Project Proposal: On-device Speech Translation

Team Jobless
Jiyang Tang (jiyangta), Ran Ju (ranj), Tinglong Zhu (tinglonz), Xinyu Lu (xinyulu2)

1 Motivation

This project aims to develop on-device speech translation. Language barriers are a common obstacle faced by travelers around the world. In an increasingly globalized society, the ability to communicate with people from diverse linguistic backgrounds is crucial for social interaction, tourism, business, and emergency situations. Traditional phrasebooks and online translation services can be cumbersome, especially in areas with limited internet connectivity. On-device speech translation aims to bridge this gap. On-device machine learning has gained popularity due to its potential to enhance privacy, reduce latency, and operate without the need for a constant internet connection. Deploying machine learning models on user devices reduces the reliance on centralized servers, improving the user experience and alleviating concerns about data security and privacy. With on-device speech translation, users can have greater confidence in their privacy. Personal conversations and sensitive information are not transmitted to external servers, reducing the risk of data breaches or unauthorized access. What's more, in remote or rural areas where internet access is limited, on-device speech translation remains functional, ensuring that travelers can still communicate effectively. This enhances the inclusivity of the application and its usability in diverse environments. Additionally, by processing translations locally, the traveler's assistant application consumes fewer network resources, making it cost-effective and environmentally friendly. It also reduces the burden on centralized translation servers, ensuring more equitable access for users worldwide.

2 Task Definition and Problem Setup

Specifically, given a segment of audio in the source language, a speech translation system first transcribes the utterance into text in the source language and then translates it into text in the target language. There are two paradigms for such systems. Cascaded systems model two sequence transduction steps using separate deep learning models, while end-to-end systems perform both steps using a single encoder-decoder model. The major advantage of the latter is that they can avoid information loss and prevent error propagation between two steps. However, these systems could require additional computing resources for both training and inference compared to cascaded ones. As a result, we aimed to investigate the accuracy-performance trade-offs of these architectures and some on-device machine learning techniques in this project, which will be discussed in later sections. We plan to use ESPnet (Watanabe et al., 2018) to conduct experiments To simulate a resource-constrained inference environment, we restrict ourselves to using 1 Intel CPU core with about 2.5GHz clock speed and 2GB of RAM. In addition, this system requires a single-channel microphone with at least 16000 Hz sampling rate. The audio is first recorded into waveform format and then MFCC or FBank features are extracted before running the neural network. Since we aim to investigate different model compression techniques, some level of off-device training is required. Therefore, we would like to receive $150 AWS credits per person ($600 in total) for this purpose. In terms of the timeline, we plan on benchmarking one or two existing speech translation models using pre-trained checkpoints before Oct 13th. This includes a cascaded and an end-to-end system with similar model sizes. Then each member of the team will start working on one high-priority research question and should obtain some initial results before Oct 27th (see later sections for more details). Some of these research questions may require off-device training, but we should have a basic report by Nov 16th. It's also crucial to have a good idea of what hardware is required to run the final product. After that the team can start investigating additional research questions.

3 Research Questions and Experimental Design

One of the most noticeable challenges of deploying deep learning models on devices for real-time applications is their computation and storage overhead. Currently, there are two promising techniques to address this challenge, quantization and pruning. In this research, one of our research problems is "Can the application of quantization and pruning to deep learning models retain high translation accuracy while significantly reducing the model's size and computational demands?" Utilizing quantization, we would like to reduce the precision of the model's weights and then reduce the storage requirements without a huge decline in accuracy. At the same time, pruning eliminates the redundant or non-essential neurons. Combining these techniques can pave the way for lightweight, efficient systems suitable for real-time deployment on resource-constrained devices.

Besides, the choice of representation and interchange formats are pivotal for optimizing and deploying neural network models. Torch 2.0 with TorchScript and Open Neural Network Exchange (ONNX) are two primary methods. Our research object is "Which of Torchscript or ONNX offers a more robust, efficient framework for optimizing and deploying the speech recognition and translation models?" This comparison will give practical implications of choosing one over the other in terms of benchmarking.

