Sentence-level Chinese-English Audio Fusion Based on Deep Learning

Shengxin Chen 31520221154194,[1] Timin Gao 31520221154201, [1] Yu Cai 31520221154192 [1]
Jiayan Lin 31520221154213, [1] Linhuang Xie 31520221154228 [1]

[1] School of Informatics

Abstract

Listening to English audio is a good way to practice English listening. Although translation subtitles are widely used, there are relatively few audio translation tools on the market. To the best of my knowledge, there is currently no sentence-level Chinese-English audio translation tool, that is, there is no mature audio translation tool for one sentence of English and one sentence of Chinese. To this end, we developed a Sentence-level Chinese-English Audio Fusion system to fill this market gap. In view of the powerful capabilities of deep learning, in this study, we develop a cascaded audio translation system based on deep learning techniques. It contains three modules: speech recognition, machine translation, and speech generation. We use the Wav2vec 2.0 framework for speech recognition, the MarianMT model for machine translation, and the VITS model for speech synthesis. After preliminary experiments, the model has satisfactory accuracy and impressive results. The code is available at: https://github.com/CCscenery/Sentence-level-Chinese-English-Audio-Fusion-Based-on-Deep-Learning/tree/main

Introduction

In today's world, text translation technology has become an essential tool for many individuals. It allows us to easily understand written content in a language that is not our own, making it a valuable resource for reading and writing. However, text-to-text translation has its limitations when it comes to developing speaking and listening skills in a foreign language. This is where Speech-to-speech translation comes into play. Translating spoken language from one language to another, allows us to practice our listening and speaking skills in a more authentic and immersive. It gives us the opportunity to hear native speakers conversing and allows us to practice our own speaking in a way that is more similar to a real-life conversation. In particular, we are interested in studying English-speech to Chinese-speech translation as a way to improve our listening skills in our spare time. Overall, Speech-to-speech translation is a valuable tool for language learners looking to improve their speaking and listening abilities in a foreign language. In general, our group believes that Speech-to-speech translation is an important tool for language learners to improve their foreign language spoken and listening abilities, so we want to implement a speech-to-speech translation system with deep learning methods.

Speech-to-speech translation (S2ST) aims at translating speech from one language into speech in another language. S2ST technology can not only enable communication between people speaking different languages but also help knowledge sharing across the world. Generally, S2ST mainly includes an end-to-end system and a cascade system.

Recently, work on S2ST without relying on intermediate text representation is emerging, such as end-to-end direct S2ST (Kano, Sakti, and Nakamura 2021) and cascade S2ST based on discrete speech representation (Tjandra, Sakti, and Nakamura 2019). However, as of today, publicly available corpora directly suitable for such research are extremely limited.

However, the traditional cascading method is mostly used in industry. Its frame is shown in Figure 1. Generally, S2ST can be realized through the connection of three systems: automatic speech recognition (ASR), machine translation (MT), and text-to-speech synthesis (TTS) (Lavie et al. 1997)

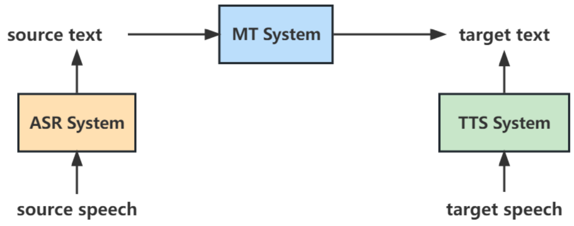

Figure 1: The Framework for Cascading Speech-to-speech Translation

- Automatic Speech Recognition (ASR)

  Automatic Speech Recognition is a rapidly developing field that aims to enable computers to automatically transcribe spoken language into text. With a wide range of applications, including voice-to-text transcription for dictation, voice commands for virtual assistants and smart devices, and language translation, ASR technology has the potential to significantly improve the efficiency and accuracy of spoken language communication.

