VNU Journal of Science:
Computer Science and Communication Engineering

Journal homepage: http://www.jcsce.vnu.edu.vn/index.php/jcsce

Original Article

ASR - VLSP 2021: An Efficient Transformer-based Approach for Vietnamese ASR Task

Truong Tien Toan

Rikkeisoft, Handico Tower, Pham Hung, Nam Tu Liem, Hanoi, Vietnam

Received 28 December 2021
Revised 24 March 2022; Accepted 5 May 2022

Abstract: Various techniques have been applied to enhance automatic speech recognition during the last few years. Reaching auspicious performance in natural language processing makes Transformer architecture becoming the de facto standard in numerous domains. This paper first presents our effort to collect a 3000-hour Vietnamese speech corpus. After that, we introduce the system used for VLSP 2021 ASR task 2, which is based on the Transformer. Our simple method achieves a favorable syllable error rate of 6.72% and gets second place on the private test. Experimental results indicate that the proposed approach dominates traditional methods with lower syllable error rates on general-domain evaluation sets. Finally, we show that applying Vietnamese word segmentation on the label does not improve the efficiency of the ASR system.

Keywords: Vietnamese automatic speech recognition, Transformer.

1. Introduction

Speech recognition is a process that converts audio signals received from the microphone into a sequence of words. Unfortunately, acquiring and analyzing features of speech is not an easy task. Typical challenges researchers encounter when they solve the speech recognition problem are:

- Pronunciation speed is often chaotic, so it is unpredictable.
- The length of the output sequence is frequently changed.
- Each person has a unique voice expressed through the region, ethnicity, loudness, the intensity of sound, and timbre.
- Environmental noises and receiver noises significantly affect the recognition results.

Vietnamese is one of the most globally popular languages spoken by a hundred of million. Each Vietnamese syllable can be considered a combination of initial, final, and tone components. Unlike English, the tone is an important feature because Vietnamese is a tonal language. Many words have similar pronunciations but different meanings due to

* Corresponding author.
E-mail address: toantt@rikkeisoft.com

https://doi.org/10.25073/2588-1086/vnucsce.325

various tones, for example, me, mè, mé, mẻ, mẽ, mẹ. Furthermore, Vietnamese articulation is significantly influenced by regional characteristics. These things make solving the Vietnamese speech recognition problem more arduous.

This year, Vietnamese Language and Speech Processing (VLSP) continues to organize the 2021 ASR tasks to find the best Vietnamese ASR systems on two sets of evaluation data corresponding to two different tasks collected mainly from online lectures. The first task focuses on a complete pipeline construction of the ASR model from scratch, while the second deals with the spontaneous speech in distinct real-life scenarios, e.g., meeting conversation and lecture speech. The training data of the first task is fixed. However, participants can utilize all available data sources to develop their second task models without limitations.

Word segmentation is one of the critical tasks in solving Vietnamese language processing problems. Similar to some East Asian languages such as Chinese and Japanese, words in Vietnamese may include one or more syllables, so it is impossible to use white space as a separator like English.

This paper first presents the workflow to build the VLSP 2021 ASR system. Furthermore, we show that our approach outperforms traditional approaches in clean and noisy environments through many experiments. Our last contribution is considering the influence of the word segmentation procedure on Vietnamese speech recognition.

The organization of the paper is as follows. First, we examine related works and previous approaches in section 2. Then, the methodology and experimental results are described in section 3 and section 4, respectively. Finally, section 5 concludes our paper and provides some potential future works.

2. Related Works

From the innovation point of view, speech recognition incorporates a long history with a few waves of significant developments.

In the 2010s, Hidden Markov models (HMMs) are broadly utilized in numerous ASR systems. [1] proposed Kaldi speech recognition toolkit, which is based on HMMs. Kaldi is a fantastic framework for speech recognition that provides acceptable performance. In the 2019s, the Seq2Seq architecture dominated others in many natural language processing problems, including speech recognition tasks. [2] show that while Transformer-based acoustic models have superior performance with the supervised dataset alone, semi-supervision improves all models across architectures and loss functions and bridges much of the performance gaps between them. In addition, they reach a new state-of-the-art for end-to-end acoustic models decoded with an external language model in the standard supervised learning setting, and a new absolute state-of-the-art with semi-supervised training.[3] proposed methods that achieve a reliable word error rate by leveraging local context. Recently, [4] and [5] show that learning powerful representations from speech audio alone can reach significant improvement and outperform the best semi-supervised approaches while only using a little labeled data.

