NATIONAL RESEARCH UNIVERSITY
HIGHER SCHOOL OF ECONOMICS

Faculty of Computer Science
Bachelor's Programme "Data Science and Business Analytics"

# Programming Project Report on the Topic:
# Software for Identification of a Speaking Person in a Video

**Fulfilled by**:
Student of the Group БПАД212
Kyoseli Timur Gyokkhanovich

**25/05/2024**

*(signature)* *(date)*

**Assessed by the Project Supervisor:**

Ignatov Andrei Dmitrievich
Professor, Research Assistant
Faculty of Computer Science, HSE University

**25/05/2024**

*(signature)* *(date)*

Zachinyaeva Mariya Alekseevna
Expert CIPE of Entrepreneurship
Center of Internships, Projects, and Entrepreneurship

**25/05/2024**

*(signature)* *(date)*

**Moscow 2024**

## Table of Contents

# Abstract

The project involves developing software to identify the speaker based on audio and video information and build captions on the video where each speech is assigned to a speaking person. It is assumed that the result of the program will be a distinction of speech by people in the video, as well as some analysis of the result.

# Аннотация

Данный программный проект подразумевает разработку программного обеспечения для идентификации спикера на основе аудио и видео информации и построения субтитров для видео, где каждая речь подписывается для говорящего человека. Предполагается, что результатом программы будет различие монологов по лицам на видео, а также некоторый анализ результата.

**Keywords:** ***voice recognition, face recognition, caption building, NLP, transcription, diarization***

# Introduction

There is a plethora of situations when people cannot hear the voices on the video (e.g., hearing impairment, noisy environment) but still want to understand what people are saying on the screen of their smartphones or computers. One of the most popular video sources is YouTube, and it was introduced in 2007 and since then has refined its caption building algorithm. Today it can be considered one of the best subtitle generators, however for the purpose of efficacy, it doesn't recognize the talking person, only recognizing his/her speech. So, our program is designed to fill that gap and build a more advanced caption builder using open-source libraries, such as OpenCV and face_recongnition.

Initially, we have been testing the software only on the Russian YouTube/VK Video show "Loud Question." In each episode, there are four main hosts and one guest. During the show, they put on the headphones with loud music on and the guest reads the question and must explain the answer to the hosts while shouting or using their hands. We used a 15-minute sample from one of the episode and ran two tests, one with plain diarization, which used identified speakers by itself, and then with face recognition to get the real or most possible speaker based on the person in the frame while the speech. As a result, just diarization yielded worse results than with face recognition, however the latter required more time to compute.

# Literature Review

To the best of our knowledge, there were no publicly available projects which used WhispeX and Face recognition in speaker diarization. Transcribing is an extremely popular job, especially among journalists. As a consequence, there are many free and pricy tools that are available online. Some of which are: Podsqueeze, Otter.ai, Transkriptor, Kapwing and Trint. Since there are also a lot of companies which offer human transcription, we will only compare the online tools.

| | Is it free? | Has own environment? | Has browser add-on? | Has a phone app? | Has a time imit, or usage limit on ranscription? | as multiple language support? |
|---|---|---|---|---|---|---|
| Podsqueeze | No, $60/user (7 files per month) | Yes | Yes | No | Yes | Yes |
| Otter.ai | No, except Basic free plan. | Yes | Yes, Chrome only. | Yes | Yes | Yes |
| Transkriptor | No | Yes | Yes | No | Yes | Yes |
| Kapwing | No | Yes | No | No | Yes | Yes |
| Trint | 80/month (7 files per month) | Yes | No | Yes | Yes | Yes |
| Our software | Yes | No | No | No | No | Yes |

**Table 1. Popular company's diarization compassion.**

Therefore, the market for the video and audio transcript is extremely competitive, and many companies make their products as convenient as possible, such as making a phone application or browser add-on. Unquestionably, in order to attract the most customers the product needs to be as convenient in use as possible, but why not save the money on paying the rent for the servers and data centres by providing a program which uses the customer's computer power.