  ASR systems typically consist of three main components: a feature extractor, an acoustic model, and a language model. The feature extractor converts the raw speech signal into a set of numerical features that capture the characteristics of the speech, such as spectral and prosodic information. The acoustic model then uses these features to predict the transcription of the speech, based on the probability of each possible transcription given the input features. The language model considers the grammatical and structural constraints of the language being spoken and helps to improve the overall accuracy of the transcription.

There are two main approaches to ASR: traditional rule-based systems and modern machine learning-based systems. Traditional rule-based systems rely on handcrafted rules and algorithms to recognize speech, while machine learning-based systems use data-driven methods to learn from large amounts of annotated speech data. In recent years, machine learning-based ASR systems, particularly those based on deep neural networks (DNNs), have achieved state-of-the-art performance and have been widely adopted in the industry.

- Machine Translation (MT)

  Machine Translation is the task of automatically translating text from one language to another. It has a wide range of applications, including language translation for websites, documents, and messaging applications, as well as multilingual information retrieval and machine-aided human translation. Machine translation is mainly divided into statistical machine translation (SMT) and neural machine translation (NMT)

  Statistical Machine Translation (SMT) is a method for building machine translation systems that rely on large-scale language corpora. It effectively reduces reliance on human intervention, and can flexibly handle language structure through the use of formalized grammar models. In recent years, SMT has evolved from word-based machine translation to phrase-based translation.

  Neural Machine Translation (NMT) is a newer machine translation technology that has emerged in recent years and has become the mainstream translation technology in the language translation industry. As a completely new machine translation model, NMT uses deep learning neural networks to acquire the mapping relationship between natural languages and directly translates from the source language to the target language, effectively avoiding the complex conversion process in traditional SMT translation. NMT has also been widely recognized by scholars for its simplicity, high efficiency, and excellent translation performance.

- Text-to-speech synthesis (TTS)

  Text-to-speech synthesis is a technology that can convert any input text into corresponding speech. There are both traditional, non-end-to-end methods and the more popular end-to-end methods based on neural networks for speech synthesis. Non-end-to-end methods are typically composed of three parts: a text analysis frontend, an acoustic model, and a vocoder. First, the text front end converts the text into a standard input. Then, the acoustic model transforms the standard input into intermediate acoustic features for modeling the long-term structure of speech. The most common intermediate acoustic features are spectrograms, vocoder features, or language features. Finally, the vocoder fills in low-level signal details and converts the acoustic features into time-domain waveform samples. On the other hand, end-to-end synthesis systems directly input text or phoneme characters and output audio waveforms. End-to-end systems reduce the requirement for linguistic knowledge and can be easily replicated across different languages to implement batch synthesis systems for dozens or even more languages. Furthermore, end-to-end speech synthesis systems show strong and rich expressive capabilities for pronunciation styles and rhythms.

Thanks to the mature research of each part, the cascading method will reduce the difficulty of task implementation, but the control of propagation errors and the loss of paralinguistic information are still worthy of research.

Related Work

End-to-end system. The end-to-end system directly translates from voice to voice. Most recently, researchers have built one-stage S2ST systems (Jia et al. 2019) that jointly optimize intermediate text generation and target speech generation steps (Kano, Sakti, and Nakamura 2021) or further remove the dependency on text completely (Tjandra, Sakti, and Nakamura 2019) Not relying on text generation as an intermediate step allows the systems to support translation into languages that do not have standard or widely used text writing systems (Zhang et al. 2021). The advantage of this method is fast, but the disadvantage is that task modeling is complex and difficult.

Cascade system.

The Cascade system divides the translation process into three steps: speech recognition, machine translation, and speech generation. The task is completed by cascading these three parts:

- Automatic Speech Recognition (ASR)

  Automatic Speech Recognition (ASR) End-to-end (E2E) automatic speech recognition (ASR) models have received increasing attention in recent years, such as connectionist temporal classification (CTC) (Graves et al. 2006), recurrent neural network transducer (RNN-T) (Graves 2012; Graves, Mohamed, and Hinton 2013), and attention-based encoder-decoder (AED) (Chorowski et al. 2014). One of the most appealing aspects of E2E models is their simplified training procedure compared to traditional hybrid ASR frameworks.

