Final Report Title

Andy Kiuchi
Department of Computer Science
EPITA
14-16 Rue Voltaire, 94270 Le Kremlin-Bicêtre
andy.kiuchi@epita.fr

Martin Jarnier
Department of Computer Science
EPITA
14-16 Rue Voltaire, 94270 Le Kremlin-Bicêtre
Martin.Jarnier@epita.fr

Traore Nandy
Department of Computer Science
EPITA
14-16 Rue Voltaire, 94270 Le Kremlin-Bicêtre
Nandy.Traore@epita.fr

Abstract

We have decided to develop a virtual meeting transcription system because it seemed to us that few such tools existed and the demand was strong, especially in this new era where teleworking is omnipresent in a company's life. The goal is to transcribe as faithfully as possible for someone who was not present what happened in a meeting, what the main ideas were, and what the atmosphere was like in order to know which topics need to be addressed or taken more seriously than others. Imagine a specific case where, during a busy period in the IT department, some department heads or managers are unable to attend a client meeting. Having a summary of about one page for a meeting lasting over 2 hours is a significant time and efficiency saver, and it also facilitates the transmission of information between teams. Of course, this transmission is one-way because communication can only flow in one direction, but even outside of exceptional cases, this tool could inform employees about major decisions taken in meetings that are not accessible to them. This creates a sense of involvement among employees who feel included in the company's life. The main outlines of this project involve using audio from the meeting in order to perform NLP features to collect as much information as possible to transmit to the recipient, so a speech-to-text (STT) system will be used before this extraction.

1 Introduction

1.1 Who-did-what statement

- Andy Kiuchi was responsible for gathering datasets from the web, preprocessing the data, conducting evaluations, and working on the diarization file;
- Martin Jarnier trained the models, conducted the Summarization and the Sentiments Analysis.
- Traore Nandy State of the art: Analyze articles to understand the current detailed overview of the landscape of our project's specific domain.

1.2 Project presentation

Our project aimed to bridge the gap between the increasing demand for efficient remote communication tools and the current offerings in virtual meeting transcription and analysis. Our system stands out by not just transcribing the spoken word but by analysing and understanding the context, emotions, and key points discussed during a meeting. This multi use approach to meeting analysis allows for a richer understanding of the audio. It can streamline the workflow by providing executives and team members with concise and informative summaries of meetings, highlighting action points and key decisions. As we reflect on the goals set at the outset of this project and what we have achieved, We were unable to create a bot on the desired platforms that records conversations, mainly due to licensing issues.

1.3 Achievements

Our milestone reached during the project:

- **Summarization:** we use a summarization feature that condenses a meeting recordings into concise summaries.
- **Comprehensive Sentiment and Speaker Importance Analysis:** By incorporating Sentiment Analysis, our system can now analyze the sentiment of conversations and assess the contribution and importance of each speaker in the meeting. This allows for a deeper understanding of the meeting dynamics and identifies key contributors and potential areas of conflict or consensus.

2 Related works

This is our *detailed* overview of the landscape of our project's specific domain.

- **Speech to Text:** For speech-to-text, the main alternative is the article "Wav2Seq: Pre-training Speech-to-Text Encoder-Decoder Models Using Pseudo Languages" by Felix Wu, Kwangyoun Kim, Shinji Watanabe, Kyu Han, Ryan McDonald, Kilian Q. Weinberger, Yoav Artzi.

  As they present themselves, Their idea is to use Wav2Seq as an approach for self-learning to pre-train models for speech recognition and translation by pre-training both parts of encoder-decoder models for speech data. They induce a pseudo language as a compact discrete representation and formulate a self-supervised pseudo speech recognition task — transcribing audio inputs into pseudo subword sequences. This process stands on its own or can be applied as low-cost second-stage pre-training.

  Wav2seq has 3 tasks: automatic speech recognition (ASR), spoken named entity recognition (SNER), and speech-to-text translation (ST).

