LEARNING ASR-ROBUST CONTEXTUALIZED EMBEDDINGS FOR SPOKEN LANGUAGE UNDERSTANDING

Chao-Wei Huang *Yun-Nung Chen*

National Taiwan University, Taipei, Taiwan
r07922069@ntu.edu.tw y.v.chen@ieee.org

ABSTRACT

Employing pre-trained language models (LM) to extract contextualized word representations has achieved state-of-the-art performance on various NLP tasks. However, applying this technique to noisy transcripts generated by automatic speech recognizer (ASR) is concerned. Therefore, this paper focuses on making contextualized representations more ASR-robust. We propose a novel confusion-aware fine-tuning method to mitigate the impact of ASR errors on pre-trained LMs. Specifically, we fine-tune LMs to produce similar representations for acoustically confusable words that are obtained from word confusion networks (WCNs) produced by ASR. Experiments on multiple benchmark datasets show that the proposed method significantly improves the performance of spoken language understanding when performing on ASR transcripts[1].

***Index Terms*—** spoken language understanding, contextualized embedding, ASR robustness

1. INTRODUCTION

A spoken language understanding (SLU) module serves an important role in a spoken dialogue system, which aims at extracting semantic concepts from spoken utterances and provides structured information for accessing the backend database. Typical tasks of SLU include intent detection and slot filling. These two tasks focus on predicting speaker's intent and extracting semantic concepts as constraints for the natural language. A movie-related example utterance "*find comedies by James Cameron*" shown in Figure 1 has two slot-value labels and a specific intent for the whole utterance.

Applying deep learning techniques has been shown to boost the performance of SLU [1, 2, 3, 4]. Most prior work focused on applying understanding models on *manual* transcripts, ignoring the errors introduced by automatic speech recognizers (ASR). Hence, several methods were proposed to address this problem. Simonnet et al. [5] simulated ASR errors and trained SLU models for better handling of the errors. The prior work leveraged information from lattices or word confusion networks [6, 7, 8, 9, 10, 11], and Zhu et al. [12] applied domain adversarial training for ASR-error adaptation, demonstrating the importance of incorporating ASR errors for better SLU performance.

Word	*find comedies by james cameron*
Slot	genre: *comedy*, director: *James Cameron*
Intent	find_movie

Fig. 1: An annotated utterance example.

Deep contextualized word representations recently have achieved great success among language understanding tasks [13, 14, 15]. Nevertheless, they may be less robust to noisy texts, such as the recognized results. In this paper, we investigate the impact of ASR errors on contextualized embeddings and further propose a novel confusion-aware fine-tuning method to alleviate this problem. To our best knowledge, there is no prior work that learned contextualized word embeddings and considered the errors produced from spoken language for better robustness. Our contributions are 3-fold:

- This is the first attempt to learn contextualized word embeddings specifically for spoken language.
- The proposed approach achieves better performance on the benchmark spoken language understanding tasks.
- The proposed method shows better robustness to ASR errors.

2. LEARNING ASR-ROBUST CONTEXTUALIZED EMBEDDINGS

Preparing datasets for SLU takes a lot of effort. SLU datasets are typically smaller than NLU datasets since it's much more labor intensive to collect labeled spoken utterances. Hence, the goal of this paper is to learn ASR-robust contextualized embeddings such that the downstream SLU models trained on manual transcripts can perform well on automatic transcripts, using only unlabeled spoken utterances to adapt.

To enable the embeddings to adapt ASR errors for improving SLU, our proposed method consists of three stages:

[1]Code available at: https://github.com/MiuLab/SpokenVec

1) language model pre-training on general domain corpora $\mathcal{D}_{\text{LM}}$, 2) confusion-aware language model fine-tuning on the text from the target SLU task, where the text can be either manual transcripts $\mathcal{D}_{\text{trs}}$ or automatic transcripts $\mathcal{D}_{\text{asr}}$, and 3) training a language understanding model with the fine-tuned LM on labeled SLU data $\mathcal{D}_{\text{SLU}}$, which consists of the manual transcripts $\mathcal{D}_{\text{trs}}$ with their corresponding labels.

