Group 7: Russian to English Automatic Subtitling System

Darya Shyroka, Ladan Naimi, Shreyas Shankar, Mukhamedali Zhadigerov

Outline

Initial Idea

Approaches

Literature Review

Engineering Methods

Evaluation Metrics

Data

Plan

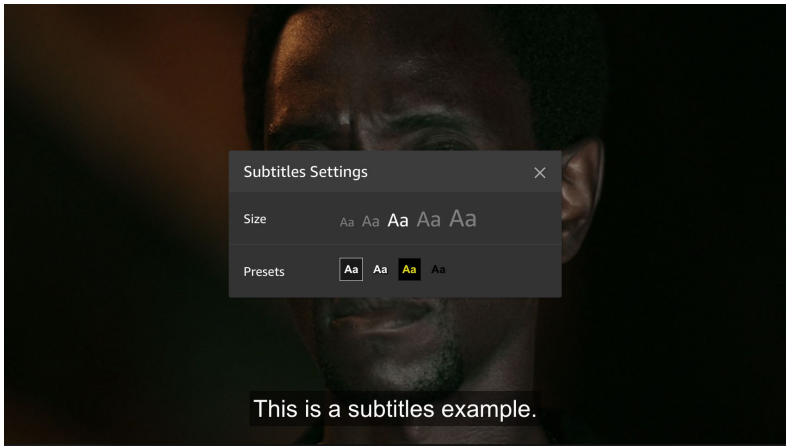
Challenges

Project Status

Acknowledgements

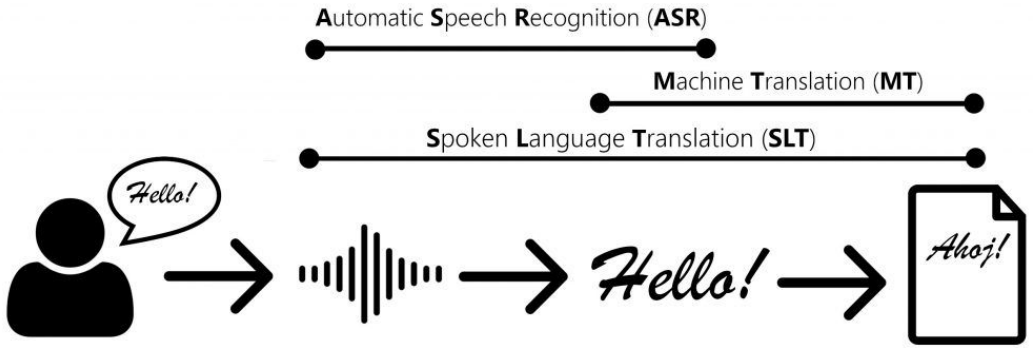
Initial Idea

- A model that could take Russian audio as input and output English text
- -> Use to develop an automatic subtitle generator that could take a video in Russian and add English subtitles onto it.

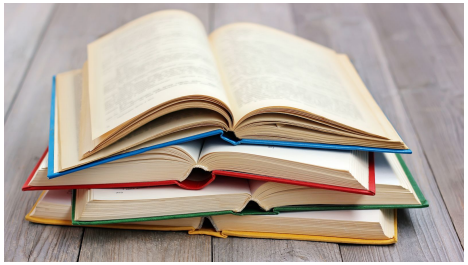


Approaches

- End-to-End: Spoken Language Translation (SLT) (Fairseq model):
  - Russian speech -> English text directly, without translation in between
  - Drawback: not a lot of data available (only 16 hours in CoVost dataset), Fairseq documentation is difficult to read
- Pipeline: ASR model + MT model
  - Separate steps with two different models - train separately, easier to tune hyperparameters, more data available, more pretrained models available
  - We ended up choosing this approach

