INTERSPEECH 2023
20-24 August 2023, Dublin, Ireland

MMSpeech: Multi-modal Multi-task Encoder-Decoder Pre-training for speech recognition

Xiaohuan Zhou, Jiaming Wang*, Zeyu Cui, Shiliang Zhang, Zhijie Yan, Jingren Zhou, Chang Zhou*†

DAMO Academy, Alibaba Group, China
{shiyi.zxh, wangjiaming.wjm, zeyu.czy, sly.zsl, zhijie.yzj, jingren.zhou, ericzhou.zc}@alibaba-inc.com

Abstract

In this paper, we propose a novel multi-modal multi-task encoder-decoder pre-training framework (MMSpeech) for Mandarin automatic speech recognition (ASR), which employs both unlabeled speech and text data. The main difficulty in speech-text joint pre-training comes from the significant difference between speech and text modalities, especially for Mandarin speech and text. Unlike English and other languages with an alphabetic writing system, Mandarin uses an ideographic writing system where character and sound are not tightly mapped to one another. Therefore, we propose to introduce the phoneme modality into pre-training, which can help capture modality-invariant information between Mandarin speech and text. In addition, a much larger amount of unsupervised text data 292G is utilized for pre-training, which brings significant improvements. Experiments on AISHELL-1 show that our proposed method achieves state-of-the-art performance, with a more than 40% relative improvement.

Index Terms: ASR, pre-training, encoder-decoder

1. Introduction

Recently, research on pre-training has been widely investigated and greatly improved the performance of downstream speech tasks, such as automatic speech recognition (ASR). In general, pre-training methods for ASR can be roughly divided into two branches, namely encoder pre-training and encoder-decoder pre-training methods. For encoder pre-training methods, a large amount unlabeled speech data is used to help the encoder learn the ability of extracting the universal speech representation [1, 2, 3, 4]. For example, Data2Vec [4] is trained by generating representations using the teacher encoder based on the full input and then regressed by the student encoder of the same architecture based on a masked version of the input. As only the encoder is pre-trained, they usually employ connectionist temporal classification (CTC) [5] based models for downstream ASR tasks. Since encoder-decoder based ASR models [6, 7] usually obtain better recognition performance, encoder-decoder pre-training methods are further proposed. These methods are usually optimized within a multi-task learning framework. For example, Speech2C [8] introduces two tasks using speech-only data via pseudo codes. One is to predict the pseudo codes via masked language modeling and the other is to learn the reconstruction of pseudo codes. Recently, multi-modal pre-training has achieved great success in both cross-modal and single-modal downstream tasks. SpeechT5 [9], STPT [10], and SpeechUT [11] leverage unlabeled speech and text data for speech-text joint pre-training. To mitigate the discrepancy between speech and text, they propose to map text and speech into a shared representation space for joint pre-training, where the shared representation space can be implicitly learnable vectors or artificially defined phonemes. However, there are still some aspects needed to be further investigated: (1) Existing encoder-decoder pre-training works are mainly exclusively for English while almost none for Mandarin; (2) Unlabeled text data is underestimated and less explored in speech pre-training literature; (3) The complementarity between tasks of different works is not fully exploited.

To be more specific, Mandarin is an ideographic rather than a phonetic language like English, which contains a large number of homophones [12]. More significant differences exist between Mandarin speech and text, and it is hard to learn a shared representation space for them implicitly. Therefore, we introduce the phoneme modality into pre-training, which is a natural bridge to alleviate the problem of homophones and capture modality-invariant information between speech and text; that is, both speech and text can be uniquely mapped to a phoneme sequence. On the other hand, previous works have focused on collecting speech data which has been scaled up to 10k hours. However, the number of text data remained the same, always using the LibriSpeech LM corpus (1.8G). It is valuable to explore the value of text data since it has lower costs to acquire and store text data than speech data. In this paper, we introduce 292G text data from M6-Corpus [13] for pre-training, which is much larger than previous works [9, 14, 10]. Experiments demonstrate that enough unlabeled text data is also useful like unlabeled speech data.