In this paper, we focus on the task of intent detection, which is an utterance-level multi-class classification problem. More formally, given an utterance $x = \{w_1^x, w_2^x, ..., w_{|x|}^x\}$, the goal is to predict its corresponding intent I_x. The input utterance x can be either manually transcribed texts, denoted as x_{trs}, or ASR-recognized results, denoted as x_{asr}. The proposed approach is detailed below.

2.1. Embeddings from Language Model (ELMo)

Peters et al. [13] proposed ELMo to extract context-dependent word embeddings from a pre-trained LM, and the contextualized embeddings were proved to be able to improve the performance of downstream NLP tasks. In this paper, we adopt the same model architecture as in the original work, which consists of a CNN character encoder and two bidirectional LSTMs [16]. Same strategy of combining hidden states from different layers is applied [13], which computes the representation e_t for a word w_t^x in the sentence x as:

$$e_t = \gamma \sum_{i=0}^{2} \alpha_i \cdot h_{t,i}^x,$$

where $h_{t,i}^x = [\overleftarrow{h_{t,i}^x}; \overrightarrow{h_{t,i}^x}]$ is the concatenation of the i-th layer output from both directions at the time t, α_i is the weight for the i-th layer, and γ is a scaling factor. α_i and γ are scalar parameters learned along with downstream tasks. The ELMo model is pre-trained on the general-domain textual data $\mathcal{D}_{\text{LM}}$.

2.2. Language Model Fine-Tuning

One advantage of pre-training a language model is that it can leverage large amounts of unlabeled text corpora. Usually the data is general such as Wikipedia. However, the data distribution of the target task may be different from that used in pre-training, posing a domain mismatch problem. Howard et al. [17] proposed to fine-tune the pre-trained LM with sentences from the downstream dataset and showed that it boosts the performance of the downstream task. Chronopoulou et al. [18] also demonstrated the effectiveness of the fine-tuning method.

In order to adapt the pre-trained LM to the target data, the fine-tuning technique is applied. Given an utterance $x = \{w_1, w_2, ..., w_{|x|}\}$, the bidirectional language modeling loss can be written as:

$$\mathcal{L}_{\text{LM}} = \frac{1}{|x|} \sum_{t=1}^{|x|} -\log p(w_t \mid w_{<t}) - \log p(w_t \mid w_{>t}),$$

where $p(w_t \mid w_{<t})$ and $p(w_t \mid w_{>t})$ are probabilities of w_t predicted by the forward LM and the backward LM respectively.

Language model fine-tuning can be performed on both manual transcription and recognized results

2.3. Confusion-Aware Fine-Tuning

Taking ASR transcripts as inputs may introduce an issue that words in an utterance may be misrecognized. For instance, *fair* and *fare* are acoustically similar, so an ASR system may fail to distinguish between them, resulting in a substitution error. Such recognition errors might be recovered by human, because human are aware of the acoustic confusability of words. However, the errors may significantly degrade the testing performance when the models are trained on manual transcripts. In order to enhance the ASR robustness in contextualized word embeddings, this section integrates the acoustic confusion into our LM.

We propose a confusion-aware fine-tuning method to mitigate this problem from pre-trained LMs, which aims at making the LM consider multiple acoustically confusable words. Let $c = \{w_{t_1}^{x_1}, w_{t_2}^{x_2}\}$ denote an acoustic confusion, i.e., two words with similar pronunciation in two different utterances x_1 and x_2, and $C = \{c_1, c_2, \cdots, c_{|C|}\}$ denote the set of all acoustic confusions in x_1 and x_2. We introduce a new loss term called confusion loss:

$$\mathcal{L}_{\text{conf}} = \frac{1}{|C|} \sum_{c \in C} \sum_{i=0}^{1} 1 - \frac{h_{t_1,i}^{x_1} \cdot h_{t_2,i}^{x_2}}{\left\|h_{t_1,i}^{x_1}\right\| \left\|h_{t_2,i}^{x_2}\right\|},$$