In the previous years, most of the Vietnamese ASR systems were based on hybrid architectures, e.g., [6-7]. These models are trained on thousands of hours of speech data and reach acceptable performances. In VLSP 2021, most approaches are based on end-to-end which dominates the traditional in many evaluations. End-to-end methods become de facto the standard in solving Vietnamese ASR problem [8].

3. Methodology

This section describes our effort to collect and label a large corpus that includes Vietnamese human speech in various conditions. In addition, we present effective data augmentation methods such as SpecAugment, background noises, and simulated reverberation. Finally, we propose a Transformer-based model trained with many advanced techniques to solve the Vietnamese ASR problem.

3.1. Data Preparation

Data preparation is the most crucial stage in AI systems building. Speech data has its typical characteristics compared to other data types. Therefore, building a speech recognition system faces many challenges, as mentioned in section 1. Training a highly reliable ASR model requires much data, including utterances in many real-world conditions. Commercial systems typically require thousands of hours of labeled audio. Because Vietnamese speech recognition public data is minimal, we collect meeting conversations, audio calls from differentsources on the internet and then perform a manual labelling process.

We first crawl audio from various sources such as YouTube, VnExpress, Facebook. The audio files are then segmented by a Voice Activity Detection (VAD) module. After that, all utterances are converted into standard type as the configuration is in table 1. Finally, hand-operated labeling is performed by 40 collaborators over three months. The total duration of the audio segments is approximately 2700 hours. Combined with the VLSP 2021 data, we have about 3000 hours of audio. Northern speech data accounts for about 60%.

Table 1. Audio configuration.

Parameters	Value
Sample rate	16000
Bit-depth	16
Channel	mono

Table 2. Our labeled data (hours)

Region	Duration
Northern	1600
Central	400
Southern	700
VLSP 2021	240
Total	2940

3.2. Proposed Pipeline

This part presents the introduced model, which contains five main components, e.g., Data Augmentation, Feature Extraction, Acoustic Model, Language Model, and Decoding.

3.2.1. Data Augmentation

Building a valuable ASR system requires enormous transcribed audio data, consisting of utterances in numerous noisy situations. However, data preparation for speech recognition is a costly process. In order to address this obstacle, we adopt multiple modern data augmentation methods in the paper. Data augmentation may be a standard procedure embraced to extend the amount of training data, avoid overfitting and improve the robustness of the model.

To simulate data in real-world environments, we utilize the *Room Impulse Response and Noise Database* [9], which includes 19.5 hours of simulated and actual room impulse responses, 0.9 hours of real isotropic noises, and 5.9 hours of point-source noises[1]. In addition to RIRs, we also randomly manipulate other speech signal transformations, including shifting time, changing pitch, and adjusting speed.

3.2.2. Feature Extraction

Feature extraction is one of the most important steps in generating every ASR system where particular features such as power, pitch, and vocal tract from a speech signal are handled. Accordingly, we extract the Mel-Filterbank feature, which is employed as the input for the acoustic model.

As a modern and straightforward augmentation method, SpecAugment [10] has been utilized to enhance the performance of many ASR systems. The augmentation rules comprise warping the features, masking blocks of frequency channels, and masking blocks of time steps. Thus, we manipulate SpecAugment for our speech recognition model to achieve better.

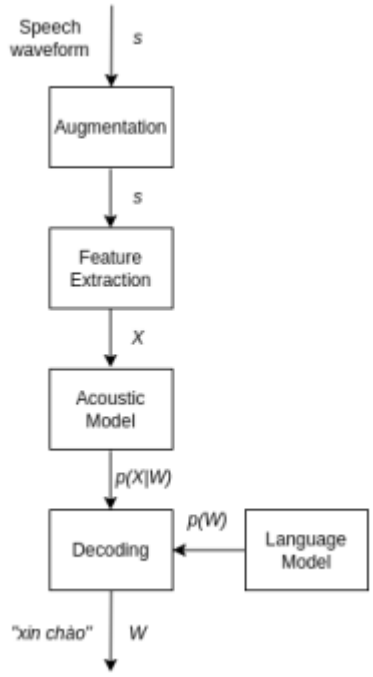

Figure 1. The proposed ASR system.

