Robust Spoken Language Understanding with Acoustic and Domain Knowledge

Hao Li[*]
Chen Liu[*]
Shanghai Jiao Tong University, China
lh575526@sjtu.edu.cn
chris-chen@sjtu.edu.cn

Su Zhu
Shanghai Jiao Tong University, China
paul2204@sjtu.edu.cn

Kai Yu[†]
Shanghai Jiao Tong University, China
kai.yu@sjtu.edu.cn

ABSTRACT

Spoken language understanding (SLU) converts user utterances into structured semantic forms. There are still two main issues for SLU: robustness to ASR-errors and the data sparsity of new and extended domains. In this paper, we propose a robust SLU system by leveraging both acoustic and domain knowledge. We extract audio features by training ASR models on a large number of utterances without semantic annotations. For exploiting domain knowledge, we design lexicon features from the domain ontology and propose an error elimination algorithm to help predicted values recovered from ASR-errors. The results of CATSLU challenge show that our systems can outperform all of the other teams across four domains.

CCS CONCEPTS

• **Computing methodologies → Natural language processing**.

KEYWORDS

Spoken Language Understanding, Robustness

ACM Reference Format:
Hao Li, Chen Liu, Su Zhu, and Kai Yu. 2019. Robust Spoken Language Understanding with Acoustic and Domain Knowledge. In *2019 International Conference on Multimodal Interaction (ICMI '19), October 14–18, 2019, Suzhou, China.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3340555.3356100

1 INTRODUCTION

The spoken language understanding (SLU) is crucial for voice user interfaces, with the widespread adoption of smart devices like Google Home, Amazon Alexa, Apple Siri, and Microsoft Cortana. It is usually designed as a pipeline structure. An automatic speech recognition (ASR) module converts the audio signal into the text, followed by an SLU module parsing it into the corresponding meaning representations for certain narrow domains. Most of the previous works focus on semantic parsing on a given text by ignoring speech recognition errors [3, 11, 14, 16, 18, 23]. Although the speech recognized text contains ASR uncertainties[6, 7, 19, 22], it is still insufficient for SLU by using only textual features[5]. Besides ASR-errors, the data sparsity problem for new and extended domains become a new challenge[2].

In this paper, we proposed a robust SLU system, leveraging both acoustic and domain knowledge. We first formulate the SLU as a joint sequence labelling and sentence classification task. For acoustic knowledge, we train an end-to-end ASR model with a large number of utterances without semantic annotations, which provides high-quality audio features. Both audio and textual features are exploited to improve the robustness of the SLU model to ASR-errors. For domain knowledge, we design lexicon features from the domain ontology to improve the generalization capability for unseen values. We also proposed an error elimination algorithm depending on the domain ontology to help recover the predicted values from ASR-errors.

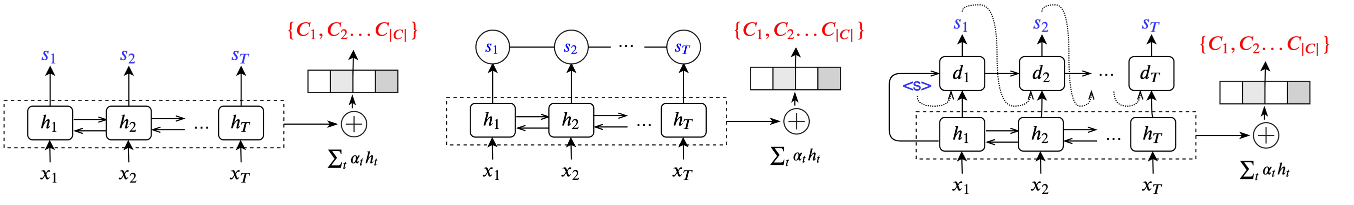
Our approaches are evaluated in the CATSLU [24] challenge with four different domains. Our systems outperform all of the other teams across four domains in terms of the joint accuracy and win two first places in terms of F1-score.

2 PROBLEM FORMULATION

Our goal is to learn an SLU parser from instances of user utterances paired with their structured meaning representations in a narrow domain. The user utterance consists of the speech signal, text recognized by an ASR system and text transcribed by humans. Let $o = o_1 \cdots o_{|o|}$ denote a speech signal, $a = a_1 \cdots a_{|a|}$ its ASR text (it may contain ASR errors as shown in Table 1), $u = u_1 \cdots u_{|u|}$ its human-transcript,

[*] Both authors contributed equally to this research.
[†] The corresponding author is Kai Yu.

