conversion

A defining feature of Seed-VC is its zero-shot voice conversion capability. Without requiring fine-tuning for new speakers, the model generalizes across unseen voices by relying on pre-trained feature extractors and diffusion priors. This property makes it particularly suitable for automatic dubbing systems.

2.7.7 How it's used

All audio segments from each speaker based on GPT-4's diarization is collected into a singular audio file. Then, Seed-VC loops through each synthesized segment, and uses the classified speaker's audio file as reference. Then, `length_adjust` is used as explained earlier. The chosen parameters are:

- `difusion_steps = 25`
- `inference_cfg_rate = 0.7`

2.8 Autodubbing Evaluation Metrics

As there currently are no objective metrics that measure the quality of dubbing systems exist, it is difficult to evaluate the outputs of this system. Using quantitative metrics such as translation, transcription and diarization accuracy is not enough to evaluate the quality and *feel* of the outputted audio. This is why this project uses a mix of quantitative and qualitative metrics to try to evaluate the results more thoroughly. The quantitative metrics used are simple accuracy scores, calculated as:

$$acc(S) = \frac{\sum_i^n p(s_i)}{n}$$

Where $s_i..s_n$ are all segments in S, and p is a predicate function that returns 1 if the predicate is fullfilled and 0 otherwise. Three predicate functions are used:

- $p_{transcrib}$: Segment is transribed correctly
- $p_{translat}$: Segment is translated correctly
- p_{diar} : Speaker identified correctly in segment

And thus we get transription, translation and diarization accuracies. In addition to the quantitative metrics, a Personal Rating metric is introduced to assess the quality of the video. Here the rating is on a scale of 1-10, where 1 is non-comprehensive, and 10 is a natural sounding output that preserves the message, emotion and delivery style of the original input.

3 Results

A series of videos were processed through the autodubbing pipeline, and the results are summarized in Figure 3.1. Furthermore, runtime benchmarks are observed in Figure **??**
Among the analyzed videos, one of the best outcomes was observed in the ”Job Interview” video. The Whisper model achieved near-perfect transcription accuracy, with the exception of a single segment where the desired output, ”Well. Tell me about yourself,” was mistakenly transcribed as ”We’ll tell you about yourself.” Despite this minor error, the rest of the pipeline performed seamlessly, resulting in a final video with accurate translations and voice conversion that preserved the essence of the source voices.

In the case of the Charlie Chaplin speech, the pipeline also delivered strong results. Every segment was transcribed, translated, and diarized correctly, which was expected given the simplicity of a single-speaker video. The output effectively captured Charlie Chaplin’s voice. However, the synthesized audio introduced white noise, likely originating from the original video. Additionally, towards the emotional crescendo at the end of the speech, the voice conversion model failed to capture the heightened expressiveness of the speaker due to limitations in the pipeline’s architecture. This limitation will be discussed in more detail in the discussion section.

The results from the Rick and Morty scene demonstrated accurate transcription and translation of the input audio. However, the presence of three speakers in the same conversation posed significant challenges for GPT-4’s diarization model. This led to errors in assigning voices, such as Jerry being incorrectly assigned Rick’s voice. Consequently, when Rick interrupts Jerry mid-sentence, the voice does not change as expected, detracting from the authenticity of the scene.

For the Spongebob scene, which involves a dialogue between only two characters, the models performed closer to expectations. The pipeline processed the scene almost flawlessly; however, a few segments were misclassified by GPT-4, causing Spongebob to momentarily sound like Patrick. These errors, though limited, underscore the need for improved diarization accuracy in simpler scenarios.

In the Ed, Edd, and Eddy scene, the pipeline successfully completed transcription, translation, and diarization without significant issues. However, the synthesized voices—particularly Double-D’s—lacked naturalness and authenticity. Additionally, non-verbal sounds, such as Ed’s laughter, were rendered unnaturally, highlighting a limitation of the text-to-speech model in conveying emotion during non-verbal sequences like ”hahaha.”

The Piers Morgan interview presented the most significant challenges for the pipeline. This video involved three participants frequently interrupting and speaking over one another, creating difficulties in transcription. Whisper struggled to accurately separate overlapping speech, resulting in segments that combined dialogue from multiple speakers or entirely omitted background sentences. The diarization task also proved problematic, as GPT-4 struggled to accurately assign voices among three speakers based solely on textual context clues. These challenges highlight limitations in handling complex multi-speaker scenarios.