Table 3. Feature extraction configuration:

Parameters	Value
Frame length	25 ms
Frame shift	10 ms
Number of mel-bins	80
Sample frequency	16000
Window type	Hamming

3.2.3. Acoustic Model

The acoustic model works like the human ear, and its input is the feature of speech, X. The output is the conditional probability of X given the input word sequence W, P(X|W).

Recently, the acoustic model has been based chiefly on the encoder-decoder architectures. Our acoustic model is the Transformer [11] that is the most well-known encoder-decoder architecture. However, we apply a pre-norm version of Transformer where the normalization module in each block is performed before utilizing other modules instead of post-norm as initially. This method boosts up the recognition results and helps the model converge faster.

Practically, the Mel-Filter Banks feature of an utterance can be a sequence of vectors up to thousands of lengths. Using that as input for the acoustic model is time-consuming and computationally intensive. Therefore, it is necessary to have a front-end, usually a convolutional neural network with three to five layers, to diminish the length of the input feature.

3.2.4. Language Model

The language model simulates the human brain that stores information related to the knowledge of a language. Regular language models are n-gram, RNN, Transformer. In order to save the inference time, we use the n-gram language model for our commercial speech recognition system. We only leverage the label of training data for generating language models because these sentences are spoken texts that are more suitable for speech recognition than other formal texts such as newspapers and novels.

3.2.5. Decoding

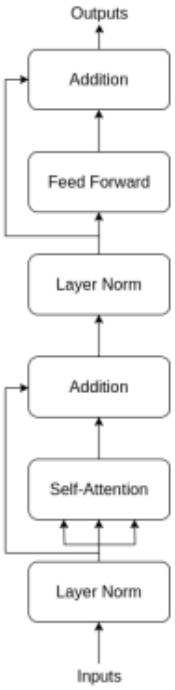

Figure 2. The pre-norm Transformer block.

The posteriors of the acoustic model and the perplexity of the language model are both leveraged to explore the best transcription of input utterance in the decoding stage. For detail, given the input signal X, we use beam search during decoding to find the best word sequence W^* of X. That can be computed by below equation:

$$W^* = \underset{W}{\operatorname{argmax}}((1-\alpha)\log P(X|W) + \alpha \log P(W) + \beta|W|)$$

Where W are all possible transcription of X, α is language model weight, β is the word insertion penalty which is a similar component in [12]. α and β are hyper-parameters that can be found through fine-tuning the development set. P(X|W) is the conditional probability of X given the sequence W, which is also the output of the acoustic model. P(W) is the probability of W in the corpus, which the language model computes.

4. Experiments

In this section, we describe our experiments and discuss the advantages and disadvantages of the system.

4.1. Evaluation Metrics

The quality of the models are evaluated by the Syllable Error Rate (SER) metric.

$$SER = \frac{S + D + I}{N} = \frac{S + D + I}{S + D + C}$$

Where,

- S is the number of substitutions.
- D is the number of deletions.
- I is the number of insertions.
- C is the number of correct syllables.
- N is the number of syllables in the reference (N = S + D + C).

4.2. Data

We use the Kaldi speech recognition toolkit [1] as a baseline to compare with the proposed approach. The collected corpus is utilized as the training set while the development set of ASR task 2 is manipulated as our development set. All hyperparameters are tuned using the development set, which includes 2.5 hours of labeled data.

In addition to the in-domain private test set sent by the organizers, we prepare two more test sets, including utterances in the general domain, to evaluate the performance of the models in actual conditions.

Because of deliberating the effectiveness of applying Vietnamese word segmentation on speech recognition, we generate another version of the training dataset where the VnCoreNLP toolkit [13] segments each transcription. The segmented version has a further post-processing step in the testing stage to eliminate the concatenation mark of syllables in a word, e.g., "tôi yêu lập_trình"

→ "tôi yêu lập trình". Both the segmented and unsegmented models are evaluated on the same evaluation sets for a fair comparison.

Table 4. Our datasets (hours):

Set	Duration
Train	2940
Dev	2.5
Private	2.2
Our clean	4.5
Our noise	5.5

4.3. Model Settings

According to the two versions of models, there are two

corpora where vocab and language models are trained. The vocab models are created by the SentencePiece toolkit [14] with 3000 sub-word units. The 4-gram language models are generated using the same vocab of the acoustic models by the KenLM toolkit [15].