  Our difference is that we use Whisper, an ASR model from OpenAI. Unlike Wavseq, which is trained mainly with two datasets, LibriSpeech (1000 hours of audio) and LibriLight (60,000 hours of audio), Whisper is trained on 680,000 hours of noisy speech training data, including 96 other languages and 125,000 hours of English translation data. Also, Whisper uses voice activity detection (VAD) to segment the audio into speech segments before transcription. This segmentation allows WhisperX to efficiently process long audios

and improve transcription accuracy, whereas Wav2seq passes through HuBERT to reduce the size of audio and applies average pooling to reduce the sequence length, and then uses k-means clustering to discretize these hidden feature vectors as the length that Wav2seq can take is limited. Also, thanks to VAD, Whisper is faster than Wav2seq since WhisperX uses batch transcription, making it faster.

- **Summarization:** Most research on summarization has focused on formal written documents, which isn't ideal for summarizing meetings because the structure of a meeting isn't as linear as that of a text.

  For example, not only do you have multi-party conversations, but you can also jump from one subject to another. Then, one person may want to return to the previous subject, but the others who you think were your friends, might decide to completely ignore you and continue talking about the last chapter of Jujutsu Kaisen, even though you've already told them that you're ten chapters behind.

  Conversations during meetings are spontaneous, often unprepared interactions aimed at real-time collaboration. This often results in transcripts with low information density and a considerable amount of "noise" due to fragmented, repetitive, and wandering language. Meeting transcripts are typically much longer than documents used for text summarization, containing a lot of repetition, various speech styles, and the possibility of overlapping speech.

  One of the state-of-the-art articles is "A Hierarchical Network for Abstractive Meeting Summarization with Cross-Domain Pretraining" by Chenguang Zhu, Ruochen Xu, Michael Zeng, Xuedong Huang from the Microsoft Cognitive Services Research Group. The paper introduces a hierarchical neural network, HMNet, for generating abstract summaries of meetings from transcripts.

  The model utilizes a two-level structure to encode long transcripts and a role vector to represent each participant. It is pretrained on news summary data and fine-tuned on meeting summary data. In the end, we chose to use "bart-large-cnn-samsum" as it had the best benchmark performance for the size that we could run.

- **Sentiment Analysis:** There are many techniques for sentiment analysis, including lexicon-based approaches, rule-based methods, part-of-speech analysis, term position analysis, statistical techniques, supervised and unsupervised machine learning methods, as well as deep learning methods such as LSTM, CNN, RNN, DNN, DBN, BERT, and other hybrid approaches.

  One of the state-of-the-art approaches in sentiment analysis is "SenticNet 7: A Commonsense-based Neurosymbolic AI Framework for Explainable Sentiment Analysis" by Erik Cambria, Qian Liu, Sergio Decherchi, and Frank Xing, which utilizes neurosymbolic reasoning based on common sense for explainable sentiment analysis. They use auto-regressive language models to construct a symbolic representation like hierarchical knowledge graph, which is then used with linguistic patterns to extract text polarity in natural language. The advantages include being unsupervised (as it does not require training on labeled data), reproducible (since each reasoning step can be explicitly recorded and replicated), interpretable (as the process of generalizing input words and expressions into corresponding primitives is fully transparent), reliable, and explainable (because classification outputs are explicitly linked to emotions and input concepts). As a result, the model has a better understanding of the text and can capture nuances in the text with greater accuracy.

  In our case, we decided to use a lexicon-based approach as it is lighter to deploy and relatively fast. More precisely, we opted for the Vader sentiment analysis tool as it has been improved for speed and performance, reducing the time complexity from O(N 4) to O(N) and can be installed with a simple pip install.

3 Project

To make the project functional and truly useful for consumers, we mainly focused on adding new features that allow for the extraction of maximum information from meetings. Here we will outline the main functionalities that have ben implemented:

3.1 Speech to Text

This feature primarily serves as a tool to facilitate other operations, allowing for the extraction of text from audio recordings to carry out operations on it. For this task, we use the WhisperNet model, which is based on recurrent neural networks (RNNs), specifically designed for sequence processing such as speech. These networks are particularly effective in taking into account the temporal context of the data, which is crucial for speech recognition.

