BUILDING ROBUST SPOKEN LANGUAGE UNDERSTANDING BY CROSS ATTENTION BETWEEN PHONEME SEQUENCE AND ASR HYPOTHESIS

Zexun Wang[1†], *Yuquan Le*[1,2†], *Yi Zhu*[1,3], *Yuming Zhao*[1], *Mingchao Feng*[1], *Meng Chen*[1], *Xiaodong He*[1]

[1]JD AI, Beijing, China, [2]Hunan University, [3]LTL, University of Cambridge
leyuquan@hnu.edu.cn, yz568@cam.ac.uk
{wangzexun3,zhaoyuming3,fengmingchao,chenmeng20,xiaodong.he}@jd.com

ABSTRACT

Building Spoken Language Understanding (SLU) robust to Automatic Speech Recognition (ASR) errors is an essential issue for various voice-enabled virtual assistants. Considering that most ASR errors are caused by phonetic confusion between similar-sounding expressions, intuitively, leveraging the phoneme sequence of speech can complement ASR hypothesis and enhance the robustness of SLU. This paper proposes a novel model with **C**ross **A**ttention for **SLU** (denoted as **CASLU**). The cross attention block is devised to catch the fine-grained interactions between phoneme and word embeddings in order to make the joint representations catch the phonetic and semantic features of input simultaneously and for overcoming the ASR errors in downstream natural language understanding (NLU) tasks. Extensive experiments are conducted on three datasets, showing the effectiveness and competitiveness of our approach. Additionally, We also validate the universality of CASLU and prove its complementarity when combining with other robust SLU techniques.

Index Terms— Spoken Language Understanding, NLU Robustness, Cross Attention Network, Phoneme Embedding

1. INTRODUCTION

Spoken Language Understanding (SLU) is the critical technology of voice-enabled virtual assistants, e.g. Apple Siri and Amazon Alexa. It serves as a bridge to allow machines to interact with humans effectively and has obtained increasing attention in recent years. The SLU generally involves two main modules, Automatic Speech Recognition (ASR) and Natural Language Understanding (NLU). The ASR engine is utilized to transcribe human speech into the text. Then the NLU module is applied to ASR output to comprehend the user's requests, typically including intent classification and slot filling tasks. Owing to the remarkable success in various fields, deep learning techniques are widely explored for SLU [1, 2, 3]. Although these methods have made rapid progress, the performance is still compromised inevitably when facing ASR errors [4]. Hence, building SLU robust to ASR errors is essential for improving end-user experience in virtual assistants.

Previous works related to robust SLU mainly deal with ASR transcripts directly. Various approaches have been investigated to correct ASR hypothesis [5, 6], or leverage ASR output information [7, 8, 9, 10], especially N-best hypothesis [11, 12, 13, 14] directly within downstream NLU model. Despite the success of these methods, using only texts generated by ASR module unavoidably loses useful speech information like pronunciation and prosody. Moreover, many ASR errors are even elicited by phonetic confusion of similar-sounding words that are incorrectly transcribed to each other. For example, "*buy a computer*" may be mis-recognized to "*by a computer*", which would confuse the downstream task. To this end, there have been research efforts drawing upon speech information exploiting different forms to improve SLU robustness [15, 16, 17, 18, 19, 20]. Among them, phoneme is considered as a clean form[1] of speech representation complementary to text. As the atom speech unit of a language, phoneme can capture complex phonetic properties and interactions, so the ASR transcription in phoneme level should be more similar to the correct utterance than character or word level [19], and the Phoneme Error Rate (PER) will be smaller than Character Error Rate (CER) and Word Error Rate (WER).