The Kurzgesagt video demonstrated great results of the autodubbing system. The dubbed audio sounds much like the original, and even captures some of the background voice effects. It demonstrates natural sounding calm speech, where the voice closely resembles the original

Video	Total Segments	Transcription Accuracy	Translation Accuracy	Diarization Accuracy	Personal Rating (out of 10)
Job Interview	29	96%	100%	100%	10
Ed, Edd, & Eddie	17	100%	100%	100%	7
Spongebob	14	100%	100%	90%	7.5
Piers Morgan Heated Interview	33	76%	100%	82%	4.5
Charlie Chaplin The Great Dictator	26	100%	100%	100%	9
Rick and Morty	16	100%	100%	56%	6
Kurzgesagt	25	100%	92%	100%	9.5
Weather Forecast	11	100%	100%	100%	7

Figure 3.1: Results

Video	Video Length	Transcription Runtime	VC Runtime
Job Interview	1:45	7m 7s	21m 49s
Ed, Edd, & Eddie	1:17	6m 57s	19m 38s
Spongebob	00:53	5m 21s	17m 45s
Piers Morgan Heated Interview	2:14	11m 5s	32m 24s
Charlie Chaplin The Great Dictator	3:26	13m 21s	39m 32s
Rick And Morty	1:10	6m 13s	18m 12s
Kurzgesagt	2:09	7m 28s	36m 9s
Weather Forecast	1:08	6m 35s	17m 40s

Figure 3.2: Runtime Results

speaker. One small failure this particular example shows is a translation error. For all the other examples, the translations have on point. However, in this example, the word ”patron” in the context of supporting fans was translated by the DeepL model to ”lånere” or ”loaners” in English, which is a bit misleading. For this single mistake, the personal rating goes from 10 to 9.5.

For the weather forecast video, all models correctly transcribe, translate and diarize. Furthermore, the voice conversion model accurately captures the essence of the speakers, however, the audio/video synchronization is not optimal. There are multiple examples where audio includes speech, but the video shows the person not moving their mouth. This brings down the quality of the dubbed product, and is most likely due to the regrouping of segments done by Chat GPT-4 which will be discussed later.

4 Discussion

4.1 Technical Limitations

The results of the autodubbing pipeline reveal both promising potential and significant challenges that require further attention. This discussion explores key limitations observed during the project, highlighting areas for improvement and considerations for future work.

One of the most pressing issues is the sensitivity of audio-video synchronization to small inaccuracies in timestamping. Whisper's transcription accuracy directly influences the alignment between dubbed audio and video. Even slight misalignments in timestamps can lead to a noticeable desynchronization, which detracts from the viewing experience. This emphasizes the importance of precise segment timing throughout the pipeline. For this specific reason, a modified version of Whisper was used in the autodubbing pipeline, called stable-ts. This user-modified version of Whisper implements technologies such as silence suppression and a special type of Voice Activity Detector (VAD) called Silero VAD. These technologies help Whisper provide more accurate timestamps. However, even then, the timestamps never got to a perfect state, resulting in errors that range between 50 to 300 milliseconds, which can become quite noticeable on a film.

A critical limitation of the diarization approach lies in its reliance on GPT-4, which performs speaker classification based solely on textual context. While this approach can work reasonably well for videos with two speakers, it struggles significantly in multi-speaker scenarios. For example, videos with overlapping dialogues or three or more speakers frequently result in misclassifications, as the model lacks the audio-based context required to accurately distinguish between speakers. This issue has downstream consequences on the voice conversion step, where accurate diarization is vital for grouping sufficient speech samples from a single speaker. These errors in diarization cause the speaker to either sometimes change voices when they are not supposed to, or the other way around. A better choice for this could have been to use speaker recognition models to categorize each speaker, however, at the time of working on this project, no sufficient models that provided accurate results were found. Although there is no doubt that if such a model were to be trained well enough, it would make the autodubbing system perform far better at tasks that include more than two speakers.

Furthermore, the regrouping of segments done by GPT-4 was originally done to counteract a specific problem. Before regrouping, because of the small inaccuracies of Whisper timestamps, sometimes the next segment starts playing just as the current segment is about to end, resulting in a small time frame where two voices can be heard (even though it is the same speaker.) This regrouping however, introduced a new problem. Sometimes, there are small pauses between segments (1-2 seconds), and when regrouping is done between said segments, the silence portion disappears, resulting in outputs like the one seen in the weather forecast example (this can also sometimes be seen in the other examples.) This means that by introducing segment regrouping, we sacrifice synchronization quality to enhance the naturalness of the audio, such that the same speaker does not make two voices.