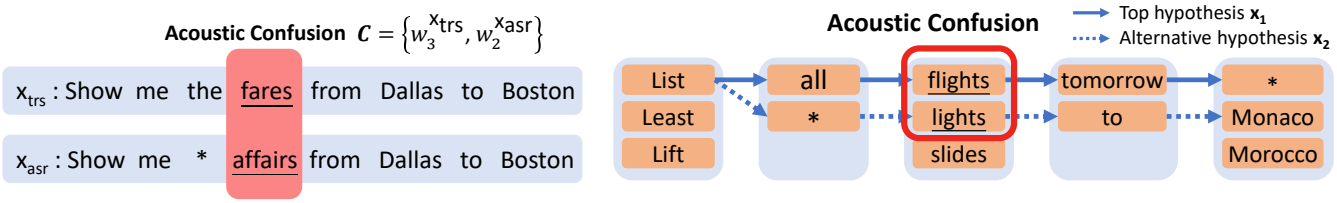
which is the cosine distance between the LM hidden states corresponding to words. Note that we empirically find that including only the first two layers in loss computation works the best. Two approaches are designed for extracting acoustic confusions.

2.3.1. *Supervised Confusion Extraction*

Assuming that both ASR transcripts x_{asr} and manual transcripts x_{trs} of a spoken utterance are accessible, we align x_{asr} with x_{trs} with respect to minimum edit-distance criterion to extract acoustic confusions as shown in Figure 2a. By minimizing $\mathcal{L}_{\text{conf}}$, we directly force the LM to produce representations for an erroneous word similar to its correct counterpart. This method is called *supervised confusion extraction* considering that it requires the availability of manual transcripts $\mathcal{D}_{\text{trs}}$.

2.3.2. *Unsupervised Confusion Extraction*

Considering the scenario where only audio recording of a spoken utterance is available, we can apply an ASR on the recording and construct a word confusion network (WCN). Then a list of n-best hypotheses is generated and aligned using WCN,



(a) Supervised confusion extraction with aligned utterances

(b) Unsupervised confusion extraction using a WCN generated by ASR

Fig. 2: Illustration of different extraction approaches. $*$ denotes a blank symbol for alignment purpose.

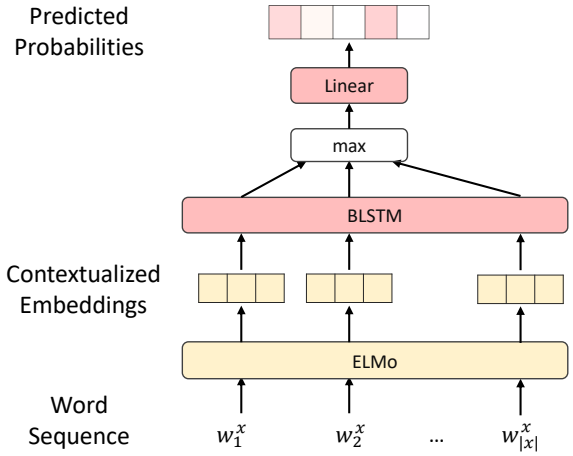


Fig. 3: Illustration of our SLU model architecture.

and the acoustic confusions can be obtained as depicted in Figure 2b.

An important advantage of this approach is that it does not require any labeled utterances or manual transcripts; therefore, we can leverage unlabeled audio recordings to fine-tune LMs in an unsupervised fashion.

2.4. Joint Objective Function for Fine-tuning

In the fine-tuning stage, we minimize the joint objective function including the LM loss and confusion-aware loss:

$$\mathcal{L}_{\text{FT}} = \mathcal{L}_{\text{LM}} + \beta \mathcal{L}_{\text{conf}},$$

where β is a hyperparameter to balance the contribution of two loss functions. The procedure enables our model to incorporate not only the target domain information but the acoustic information for better robustness to ASR errors.