Inspired by this, we propose a novel deep learning-based model CASLU to harness phoneme-level information to complement text for more robust SLU. In particular, we utilize cross-attention [21] to explicitly model the fine-grained interactions between ASR phonemes and hypotheses. To the best of our knowledge, there has been rare work exploring this before. The main contributions of this paper are two-fold: (1) We propose CASLU to explicitly capture the fine-grained correlations between ASR phonemes and hypotheses for robust SLU. (2) Experimental results on three datasets demonstrate the effectiveness and competitiveness of CASLU. We also validate the universality of CASLU with different text and phoneme encoders, and prove its complementarity combining with other robust SLU techniques.

† Equal contribution.

[1]For example, audio signals or features may vary greatly across people and could be distorted by different noise sources.

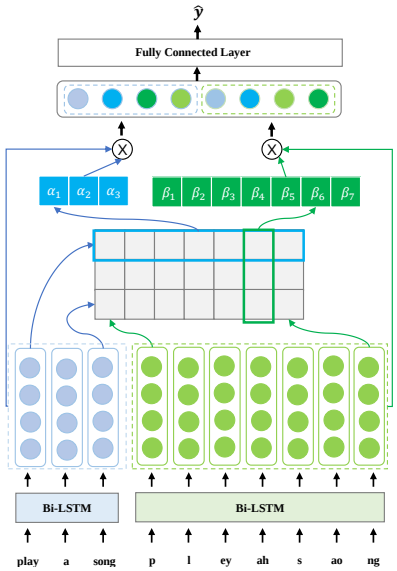


Fig. 1. The architecture of our proposed model CASLU.

2. MODEL

This section introduces our proposed model CASLU in detail (see Figure 1). The input is an utterance X which contains a sequence of words as well as its corresponding phoneme sequence: $X^w = [x_1^w, x_2^w, ..., x_m^w]$ and $X^p = [x_1^p, x_2^p, ..., x_n^p]$, where $x_i^w \in V_w$, V_w is the word vocabulary, m is the sequence length of text sequence, $x_j^p \in V_p$, V_p is the phoneme vocabulary, and n is the length of the phoneme sequence. The goal of the model is to interpret the user's utterance request to its corresponding class y.

2.1. Text and phoneme Encoder

The text encoder utilizes Bi-LSTM [22] to extract word information in context. Each word x_i^w in X^w is mapped into its corresponding word vector $\mathbf{e}_i^w \in \mathbb{R}^{d_w}$, where d_w is the dimensionality of word embedding. Then a Bi-LSTM encoder computes contextualized hidden representations as follows:

$$\overrightarrow{\mathbf{h}_i^w} = \overrightarrow{\mathbf{LSTM}}(\mathbf{e}_i^w),\ \overleftarrow{\mathbf{h}_i^w} = \overleftarrow{\mathbf{LSTM}}(\mathbf{e}_i^w),\ \mathbf{h}_i^w = [\overrightarrow{\mathbf{h}_i^w}, \overleftarrow{\mathbf{h}_i^w}] \quad (1)$$

where $\overrightarrow{\mathbf{h}_i^w}$ and $\overleftarrow{\mathbf{h}_i^w}$ are the forward and backward hidden states for word x_i^w. $\mathbf{h}_i^w$, the concatenation of hidden states in both directions, is used as the word hidden representations.

Similar to text encoder, the phoneme sequence is fed into phoneme encoder with another Bi-LSTM to acquire contextualized phoneme hidden representations $\mathbf{h}_j^p$ for each phoneme. The process is similar to Equation 1.