Before applying the WhisperNet model, we perform preprocessing on the audio recordings. This involves extracting the most relevant acoustic features, such as Mel-frequency cepstral coefficients (MFCC) or spectrograms. These techniques allow us to capture the key information contained in speech, thus facilitating the model's task in converting audio to text.

3.2 Summarization

The key feature of our project has been to efficiently summarize conversations from meetings. To achieve this, we have chosen the BART-large-CNN-SAMSUM model. This model stands out not only for the quality of its results, capable of concisely synthesizing complex exchanges, but also for its portability, a major asset for our application.

However we encountered certain difficulties, especially when the model had to process transcriptions generated by Whisper. These, although faithful to oral speech, were not always well-suited to summarization due to their format. Another challenge was measuring the accuracy of the summaries produced. Indeed, although the model generates grammatically correct texts in English, it was not always focused on the key points of the discussions.

The BART model, for Bidirectional and Auto-Regressive Transformers, relies on a cutting-edge architecture that combines both auto-regressive and bidirectional approaches for text understanding and generation, making it particularly suited to automatic summarization tasks. The choice of the large-CNN-SAMSUM variant of this model is motivated by its specialization in processing and summarizing dialogues, a specificity that aligns perfectly with the objectives of our project.

3.3 Sentiment Analysis

Sentiment analysis is an esential feature to gain an overall insight into the mood and interaction in a discourse or conversation. Indeed two reports can be completely different depending on the intonation of the text and the way the sentences are turns. For this, we use the VADER Sentiment library, which performs corpus analysis by detecting positive negative, and neutral terms.

In our approach, we apply this method at different scales: first on each sentence of each speaker, then on the entire conversation. This allows us to obtain the sentiments of each participant as well as an overview of the global sentiments of the conversation. This approach provides us with a maximum of contextual information, reducing potential errors in sentiment prediction.

By analyzing each sentence individually, we can detect emotional nuances and variations in tone between speakers. By aggregating this information over the entire conversation, we obtain an overview of emotional trends and group dynamics. This can be extremely useful for understanding participants' reactions, identifying points of friction and facilitating more effective communication.

3.4 Speaker Importance

The purpose of this feature was inspired by our own experiences in work groups. This is a tool to detect potential slackers present in meetings who do not actively participate in the conversation but try to make it appear otherwise.

Our goal was to detect the importance of each speaker in the meeting. For this, we set up a process that unfolds as follows:

We first detect the important words of the entire conversation regardless of the speaker. This is done by the en_core_web_sm model with the spaCy library. This works by using a weighting algorithm called TF-IDF (Term Frequency-Inverse Document Frequency). SpaCy calculates the TF-IDF score for each word in the text. TF-IDF is a statistical measure that evaluates the importance of a word in a document relative to the entire corpus. It considers both the frequency of the word in the document (TF) and its rarity in the entire corpus (IDF). We make sure not to take into account recurrent linking words ("and," "but," "or").

Subsequently, depending on the total length of the conversation, we establish a limit and a ratio of words pronounced and important words. If this ratio is below our threshold, then the speaker is considered "inactive" in the conversation and is flagged at the end of the analysis.

The limit of this method is the lenght of the discussion, we cant be accurate if we dont have enough data to compute accurate TF-IDF.