- Machine Translation (MT)

  Machine Translation (MT) Over the past decade, the state of the art in machine translation has been greatly improved through the use of neural machine translation (NMT) (Bachman, Alsharif, and Precup 2014) and Transformer-based models (Vaswani et al. 2017). These models often achieve state-of-the-art (SOTA) translation performance using large-scale corpora (Ott et al. 2018).

Along with the advancement of NMT, consistency training (Bachman, Alsharif, and Precup 2014) has been widely adopted and has shown great potential for improving NMT performance. It simply regularizes NMT model predictions to be invariant to small perturbations applied to the inputs and hidden states (Chen et al. 2021) or the model randomness and variance present in the training process (Wu et al. 2021).

- Text-to-speech synthesis (TTS)

Text-to-speech synthesis (TTS) is a technology that can convert any input text into corresponding speech. Neural network-based autoregressive TTS systems have demonstrated the ability to synthesize realistic speech (Shen et al. 2018), but their sequential generative process makes it challenging to fully utilize modern parallel processors. To address this issue and improve synthesis speed, several non-autoregressive methods have been proposed. One such method involves extracting attention maps from pre-trained autoregressive teacher networks (Ren et al. 2019) in the text-to-spectrogram generation step, which aims to reduce the difficulty of learning alignments between text and spectrograms. More recently, likelihood-based methods have further eliminated the reliance on external aligners by estimating or learning alignments that maximize the likelihood of target Mel-spectrograms (Zeng et al. 2020). Additionally, generative adversarial networks (GANs) (Goodfellow et al. 2020) have been explored in second-stage models for TTS.

The cascading method sequentially combines the above three parts. This approach has several advantages:1)It allows for the decoupling of the voice translation task into separate modules, which makes it easier to design and optimize each component.2)Both speech recognition and machine translation have a long history of research and development, so they have a wealth of mature technologies that can be leveraged to improve the effectiveness of speech-to-speech translation. However, cascading systems also have some drawbacks compared to end-to-end systems:1)Error propagation: If there is an error in the output of one module, it will be carried forward to the next module, potentially leading to further errors down the line.2)Low translation efficiency: Cascading systems may not be able to meet the real-time requirements of certain voice translation tasks due to their relatively slow processing speed. On the whole, a cascading method is a useful approach for speech-to-speech translation, but it is important to consider its limitations and trade-offs when deciding which method to use.

Proposed Solution

Overall System

We disassemble the model into three parts: ASR, MT, and TTS. We also divide TTS into Chinese speech synthesis and English speech synthesis. Chinese speech is our final output, while English speech synthesis is to convert the English text after speech recognition into audio. The purpose of this is to complete our original intention of designing this system, that is, one English sentence corresponds to one Chinese sentence. The advantage of this is that the output text has punctuation marks. We can use punctuation marks to split the audio and complete our sentence-level audio translation. So the general process of this model is to use wav2vec 2.0 to identify English audio as English text and then use MarianMT to machine translate it into Chinese text. Then we use punctuation marks to segment, and use VITS model and pyttx3 library to output and splice Chinese and English text sentence by sentence.

Automatic Speech Recognition (ASR)

In the ASR part, we use Wav2vec 2.0. It is a cutting-edge speech recognition model that has been developed to effectively convert spoken language into text. The architecture of Wav2vec 2.0 is shown in the Figure 2 . It utilizes transfer learning techniques, which involve using pre-trained models as a starting point and fine-tuning them for a specific task, in this case, speech recognition. This allows for the rapid training and deployment of new models, making Wav2vec 2.0 a highly efficient and effective tool for speech-to-text conversion. One of the key techniques employed by Wav2vec 2.0 is called Masked Language Modeling (MLM), which was originally introduced in the BERT model. MLM consists of two main tasks: the contrastive learning task and the quantization task.