In this way, we propose a novel multi-modal multi-task encoder-decoder pre-training framework (MMSpeech) for Mandarin ASR. Five tasks are employed for multi-task pre-training. Firstly, to bridge the gap between Mandarin speech and text and utilize unsupervised text data better, we propose to introduce the phoneme modality into the pre-training. The masked speech prediction (MSP), phoneme prediction (PP), and phoneme-to-text (P2T) tasks are introduced to build relationships among speech, phonemes, and text. The MSP and PP tasks are two encoder pre-training tasks that predict phonemes based on speech. The P2T task utilizes much text data to build a relation between phonemes and text. Experiments prove that the improvements achieved by the P2T task pre-training can not be replaced by an external language model (LM), demonstrating P2T not only learns the grammatical rules of text but also learns the connection between pronunciation information and text. Furthermore, we are the first to introduce the speech-to-pseudo-codes (S2C) task [15, 8] proposed for speech-only pre-training into the speech-text joint pre-training and prove their complementarity. We consider that the S2C task translating speech to pseudo-codes within a sequence-to-sequence manner

* Equal contribution. † Corresponding author.

10.21437/Interspeech.2023-791

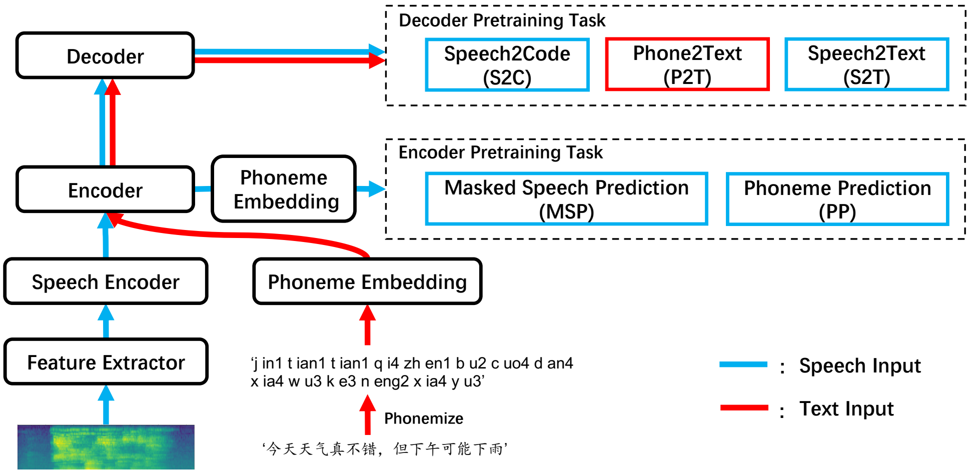

Figure 1: *MMSpeech. The modules S2T are used in the final ASR model and the others are used to auxillate the pre-training. The blue, red lines represent speech and text data flow in the model. Note that the parameters of the two phoneme embedding are shared.*

can enhance the ability of the decoder to locate and encapsulate speech information. Finally, we introduce the downstream speech-to-text (P2T) task to further improve the pre-training performance. Experiments on AISHELL-1 corpus show that our proposed method achieves the state-of-the-art (SOTA) performance, with a more than 40% relatively improvement compared with other pre-training methods.

2. Methods

In this section, we elaborate the proposed MMSpeech based on encoder-decoder framework. As illustrated in Fig. 1, MMSpeech consists of five tasks with speech and text data.

2.1. Model Architecture

As shown in Fig. 1, MMSpeech mainly employs the encoder-decoder architecture. The encoder network consists of a speech feature extractor, a speech encoder and a shared encoder. Specifically, the speech feature extractor is a multi-layer convolutional network while the speech encoder and the shared encoder are multiple transformer layers with multi-head self attention [16]. The decoder network is also multiple transformer layers, which is similar to the encoder except for masked self-attention and cross-attention.

2.2. Encoder Pre-Training Tasks

For encoder pre-training, we introduce self-supervised masked speech prediction (MSP) and supervised phoneme prediction (PP) tasks which utilize unlabeled speech data and supervised data to build a relation between speech and phonemes.