The computational demands of Seed-VC present another challenge, particularly on the hardware used during this project. Looking at the results in Figure **??**, we see how short videos can take anything from 17 to 39 minutes, making the pipeline impractical for longer videos. While this limitation did not impede the processing of shorter test cases, it raises concerns about the scalability of the system for real-world applications. Even the transcription runtimes are dispro-

portionately large, however significantly faster runtimes can be achieved when using some of the smaller Whisper models.

Whisper's transcription performance, as the first stage of the pipeline, is critical. In videos with overlapping speakers or significant background noise, transcription errors were observed. This stage is foundational, as inaccuracies propagate through the pipeline, affecting translation, diarization, and ultimately the dubbed output. Ensuring robust performance in this phase is vital to the overall effectiveness of the pipeline. In this project, the ”medium” sized Whisper model was used, however, if more computational power is available, then choosing the ”large” model is definitely a good idea if the goal is to minimize transcription error as much as possible.

Another significant challenge lies in processing expressive or emotional speech. Expressive elements, which are crucial to the emotional authenticity of the dubbed content, often sound unnatural or are entirely omitted in the synthesized audio. For example, laughter sequences synthesized by the text-to-speech system lacked the spontaneity and variability of natural human expression. This is mostly because the VC model uses the whole collection of the speaker's line as reference. When this happens, the emotion that is brought by the individual segment is lost over the collection of many. The reason why this approach was preferred, is because when using VC iteratively on each individual segment (without any other information), the same speaker can sometimes sound completely different than two seconds ago. This happens because the VC model sometimes has to ”guess” how the person sounds like based on a very short segment such as ”Oh” or ”Hello!”, which does not include enough information about the speaker's voice. It can be argued that a middle ground would have been best, where instead of using the whole collection of speaker lines, one could maybe have used a short window around the current segment as the target voice, such that the emotion around the current segment is not ”averaged out.”

A challenge that both Whisper and Seed-VC share is handling non-verbal sounds. For Whisper, some non-verbal sounds such as laughter can go completely undetected through the model, without being stored in the transcription. This might be due to the fact that some non-verbal noises are difficult to transcribe, which can be problematic in a text-based solution like the one this project uses. A new problem arises when the transcription model actually picks up the non-verbal speech; The text-to-speech model does not know how to properly make the noise naturaly like humans. This results in moments like the one in the Ed, Edd, and Eddie scene, where Ed sounds very robotic and unnatural when laughing. It could be argued that with a better VC approach, some of the original emotion would stay in tact, resulting in slightly more natural sounding audio sequences. However, this still highlights an important challenge of using a text-based solution for dubbing: the struggle of synthesizing non-verbal speech from text.

Seed-VC offers real-time voice conversion, and some of the previously mentioned challenges could have been solved if this technology was used, however, it requires a lot more computational power which was not available during the time the pipeline was being constructed. Furthermore, using real-time voice conversion would require rethinking the whole architecture of the pipeline, as some steps would be redundant (for example, diarization would not be needed if Seed-VC operated on the reference audio directly from the target audio in real-time)

Lastly, the lack of standardized evaluation metrics for dubbed content poses a broader challenge in assessing the system's performance. While transcription accuracy, translation quality, runtime, and diarization correctness can be quantitatively measured, evaluating the naturalness and emotional fidelity of synthesized voices remains subjective. The difficulty in objectively assessing these elements underscores the need for more comprehensive evaluation frameworks for autodubbing systems.

In summary, the autodubbing pipeline demonstrates notable strengths, such as its ability to handle simpler scenarios with minimal errors, but also faces significant limitations. Issues like timestamp precision, computational inefficiency, and the reliance on textual context for diarization highlight the challenges of developing a fully automated system. Addressing these limitations through improved models, multi-modal approaches, and hardware optimization will be essential for advancing the field of autodubbing and enhancing its applicability to real-world scenarios.

4.2 Ethical Considerations and Broader Impact

The use of AI for autodubbing, while innovative, raises several ethical concerns and potential societal impacts that need to be carefully addressed. Automated dubbing technologies could reduce the demand for human voice actors and audio engineers, particularly in smaller-scale productions or lower-budget media. While the technology can make localization more accessible and affordable, it also raises the risk of job displacement in creative industries. However, it is important to consider that such systems could also be used to assist professionals rather than replace them, potentially enabling faster turnaround and broader language coverage with human oversight.