In the contrastive learning task, the model is trained to identify the true, unmasked feature at a given position in the input, while simultaneously rejecting unmasked features at other positions. This helps the model to focus on the most relevant features for speech recognition, improving its accuracy and performance. In the quantization task, the true unmasked features are transformed into a discrete space, allowing for more efficient processing and representation of the data. To ensure diversity in this process, the entropy of the averaged softmax distribution is maximized, resulting in the unmasked features being evenly distributed in the discrete space. Overall, the combination of these techniques allows Wav2vec 2.0 to achieve impressive results in a speech to text conversion, making it a valuable tool for a wide range of applications.

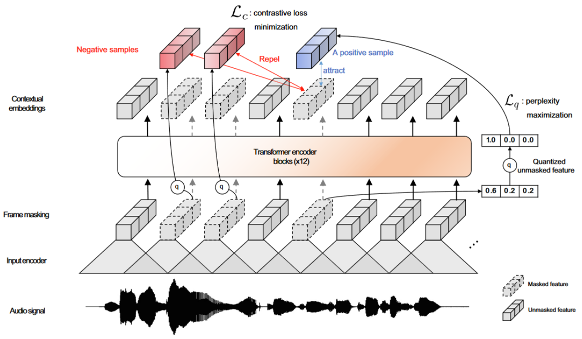

Figure 2: The architecture of architecture Wav2vec 2.0

Machine Translation (MT)

In this section, we will introduce the method we used originally and the method we use now because it is unfortunate that although the original method has improved the score, the actual translation effect may not be good. We think this is because the parallel corpus is not enough to support the normal translation effect or for some other reasons. So we tried to use a larger dataset, but our equipment didn't support us to train this model. In order to ensure the complete functionality of the whole project, we adopt a new method to use a pre-trained model and explore the effectiveness of the original method under the premise of ensuring the integrity of the result.

- Original method

  We used the StrokeNet (Wang, Liu, and Zhang 2022) and the concept of a bidirectional translation model just like BiBERT (Xu, Van Durme, and Murray 2021) to ensure full utilization of contextualized embeddings for En→Zh on the WMT'18 dataset. The dataset consists of sentences in total from domains including broadcast, newswire, and web data.

  Our model configuration is Transformer and implemented using the Fairseq framework, configuration is a six-layer transformer architecture with FFN dimension size 2048 and 4 concerns. We use the 768 embedding dimensions to match the dimensions of the pretrained language model. The evaluation metric is the commonly tokenized BLEU (Papineni et al. 2002) score .

- New method

  In this section, we use the output (contextualized embeddings) of the last layer of pre-trained language models on building NMT models and dual translation train as above. We implemented using a pre-trained Huggingface model on a smaller En→Zh IWSLT'17 dataset. The dataset consists of 200K parallel sentences in total.

  Our model configuration is based on MarianMT model, which supports fast training and translation. Models were originally trained by Jörg Tiedemann using the Marian C++ library, which supports fast training and translation. All models are transformer encoder-decoders with 6 layers in each component.

  We only use the last layer of the model, introduced layer-aware attention mechanism to capture compound contextual information from mode. We use it to extracting with embedded source sentences from the last layer of frozen pre-trained language models and feed them to an embedding layer of NMT encoders. Instead of randomly initializing the source embedding layers, we use the outputs of these pre-trained models and do not allow these parameters to be updated during training.

  Pre-trained monolingual language models can improve the performance of machine translation systems, but machine translation is inherently a bilingual task. We hypothesize that a pre-trained language model with its training data consists of a mixture of texts from source and target languages. In other words, we expect source and target language data to enrich each other's contextual information to better facilitate bidirectional translation. Therefore we use a bilingual pre-trained language model, called BIBERT.