2.2.1. Masked speech prediction

The MSP task utilizes unlabeled speech for encoder pre-training by masked language modeling [17]. We choose the phoneme distributions as the predicted target [10] rather than hidden states like [2, 3, 4] since phonemes containing only pronunciation information are a bridge between speech and text. Specifically, as shown in Fig. 1, a speech feature extractor first extract latent speech representations from speech input $\mathbf{X}$[1], which can be represented as $\mathbf{Z} = (\mathbf{z}_1, \ldots, \mathbf{z}_T)$. Then the target and predicted phoneme distributions can be computed. For the target phoneme distribution, the latent speech representations $\mathbf{Z}$ are fed to the speech encoder and shared encoder to build contextualized speech representations $\mathbf{H} = \left(\mathbf{h}_1, \ldots, \mathbf{h}_{T'}\right)$. $T^{'}$ is the down sampling size of T. We compared $\mathbf{H}$ with a learnable phoneme embedding $\mathbf{E} = (\mathbf{e}_1, \ldots, \mathbf{e}_I)$ and get the the target phoneme distribution $p(\mathbf{e}_i|\mathbf{h}_t)$. Note that the phoneme embedding $\mathbf{E}$ is actually frozen in the MSP task, which will be described in Section 2.2.2. As for the predicted phoneme distribution, $\widetilde{\mathbf{Z}}$ is a masked version of $\mathbf{Z}$ obtaining by a span mask strategy[4]. $\widetilde{\mathbf{H}}$ is the corresponding contextualized speech representations and $p(\mathbf{e}_i|\widetilde{\mathbf{h}}_t)$ is the predicted phoneme distribution. Based on the target and predicted phoneme distributions, the masked KL divergence loss can be computed as follows:

$$\mathcal{L}_{\text{MSP}} = -\sum_{t=1}^{T^{'}}\sum_{i=1}^{I} p(\mathbf{e}_i|\mathbf{h}_t)\text{log}\frac{p(\mathbf{e}_i|\widetilde{\mathbf{h}}_t)}{p(\mathbf{e}_i|\mathbf{h}_t)} \tag{1}$$

2.2.2. Phoneme prediction

There exists collapse problem in the MSP task namely producing similar phoneme distribution for all masked frames resulting in a trivial task. Therefore, we introduce the PP task with paired speech-text data to guide the phoneme embedding learning. Based on the distribution $p(\mathbf{e}_i|\mathbf{h}_t)$ and the target phoneme sequence converted from the corresponding text, we employ the CTC loss to optimize the PP task:

$$\mathcal{L}_{\text{PP}} = -\sum_{\mathbf{e}_{t,j}\in\pi}\prod_{t=1}^{T'} p(\mathbf{e}_{t,j}|\mathbf{h}_t) \tag{2}$$

where π denotes all possible augmented sequence with the blank symbol of the target phoneme sequence. We also share the phoneme embedding with the P2T task in Section 2.3.1. To further alleviate the collapse problem, the phoneme embedding is frozen in the MSP task and updated by the PP and P2T tasks.

2.3. Encoder-Decoder Pre-Training Tasks

For encoder-decoder pre-training, we introduce self-supervised phoneme-to-text (P2T) and speech-to-pseudo-code (S2C) tasks which utilize unlabeled text data and unlabeled speech data.

[1] In this paper, the speech input is the log Mel-filterbank feature rather than the raw audio waveform [2, 3, 4]

2.3.1. *Phoneme-to-text*

We propose to use the P2T task utilizing large-scale unlabeled text data for pre-training, which is a modified version of text-infilling [18, 9]. We convert input Mandarin texts into phoneme sequences via an open-sourced "pypinyin" python package, which reduces the difference between Mandarin speech and text inputs and makes it easier for them to share an encoder [14, 10]. We also add noise to the phoneme sequences by masking or replacing token spans. Then the phoneme embedding and shared encoder is employed to extract phoneme features $\mathbf{H}^e$. Note that we share the phoneme embedding here with the PP task as described in Section 2.2.2. The decoder reconstructs the text sequence like the text-filling task based on $\mathbf{H}^e$. The task is optimized by maximizing cross entropy:

$$\mathcal{L}_{P2T} = -\sum_{l=1}^{l=L} logp(\mathbf{y}_l^t|\mathbf{y}_{l-1}^t, \mathbf{H}^e) \tag{3}$$

2.3.2. *Speech-to-pseudo-code*

For the S2C task, we utilize unlabeled speech data and introduce a reconstruction loss following Speech2C [8], which generates pseudo-codes $\mathbf{Y} = (\mathbf{y}_1^c, \ldots, \mathbf{y}_L^c)$ in an autoregressive fashion:

$$\mathcal{L}_{\text{S2C}} = -\sum_{l=1}^{L} \text{log}p(\mathbf{y}_l^c|\mathbf{y}_{l-1}^c, \mathbf{H}) \tag{4}$$