2.2. Interaction Layer

After encoding text and phoneme sequences, it is now crucial to fuse their representations in order to effectively leverage information from both sides. The cross attention block has been proved to be capable of capturing fine-grained correlation of any pixel pair between two images [21]. Motivated by this, we first calculate a correlation map $\mathbf{C} \in \mathbb{R}^{m \times n}$ between word hidden representations $\mathbf{h}_i^w$ and phoneme hidden representations $\mathbf{h}_j^p$ through cosine distance:

$$C_{ij} = \frac{(\mathbf{h}_i^w)^T \mathbf{h}_j^p}{||\mathbf{h}_i^w||_2 ||\mathbf{h}_j^p||_2} \quad (2)$$

where C_{ij} can be seen as the semantic relevance between word x_i^w and phoneme x_j^p, and the matrix $\mathbf{C}$ characterizes the fine-grained correlations among words and phonemes. Specifically, the vector of the i-th row in $\mathbf{C}$:

$$\mathbf{r}_i = \begin{bmatrix} C_{i,1} & C_{i,2} & ... & C_{i,n} \end{bmatrix}^T, \mathbf{r}_i \in \mathbb{R}^n \text{ and } i \in \{1, ..., m\}$$

represents the correlations between the i-th word x_i^w and *every* phoneme in the phoneme sequence X^p. Correspondingly, the j-th column vector $\mathbf{c}_j \in \mathbb{R}^m$ denotes the relationship between the j-th phoneme x_j^p and the whole word sequence X^w.

We further apply convolution operation to *each* row and column vector of $\mathbf{C}$ individually to fuse local correlations between words and phonemes into cross attention weight. Taking the row vector $\mathbf{r}_i$ as an example, which represents interactions of the word x_i^w over phoneme sequence, its attention weight α_i is calculated by employing a convolution layer with a single text kernel $\mathbf{k}^w \in \mathbb{R}^n$ followed by softmax function:

$$\alpha_i = \frac{\exp(\mathbf{k}^{w^T} \mathbf{r}_i)}{\sum_{i'=1}^{m} \exp(\mathbf{k}^{w^T} \mathbf{r}_{i'})}, i \in \{1, ..., m\} \quad (3)$$

Similarly, a phoneme kernel $\mathbf{k}^p \in \mathbb{R}^m$ operates on column vectors $\mathbf{c}_j$ to obtain the attentions β_j for all input phonemes.[2]

Finally, after aggregating the feature map $\mathbf{C}$ into attentions for all words and phonemes with convolution kernels, the *phonetic-aware text representation* $\mathbf{t}$ and *lexical-aware phoneme representation* $\mathbf{p}$ can be computed as follows:

$$\mathbf{t} = \mathbf{h}^w \boldsymbol{\alpha}, \quad \mathbf{p} = \mathbf{h}^p \boldsymbol{\beta} \quad (4)$$

where $\mathbf{h}^w = [\mathbf{h}_1^w, ..., \mathbf{h}_m^w]$ stacks all the word hidden representations and $\boldsymbol{\alpha} = [\alpha_1, ..., \alpha_m]^T$, and $\mathbf{h}^p$ and $\boldsymbol{\beta}$ are defined analogously.

We concatenate $\mathbf{t}$ and $\mathbf{p}$ and feed the resulted vector into a fully connected layer with *softmax* function to predict the probability of intent class. The standard cross-entropy is applied to calculate the classification loss.

Datasets	Train	Dev	Test	Intents
Snips	11769	1312	700	7
TREC	4904	548	500	6
Waihu	7141	906	886	16

Table 1. Dataset statistics.

[2]In practice, the length m and n of kernels $\mathbf{k}^w$ and $\mathbf{k}^p$ are the maximum lengths of all text and phoneme sequences.

Models	Snips		TREC		Waihu	
	Trans(%)	ASR(%)	Trans(%)	ASR(%)	Trans(%)	ASR(%)
B1: Bi-LSTM w/ trs	**96.24**	73.10	84.13	62.27	72.16	67.95
B2: Bi-LSTM w/ asr	94.57	90.78	83.40	69.40	70.09	68.47
B3: Multi-input [18]	94.50	92.00	83.60	72.00	70.69	68.89
CASLU	95.50	**92.57**	**84.47**	**73.60**	**73.55**	**69.22**
CASLU w/o $\mathbf{t}$	92.14	91.63	82.60	71.35	70.15	68.79
CASLU w/o $\mathbf{p}$	94.71	91.43	81.40	71.20	73.13	68.62

Table 2. Performance comparison in accuracy of CASLU and other baselines. *Trans* and *ASR* mean evaluating on transcription x_{trs} and 1-best ASR hypothesis x_{asr} respectively. *CASLU w/o* $\mathbf{t}$ *(*$\mathbf{p}$*)* denotes omitting $\mathbf{t}$ ($\mathbf{p}$) in the final prediction layer.