Furthermore, even though autodubbing can make media more accessible across languages and cultures, there is a risk of cultural elements being diluted or misrepresented in the translation and synthesis process. Careful consideration must be given to preserving linguistic and cultural nuances, as well as accurately reflecting the original artistic intent.

4.3 Error Management and Human Oversight

While the system demonstrates significant capabilities, errors can still arise, particularly in transcription, translation, and diarization. The severity of these errors can range from minor timing discrepancies to complete voice misassignments. Therefore, the following strategies should be considered for error mitigation:

Human Review as Quality Control: Human oversight can serve as a final validation step, particularly for critical content. Automated metrics can be paired with manual review to ensure quality consistency.

- Confidence Scoring: Implementing confidence scores from each component (Whisper transcription confidence, GPT-4 certainty in diarization, etc.) can help prioritize which segments require human review.
- Fallback Mechanisms: When confidence scores fall below a threshold, the system could fall back to simpler alternatives or flag the output for manual correction.
- Data Augmentation and Fine-Tuning: Customizing models with domain-specific datasets could improve performance, especially for complex multi-speaker environments.

It is definitely feasible for errors to be corrected when human oversight is available, especially since the system works in segments. Simply put: if critical errors occur, find the segment where the error takes place and manually correct the error. This is not definitive however, as some errors such as distorted voices or general errors in the VC models (for example unwanted background noise captured by VC) are hard to correct.

4.4 Future Work

Most other autodubbing works take extensive measures to ensure synchronization and temporal alignment, for example, by tweaking the NMT models to output text that is expected to be

uttered in the same length/way/shape as the original audio. In this project, this is not done, in fact the NMT part of the pipeline is completely unrestricted, and translates the same way as it would translate in any other task. The way synchronization is handled is by heavily relying on the start and end timestamps from Whisper, and adjusting the length of the synthesized audio based on this duration. For future work, it is recommended to either look at some of the methods used in the paper ”From Speech-to-Speech Translation to Automatic Dubbing” (1). Here we see a more mathematical approach to prosodic alignment, which could lead to a more fluid and natural-looking video output. Similar approaches where NMT holds the responsibility of length control can be found in the VideoDubber paper (11). Otherwise, something else that can be done is finding methods to optimize the timestamps of transcribed segments. Theoritically, if timestamps were close to 100% accurate, then natural sounding audio could be achieved even when using segments as small as a couple of words at a time. Furthermore, a different diarization method that is not textual-based is essential if the system is to handle multi-speaker videos. At the time of writing this project, no effective pre-trained speaker recognition models were to be found, which is why the approach of textual diarization was used. Another option could be using Voice Conversion in real-time if efficient methods for this are found. Using this method, there would be no need for diarization, as the reference audio would always be the audio that is currently running. Finally, video modification is not considered in this project, however it could be interesting to explore methods where mouth movements are automatically altered to match the corresponding speech/language. A paper by Dan Bigioi explores concepts such as Facial Animation Generation in the context of automatic dubbing and discusses challenges such as how to bridge the uncanny valley that results from current AI generated facial expression. (2).

5 Conclusion

The autodubbing pipeline constructed in this project integrates state-of-the-art AI technologies—Whisper for transcription, GPT-4 for textual diarization, DeepL for translation, Google Cloud Text-to-Speech for voice synthesis, and Seed-VC for voice conversion—to automate the dubbing process. The results demonstrate the feasibility of such a system, but they also expose significant challenges that need to be addressed for the pipeline to achieve practical applicability.

One of the primary strengths of the pipeline lies in its ability to process simple scenarios effectively. For example, in single-speaker videos like the Charlie Chaplin speech, the system demonstrated high transcription and diarization accuracy, producing convincing voice conversion outputs. However, even in these ideal conditions, limitations such as noise artifacts in synthesized audio and a lack of emotional expressiveness were observed. These issues underscore the need for models that can better capture and reproduce the nuanced features of human speech.

For multi-speaker and overlapping-dialogue scenarios, the pipeline's weaknesses become more apparent. GPT-4's reliance on textual context for diarization leads to misclassifications, particularly when more than two speakers are involved. This has downstream effects on voice conversion, where inadequate reference voices result in inconsistent or unnatural outputs. The reliance on text-only diarization highlights the necessity for multi-modal approaches that incorporate audio features for more accurate speaker identification.