Pseudo-codes are discrete token sequences generated by using an external model to annotate unlabeled speech. Now we describe how to generate pseudo-codes. Specifically, we first use the pre-trained encoder to extract a sequence of hidden states. Then K-means clustering is applied to discretize these hidden states and obtain a sequence of hidden units. To shorten the sequence length like [15], we deduplicate these units (e.g., "1 1 1 2 3 3" will be processed to "1 2 3") and further adopt a byte-pair encoding (BPE) [19] model to generate a new sequence of hidden units referred to as pseudo-codes.

2.4. Multi-task Pre-Training

We additionally introduce the downstream speech-to-text (S2T) task to further improve the pre-training performance:

$$\mathcal{L}_{\text{S2T}} = -\sum_{l=1}^{L} \text{log}p(\mathbf{y}_l^t|\mathbf{y}_{l-1}^t, \mathbf{H}) \tag{5}$$

With the S2T task, we can actually obtain excellent recognition results without fine-tuning and directly estimate the quality of pre-training in the pre-training.

Finally, the overall pre-training loss for MMSpeech can be defined as the weighted summation of five tasks losses:

$$\mathcal{L} = \lambda_1\mathcal{L}_{\text{MSP}} + \lambda_2\mathcal{L}_{\text{PP}} + \lambda_3\mathcal{L}_{\text{S2C}} + \lambda_4\mathcal{L}_{\text{P2T}} + \lambda_5\mathcal{L}_{\text{S2T}} \tag{6}$$

where λ_1, ... λ_5 are the task weights.

3. Experiments

3.1. Experiment Settings

Data We evaluate our methods on the Mandarin ASR task. For unlabeled speech data, AISHELL-2 [20] and WenetSpeech [21] corpus are separately considered. For unlabeled text data, we use M6-Corpus [13] including 292GB texts. For paired speech-text data, we use AISHELL-1 [22] corpus. All speech data is extracted to 80-dimensional log Mel-filterbank features while specAugment [23] and speed perturbation [24] are employed. A 21128-token BERT tokenizer[17] is utilized to tokenize Mandarin texts.

Model configuration We implement MMSpeech based on OFA [25]. We design MMSpeech in two different configurations, namely BASE and LARGE. For BASE architecture, the speech feature extractor is a two-layer convolutional network (CNN) while each layer has 768 channels with strides (2,2) and kernel widths (3,3). The speech encoder, shared encoder and decoder all contain 6 transformer layers with model dimension 768, inner dimension 3072 and 12 attention heads. For LARGE architecture, the transformer layer numbers of the speech encoder/shared encoder/decoder are changed to 12, the model dimensions are changed to 4096 and the attention heads are changed to 16, while other components are kept the same.

Training detail We use the Data2Vec model to generate pseudo-codes for the S2C task by default, which has a similar architecture with MMSpeech-BASE encoder but with a 1D CNN [26] for feature extractor, and are pre-trained with 1000 hours AISHELL-2 audio. To stabilize the multi-task pre-training, we proposed to first train the model with the P2T task until convergence, and then conduct the multi-task pre-training base on it like [10]. For multi-task learning, we adjust the number of samples in a mini-batch to decide the task weights $\lambda_1, \ldots \lambda_5$. The sample rates in a mini-batch for each task are 4:4:2:1:1 for the MSP, S2C, P2T, PP, and S2T tasks, respectively. During inference, we use an external language model (LM) for shallow fusion [27] by default. The LM is implemented as a 12-layers Transformer and trained with the M6-Corpus text data. More details are shown in our code[2].

3.2. Results

We evaluate ASR performance with the word-error-rate (WER) metric. Table 1 presents AISHELL-1 recognition results of models under the BASE configuration. For comparision, we also evaluate the pre-trained Data2Vec metioned in Section 3.1, which is fine-tuned with a CTC loss. The speech data used for all pre-training methods is AISHELL-2 audio. As shown in Table 1, our proposed MMSpeech outperforms the model without pre-training and Data2Vec significantly no matter with or without LM fusion. Furthermore, MMSpeech can directly obtain the excellent recognition results even without fine-tuning (FT) as shown in the fourth row in Table 1.