3. EXPERIMENTS

3.1. Dataset

Experiments are conducted on three datasets, and their statistics are shown in Table 1: (1) **Snips** is a benchmark dataset for SLU intent detection. It comprises pairs of user commands and intents such as GetWeather and PlayMusic. We held out a validation set from the training set. (2) **TREC** is a question classification dataset containing six fact-based question types such as HUMAN and LOCATION. (3) **Waihu** is a Chinese dataset for intent classification, which was collected from our online voice-enabled customer service bot[3], so the audio is from real users and the ASR hypotheses and phoneme sequences are generated by our online ASR engine. As for Snips and TREC with only text, since speech transcription and annotation are expensive and labor-intensive, we follow a similar strategy as described in [18] to create noisy speech corpus form them. We use Amazon Polly[4] to convert the raw text to speech and apply Speech Synthesis Markup Language (SSML) tags and ambient noise[5] to get noisy speech data. Then, we use Amazon Transcribe[6] to transcribe the audio. Since our hypothesis is to create a model that is robust to ASR errors, we keep only hypotheses containing ASR errors. Therefore, the phoneme sequences from ASR systems should link the phonemes that the system often confuses.

3.2. Training Details

For all datasets, the maximum length of text and phoneme sequence is set to 40 and 80 respectively. Masks are applied to zero out the effect of paddings. The model uses one layer of Bi-LSTM with 150 hidden nodes. The embeddings of word and phoneme are randomly initialized for simplicity. The batch-size is set to 64. Adam [23] is used as the optimizer with a learning rate of 0.001. We train the model for 20 epochs and choose the best model on the validation set. We run three trials for each experiment with different random seeds and report the average score to avoid bias introduced by training randomness.

[3]http://yanxi.jd.com.

[4]https://aws.amazon.com/cn/polly.

[5]www.pacdv.com/sounds/ambience_sounds.html.

[6]https://aws.amazon.com/transcribe.

3.3. Main results

For evaluation, we use classification *accuracy* as metric. We test on both manual transcriptions x_{trs} and ASR hypotheses x_{asr} on account of the criterion from [15] that *model robustness towards ASR errors is only improved given increased performance tested on a set with ASR errors and no performance degradation on a set without ASR errors*. Experimental results are reported in Table 2. The **B1** and **B2** are the baselines that use *only* manual transcriptions (clean text) and 1-best ASR hypotheses (noisy text) as input respectively. The **B3** concatenates $\mathbf{t}$ and $\mathbf{p}$ averaged over $\mathbf{h}^w$ and $\mathbf{h}^p$ rather than the weighted sum in Equation 4. It's a special case of cross-attention where attention weights are just uniform. In the first part of Table 2, we observed that: (1) CASLU comprehensively beats all baselines on the test sets of ASR hypotheses ($p<0.05$); (2) When evaluating on manual transcriptions, CASLU even performs better on TREC and Waihu with only a slight drop on Snips. Both results indicate our model enhanced by phoneme sequence can achieve stronger robustness to ASR errors. The second part shows the ablation study. It has the same architecture as the full CASLU, except that it only feeds the representation of phoneme/text into the final classifier layer. After removing either text or phoneme representation, the performance degrades in both scenarios, which suggests that representations in both modalities complement each other. However, it is noteworthy that CASLU w/o $\mathbf{t}$ or CASLU w/o $\mathbf{p}$ are both better than B2 on ASR hypotheses, which means that after fine-grained interaction, either phonetic-aware text representation or lexical-aware phoneme representation has fused the information from the other.

