LAB4: HOW DO YOU TURN THIS ON ON-DEVICE AUTOMATIC SPEECH RECOGNITION

Yi-Ting Yeh and Ting-Rui Chiang
Carnegie Mellon University
{yitingye,tingruic}@cs.cmu.edu

1 RELATED WORK

Karita et al. (2019) provide a comprehensive study on speech models based on Transformer and RNN architecture. They compare the performance of Transformer and RNN models on automatic speech recognition (ASR), speech translation (ST), and text-to-speech (TTS). The results show Transformer ASR models have significant performance gain over RNN-based models. They can even outperform the hybrid DNN-HMM models implemented in Kaldi. The authors provide the Transformer training tips they observed in the experiments and describe the training recipes implemented in ESPnet, which is the main toolkit we use in our project. Transformer's weaknesses such as the higher decoding complexity are also discussed, which will be our potential future direction.

Higuchi et al. (2020) propose using a masked language model (MLM) over a CTC ASR model for non-autoregressive decoding. During traning, they train the MLM jointly with the CTC model. During inference, they first use the CTC model to gegnerate a initial prediction where tokens with low confidence are masked. Afterward, the they use MLM model iteratively to complete the prediction. They show their non-autoregressive can be as accurate as auto-regressive models, while the computation time required is much less. Since the authors claim that this CTC with MLM model has been implemented in the espnet library, we may try it if time permits.

He et al. (2019) specialize a Transducer-based model for on mobile device. To improve the efficiency of inference, they utilize use the state caching techniques. They also parallelize the computation of the two components in the encoder. To reduce the memory usage, they linearly quantize the model without an explicit "zero point" offset. To utilize the prior knowledge about possible speech content, e.g. contacts, app names, they reweight the predictions with an weighted finite state transducer as a language model. They also augmentate their training data with text instances that involves text normalization. By combining the above techniques, they boost the performance on several benchmark dataset, and accelerate the speed by two times. We could consider including techniques in our project if time permits.

Lee et al. (2021) consider the on-demand layer pruning problem: training a ASR model and removing some of the layers without any fine-tuning. Authors uses singular vector canonical correlation analysis (SVCCA) (Raghu et al., 2017) to show the effect of two regularization methods stochastic depth Huang et al. (2016) and intermediate CTC Lee & Watanabe (2021) in the context pruning. Based on the analysis, they develop an iterative search method for pruning layers and inducing submodels which match the original performance. In our project, we could use the proposed method to prune our models and SVCCA would be a good tool for analyzing the results.

2 BASELINES

2.1 EXPERIMENTAL SETTINGS

Baseline models: We will use two different architecture as our baselines: (1) a transformer-based model and (2) a LSTM-based model. Specifically, we will use two trained model from ESPNet-Zoo [1]. The encoder and the decoder of the transformer-based model have 17 and 5 layers of multi-head attention modules respectively, while the encoder and the decoder of the LSTM-based model have

[1] `Shinji Watanabe/librispeech_asr_train_asr_transformer_e18_raw_bpe_sp_valid.acc.best` and `kamo-naoyuki/mini_an4_asr_train_raw_bpe_valid.acc.best`.

Model	WER	Latency (second)	Energy (watt)
Transformer - CTC	3.23	91.180	5.787
Transformer - AR	54.18	99.378	5.82
Transformer - Hybrid	4.33	103.448	5.778
LSTM - CTC	100	51.851	5.3789
LSTM - AR	100	10.265	5.786
LSTM - Hybrid	100	143.83	5.885

Table 1: Evaluation Results of Baselines

4 and 1 layers of LSTM modules respectively. Both of the two models are trained jointly with the CTC loss and an auto-regressive conditional LM loss. The computation of CTC loss involves only the outputs from the encoder, while the computation of the auto-regressive conditional LM loss involves the decoder. Since both of the loss functions are used, they can be used to predict text in three different models: (1) CTC model : Predict with greedy CTC decoding. (2) Auto-regressive (AR) model: Predict with the auto-regressive conditional LM. (3) Hybrid model: Both of CTC prediction and the auto-regressive prediction are used.

Dataset: We choose the clean split of LibriSpeech (Zhang et al., 2020). We will use the 100-hour clean training split to fine-tune our model, and test it on the clean testing set.

2.2 Evaluation

We will evaluate the accuracy and efficiency in different ways. For the accuracy, we will evaluate the word error rate (WER) using a server on the full testing dataset. For the efficiency and power consumption, we will estimate the average latency and power usage on Raspberry Pi 4 over a randomly sampled subset of the testing dataset. The results are in Table 1.

2.3 Discuss

The results are aligned with our hypothesis:

1. The LSTM-based model has WER close to 100%. A transformer-based model is required for reasonable performance.
2. Using the CTC mode for decoding is faster than using the auto-regressive LM mode. Moreover, the performance gap between the two modes is acceptable.

Therefore, we do not need to change our plan on our project.

References

Yanzhang He, Tara N Sainath, Rohit Prabhavalkar, Ian McGraw, Raziel Alvarez, Ding Zhao, David Rybach, Anjuli Kannan, Yonghui Wu, Ruoming Pang, et al. Streaming end-to-end speech recognition for mobile devices. In *ICASSP 2019-2019 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6381–6385. IEEE, 2019.

Yosuke Higuchi, Shinji Watanabe, Nanxin Chen, Tetsuji Ogawa, and Tetsunori Kobayashi. Mask ctc: Non-autoregressive end-to-end asr with ctc and mask predict. 2020.

Gao Huang, Yu Sun, Zhuang Liu, Daniel Sedra, and Kilian Q Weinberger. Deep networks with stochastic depth. In *European conference on computer vision*, pp. 646–661. Springer, 2016.

Shigeki Karita, Nanxin Chen, Tomoki Hayashi, Takaaki Hori, Hirofumi Inaguma, Ziyan Jiang, Masao Someki, Nelson Enrique Yalta Soplin, Ryuichi Yamamoto, Xiaofei Wang, et al. A comparative study on transformer vs rnn in speech applications. In *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, pp. 449–456. IEEE, 2019.

Jaesong Lee and Shinji Watanabe. Intermediate loss regularization for ctc-based speech recognition. In *ICASSP 2021-2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pp. 6224–6228. IEEE, 2021.

Jaesong Lee, Jingu Kang, and Shinji Watanabe. Layer pruning on demand with intermediate ctc. *arXiv preprint arXiv:2106.09216*, 2021.

Maithra Raghu, Justin Gilmer, Jason Yosinski, and Jascha Sohl-Dickstein. Svcca: Singular vector canonical correlation analysis for deep learning dynamics and interpretability. *arXiv preprint arXiv:1706.05806*, 2017.

Yu Zhang, James Qin, Daniel S Park, Wei Han, Chung-Cheng Chiu, Ruoming Pang, Quoc V Le, and Yonghui Wu. Pushing the limits of semi-supervised learning for automatic speech recognition. *arXiv preprint arXiv:2010.10504*, 2020.