Moreover, speech recognition relies heavily on the efficient representation and processing of audio signals. The subsampling rate of encoders and feature extraction methods play an important role in the process. A hypothesis for our project is that using an appropriate small encoder subsampling rate and Mel-Frequency Cepstral Coefficients (MFCC) feature sampling rate will not affect the accuracy and efficiency of the speech recognition model. We will conduct experiments with variations in the rates to verify our hypothesis. Encoder subsampling reduces the temporal resolution of the sequence in deep learning models, it allows the models to capture long-term dependencies effectively. MFCC represents the short-term power spectrum of sound, and the rate could have implications for capturing phonetic nuances. Exploring the interplay between these rates will help us discern an optimal balance where the model could both maintain high translation accuracy and operate efficiently.

Also, we would like to explore the effect of the Attention mechanism and the convolutional neural networks (CNNs). Both of the two techniques have profound success in a diverse range of tasks. However, their intrinsic differences in handling the trade-off of accuracy and resources of auto speech translation need a comprehensive exploration. This research proposes to evaluate and contrast the performance, parameter efficiency, and other benchmarking of these two architectures. Our preliminary hypothesis suggests that the attention mechanism might excel in accuracy while CNNs might be inherently more robust and parameter-efficient for tasks.

Dataset For the dataset, we plan to use CVSS (Jia et al., 2022) for experiments that require training. CVSS is a multilingual-to-English speech-to-speech translation corpus, with sentence-level speech and text label pairs. We will use the Spanish-to-English language pair from the CVSS-C subset where all the translation speeches are from a single speaker. To maintain consistency, we will select pre-trained checkpoints that are trained on this subset when training is not required.

Evaluation metric Referring to ASR and translation, it is pivotal to employ robust evaluation metrics that effectively encapsulate the quality of generated outputs efficiently. We would like to choose BLEU and F1 scores as our metrics. BLEU (Bilingual Evaluation Understudy) is a widely accepted metric in the translation community. It gauges the congruence of n-grams between machine-generated translations and their reference counterparts. However, BLEU focuses on attention and it can ignore fluency or semantic coherence. That's why we also need the F1 score. It offers a harmonic blend of precision and recall. Combined with BLEU, they can provide a balanced and comprehensive perspective on the performance of our ASR and translation systems, offering insights into our model.

Ablations For our ablation study, we will frame it from the following perspective:

- Baseline: This model includes all features and components. It serves as a reference point against which the performance of other models (where components have been removed) will be compared. Quantization: We will remove or modify the quantization component of the model. By comparing the model's performance against the baseline, we can assess the contribution of quantization to the overall model performance. Pruning: Similarly, we will modify or remove the pruning component to discern the role of pruning in our model's effectiveness. Inference engine: We will change our specific inference engine in the ablation study. The impact of it can be quantified by evaluating the performance and benchmarking change.

- Lower subsampling rate: We would like to conduct experiments with different subsampling rates and compare the performance and benchmarking of the models.
- Hardware: For this research, we will use 1 Intel CPU core with about 2.5GHz clock speed and 2GB of RAM as the hardware.

4 Related Work and Baselines

4.1 Related work

For speech translations (ST), there are mainly two approaches: cascaded ST system and end-to-end (E2E) ST systems (E2E-ST), we may try both approaches in this project. For the cascaded ST, the idea is to split the speech translation task into smaller and feasible sub-tasks: speech translation and machine translation (Xu et al., 2023). While E2E-ST aims to solve the speech translation problem with an E2E model. Both approaches have their problems. Cascaded systems have the issue of error accumulation while the E2E ST model is difficult to learn due to the cross-lingual and cross-modal mapping in a single model (Xu et al., 2023).

For cascaded systems, the latest systems are composed of Transformer/Transformer Variants (including Speech-Transformer, Conformer, self-supervised learned Transformer, like Wave2Vec, HuBERT, etc.) based ASR, and Transformer-based MT (since Transformer architecture outperforms all the other model architectures in sequence generation task) (Xu et al., 2023).