3.4. Discussion

Universality of our method. To validate the effectiveness of CASLU on other neural structures, here we implement other variant models by replacing Bi-LSTM encoder with GRU/LSTM/Bi-GRU/CNN. We keep the model structure and hyper-parameters consistent for fair comparison. For simplicity, we perform experiments on Snips and Waihu datasets. Table 3 shows that our proposed models still outperform the

		Snips		Waihu	
		Trans(%)	ASR(%)	Trans(%)	ASR(%)
GRU	w/ asr	94.21	90.31	65.46	63.66
	Multi-input	94.53	90.50	67.23	65.35
	CASLU	**95.17**	**91.55**	**70.62**	**67.47**
LSTM	w/ asr	94.86	89.84	65.23	62.53
	Multi-input	95.14	90.67	66.76	64.80
	CASLU	**95.35**	**91.26**	**68.84**	**66.89**
Bi-GRU	w/ asr	94.43	91.33	70.03	68.28
	Multi-input	95.10	91.71	72.80	69.47
	CASLU	**95.24**	**92.38**	**73.36**	**70.71**
CNN	w/ asr	95.29	91.95	73.55	70.13
	Multi-input	95.95	92.10	73.65	70.81
	CASLU	**96.29**	**92.57**	**73.97**	**70.88**

Table 3. Test accuracy of variant models with other encoder architectures, shown in the leftmost column.

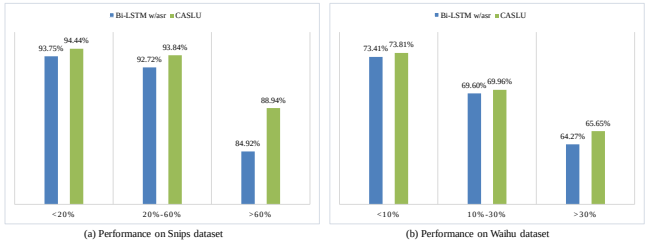
Models	Snips	Waihu
CASLU	92.57	69.22
CASLU + VAT	92.67	**71.75**
CASLU + N-Best	**93.48**	69.90

Table 4. Test accuracy (%) of the combined methods. (Sign Test, with p-value$<$0.05)

corresponding baselines, which demonstrates that our method can be a unified framework for SLU.

Combining with other techniques. Recently, many attractive techniques of robust SLU have emerged. These techniques focus on utilizing more information, such as the classification probability distribution [15] or ASR N-best hypotheses [11]. In [15], virtual adversarial training (VAT) is applied with a Kullback–Leibler (KL) divergence term added to minimize the distance between predicted label distributions of transcriptions and ASR hypotheses to train a robust SLU model. In [11], the authors uses N-best ASR hypotheses by concatenating their texts or embeddings to improve the SLU system robustness. It is very meaningful and interesting to study whether our method can combine with them and get further gain. Therefore, we implement two combined models: CASLU+VAT and CASLU+N-best. CASLU+VAT is the integration of CASLU and VAT with 1-best hypothesis as adversarial example. CASLU+N-best exploits N-best ASR hypothesis texts and phonemes as input, contrary to just using top ASR result. In both settings the numbers of parameters are the same as CASLU, and the difference lies in training objectives and input data. Table 4 shows that both models have achieved further improvements on the two datasets, which corroborates the necessity of phoneme information and the complementarity of our method.

Performance at different WER ranges. With the fusion of phoneme and text information, it is conceivable that our model should be more robust when there are more errors in ASR hypothesis. We verify this assumption by stratifying the Snips and Waihu test sets into three buckets based on the WER score of each instance. Figure 2 shows the results



(a) Performance on Snips dataset (b) Performance on Waihu dataset

Fig. 2. Performance comparison in accuracy of Bi-LSTM w/asr (B2) and CASLU at different WER ranges.