The front-end is a 3-layers 1D convolution neural network with a kernel size of three and a 2-length striding window. The initialization parameters of the transformer model are presented in table 5. Our acoustic model's total number of trainable parameters is approximately 100 million.

4.4. Training Setup

In the training stage, the label smoothing cross-entropy criterion is applied. In addition, we use Adam optimizer with beta factor (0.9, 0.98). Moreover, some techniques such as gradient clipping, weight decay are also utilized to speed up the training process. All the parameters used in the training procedure are listed in table 6.

We use a powerful machine that contains 6 x RTX 3090, 24GB VRAM, 256GB of RAM, CPU Xeon E5 48 cores to train the models.

Table 5. Acoustic model configuration:

Parameters	Encoder	Decoder
Num layers	12	6
Dim	768	256
Num heads	4	4
Dim feed forward	3072	1024
Dropout	0.1	0.1

Table 6. Hyper-parameters of the training procedure:

Parameters	Value
Batch size	48
Smoothing factor	0.1
Warmup steps	100,000
Learning rate	0.001

4.5. Results

After training, we average the parameters of the top-10 checkpoints. This technique is almost always helpful because it helps create an ensemble model by averaging the parameters of the models over the epochs.

By fine-tuning the hyper-parameters on the development set, we figure out the best used for decoding as shown in table 7.

Table 7. Hyper-parameters tuning result:

Parameters	Best Value
α	0.2
β	0.4
Beam size	100

Table 8 compares the effectiveness of participant's methods on the private test set of ASR task 2. Although, we only use a single model without improving accuracy by ensemble multiple architectures together. Our simple approach still proved effective by taking second place on the leaderboard. That makes implementing the model in real-life applications free of resources and running time while ensuring good performance.

Table 8. SER on the private test set.

Team	SER
Lightning	4.17%
Ours	**6.72%**
VC-Tus	8.83%
LAB-914-ASR	9.88%
VB_ASR	13.19%
D2_Speech	14.09%
CHC-79	18.05%
DAL	18.99%
eve	28.60%

Table 9 shows the SER that our systems achieve on the general-domain evaluation sets. The Transformer-based method outperforms the baseline by a significant performance gap. The unsegmented model reaches a lower SER than the segmented version. The result shows that utilizing the Vietnamese word segmentation does not affect the recognition achievement.

Both the Transformer-based and Kaldi are trained on the same machine as mentioned in section 4.4. However, we perform inference procedures on another computer, including CPU Xeon E5 2.2 GHz and GPU GTX 1080 Ti. Table 10 presents the measurements of the models, where the last two rows indicate the processing time of the models on our evaluation sets. The table shows that our approach outperforms the Kaldi method in the GPU-based training and testing stages.

Table 9. SER on our general-domain test sets:

Method	Clean	Noise
Baseline (Kaldi)	6.0%	8.2%
Segmented	4.6%	7.5%
Unsegmented	**4.2%**	**7.2%**

Table 10. Performance on our 10-hour sets:

	Ours	Kaldi
Training	2 weeks	1 month
Inference (GPU)	30 minutes	40 minutes
Inference (CPU)	5 hours	2 hours

4.6. Drawbacks

Although the model gets many remarkable achievements, it still has limitations. Firstly, like many other systems, our best speech recognition system also makes mistakes when facing regional utterances as input. The reason is the lack of data collection in the central and southern regions compared to the north.

Secondly, the model only works well when dealing with utterances of less than 40 seconds in duration, and longer segmentations require a further preprocessing step to split them into shorter utterances.

Finally, our system cannot run as well as the conventional systems on low-resources which do not include GPUs.

5. Conclusion

5.1. Summary

We propose a practical Transformer-based approach to solve VLSP ASR shared task. Using a single model with start-of-the-art augmentation and training techniques outperform other methods by reaching a promising syllable error rate. In addition, we also show that applying word segmentation is not effective in solving speech recognition compared to other problems in natural language processing.

5.2. Future Work

We first collect more regional utterances to enhance the model's generalization for future improvements. Furthermore, we will manipulate advanced methods like [16] and [17] to reduce the inference time. Next, we can experiment with more state-of-the-art architectures such as Conformer [3] and ContextNet [18]. Finally, we utilize pre-trained embeddings for audio comprise wav2vec 2.0 [4] and HuBERT [5].