Table 1: *WER on the AISHELL-1 dev/test set when the unlabeled speech data is AISHELL-2 audio and the architecture is the* BASE.

Model	dev		test	
	w/o LM	with LM	w/o LM	with LM
w/o pre-training	6.4	5.2	6.8	5.7
Data2Vec	3.8	3.7	4.1	3.9
MMSpeech	**2.4**	**2.1**	**2.6**	**2.3**
– w/o FT	2.5	2.3	2.6	2.3

We also compare our methods with the previously published Mandarin pre-training models[3] in Table 2, where the Wav2Vec 2.0[2] and HuBERT[3] are ASR systems which are pre-trained with WenetSpeech audio and fine-tuned on

[2] https://github.com/OFA-Sys/OFA/blob/main/README_mmspeech.md

[3] https://github.com/TencentGameMate/chinese_speech_pretrain

AISHELL-1. Here for MMSpeech, we use the same speech data for pre-training and the above HuBERT-BASE to generate pseudo codes for S2C. Table 2 shows the WER results rescoring with an LM. We achieve improvement from Table 1 since the WenetSpeech dataset contains ten times the audio data of AISHELL-2. Our proposed MMSpeech outpeforms Wav2Vec 2.0[2] and HuBERT[3] under the BASE or LARGE setting. Besides, we obtain the SOTA performance on the AISHELL-1 dev/test set, achieving a relative 48.3%/42.4% WER decrease.

Table 2: *WER on the AISHELL-1 dev/test sets compared with the published models pre-trained with the WenetSpeech audio.*

Model	encoder size	dev	test
Wav2Vec 2.0	BASE	4.2	4.7
HuBERT	BASE	4.1	4.3
MMSpeech	BASE	**2.0**	**2.1**
Wav2Vec 2.0	LARGE	3.8	4.1
HuBERT	LARGE	3.1	3.3
MMSpeech	LARGE	**1.6**	**1.9**

Table 3: *Ablation study based on MMSpeech in Table 1.*

	dev		test	
	w/o LM	with LM	w/o LM	with LM
MMSpeech	**2.4**	**2.1**	**2.6**	**2.3**
– P2T	3.4	2.7	3.8	3.0
– MSP	2.9	2.4	3.2	2.6
– S2C	2.6	2.3	2.8	2.5
– MSP&S2C	2.7	2.4	3.1	2.7
– PP	3.1	2.5	3.5	2.8
– S2T	2.9	2.4	3.3	2.7

3.3. Ablation study

To better understand MMSpeech, we conduct an ablation study by removing different pre-training tasks for multi-task learning. We use the configuration of Table 1 and present the result on the AISHELL-1 dev/test set. As shown in Table 3, we have the following findings: (1) In the second row, we observe significant performance degradation when pre-training without the P2T task. Compared with the fifth row, unlabeled text data plays a more critical role than unlabeled speech data in MMSpeech. It is different from the conclusion of SpeechT5[9], since our unlabeled text data is huge, 292GB, while only 1.8GB of texts are used previously[9, 10]. Even decoding with an LM, introducing the P2T task still contributes an avarage 0.65 WER reduction different from STPT [10], which proves the Mandarin P2T task not only learns linguistic information from text data but also acts as a supplement to the S2T modeling. (2) In the third to fifth row, we present the results without the unlabeled speech tasks. The results in the fourth row prove that the S2C task can benefit the speech-text joint pre-training, while it should be optimized with the MSP task jointly. As shown in the fifth and the third rows, the model pre-trained without unlabeled speech data outperforms the model pre-trained with an additional S2C task when decoding without LM. (3) In the sixth row, we remove the PP task for the joint pretraining, which causes the MSP learning without guidance and significant WER increase. Furthermore, the training doesn't converge when removing the PP and S2C tasks together and we observe that all predictions in the MSP collapse into one or two target phonemes. (4) In the last row, we remove the supervised S2T task during pre-training but keep the same number of training steps. The WER has an avarage 0.6 increase after fine-tuning.