Transcription text: i want to add a song by jazz brasileiro

Transcription phoneme: ay w-aa-n-t t-uw *ae-d* ah s-ao-ng b-ay jh-ae-z b-r-ae-s-ah-l-iy-r-ow

ASR text: i want to i had a song by just presented right

ASR phoneme: ay w-aa-n-t t-uw ay *hh-ae-d* ah s-ao-ng b-ay jh-ah-s-t p-r-iy-z-eh-n-t-ah-d r-ay-t

Ground-Truth: AddToPlaylist ***Bi-LSTM w/ asr:*** PlayMusic

Multi-input: PlayMusic ***CASLU:*** AddToPlaylist

Table 5. An Example with incorrect ASR hypothesis, correctly classified using CASLU model.

of CASLU and its counterpart without phoneme input (B2: Bi-LSTM w/ asr) on the three buckets. We can see that not only does CASLU outperform B2 at all ranges, but the margin becomes also larger when WER increases. For instance, CASLU is better than B2 by over four absolute points on Snips when WER is over 60%, which demonstrates the strong robustness of our model to ASR errors.

Case Study. Table 5 presents an example where the ASR hypothesis contains errors, but the similarity between *ae-d* and *hh-ae-d* in the phoneme sequence helps model for the correct classification. This clearly manifests the benefit of introducing and fusing phoneme information in our model.

4. CONCLUSIONS

In this paper, we propose a novel method to enhance the robustness of SLU by fusing the information from phoneme sequence and ASR hypothesis. A fine-grained interaction module based on cross attention is devised to obtain phonetic-aware text representations and lexical-aware phoneme representations. Then two complementary representations are combined seamlessly to pursue better NLU performance. Extensive experiments were conducted to prove the effectiveness and versatility of our method. In the future, we will explore more information fusion approaches to facilitate this task.

5. ACKNOWLEDGEMENT

This work is supported by the National Key R&D Program of China under Grant No. 2020AAA0108600.

6. REFERENCES

[1] Kaisheng Yao, Baolin Peng, Yu Zhang, Dong Yu, Geoffrey Zweig, and Yangyang Shi, "Spoken language understanding using long short-term memory neural networks," in *SLT*. IEEE, 2014, pp. 189–194.

[2] Chih Wen Goo, Guang Gao, Yun Kai Hsu, Chih Li Huo, Tsung Chieh Chen, Keng Wei Hsu, and Yun Nung Chen, "Slot-gated modeling for joint slot filling and intent prediction," in *NAACL*, 2018, pp. 753–757.

[3] Grégoire Mesnil, Yann Dauphin, Kaisheng Yao, Yoshua Bengio, Li Deng, Dilek Hakkani-Tur, Xiaodong He, Larry Heck, Gokhan Tur, Dong Yu, et al., "Using recurrent neural networks for slot filling in spoken language understanding," in *TASLP*. IEEE, 2014, pp. 530–539.

[4] Atsunori Ogawa and Takaaki Hori, "Error detection and accuracy estimation in automatic speech recognition using deep bidirectional recurrent neural networks," *Speech Communication*, vol. 89, pp. 70–83, 2017.

[5] Yik-Cheung Tam, Yun Lei, Jing Zheng, and Wen Wang, "ASR error detection using recurrent neural network language model and complementary ASR," in *ICASSP*, 2014, pp. 2312–2316.

[6] Yue Weng, Sai Sumanth Miryala, Chandra Khatri, Runze Wang, Huaixiu Zheng, Piero Molino, Mahdi Namazifar, Alexandros Papangelis, Hugh Williams, Franziska Bell, et al., "Joint contextual modeling for asr correction and language understanding," in *ICASSP*. IEEE, 2020, pp. 6349–6353.