ICMI '19, October 14–18, 2019, Suzhou, China
© 2019 Association for Computing Machinery.
ACM ISBN 978-1-4503-6860-5/19/10…$15.00
https://doi.org/10.1145/3340555.3356100



(a) BLSTM (b) BLSTM-CRF (c) BLSTM-focus

Figure 1: Model architectures for joint sequence labelling and sentence classification.

Table 1: A data instance of user utterance (human-transcript and ASR text) and its semantic annotations.

u	What's the weather of Suzhou
a	What's the weather off Suizhou
y	inform(city="Suzhou");request(weather)
s, c	What's:O the:O weather:O of:O Suzhou:B-inform(city) => request(weather)

and $y = y_1 \cdots y_{|y|}$ its meaning representation. We wish to estimate $p(y|o, a, u)$, while u is not valid in the test stage.

The label y is annotated on the human-transcript u, which contains a set of *act(slot=value)* triples[1], as shown in Table 1. Since each value appears in the human-transcript, we can make an alignment for each *act(slot=value)* triple. Thus, y is converted into two parts: a sequence of tags[2] for each word in u, $s = s_1 \cdots s_{|u|}$, and a set of sentence classes $c = \{c_1 \cdots c_{|c|}\}$ (i.e. triples without value), as shown in Table 1. Therefore, we decompose $p(y|o, a, u)$ into a two-stage prediction process:

- Joint sequence labelling and sentence classification: We train the joint model by estimating $p(s, c|o, u)$, and replace u with a in the test stage.
- ASR-error elimination for predicted values: In the test stage, values of extracted *act(slot=value)* triples may contain ASR errors. It is essential to eliminate the ASR errors to obtain true values.

3 AUDIO-TEXTUAL SLU

Joint sequence labelling and sentence classification on text

Basic models for single domain. Long short-term memory (LSTM) based models are applied in joint training of sequence labelling and sentence classification [11, 20]. Let $x = (x_1 \cdots x_T)$ denote the input word sequence which is human-transcript u at the training stage and ASR text a at the test stage. We use bidirectional LSTM (BLSTM) [4] to capture both past and future contextual information. Each word x_t is mapped to a fixed-dimensional vector by a word embedding function $\psi(\cdot)$, then the hidden vectors are recursively computed at the t-th time step via:

$$\overrightarrow{\mathbf{h}}_t = \mathrm{f}_{\mathrm{LSTM}}(\psi(x_t), \overrightarrow{\mathbf{h}}_{t-1}), t = 1, \cdots, T \tag{1}$$

$$\overleftarrow{\mathbf{h}}_t = \mathrm{b}_{\mathrm{LSTM}}(\psi(x_t), \overleftarrow{\mathbf{h}}_{t+1}), t = T, \cdots, 1 \tag{2}$$

$\mathbf{h}_t = [\overrightarrow{\mathbf{h}}_t; \overleftarrow{\mathbf{h}}_t]$, where $[\cdot;\cdot]$ denotes vector concatenation, and $\mathrm{f}_{\mathrm{LSTM}}$ and $\mathrm{b}_{\mathrm{LSTM}}$ are the LSTM functions for forward and backward passes respectively. As illustrated in figure 1, three models are applied for the sub-task of sequence labelling:

- **BLSTM**: Each $\mathbf{h}_t$ is utilized to compute the probability distribution over the slot labels by a linear output layer and the *softmax* function.
- **BLSTM-CRF**: A CRF (conditional random field) layer is added on the top of the slot output layer to model the label dependency.
- **BLSTM-focus**: The focus mechanism [23] with an LSTM decoder is added on the top of BLSTM encoder, which can also help build the label dependency.

We build another linear output layer for the sub-task of sentence classification. Its input is a weighted sum of the BLSTM hidden vectors, i.e., $\sum_{t=1}^{T} \alpha_t \cdot h_t$. The weights are calculated by a typical attention mechanism [1], i.e., $\alpha_t = \frac{\exp(a(h_T, h_t))}{\sum_{i=1}^{T} \exp(a(h_T, h_i))}$, where a is a feed-forward neural network. Since an utterance may have multiple classes, we use the *sigmoid* function to estimate the existence probability for each class label.