  Therefore, we also use a different from the ordinary one-way translation model, we use a two-way translation model, it is, one model can translate En → Zh and Zh → En. So our first processing step is mixing source and target language data.

Text-to-speech synthesis (TTS)

In the TTS part, we use Variational Inference with adversarial learning for end-to-end Text-to-Speech(VITS). VITS mainly includes three parts: Conditional Variational AutoEncoder (VAE), alignment estimation generated from variational inference, and generation of confrontation training.

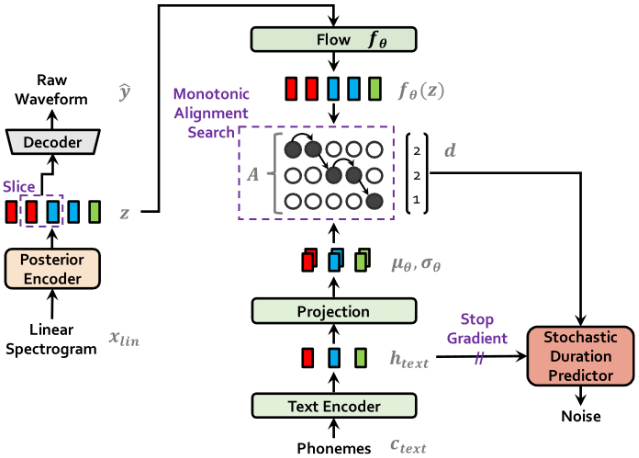

Figure 3: The training procedure of the VITS model

The training procedure is shown in the Figure 3. First, put the text and code the isolated text into context-sensitive features through the Text Encoder module. Then, a priori encoder module is used to obtain a priori distribution (a hidden variable is predicted based on the condition of text and speaker identity). At the other end, the input voice waveform is converted into a linear spectrum without parameters. After the Poster Encoder module, the output is also a posterior distribution of hidden variables.

In order to train the model and make the distance between the generated waveform and the original waveform close, the following operations are carried out in VITS:

1. Because the output length of the posterior encoder is the length of the spectrum. The output length of a priori encoder is the length of the text. In order to calculate the distance between the two prior distributions and the posterior distribution, the dimensions of the two modules are aligned first. Since there is no a posteriori encoder when reasoning, we need a time length predictor to model the process of just dynamic planning with a time length predictor, send the text en-

coder output to the time length predictor, predict a time length, and then expand the prior distribution.

2. The features generated by the posterior encoder are sent to the decoder to generate waveforms, calculate the Mel spectrum, and calculate L1 loss with the real spectrum.
3. The duration predictor is based on the VAE structure, and there will still be a variational lower-bound loss
4. In the GAN training, the discriminator and the generator have conflicts (the discriminator judges whether the output is from the decoder or the real waveform)

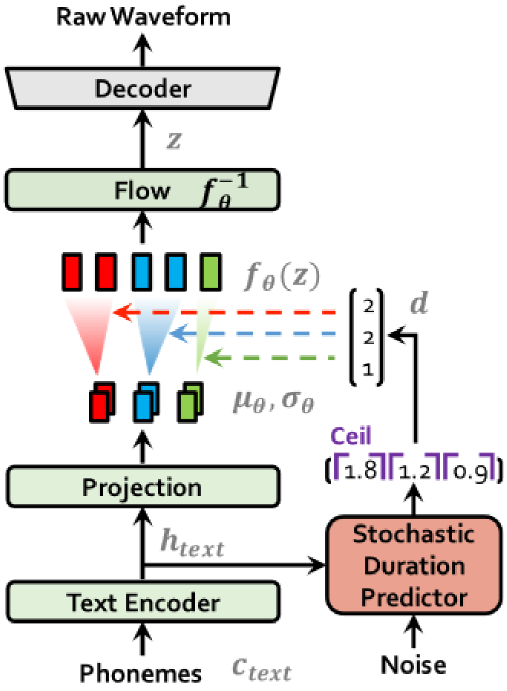

Figure 4: The inference procedure of the VITS model

The inference procedure is shown in the Figure 4. Text input is represented by a text encoder, which is divided into two ways.