2.5. Spoken Language Understanding (SLU)

To further build an SLU model that leverages ASR-robust contextualized embeddings, we employ a biLSTM as our SLU model, where the biLSTM takes contextualized word embeddings $\{e_t\}_{t=1}^{|x|}$ as the input, and the outputs of the last biLSTM layer are max-pooled, linearly transformed and soft-maxed to obtain the predicted probabilities for each class. The overall architecture is illustrated in Figure 3. During training, we use cross entropy as the loss function. Weights of the ELMo model are fixed during this stage except for α_i and γ. The SLU model is trained on $\mathcal{D}_{\text{SLU}}$ and evaluated on automatic transcripts. The trained SLU model is expected to achieve better performance on automatic transcripts due to the integration of ASR-robust contextualized word embeddings.

3. EXPERIMENTS

		train	test	intents	WER
ATIS		4478	893	21	15.55%
SmartLights	close	1765		6	45.61%
	far	1765		6	71.02%
Snips		13084	700	7	45.56%

Table 1: Dataset statistics.

3.1. Setup

Three SLU datasets used in the experiments as listed below:

- ATIS (Airline Travel Information Systems) [19, 20, 21] is a benchmark dataset widely used in language understanding research. The dataset contains audio recordings of people making flight reservations with corresponding manual transcripts.
- Snips SmartLights [22] contains spoken commands for smart light assistant. The dataset comes with two kinds of microphone settings, *close field* and *far field*.
- Snips [23] is a dataset for benchmarking NLU systems. This dataset is larger than ATIS and Snips SmartLights. We use a commercial text-to-speech system [2] to synthesize audio from text data.

The dataset statistics are shown in Table 1.

For ATIS, we train an ASR system on WSJ [24] using the *s5* recipe from Kaldi [25]. For the other datasets, we use an ASR model released in Kaldi to provide better recognition results [3]. We use the ASR system to recognize audio recordings and extract acoustic confusions for fine-tuning.

[2] https://cloud.google.com/text-to-speech/

[3] https://kaldi-asr.org/models/m1

Model		ATIS		SmartLights				Snips	
				close		far			
		Manual	ASR	Manual	ASR	Manual	ASR	Manual	ASR
(a)	Oracle	96.47	94.75	90.60	74.05	82.94	56.96	96.38	91.79
(b)	Context-independent	93.60	90.35	95.67	65.71	95.67	44.55	96.29	72.70
(c)	Pre-trained ELMo	96.65	93.27	97.01	64.53	97.01	44.46	96.29	77.86
(d)	(c) + fine-tune, $\mathcal{L}_{\text{LM}}$ only	96.91	94.27	95.91	66.33	95.66	46.22	96.38	87.74
(e)	(c) + fine-tune, $\mathcal{L}_{\text{FT}}$ (sup-conf)	96.61	**95.65**	95.92	**67.99**	95.53	46.57	97.01	88.52
(f)	(c) + fine-tune, $\mathcal{L}_{\text{FT}}$ (unsup-conf)	97.02	95.39	95.98	67.98	95.79	**47.38**	97.04	**89.55**

Table 2: Results of intent detection tasks (%). **Manual** and **ASR** indicate evaluating on x_{trs} and x_{asr} respectively. **close** and **far** represent different microphone settings. *sup-conf* stands for supervised confusion extraction, and *unsup-conf* stands for unsupervised confusion extraction. The best numbers for each dataset are marked in bold.

3.2. Model and Training Details

The pre-trained weights of ELMo from [13] are adopted. The size of contextualized representations is 1024. Our SLU model has two layers with 300-dimensional hidden states.