Domain adaptation. For domain adaptation, we consider both domain-invariant and domain-specific features [8, 9, 12, 22]. We use two BLSTMs to capture the domain-invariant and domain-specific features, respectively. Following [8], the shared BLSTM are also trained adversarially to produce domain agnostic representations.

Leveraging domain ontology

Lexicon features. In the ontology of each domain, there are several lexicons contain all possible values of certain slots. To leverage the domain ontology, we design *N-gram* lexicon features as external features besides word embeddings.

[1] All possible values for each slot are predefined in the domain ontology. Note that the value of each triple can be empty, e.g. "request(weather)".
[2] A value may consist of continuous words, so Inside/Outside/Beginning (IOB) representation is exploited.

Let $\mathcal{V}_l = (v_1 \cdots v_{M_l})$ denote a lexicon of the current domain, where M_l is the number of values. Each value v_i can be represented as a word sequence $(w_{i,1} \cdots w_{i,|v_i|})$. Thus, the n-gram candidates of $\mathcal{V}_l$ can be represented as a set, $\mathcal{V}_l^n = \{(w_{i,j} \cdots w_{i,j+n-1}) \mid i = 1, \cdots, M_l;\ j = 1, \cdots, |v_i| - n + 1\}$.

Given an utterance $x = (x_1 \cdots x_T)$, we have an n-gram starts at the t-th time step, i.e., $(x_t \cdots x_{t+n-1}), t = 1, \cdots, T$. The indication of whether $(x_t \cdots x_{t+n-1})$ exists in $\mathcal{V}_l^n$ returns a binary feature at the t-th time step. We consider n from 1 to 3. If the current domain has L lexicons, the dimension of lexicon features should be $n \times L$.

ASR-error elimination for predicted values. In the test stage, ASR texts are fed into the model to get aligned labels, which can produce the corresponding *act(slot= value)* triples. However, ASR-errors may be retained in the predicted values. To tackle this problem, we propose an error elimination algorithm for the predicted values based on the domain ontology and a pronouncing dictionary, as shown in algorithm 1.

Algorithm 1 ASR error elimination with domain ontology

```
Input: Predicted triples $y$; domain ontology $ont$; pronouncing dictionary $\mathcal{D}$; similarity score threshold $thr$.
Output: Predicted triples after error elimination $\tilde{y}$.
Build a reversed mapping $v2s$ of $ont$.
$\tilde{y} \leftarrow []$
for $(act, slot, v)$ in $y$ do
    if $v$ not in $ont[slot]$ then
        if $v$ in $v2s$ and len($v2s[v]$) = 1 then
            $\tilde{y}$.append($(act, v2s[v], v)$)
        else
            $\tilde{v}, min\_dist \leftarrow$ minEditDist($v, ont[slot], \mathcal{D}$)
            if $min\_dist \leq thr$ then
                $\tilde{y}$.append($(act, slot, \tilde{v})$)
            end if
        end if
    else
        $\tilde{y}$.append($(act, slot, v)$)
    end if
end for
```

Given a predicted triple *act(slot=value)*, if *value* is not valid for *slot* according to the domain ontology, this triple should be considered for error elimination. We step further to replace the wrongly predicted value with the most *similar* one for *slot* according to the domain ontology, while we also set a similarity threshold to reject the predicted triple. Since the value error mainly comes from ASR errors, we measure the similarity of two values by the edit distance of their phoneme sequences relying on a pronouncing dictionary.

Moreover, the prediction of slots may also be wrong. For example, in the *weather* domain, the model may be confused about whether "Suzhou" is a "city" or "district". Therefore, when an error occurs, we first replace the predicted slot with the slot that the value uniquely belongs to.