3.4. Analysis

Impact of unsupervised text task. To analyze why the P2T task is effective, we investigate two main factors: text data amount and input features for P2T. As shown in Table 4, both reducing text data amount and replacing P2T with text-infilling (T2T) bring performance degradation. The impact of latter is more significant since phoneme can bridge the gap between Mandarin speech and text. Besides, unlike STPT [10] which conducts experiments on the English dataset, our experiments show the improvement of P2T can not be replaced by an LM when using the same amount of text data (1.8G). It indicates that P2T is more effective to Mandarin tasks due to the larger difference between Mandarin speech and characters.

Generalizability of pretrained models. We evaluate on the AISEHLL-2 test set to validate the generalizability of MMSpeech. We compare our results to the ASR models trained with AISHELL-2 supervised data without pre-training. Table 5 shows that our model without fine-tuning and using only AISHELL-1 (178h) supervised data during pre-training can achieve a comparable results with the model trained with AISHELL-2 (1000h) in the fourth row. Besides, MMSpeech after fine-tuning with the AISHELL-2 train set outperforms the published SOTA results on the AISHELL-2 test set.

Table 4: *Comparision of the unsupervised text tasks. "()" indicates the WER is measured with an external LM.*

Model	unsup-text	dev	test
w/o pre-training	-	6.4 (5.2)	6.8 (5.7)
+ P2T	292G	2.7 (2.4)	3.1 (2.7)
+ P2T	1.8G	3.0 (2.8)	3.5 (3.2)
+ T2T	292G	3.7 (3.6)	4.2 (3.9)

Table 5: *MMSpeech-*BASE *evaluated on AISHELL-2 test set.*

Model	unsup-speech	iOS	Mic	Android
Transformer[4] [7]	-	7.5	8.6	8.9
Conformer[5] [28]	-	5.3	5.6	5.7
MMSpeech	WenetSpeech	**3.9**	**4.5**	**4.0**
- w/o FT	WenetSpeech	6.2	7.1	6.5

4. Conclusion

MMSpeech tries to find a good practice for building a large Mandarin pre-training model. Previous works have overlooked both language differences and the use of unsupervised text data, resulting in a lack of a good Mandarin-optimized pre-training model in the community. We carefully design the pre-training tasks as well as the model architecture so that MMSpeech can utilize a much larger amount of unsupervised text data to mitigate the discrepancy between speech and text caused by the larger number of Homophones in Mandarin compared to English. Experiments show a significant performance improvement (relatively 40% compared to pre-SOTA), which verifies our arguments.

[4] Transformer results from https://github.com/espnet/espnet

[5] SOTA results from https://paperswithcode.com/

5. References

[1] S. Schneider, A. Baevski, R. Collobert, and M. Auli, "wav2vec: Unsupervised pre-training for speech recognition," *Interspeech*, 2019.

[2] A. Baevski, Y. Zhou, A. Mohamed, and M. Auli, "wav2vec 2.0: A framework for self-supervised learning of speech representations," *NeurIPS*, vol. 33, pp. 12 449–12 460, 2020.

[3] W.-N. Hsu, B. Bolte, Y.-H. H. Tsai, K. Lakhotia, R. Salakhutdinov, and A. Mohamed, "Hubert: Self-supervised speech representation learning by masked prediction of hidden units," *TASLP*, vol. 29, pp. 3451–3460, 2021.

[4] A. Baevski, W.-N. Hsu, Q. Xu, A. Babu, J. Gu, and M. Auli, "Data2vec: A general framework for self-supervised learning in speech, vision and language," *ICML*, 2022.

[5] A. Graves, S. Fernández, F. Gomez, and J. Schmidhuber, "Connectionist temporal classification: labelling unsegmented sequence data with recurrent neural networks," in *ICML*, 2006, pp. 369–376.

[6] W. Chan, N. Jaitly, Q. V. Le, and O. Vinyals, "Listen, attend and spell," *arXiv preprint arXiv:1508.01211*, 2015.

[7] A. Mohamed, D. Okhonko, and L. Zettlemoyer, "Transformers with convolutional context for asr," *arXiv preprint arXiv:1904.11660*, 2019.

[8] J. Ao, Z. Zhang, L. Zhou, S. Liu, H. Li, T. Ko, L. Dai, J. Li, Y. Qian, and F. Wei, "Pre-training transformer decoder for end-to-end asr model with unpaired speech data," *arXiv preprint arXiv:2203.17113*, 2022.