Some companies have gone even further and provide software which automatically makes note during corporate meetings, additionally it makes a live transcript of the meetings and even provides a summarisation after the call. The pricing is generous compared to other companies' products: it has four different plans, starting with Basic free, that gives users AI meeting assistant which records, transcribes, capture slides and more but only with 30 minutes per conversation, 300 monthly transcription minutes. The Pro plan is $16.99/month which gives additional features, such as advanced search, export, playback, 1200 monthly transcript minutes with 90 minutes per conversation and let import and transcribe 10 audio and video files per month.

Other companies have created comprehensive tools that provide all sorts of instruments for not just transcription but content creating. Kapwing is a multitool that comprises a plethora of tools for video and image processing, e.g. AI subtitle generator, Audio enhancement and sophisticated video and image handling environment. The pricing varies from free basic plan to $50/month business plan with up to 900 minutes of auto-subtitle videos, 900 minutes of video translation and more.

While all the companies place an upper bound on the length of the video or total time used to transcribe the video, the software we develop will not have that bottleneck, although long and high quality videos files might not work as fast as short and medium quality ones, so there might be hardware limitations.

Some studies were also done on speaker diarization with face recognition. Although we know the number of speakers beforehand and know 4 out of 5 faces, the studies have proposed several options such as using whole body or only part of the bodies to detect a person and save the file and use it to reference and distinguish from other new speakers. Additionally, in our case, all people sit around the table, so only the upper parts of the body visible and used in identification. One study also used lip activity, and face size (the person with the larger face on the frame is likely to be the speaker)[3] to more accurately identify the correct speaker, the precision of their self-made model is 76% with 75% recall.

# Framework specificities

To identify and diarize the words said, we used the OpenAI’s Whisper[2], specifically its modification, WhisperX[1] written for Python, which supports word-level timestamps and speaker diarization. We applied this technology to identify words and get the time stamps of sentences, which were later used in person identification. Each timestamp has the beginning time and the end time. Whisper is an ASR model developed by OpenAI, trained on a large dataset of diverse audio. It produces very accurate speeches and transcriptions with time stamps for the whole speech and for each word, however sometimes they can be inaccurate and miss by several seconds. OpenAI's whisper does not natively support batching. WhisperX pipeline consists of several parts.

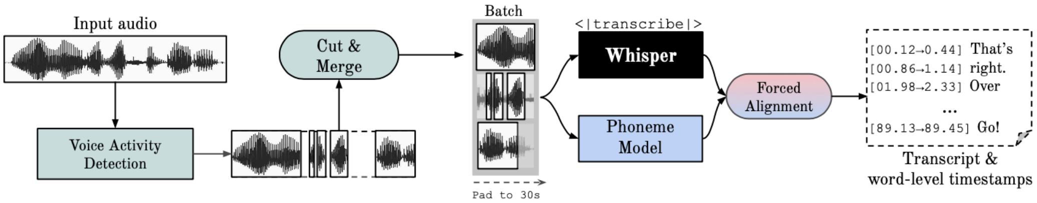

Figure 1. WhisperX pipeline. Source: https://github.com/m-bain/WhisperX

Firstly, it takes the audio file and uses a Voice activity detection (VAD), which refers to the process of identifying regions within an audio stream that contain speech, basically presegmenting the audio. Then it is clipped and blended into approximately 30-second input chunks with boundaries that correspond to minimally active speech sections. The generated chunks are then transcribed concurrently with OpenAI's Whisper and forcibly aligned with a phoneme recognition algorithm to get accurate word-level timestamps at high throughput.

To identify people, we used an open source face recognition library. First, we prepared four folders with images of the same people in each, in our case these were Anton Shastun, Arseniy Popov, Sergei Matvienko and Dmitriy Pozov. Each folder contained 2 to 3 images, as further increase could increase the runtime of the code, and such number was enough to produce applicable results. These images were checked for any similarities with the person on the current frame and then the name of the photo, which matched the current person on the frame, was assigned as the speaker.

After getting the timestamps, we used the face recognition on each part of the video and looked for the person who was recognized the most in each timestamp and the model used him as the source of the phrase/sentence. This method of identifying is suitable, since during the TV show, most of the time the camera was pointing at the person who was saying the phrase, thus giving a good chance of making a good prediction on the speaker of the current speech.