Leveraging audio information

End-to-end ASR. In order to leverage audio information, we build a Chinese char-level end-to-end ASR (E2E ASR) system. We utilize a sequence-to-sequence (seq2seq) model with the attention mechanism [1]. Following [21], the encoder includes 2 layers of 3×3 convolutions with 16 channels, 2 layers of 3×3 convolutions with 32 channels, and 2 max-pooling layers with the stride of 2. On top of the convolutional layers, an encoder contains 5 layers of BLSTM and a projection layer. The decoder consists of 2 LSTM layers and uses *Tanh attention* [15]. We pre-trained the E2E ASR model on a 2000 hours Chinese Mandarin dataset and then finetuned by all utterances in the challenge dataset. Due to the limitation of the challenge dataset, both the pre-trained E2E ASR and finetuned systems perform worse than the given 1best results, as shown in Table 2.

Table 2: CER% of dev set for different domains

system	map	music	video	weather
given 1best	19.92	25.40	13.18	29.26
pre-trained E2E ASR	52.77	52.49	49.84	56.87
+ finetune	30.7	32.2	22.8	32.7
+ 10+1-best LM rescore	**17.5**	**24.6**	**11.2**	**27.2**

LM rescore. Therefore, it is intuitive to train an in-domain Language Model (LM) and do shallow fusion [17] with seq2seq models. We also combined the *N-best* list produced by E2E ASR model and the given 1best, and used an in-domain LSTM LM to rescore these *N+1-best* candidates. The new results slightly outperform the given 1best (see Table 2).

ASR features. ASR feature can be extracted from the encoder of the E2E ASR model. Let $\mathbf{o} = o_1 \cdots o_{|o|}$ denote audio signals of an utterance. The encoder will produce a corresponding sequence of hidden vectors $\mathbf{E} = \mathbf{h}_1 \cdots \mathbf{h}_{|o|}$. We can obtain ASR feature vectors of the utterance in three ways: the last hidden vector, max-pooling and mean-pooling over time. They are exploited as additional inputs for each word in the textual SLU model.

4 EXPERIMENTS

Datasets

We use the dataset provided in the CATSLU challenge [24][3], which consisted of four dialogue domains (*map, music, weather, video*). There is also an ontology file for each domain, containing all possible *acts*, *slots* and *values.*

[3] https://sites.google.com/view/CATSLU/home

Table 3: F1-scores and joint accuracies on the development set of each domain. (F1-score/accuracy)

Method	map	music	weather	video
BLSTM	78.8/75.1	82.8/71.9	87.9/83.3	80.5/66.7
+EE	88.0/83.5	89.7/79.8	91.4/85.5	87.0/72.3
+EE +LF	88.0/83.5	92.0/85.3	91.9/87.8	90.8/80.5
+EE +LF +AF	88.8/84.2	92.5/**86.4**	92.5/87.3	90.8/80.5
+DA	88.5/83.4	92.5/85.0	91.7/86.2	91.3/81.0
BLSTM-CRF	81.1/76.8	84.2/75.1	89.3/85.2	81.2/67.7
+EE	88.5/84.0	89.8/79.8	91.9/86.8	87.9/73.3
+EE +LF	88.7/83.8	92.3/85.3	**92.7**/**88.6**	91.3/**81.0**
+EE +LF +AF	88.9/84.5	**92.7**/85.6	92.5/87.6	90.8/80.0
+DA	88.7/84.3	92.4/84.5	91.9/86.0	**91.4**/80.0
BLSTM-focus	80.9/76.1	85.6/76.1	87.6/82.3	81.6/69.2
+EE	88.8/84.4	90.6/81.6	89.2/83.1	87.8/73.3
+EE +LF	87.7/83.1	92.6/85.0	91.9/87.6	91.3/81.0
+EE +LF +AF	**89.1**/**84.8**	92.5/85.0	91.9/87.0	91.2/81.0
+DA	88.7/83.9	92.2/84.0	92.0/86.5	90.3/80.0

Experimental setup

To avoid error propagation of Chinese word segmentation [13], our SLU models are at Chinese char level. BLSTMs of SLU models are single-layer with 256 hidden units, and the dimension of char embeddings is 200. We initialize the input embedding layer by pre-training a LSTM based bidirectional language models (biLMs) with zhwiki[4] corpus. For training, the network parameters are randomly initialized with the uniform distribution (-0.2, 0.2). The dropout with a probability of 0.5 is applied to the non-recurrent connections during the training stage. The maximum norm for gradient clipping is set to 5. The learning rate is set to 0.001 and kept for 50 epochs with Adam optimizer [10] for both single-domain and domain-adaptation tasks. We respectively use the human-transcripts of training and development sets for training and validation to save the best parameters. In the test stage, ASR-text is used. The metrics used in evaluation are F1-score of *act(slot=value)* triples and utterance-level accuracy.