[7] Faisal Ladhak, Ankur Gandhe, Markus Dreyer, Lambert Mathias, Ariya Rastrow, and Björn Hoffmeister, "Latticernn: Recurrent neural networks over lattices.," in *INTERSPEECH*, 2016, pp. 695–699.

[8] Chao Huang and Yun Chen, "Learning spoken language representations with neural lattice language modeling," in *ACL*, 2020, pp. 3764–3769.

[9] Gökhan Tür, Anoop Deoras, and Dilek Hakkani-Tür, "Semantic parsing using word confusion networks with conditional random fields.," in *INTERSPEECH*. Citeseer, 2013, pp. 2579–2583.

[10] Prashanth Gurunath Shivakumar, Mu Yang, and Panayiotis Georgiou, "Spoken language intent detection using confusion2vec," *arXiv preprint arXiv:1904.03576*, 2019.

[11] Mingda Li, Weitong Ruan, Xinyue Liu, Luca Soldaini, Wael Hamza, and Chengwei Su, "Improving spoken language understanding by exploiting asr n-best hypotheses," *arXiv preprint arXiv:2001.05284*, 2020.

[12] Xinyue Liu, Mingda Li, Luoxin Chen, Prashan Wanigasekara, Weitong Ruan, Haidar Khan, Wael Hamza, and Chengwei Su, "Asr n-best fusion nets," in *ICASSP*. IEEE, 2021, pp. 7618–7622.

[13] Atsunori Ogawa, Marc Delcroix, Shigeki Karita, and Tomohiro Nakatani, "Rescoring n-best speech recognition list based on one-on-one hypothesis comparison using encoder-classifier model," in *ICASSP*. IEEE, 2018, pp. 6099–6103.

[14] A. Ogawa, M. Delcroix, S. Karita, and T. Nakatani, "Improved deep duel model for rescoring n-best speech recognition list using backward lstmlm and ensemble encoders," in *INTERSPEECH*, 2019, pp. 3900–3904.

[15] Weitong Ruan, Yaroslav Nechaev, Luoxin Chen, Chengwei Su, and Imre Kiss, "Towards an asr error robust spoken language understanding system," in *INTERSPEECH*, 2020, pp. 901–905.

[16] Dmitriy Serdyuk, Yongqiang Wang, Christian Fuegen, Anuj Kumar, Baiyang Liu, and Yoshua Bengio, "Towards end-to-end spoken language understanding," in *ICASSP*. IEEE, 2018, pp. 5754–5758.

[17] Loren Lugosch, Mirco Ravanelli, Patrick Ignoto, Vikrant Singh Tomar, and Yoshua Bengio, "Speech model pre-training for end-to-end spoken language understanding," in *INTERSPEECH*, 2019.

[18] Anjie Fang, Simone Filice, Nut Limsopatham, and Oleg Rokhlenko, "Using phoneme representations to build predictive models robust to asr errors," in *SIGIR*, 2020, pp. 699–708.

[19] Mukuntha Narayanan Sundararaman, Ayush Kumar, and Jithendra Vepa, "Phoneme-bert: Joint language modelling of phoneme sequence and asr transcript," *arXiv preprint arXiv:2102.00804*, 2021.

[20] Minjeong Kim, Gyuwan Kim, Sang-Woo Lee, and Jung-Woo Ha, "St-bert: Cross-modal language model pre-training for end-to-end spoken language understanding," in *ICASSP*, 2021, pp. 7478–7482.

[21] Ruibing Hou, Hong Chang, Bingpeng Ma, Shiguang Shan, and Xilin Chen, "Cross attention network for few-shot classification," in *NIPS*, 2019, pp. 4005–4016.

[22] Sepp Hochreiter and Jürgen Schmidhuber, "Long short-term memory," *Neural computation*, vol. 9, no. 8, pp. 1735–1780, 1997.

[23] Diederik P. Kingma and Jimmy Ba, "Adam: A method for stochastic optimization," in *ICLR*, 2015.