The computational demands of the pipeline also present a significant challenge. Processing a 1-2 minute video currently requires 30-40 minutes, largely due to the time-intensive Seed-VC model. While the pipeline is functional for short test cases, this computational bottleneck limits scalability and usability in real-world applications. Addressing this limitation will require either hardware optimization or the integration of faster voice conversion techniques, such as real-time voice conversion.

The pipeline also struggled with non-verbal sounds like laughter, where both Whisper and the text-to-speech models failed to produce natural outputs. This limitation highlights a broader challenge in text-based autodubbing systems: the difficulty of synthesizing realistic non-verbal expressions. While better voice conversion techniques could mitigate this issue to some extent, the fundamental problem lies in the text-based architecture of the pipeline.

Finally, the lack of standardized evaluation metrics for dubbed content complicates the assessment of the system's performance. While transcription accuracy, translation quality, and diarization correctness provide measurable benchmarks, the subjective nature of evaluating voice naturalness and emotional fidelity remains a challenge. Developing more robust evaluation frameworks will be critical for advancing the field of autodubbing.

In conclusion, this project demonstrates that an autodubbing pipeline integrating diverse AI models can generate meaningful results, particularly for simpler scenarios. However, challenges such as timestamp precision, computational inefficiency, and the limitations of text-based diarization and voice synthesis highlight areas for improvement. Future research should focus on multi-modal approaches, enhanced voice conversion techniques, and the development of standardized evaluation methods to push the boundaries of autodubbing systems. Addressing these issues will bring the field closer to creating scalable, high-quality solutions for global media localization.

Bibliography

[1] Amazon. From speech-to-speech translation to automatic dubbing, 2020. URL https://aclanthology.org/2020.iwslt-1.31.pdf.

[2] D. Bigioi. Multilingual video dubbing—a technology review and current challenges, 2023. URL https://www.researchgate.net/publication/374186615_Multilingual_video_dubbing-a_technology_review_and_current_challenges.

[3] G. Cloud. Text-to-speech documentation, 2024. URL https://cloud.google.com/text-to-speech/docs/voice-types#neural2_voices.

[4] DeepL. How does deepl work?, 2021. URL https://www.deepl.com/en/blog/how-does-deepl-work.

[5] DZ. Attention - what is it and how it works. 2024. URL https://dzdata.medium.com/attention-what-is-it-and-how-it-works-9ee0dcb97e53.

[6] S. gil Lee Wei Ping Boris Ginsburg2 Bryan Catanzaro2 Sungroh Yoon1. Bigvgan: A universal neural vocoder with large-scale training, 2023.

[7] Google. Attention is all you need, 2023. URL https://arxiv.org/pdf/1706.03762.

[8] M. M. B. X. D. W.-F. S. O. A. C. Y. B. Ian J. Goodfellow, Jean Pouget-Abadie. General adversial networks, 2014. URL https://arxiv.org/pdf/1406.2661.

[9] jianfch. Stable-ts whisper for precise timestamps, 2024. URL https://github.com/jianfch/stable-ts.

[10] M. I. Kamaluddin. Accuracy analysis of deepl: Breakthroughs in machine translation technology, 2024. URL https://www.researchgate.net/publication/381958486_Accuracy_Analysis_of_DeepL_Breakthroughs_in_Machine_Translation_Technology.

[11] Microsoft. Videodubber: Machine translation with speech-aware length control for video dubbing, 2023. URL https://arxiv.org/pdf/2211.16934.

[12] OpenAI. Robust speech recognition via large-scale weak supervision, 2022. URL https://cdn.openai.com/papers/whisper.pdf.

[13] OpenAI. Technical capabilities and use cases for gpt-4. 2023. URL https://openai.com/research/gpt-4.

[14] Plachtaa. Seed-vc repository, 2024. URL https://github.com/Plachtaa/seed-vc.

[15] L. Roberts. Understanding the mel spectrogram, 2020. URL https://medium.com/analytics-vidhya/understanding-the-mel-spectrogram-fca2afa2ce53.

[16] SuperAnnotate. Introduction to diffusion models for machine learning, 2024. URL https://www.superannotate.com/blog/diffusion-models.

[17] A. van den Oord, S. Dieleman, and H. Z. et al. Wavenet: A generative model for raw audio, 2016. URL https://arxiv.org/abs/1609.03499.