Evaluation on the development sets

The experimental results of different models on the development set of each domain are shown in Table 3. "+EE" means the algorithm of error elimination is applied in the test stage. "+LF" means the lexicon features are exploited. Moreover, "+AF" denotes that ASR features are included. There are 3 kinds of ASR features as mentioned in section 3, we tried all of them and chose the best on the validation. Finally, we applied the domain adaptation method ("+DA") with adversarial training technique on the basis of all of the approaches above. From Table 3, we can see that: 1) The CRF layer and focus mechanism consider more about label dependency, resulting in better performances compared to vanilla BLSTM models. 2) By recovering true values based on the domain ontology and the pronouncing dictionary, the F1 score and accuracy in all domains are improved significantly (even over 10% in some domains). This indicates the importance of domain knowledge. 3) The usage of lexicon features helps a lot especially in *music* and *video* domains. The improvement in both F1-score and accuracy can be nearly 8%. 4) ASR features are only effective in some cases, e.g., the *map* domain with BLSTM-focus model. The domain adaptation training is not effective on the development sets.

Table 4: Final results on the test set of each domain.

	map	music	weather	video
0	77.61/74.65	81.57/71.15	85.25/78.16	75.18/57.53
0*	87.43/83.08	92.84/84.91	**94.16**/88.80	**93.04**/83.91
1	87.92/83.78	92.74/85.06	92.99/86.80	92.28/82.57
2	88.07/83.84	92.63/84.76	93.35/87.44	92.18/82.75
3	88.66/**84.54**	93.13/85.36	93.72/88.16	92.49/83.49
4	**89.28**/84.47	**93.53**/86.09	93.88/**89.02**	92.77/83.55
5	89.00/**84.54**	93.42/**86.69**	93.70/88.80	92.84/**84.28**

Challenge submissions

For the challenge, we are allowed to submit five systems in total. For "BLSTM", "BLSTM-CRF" and "BLSTM-focus", we choose the one with best performance testing on the development set. Considering the limitation of data scale, we also exploit the development set as well as the train set for training models. The details of all submitted systems are:

System 1 Basic models + lexicon feat. + error elimination
System 2 System 1 + add the development set in training
System 3 System 2 + ASR rescore 1-best
System 4 System 3 + ASR features
System 5 System 4 + domain adaptation

The final results are shown in Table 4, where '0' refers to the best baseline and '0*' denotes the best performance on each domain from other teams. For joint accuracy, our system outperforms all of the other teams in four domains. As for the F1-score, we win 2 first place across four domains.

5 CONCLUSION

In this paper, we present details of our submissions to the CATSLU challenge. The results show that acoustic and domain knowledge is essential in building robust SLU systems.

ACKNOWLEDGMENTS

This work has been supported by the National Key Research and Development Program of China (Grant No.2017YFB1002102) and the China NSFC projects (No. 61573241).

[4] https://dumps.wikimedia.org/zhwiki/latest

REFERENCES

[1] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2014. Neural machine translation by jointly learning to align and translate. *arXiv preprint arXiv:1409.0473* (2014).

[2] Ankur Bapna, Gökhan Tür, Dilek Hakkani-Tür, and Larry P. Heck. 2017. Towards Zero-Shot Frame Semantic Parsing for Domain Scaling. In *Interspeech 2017, 18th Annual Conference of the International Speech Communication Association*. 2476–2480.

[3] Renato De Mori, Frédéric Bechet, Dilek Hakkani-Tur, Michael McTear, Giuseppe Riccardi, and Gokhan Tur. 2008. Spoken language understanding. *IEEE Signal Processing Magazine* 25, 3 (2008), 50–58.

[4] Alex Graves. 2012. *Supervised Sequence Labelling with Recurrent Neural Networks*. Springer Berlin Heidelberg.

[5] Parisa Haghani, Arun Narayanan, Michiel Bacchiani, Galen Chuang, Neeraj Gaur, Pedro Moreno, Rohit Prabhavalkar, Zhongdi Qu, and Austin Waters. 2018. From Audio to Semantics: Approaches to end-to-end spoken language understanding. In *2018 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 720–726.