3.5 Topics Detection

We integrated topic modeling into our project for analyzing meetings which deals with textual transcriptions of exchanges. This process started with text preparation including the removal of speaker annotations and lemmatization to simplify words to their root We then used the LDA model from the Gensim library to identify the main themes discussed.
This technique proved particularly effective in distilling key elements of conversations offering a concise summary of the exchanged content which is ideal for quickly examining the topics discussed during meetings. However we encountered some obstacles particularly in adjusting the filtering parameters for extremes which if too restrictive can significantly reduce the vocabulary available for modeling.
Managing a limited or very homogeneous corpus also posed a problem as the elimination of common terms could lead to an insufficient corpus for LDA modeling. Moreover choosing the appropriate number of topics proved complex especially for short texts leading us to propose an approach that dynamically adjusts this number based on the length of the text. To overcome these challenges we tested different configurations of the no_below and no_above parameters to find a balance between preserving significant terms and eliminating words that are too frequent or rare. We also conducted an inspection of the corpus and dictionary after filtering to ensure there were sufficient data for effective modeling. Enriching our dataset was another successful strategy to improve the diversity of information processed. Despite these hurdles the integration of topic modeling greatly enhanced our ability to analyze and understand meeting discussions

4 Conclusion

We developed a meeting transcription system that integrates features such as speech-to-text conversion, summarization, sentiment analysis, and speaker importance detection. Our goal was to create a tool to enhance accessibility and understanding of remote meetings, particularly for those unable to attend in real-time.

We successfully implemented speech recognition, summarization, and sentiment analysis. However, we encountered challenges with the interface. We were unable to create a bot on the desired platforms to record conversations, mainly due to licensing issues with Teams.

In future iterations, the improvements could be:

- **Improve sentiment analysis:** by incorporating more complex sentiment detection methods.
- **Develop bots:** on different platforms by obtaining the necessary licenses to test the application in real-time.
- **Optimize our topic detection algorithms:** for shorter discussions to improve accuracy.
- **Question Detection and Answering:** Identify questions asked during a meeting and provide summaries of the answers. An idea to implement question detection could be using keyword spotting (e.g., starting with "What," "How," "Why") and utilize models trained on question answering tasks to offer concise answers or summaries based on the context surrounding the question. Or take example from the state of the art on sentiment analysis which utilizes neurosymbolic reasoning based on common sense for explainable sentiment analysis.

Citations

[ADER-Sentiment-Analysis, 2022] . [Cambria et al., 2022] . [Wu et al., 2022] . [Bain et al., 2023] [Zhu et al., 2020]

References

ADER-Sentiment-Analysis. Ader-sentiment-analysis. `https://github.com/username/repository`, 2022.

Max Bain, Jaesung Huh, Tengda Han, and Andrew Zisserman. Whisperx: Time-accurate speech transcription of long-form audio. 2023. URL `https://ar5iv.labs.arxiv.org/html/2303.00747`.

Erik Cambria, Qian Liu, Sergio Decherchi, Frank Xing, and Kenneth Kwok. Senticnet 7: A commonsense-based neurosymbolic ai framework for explainable sentiment analysis. 2022. URL `https://aclanthology.org/2022.lrec-1.408.pdf`.

Felix Wu, Kwangyoun Kim, Shinji Watanabe, Kyu Han, Ryan McDonald, Kilian Q. Weinberger, and Yoav Artzi. Wav2seq: Pre-training speech-to-text encoder-decoder models using pseudo languages. 2022. URL `https://ar5iv.labs.arxiv.org/html/2205.01086`.

Chenguang Zhu, Ruochen Xu, Michael Zeng, and Xuedong Huang. A hierarchical network for abstractive meeting summarization with cross-domain pretraining. 2020. URL `https://aclanthology.org/2020.findings-emnlp.19/`.

Appendix

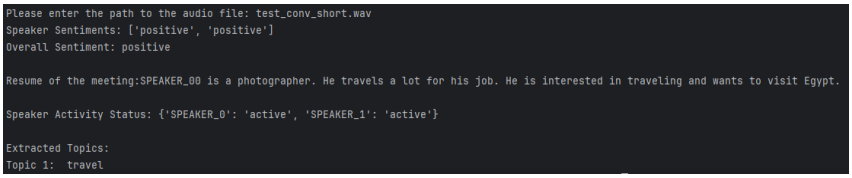

Figure 1: Our application working on a file.