Step1 After a priori encoder of the text and resampling, z (length of the text layer) is obtained

Step2 After the time length predictor, the flow model. First, sample and generate noise, and then calculate the duration through the flow model. After the time length is rounded, the value of the above text prior to encoder output heavy parameter z is expanded. The expanded z is the z of the spectral magnitude. After the waveform generator, the waveform is obtained by sampling step by step.

Experiments

In the experiment part, we use a pre-trained model for speech recognition. We mainly discuss machine translation and speech synthesis, as well as the final overall model implementation effect.

Dataset

- MT We conduct experiments on the WMT18 Zh-En benchmark. For the WMT18 Zh-En, the training data contains 50M sentence pairs. We use the scripts in Moses to tokenize. We use jieba4 to conduct. Then we apply the BPE algorithm to Chinese and English separately. For the WMT17 Zh-En, we conduct 40K BPE operations on Chinese and English joint-BPE in the StrokeNet.
- TTS Since the relevant benchmark datasets are almost all English datasets, that is, the waveform in the voice file is used as the training feature, and the English corresponding to the voice is used as the label, there are few public Chinese datasets for voice synthesis, and there are fewer training sets with the same voice color. So I unpacked the voice on the Genshin Impact game to obtain the voice source file. Because different game characters have different voice colors, I selected the game character of Pemont as the voice of the dataset, collected the relevant voice files, and used them as the dataset. In order to obtain the tag of the corresponding dataset, I first converted the voice file obtained above into text through the iFLYTEK speech recognition API and used it as the basic tag. In order to reduce label errors and filter noise and redundancy in the dataset, manual proofreading is being carried out to further improve the correspondence and accuracy between the dataset and labels. After the above steps, 2293 voice data and tags are finally obtained. According to the 4:1 ratio, 1820 of them are used as the training set and 473 are used as the test set to train the model.

Setting

- Overall system In addition, we cascaded three modules to form our model, and we also analyzed the running time of each module of the model. We divide the audio from the 30s to 180s into 11 groups and calculate their respective running time and total time.
- MT For training vanilla NMT models, the decoder input and output embeddings are shared. For the WMT18 Zh-En, we use Adam to train for 50 epochs on the basic model, with 2048 max tokens per batch, the learning rate 0.0004, weight decay of 0.00002, and dropout ratio 0.1. We warm up the learning rate for the first 1K steps and then use the inverse square root scheduler.

  For training NMT models with dual translation, we feed target sentences and expect translations in the source language. The rest hyperparameters keep the same as the vanilla models.

  For training NMT models with StrokeNet, each Chinese character is mapped to the corresponding Latinized stroke sequence. Joint vocabulary is learned from both source and target texts together. During training, all the

Table 1: The result of MT

MODEL	BLUE
basic model	21.2, 57.3/27.9/15.7/9.2
+ dual translation	22.14, 58.4/28.5/16.1/9.6
+ StrokeNet	23.41, 59.5/29.7/17.1/10.2

embeddings and softmax weights are shared. The rest hyperparameters keep the same as the vanilla models.

- TTS We set the training epoch to 1434 times, the optimizer to Adam, the learning rate to 0.0002, and the exponential continuous attenuation. Each epoch is boiled to the original 0.999875. Before training, the Chinese text in the tag is preprocessed into Pinyin, so that we can obtain the tones and phonemes of each self. At the same time, the audio is segmented to obtain the corresponding spectrum. Finally, we will get the spectrum and the corresponding text phoneme input model and then conduct training.

Result

- MT. For the WMT17 Zh-En, we generate with length penalty of 1.4 and a beam size of 5. The BLEU scores are evaluated with multi-blue provided by Moses. We use the checkpoint with the best validation BLEU for testing. The result is shown in table1.