# Software specificities

All code was run in Colaboratory by Google. Since face recognition required GPU instead of CPU power, the available cloud characteristics are: NVIDIA Tesla T4 GPU with 15 GB of RAM with Driver Version: 535.104.05 and CUDA Version: 12.2. CPU characteristics are: Intel(R) Xeon(R) CPU 2.30GHz with x86-64 architecture. Additionally, we had 12.7 GB of RAM and 78.2 GB on Disk.

# Model assessment

There are two metrics that were used to assess the project accuracy. Firstly, we assessed the accuracy of identified words said by people with Word Error Rate (WER).

WER = (S+D+I)/N, where S is the number of substitutions, D is the number of deletions, I is the number of insertions and N is the total number of words in the reference.

We calculated WER for each speech and calculated the average WER at the end. Secondly, we assessed the accuracy of identification of people with our method. The accuracy was calculated by giving 1 to the correctly identified person and 0 otherwise. At the end, the sum of 1s was divided by the total number of speeches in the video, calculated by the diarization software. The project revealed quite hopeful results. As the test data, we used a 15-minute sample from an episode with Garviil Gordeev.

# Final results

The study conducted on the accuracy of face recognition technology in a TV show setting revealed interesting findings. The Word Error Rate (WER) was found to be 0.009, indicating a high accuracy of 99.1%. However, when considering the accuracy of correctly identifying a person's face, the percentage dropped to 60%.

In comparison to using only audio for diarization, the addition of face recognition technology significantly improved accuracy. This improvement can be attributed to the challenge of identifying speakers when multiple people are talking simultaneously, a common occurrence in the TV show analyzed. The study found that when the camera focused on the speaker, the accuracy of face recognition doubled, supporting the assumption that the camera typically captures the speaker.

However, the study also identified limitations in the face recognition technology. There were instances where the camera captured the entire table with all individuals present, leading to the model's inability to recognize any faces. This contributed to the accuracy being slightly above 50%.

It is noteworthy that despite the differences in technology used, both the face recognition and diarization models yielded identical WER results. This suggests that the source of word identification had a consistent level of accuracy across both approaches.

To conclude, while face recognition technology shows promise in improving speaker identification accuracy in complex settings like TV shows, there are still challenges to overcome, such as capturing all individuals within the frame for accurate recognition. Further advancements in technology and algorithm refinement may also address these limitations in the future.

| **Model** | **Speaker Identification Accuracy** | **Compute time** |
|---|---|---|
| **Face recognition + Diarization** | **60%** | **19 minutes** |
| **Diarization** | **37%** | **14 minutes** |
| **Model from [4]** | **76%** | **N/A** |

Table 2. Model comparison.

# Conclusion

The aim of the project was realized in building a software for diarizing the speech using video with audio. The software is capable of taking video inputs and producing captions with speaker identification. Although the precision may not be as high as expected, diarization accuracy was improved by integrating face recognition technology at the expense of extra time.

Over and above, there are numerous suggestions on how to enhance the software further. Lip recognition technology could be introduced to improve distinction between speakers. This would make it easy for this application to accurately identify speakers and perform well overall. Furthermore, finding smarter ways of identifying between speakers might also contribute to improving the project’s precision.

It is also suggested that the existing model should be developed as well as tested on those data used in this project to address possible language incompatibility issues. In situations where multiple people are predicted to speak at the same time, incorporating video analysis alongside audio inputs can help eliminate ambiguities related to speaker identification. Deciding primarily based on audio, while using video analysis as an additional tool, would also lead to more accurate predictions.

The code, photo samples and video samples are available at GitHub link[10] and Yandex.Disk[11].

# References

- WhisperX GitHub: https://github.com/m-bain/WhisperX
- Whisper OpenAI GitHun: https://github.com/openai/whisper
- Face recognition GitHub: https://github.com/ageitgey/face_recognition
- Elie El Khoury, Christine Sénac, and Philippe Joly. 2014. Audiovisual diarization of people in video content. Multimedia Tools Appl. 68, 3 (February 2014), 747–775. https://doi.org/10.1007/s11042-012-1080-6
- Podsqueeze, https://podsqueeze.com/
- Otter.ai, https://otter.ai/home
- Transkriptor, https://transkriptor.com/
- Kapwing, https://www.kapwing.com/
- Trint, https://trint.com/
- Project GitHub, https://github.com/nokumade/3rd_Year_Project