References

[1] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, Glembek, N. Goel, M. Hannemann, Motlicek, Y. Qian, P. Schwarz, The Kaldi Speech Recognition Toolkit, in: IEEE 2011 workshop on automatic speech recognition and understanding, no. CONF, IEEE Signal Processing Society, 2011.

[2] G. Synnaeve, Q. Xu, J. Kahn, T. Likhomanenko, Grave, V. Pratap, A. Sriram, V. Liptchinsky, Collobert, End-to-end Asr: From Supervised To Semi-Supervised Learning With Modern Architectures, International Conference on Machine Learning (ICML 2020).

[3] A. Gulati, J. Qin, C. C. Chiu, N. Parmar, Y. Zhang, J. Yu, W. Han, S. Wang, Z. Zhang, Y. Wu, Conformer: Convolution-augmented Transformer for speech recognition, Interspeech 2020, 2020.

[4] Baevski, H. Zhou, A. Mohamed, M. Auli, Wav2vec 2.0: A Framework For Self-Supervised Learning Of Speech Representations, NeurIPS 2020.

[5] W.-N. Hsu, B. Bolte, Y.-H. H. Tsai, K. Lakhotia, R. Salakhutdinov, A. Mohamed, Hubert: Self-Supervised Speech Representation Learning By Masked Prediction Of Hidden Units, IEEE/ACM Transactions on Audio, Speech, and Language Processing Vol. 29, 2021, pp. 3451–3460.

[6] Q. M. Nguyen, T. B. Nguyen, N. P. Pham, T. L. Nguyen, VAIS ASR: Building A Conversational Speech Recognition System Using Language Model Combination, Vietnamese Language and Speech Processing conference, 2018.

[7] Q. B. Nguyen, B. Q. Dam, V. H. Nguyen, Development of Zalo Vietnamese Conversational Speech Recognition, VLSP 2019.

[8] D. V. Hai, ASR Challenge: Vietnamese Automatic Speech Recognition, VLSP 2021,

[9] Szoke,¨ M. Skácel, L. Mosner,ˇ J. Paliesek, J. Cernockỳ, Building And Evaluation of A Real Room Impulse Response Dataset, IEEE Journal of Selected Topics in Signal Processing, Vol. 13, No. 4, 2019, pp. 863–876.

[10] D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, Q. V. Le, Specaugment: A Simple Data Augmentation Method for Automatic Speech Recognition, Interspeech 2019.

[11] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, Jones, A. N. Gomez, Ł. Kaiser, I. Polosukhin, Attention Is All You Need, In: Advances in

Neural Information Processing Systems, 2017, pp. 5998–6008.

[12] Y. Wu, M. Schuster, Z. Chen, Q. V. Le, Norouzi, W. Macherey, M. Krikun, Y. Cao, Gao, K. Macherey, Google's neural machine translation system: Bridging the gap between human and machine translation, arXiv:1609.08144.

[13] T. Vu, D. Q. Nguyen, D. Q. Nguyen, M. Dras, Johnson, VnCoreNLP: A Vietnamese natural language processing toolkit, Conference of the North American Chapter of the Association for Computational Linguistics, NAACL, 2018.

[14] T. Kudo, J. Richardson, Sentencepiece: A simple and language independent subword tokenizer and detokenizer for neural text processing, Conference on Empirical Methods in Natural Language Processing, EMNLP, 2018.

[15] Heafield, KenLM: Faster And Smaller Language Model Queries, in: Proceedings of the sixth workshop on statistical machine translation, 2011, pp. 187–197.

[16] Choromanski, V. Likhosherstov, D. Dohan, Song, A. Gane, T. Sarlos, P. Hawkins, Davis, A. Mohiuddin, L. Kaiser, Rethinking attention with performers, ICLR, 2021.

[17] Wu, F. Wu, T. Qi, Y. Huang, X. Xie, Fastformer: Additive Attention Can Be All You Need, arXiv:2108.09084.

[18] W. Han, Z. Zhang, Y. Zhang, J. Yu, C.-C. Chiu, J. Qin, A. Gulati, R. Pang, Y. Wu, Contextnet: Improving Convolutional Neural Networks for Automatic Speech Recognition with Global Context, Interspeech 2020.