- TTS. Since the evaluation index of the speech synthesis model is a crowdsourcing MOS test, the rater listens to randomly selected audio samples and scores their naturalness on a 5-point scale of 1 to 5, which has certain subjectivity. So we invited some students to score the result of speech synthesis, and the average score was 4.2, corresponding to better speech synthesis, which can be heard clearly and smoothly, and students also like Genshin Impact's voice very much.

- Running time. It can be seen from the Figure5 that the total duration of audio translation is still linearly and positively correlated with the duration of input audio, which is also consistent with our conventional experience.

  The total time consumption of the model is mainly concentrated on Chinese speech synthesis and speech recognition, as shown in Figure6. If we want to improve the overall translation speed in the future and strive to move closer to real-time voice translation, we need to improve the speed of voice synthesis.

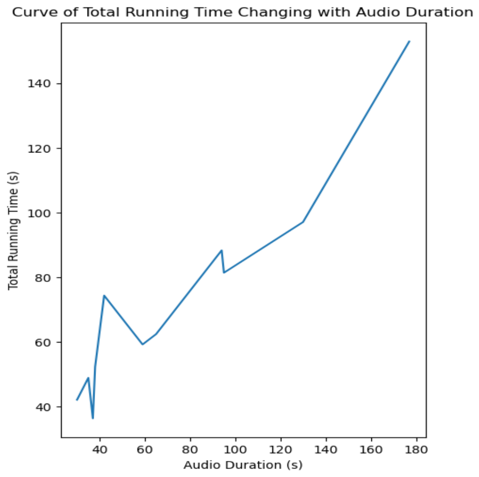

Figure 5: Curve of Total Running Time Changing with Audio Duration

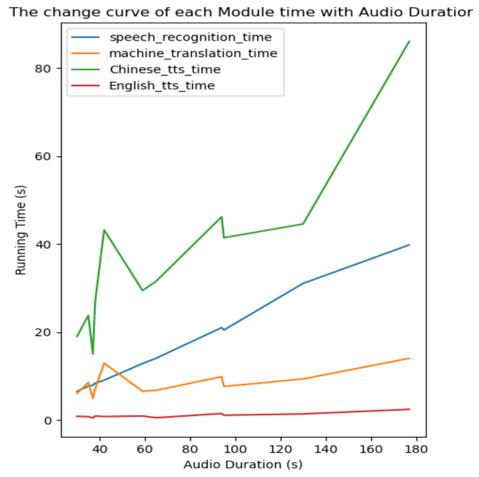

Figure 6: The change curve of each Module time with Audio Duration

Conclusion

To practice our speaking and listening, we designed and implemented a sentence-level cascade audio translation system using deep learning techniques. We divide the system into three main functional modules: speech recognition, machine translation, and speech synthesis. For these three modules, we use Wav2vec 2.0, MarianMT, and VITS respectively. The final effect is very impressive. In the future, further improving the accuracy and efficiency of the system will provide greater value to the majority of foreign language learners.

References

Bachman, P.; Alsharif, O.; and Precup, D. 2014. Learning with pseudo-ensembles. *Advances in neural information processing systems*, 27.

Chen, G.; Fan, K.; Zhang, K.; Chen, B.; and Huang, Z. 2021. Manifold adversarial augmentation for neural machine translation. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, 3184–3189.

Chorowski, J.; Bahdanau, D.; Cho, K.; and Bengio, Y. 2014. End-to-end continuous speech recognition using attention-based recurrent NN: First results. *arXiv preprint arXiv:1412.1602*.

Goodfellow, I.; Pouget-Abadie, J.; Mirza, M.; Xu, B.; Warde-Farley, D.; Ozair, S.; Courville, A.; and Bengio, Y. 2020. Generative adversarial networks. *Communications of the ACM*, 63(11): 139–144.

Graves, A. 2012. Sequence transduction with recurrent neural networks. *arXiv preprint arXiv:1211.3711*.