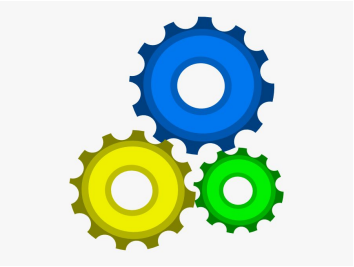


Literature Review



- The history of Russian ASR goes back to the Soviet Union with the KGB sponsoring research on speech recognition, leading to what's known today as the Speech Technology Centre (aka., SpeechPro), who are leaders in Russian ASR.
- Spontaneous conversational speech is difficult due to speech variability. It is a challenge for ASR in any language domain. SOTA WER for conversational English ASR is 8-14%.
- In 2017 they developed the current SOTA model with a WER of 16.4% involving a deep biLSTM for acoustic modelling and an RNN-based language model.
- **Wav2Vec2-Large-XLSR-53-Russian** is HuggingFace's pre-trained cross-lingual speech recognition model that is built upon the **XLSR Wav2Vec2** architecture, which is based on the encoder-decoder Transformer architecture. It reports a WER of 17.39% using Mozilla's CommonVoice-Russian test set.
- The most recent development is presented by Facebook AI Research's Speech-to-Text modelling system, which implements an end-to-end multilingual ST system using Transformers, trained on the CoVoST project datasets. However, the WER for Russian to English is around 31.4%.

Engineering Methods



- Google Colab Pro (more RAM, faster GPUs/TPUs)
- PyTorch framework, with extensive use of HuggingFace's Transformer and Datasets libraries
- Primary objective: investigate the performance of HuggingFace's cutting-edge ASR system, Wav2Vec2-Large-XLSR-53-Russian, a pre-trained cross-lingual speech recognition model
  - One half of a speech translation pipeline. The other half involves a pre-trained Russian text to English text translation model, which takes the output of the ASR model as its input
  - MT involves an encoder-decoder Transformer model, pre-trained from HuggingFace, called opus-mt-ru-en, trained on Russian subtitling data from OPUS

Evaluation

- Historically, phoneme error rate (PER)/character error rate (CER), sentence error rate (SER) and word error rate (WER) have all been used for Russian ASR evaluation, but WER is now the dominant standard for evaluation.
- Word Error Rate = (Substitutions + Insertions + Deletions) / Number of Words Spoken
- Substitutions are anytime a word gets replaced
  - For example: “twinkle” transcribed as “crinkle”
- Insertions are anytime a word gets added that wasn’t said
  - For example: “trailblazers” becomes “tray all blazers”
- Deletions are anytime a word is omitted from the transcript
  - For example: “get it done” becomes “get done”

Data

- At first, we wanted to use the OpenSLR dataset provided by Librispeech
- Due to format differences between OpenSLR and HuggingFace, we went with the Russian Common Voice dataset
- Common Voice: 111 hours of native speakers reading text in a quiet environment
- We ended up using only a subset of the data because of RAM and disk space limitations

Plan

- Develop a pipeline consisting of:
  - HuggingFace's XLSR-Wav2Vec2 to perform a basic ASR (Russian audio to Russian text), with the objective to improve the WER score compared to the pre-trained Russian-53 model
  - A pre-trained MT model: opus-mt-ru-en
- Assess if silver-standard transcriptions (ASR model output) do well in a MT model
- See whether the final output is at all usable after going through all of the steps

Challenges

a. Built from HuggingFace tutorial notebook using pre-trained Turkish ASR model on CommonVoice data
b. Compatibility of OpenSLR data with the HuggingFace Notebook
c. Configuration difference between Russian and Turkish data
d. Didn't have enough RAM to load the model, not enough disk space to load dataset
e. Needed to use TPU runtime (more disk space, 8 cores, 6GB each):
    1. Tried the model on single TPU core, but it only has 6GB RAM; the model is 12GB
    2. Tried parallelizing and using multiple cores
    3. A combination of CPU (model), TPU (runtime type)
    4. Not possible to access GPU in a TPU runtime
f. Switched back to GPU and used a fraction of the data

Project Status

- We have figured out the memory issues, and tried to train the model
- We could not make it train, because our datasets (namely tensors) are of different lengths.

Acknowledgements

- Thanks to Peter for helping us during the project.
- Thanks to Muhammad for the opportunity to try building the model
- Thanks everyone