For E2E-ST, researchers usually use a multi-task framework to train the model, since the cross-lingual and cross-modal mapping make the training process much more challenging compared to training the sub-modules independently (Xu et al., 2023). There are several training strategies: decoupled decoder, decoupled encoder, and two-stream encoder. The decoupled decoder strategy reduces the modeling burden by adding an ASR decoder (which receives the encoder's outputs as inputs) to the original E2E training pipeline. For the decoupled encoder strategy, it adds an additional semantic encoder to the original pipeline. Other than directly encoding speech into semantic features, it eases the encoder's burden by first encoding speech into acoustic features and then encoding it into semantic features. Two-stream encoder approach adds a shared encoder to jointly train ST and MT. The shared encoder takes either output from a speech encoder or machine translation text encoder, and its outputs will be used to generate translated text. This aims to map the speech encoder's feature space and the text encoder's feature space into the same one, which makes it easier to train the whole E2E-ST system.

We'll use an E2E-ST and a cascaded ST system as our baseline. For the cascaded system, we will use pretrained ASR and MT systems. For the ASR, we will use a 12-layer Transformer-based hybrid CTC/attention framework (Watanabe et al., 2017), while for the MT, we will use a 6-layer Transformer with source text encoder and translation decoder (Inaguma et al., 2020). The E2E-ST system we use was trained on a decoupled decoder scheme mentioned above (since it's easy to implement and requires less computational resources to finetune the model) and was initialized with the ASR encoder and MT decoder weights (trained on ASR-MTL and MT-MTL approach) since its performance is better than training from scratch.

The BLEU score for the cascaded systems and E2E-ST are shown in Table 1.

Model	Fisher (test)	CallHome (evltest)
E2E-ST	50.86	19.36
Cascaded	42.16	19.82

Table 1: BLEU scores of the cascaded and end-to-end translation systems.

Recently, a lot of work has been done on ASR/MT compression (most of them are quantization and pruning metrics, such as (See et al., 2016)), but little work has been done on ST model compression. We'll explore the effect of quantization, pruning, and different subsampling rates on speech translation tasks.

5 Potential Challenges

Our devices have limited processing power compared to cloud servers, which can impact the speed and accuracy of speech translation. Our adjustment is to optimize the machine learning models and algorithms for efficient on-device processing. We can also implement various model quantization and pruning to reduce memory and computation requirements. Prioritize essential features and reduce resource-intensive processes. If things go smoothly, we will extend the project by comparing end-to-end architecture and cascaded architecture. We will explore which architecture achieves better performance after applying model quantization and pruning. We will benchmark the performance of each model and try to understand and find the best trade-off between the accuracy and latency of each architecture.

6 References

References

Hirofumi Inaguma, Shun Kiyono, Kevin Duh, Shigeki Karita, Nelson Enrique Yalta Soplin, Tomoki Hayashi, and Shinji Watanabe. 2020. Espnet-st: All-in-one speech translation toolkit. *arXiv preprint arXiv:2004.10234*.

Yeting Jia, Michelle Tadmor Ramanovich, Quan Wang, and Heiga Zen. 2022. Cvss corpus and massively multilingual speech-to-speech translation. In *International Conference on Language Resources and Evaluation*.

Abigail See, Minh-Thang Luong, and Christopher D Manning. 2016. Compression of neural machine translation models via pruning. *arXiv preprint arXiv:1606.09274*.

Shinji Watanabe, Takaaki Hori, Shigeki Karita, Tomoki Hayashi, Jiro Nishitoba, Yuya Unno, Nelson Enrique Yalta Soplin, Jahn Heymann, Matthew Wiesner, Nanxin Chen, Adithya Renduchintala, and Tsubasa Ochiai. 2018. Espnet: End-to-end speech processing toolkit. In *Interspeech 2018, 19th Annual Conference of the International Speech Communication Association, Hyderabad, India, 2-6 September 2018*, pages 2207–2211. ISCA.

Shinji Watanabe, Takaaki Hori, Suyoun Kim, John R Hershey, and Tomoki Hayashi. 2017. Hybrid ctc/attention architecture for end-to-end speech recognition. *IEEE Journal of Selected Topics in Signal Processing*, 11(8):1240–1253.

Chen Xu, Rong Ye, Qianqian Dong, Chengqi Zhao, Tom Ko, Mingxuan Wang, Tong Xiao, and Jingbo Zhu. 2023. Recent advances in direct speech-to-text translation. *arXiv preprint arXiv:2306.11646*.