Graves, A.; Fernández, S.; Gomez, F.; and Schmidhuber, J. 2006. Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks. In *Proceedings of the 23rd international conference on Machine learning*, 369–376.

Graves, A.; Mohamed, A.-r.; and Hinton, G. 2013. Speech recognition with deep recurrent neural networks. In *2013 IEEE international conference on acoustics, speech and signal processing*, 6645–6649. Ieee.

Jia, Y.; Weiss, R. J.; Biadsy, F.; Macherey, W.; Johnson, M.; Chen, Z.; and Wu, Y. 2019. Direct speech-to-speech translation with a sequence-to-sequence model. *arXiv preprint arXiv:1904.06037*.

Kano, T.; Sakti, S.; and Nakamura, S. 2021. Transformer-based direct speech-to-speech translation with transcoder. In *2021 IEEE Spoken Language Technology Workshop (SLT)*, 958–965. IEEE.

Lavie, A.; Waibel, A.; Levin, L.; Finke, M.; Gates, D.; Gavalda, M.; Zeppenfeld, T.; and Zhan, P. 1997. JANUS-III: Speech-to-speech translation in multiple languages. In *1997 IEEE International Conference on Acoustics, Speech, and Signal Processing*, volume 1, 99–102. IEEE.

Ott, M.; Auli, M.; Grangier, D.; and Ranzato, M. 2018. Analyzing uncertainty in neural machine translation. In *International Conference on Machine Learning*, 3956–3965. PMLR.

Papineni, K.; Roukos, S.; Ward, T.; and Zhu, W.-J. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, 311–318.

Ren, Y.; Ruan, Y.; Tan, X.; Qin, T.; Zhao, S.; Zhao, Z.; and Liu, T.-Y. 2019. Fastspeech: Fast, robust and controllable text to speech. *Advances in Neural Information Processing Systems*, 32.

Shen, J.; Pang, R.; Weiss, R. J.; Schuster, M.; Jaitly, N.; Yang, Z.; Chen, Z.; Zhang, Y.; Wang, Y.; Skerrv-Ryan, R.; et al. 2018. Natural tts synthesis by conditioning wavenet on mel spectrogram predictions. In *2018 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, 4779–4783. IEEE.

Tjandra, A.; Sakti, S.; and Nakamura, S. 2019. Speech-to-speech translation between untranscribed unknown languages. In *2019 IEEE Automatic Speech Recognition and Understanding Workshop (ASRU)*, 593–600. IEEE.

Vaswani, A.; Shazeer, N.; Parmar, N.; Uszkoreit, J.; Jones, L.; Gomez, A. N.; Kaiser, Ł.; and Polosukhin, I. 2017. Attention is all you need. *Advances in neural information processing systems*, 30.

Wang, Z.; Liu, X.; and Zhang, M. 2022. Breaking the Representation Bottleneck of Chinese Characters: Neural Machine Translation with Stroke Sequence Modeling. *arXiv preprint arXiv:2211.12781*.

Wu, L.; Li, J.; Wang, Y.; Meng, Q.; Qin, T.; Chen, W.; Zhang, M.; Liu, T.-Y.; et al. 2021. R-drop: Regularized dropout for neural networks. *Advances in Neural Information Processing Systems*, 34: 10890–10905.

Xu, H.; Van Durme, B.; and Murray, K. 2021. Bert, mbert, or bibert? a study on contextualized embeddings for neural machine translation. *arXiv preprint arXiv:2109.04588*.

Zeng, Z.; Wang, J.; Cheng, N.; Xia, T.; and Xiao, J. 2020. Aligntts: Efficient feed-forward text-to-speech system without explicit alignment. In *ICASSP 2020-2020 IEEE international conference on acoustics, speech and signal processing (ICASSP)*, 6714–6718. IEEE.

Zhang, C.; Tan, X.; Ren, Y.; Qin, T.; Zhang, K.; and Liu, T.-Y. 2021. Uwspeech: Speech to speech translation for unwritten languages. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, 14319–14327.