[6] Dilek Hakkani-Tür, Frédéric Béchet, Giuseppe Riccardi, and Gokhan Tur. 2006. Beyond ASR 1-best: Using word confusion networks in spoken language understanding. *Computer Speech & Language* 20, 4 (2006), 495–514.

[7] Matthew Henderson, Milica Gašić, Blaise Thomson, Pirros Tsiakoulis, Kai Yu, and Steve Young. 2012. Discriminative spoken language understanding using word confusion networks. In *2012 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 176–181.

[8] Young-Bum Kim, Karl Stratos, and Dongchan Kim. 2017. Adversarial adaptation of synthetic or stale data. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. 1297–1307.

[9] Young-Bum Kim, Karl Stratos, and Ruhi Sarikaya. 2016. Frustratingly Easy Neural Domain Adaptation.. In *COLING*. 387–396.

[10] Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980* (2014).

[11] Bing Liu and Ian Lane. 2016. Attention-Based Recurrent Neural Network Models for Joint Intent Detection and Slot Filling. In *17th Annual Conference of the International Speech Communication Association (InterSpeech)*.

[12] Bing Liu and Ian Lane. 2018. Multi-Domain Adversarial Learning for Slot Filling in Spoken Language Understanding. *arXiv preprint arXiv:1807.00267* (2018).

[13] Yuxian Meng, Xiaoya Li, Xiaofei Sun, Qinghong Han, Arianna Yuan, and Jiwei Li. 2019. Is Word Segmentation Necessary for Deep Learning of Chinese Representations? *arXiv preprint arXiv:1905.05526* (2019).

[14] Grégoire Mesnil, Yann Dauphin, Kaisheng Yao, Yoshua Bengio, Li Deng, Dilek Hakkani-Tur, Xiaodong He, Larry Heck, Gokhan Tur, Dong Yu, et al. 2015. Using recurrent neural networks for slot filling in spoken language understanding. *IEEE/ACM Transactions on Audio, Speech, and Language Processing* 23, 3 (2015), 530–539.

[15] Rohit Prabhavalkar, Tara N Sainath, Bo Li, Kanishka Rao, and Navdeep Jaitly. 2017. An Analysis of" Attention" in Sequence-to-Sequence Models.. In *Interspeech*. 3702–3706.

[16] Christian Raymond and Giuseppe Riccardi. 2007. Generative and discriminative algorithms for spoken language understanding. In *Eighth Annual Conference of the International Speech Communication Association*.

[17] Shubham Toshniwal, Anjuli Kannan, Chung-Cheng Chiu, Yonghui Wu, Tara N Sainath, and Karen Livescu. 2018. A comparison of techniques for language model integration in encoder-decoder speech recognition. In *2018 IEEE Spoken Language Technology Workshop (SLT)*. IEEE, 369–375.

[18] Ye-Yi Wang, Li Deng, and Alex Acero. 2005. Spoken language understanding. *IEEE Signal Processing Magazine* 22, 5 (2005), 16–31.

[19] Xiaohao Yang and Jia Liu. 2015. Using word confusion networks for slot filling in spoken language understanding. In *Sixteenth Annual Conference of the International Speech Communication Association*.

[20] Xiaodong Zhang and Houfeng Wang. 2016. A Joint Model of Intent Determination and Slot Filling for Spoken Language Understanding. In *Proceedings of the Twenty-Fifth International Joint Conference on Artificial Intelligence*. 2993–2999.

[21] Yu Zhang, William Chan, and Navdeep Jaitly. 2017. Very deep convolutional networks for end-to-end speech recognition. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 4845–4849.

[22] Su Zhu, Ouyu Lan, and Kai Yu. 2018. Robust Spoken Language Understanding with Unsupervised ASR-Error Adaptation. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP) 2018*. IEEE, 6179–6183.

[23] Su Zhu and Kai Yu. 2017. Encoder-decoder with focus-mechanism for sequence labelling based spoken language understanding. In *2017 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 5675–5679.

[24] Su Zhu, Zijian Zhao, Tiejun Zhao, Chengqing Zong, and Kai Yu. 2019. CATSLU: The 1st Chinese Audio-Textual Spoken Language Understanding Challenge. In *2019 International Conference on Multimodal Interaction (in press)*.