[9] J. Ao, R. Wang, L. Zhou, S. Liu, S. Ren, Y. Wu, T. Ko, Q. Li, Y. Zhang, Z. Wei *et al.*, "Speecht5: Unified-modal encoder-decoder pre-training for spoken language processing," *ACL*, 2021.

[10] Y. Tang, H. Gong, N. Dong, C. Wang, W.-N. Hsu, J. Gu, A. Baevski, X. Li, A. Mohamed, M. Auli *et al.*, "Unified speech-text pre-training for speech translation and recognition," *ACL*, 2022.

[11] Z. Zhang, L. Zhou, J. Ao, S. Liu, L. Dai, J. Li, and F. Wei, "Speechut: Bridging speech and text with hidden-unit for encoder-decoder based speech-text pre-training," *arXiv preprint arXiv:2210.03730*, 2022.

[12] W. Zhou, "The homophone effect in mandarin word recognition," Ph.D. dissertation, The Ohio State University, 2015.

[13] J. Lin, R. Men, A. Yang, C. Zhou, M. Ding, Y. Zhang, P. Wang, A. Wang, L. Jiang, X. Jia *et al.*, "M6: A chinese multimodal pretrainer," *arXiv preprint arXiv:2103.00823*, 2021.

[14] Y. Tang, J. Pino, C. Wang, X. Ma, and D. Genzel, "A general multi-task learning framework to leverage text data for speech to text tasks," in *ICASSP*, 2021.

[15] F. Wu, K. Kim, S. Watanabe, K. Han, R. McDonald, K. Q. Weinberger, and Y. Artzi, "Wav2seq: Pre-training speech-to-text encoder-decoder models using pseudo languages," *arXiv preprint arXiv:2205.01086*, 2022.

[16] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *NeurIPS*, vol. 30, 2017.

[17] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *NAACL-HLT*, 2018.

[18] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, and L. Zettlemoyer, "Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension," *ACL*, 2019.

[19] P. Gage, "A new algorithm for data compression," *C Users Journal*, vol. 12, no. 2, pp. 23–38, 1994.

[20] J. Du, X. Na, X. Liu, and H. Bu, "Aishell-2: Transforming mandarin asr research into industrial scale," *arXiv preprint arXiv:1808.10583*, 2018.

[21] B. Zhang, H. Lv, P. Guo, Q. Shao, C. Yang, L. Xie, X. Xu, H. Bu, X. Chen, C. Zeng *et al.*, "Wenetspeech: A 10000+ hours multi-domain mandarin corpus for speech recognition," in *ICASSP*. IEEE, 2022, pp. 6182–6186.

[22] H. Bu, J. Du, X. Na, B. Wu, and H. Zheng, "Aishell-1: An open-source mandarin speech corpus and a speech recognition baseline," in *O-COCOSDA*. IEEE, 2017, pp. 1–5.

[23] D. S. Park, W. Chan, Y. Zhang, C.-C. Chiu, B. Zoph, E. D. Cubuk, and Q. V. Le, "Specaugment: A simple data augmentation method for automatic speech recognition," *Interspeech*, 2019.

[24] T. Ko, V. Peddinti, D. Povey, M. L. Seltzer, and S. Khudanpur, "A study on data augmentation of reverberant speech for robust speech recognition," in *ICASSP*. IEEE, 2017, pp. 5220–5224.

[25] P. Wang, A. Yang, R. Men, J. Lin, S. Bai, Z. Li, J. Ma, C. Zhou, J. Zhou, and H. Yang, "Unifying architectures, tasks, and modalities through a simple sequence-to-sequence learning framework," *ICML*, 2022.

[26] C. Wang, Y. Tang, X. Ma, A. Wu, D. Okhonko, and J. Pino, "fairseq s2t: Fast speech-to-text modeling with fairseq," in *AACL: System Demonstrations*, 2020.

[27] C. Gulcehre, O. Firat, K. Xu, K. Cho, L. Barrault, H.-C. Lin, F. Bougares, H. Schwenk, and Y. Bengio, "On using monolingual corpora in neural machine translation," *arXiv preprint arXiv:1503.03535*, 2015.

[28] A. Gulati, J. Qin, C.-C. Chiu, N. Parmar, Y. Zhang, J. Yu, W. Han, S. Wang, Z. Zhang, Y. Wu *et al.*, "Conformer: Convolution-augmented transformer for speech recognition," *Interspeech*, 2020.