In the fine-tuning stage, acoustic confusions that contain stop words are excluded, and β is set to 0.1. We set batch size to 64 and use `Adam` as the optimizer [26] with learning rate 0.001 for all stages. We fine-tune ELMo for 3 epochs and train the SLU model for 50 epochs. The Snips SmartLights dataset is very small, so we use 10-fold cross validation to evaluate the models as suggested in [22].

3.3. Baselines

We compare our method with two baselines and an oracle system as listed below.

- Context-independent: replaces the contextualized representations with traditional context-independent word embeddings. The embedding matrix is initialized randomly.
- Pre-trained ELMo: uses pre-trained ELMo weights without fine-tuning.
- Oracle: trains SLU on x_{asr} with pre-trained ELMo embeddings.

Note that our models do not utilize the information the oracle system uses, x_{asr} paired with labels.

3.4. Results

Table 2 shows the experimental results, where the reported numbers are accuracies averaged over 5 runs. All models are trained on x_{trs} except for the oracle system, and they all perform great when evaluated on x_{trs}. Rows (b) and (c) show that ASR errors degrade SLU performance considerably for both context-independent and context-dependent embeddings. When testing on x_{asr}, the performance drops 3.38% on ATIS dataset using pre-trained ELMo embeddings. The results on Snips datasets show that the performance drops more in higher WER scenarios.

Our proposed method, confusion-aware language model fine-tuning, outperforms baselines by a large margin on all datasets (rows (d)-(f)), while it maintains identical performance on x_{trs}. Results in row (d) can be viewed as an ablation to rows (e) and (f), where we exclude $\mathcal{L}_{\text{conf}}$ from the joint objective. Rows (e) and (f) show that while $\mathcal{L}_{\text{LM}}$ provides significant improvement alone, adding $\mathcal{L}_{\text{conf}}$ further boosts performance notably. The results demonstrate that the proposed method can provide ASR robustness to the SLU models.

3.5. Discussion

The research on SLU has been investigated for several years, and there are two main branches. The first branch treats SLU as a natural language understanding (NLU) task, where the prior work trained the models directly on natural language data without misrecognition [1, 2, 3, 4]. Another branch focuses on building understanding models with consideration of ASR results. While some prior work relied on ASR lattices or WCNs to provide richer information to SLU models [6, 7, 8], the work presented here focuses on using the 1-best results from ASR. Simonnet et al. proposed an ASR error simulation scheme to train robust SLU models [5], whereas we dig realistic ASR errors from recognition results. Shivakumar et al. used WCNs to extract acoustic confusions for fine-tuning context-independent word embeddings [9, 10]. Our work combines the idea of using acoustic confusions with language model fine-tuning [17] to obtain ASR-robust contextualized embeddings.

4. CONCLUSION

This paper proposes a novel confusion-aware language model fine-tuning method for learning ASR-robust contextualized embeddings. We introduce supervised and unsupervised methods for extracting acoustic confusions and integrate a confusion loss that forces LMs to consider acoustically confusable words. The experiments on SLU demonstrate that our proposed method learns contextualized embeddings that are robust to ASR errors.

5. REFERENCES

[1] K. Yao, B. Peng, Y. Zhang, D. Yu, G. Zweig, and Y. Shi, "Spoken language understanding using long short-term memory neural networks," in *SLT*, 2014.

[2] D. Guo, G. Tur, W.-t. Yih, and G. Zweig, "Joint semantic utterance classification and slot filling with recursive neural networks," in *SLT*, 2014.

[3] G. Mesnil, Y. Dauphin, K. Yao, Y. Bengio, L. Deng, D. Hakkani-Tur, X. He, L. Heck, G. Tur, D. Yu, et al., "Using recurrent neural networks for slot filling in spoken language understanding," *IEEE/ACM Transactions on Audio, Speech, and Language Processing*, vol. 23, no. 3, pp. 530–539, 2014.

[4] C.-W. Goo, G. Gao, Y.-K. Hsu, C.-L. Huo, T.-C. Chen, K.-W. Hsu, and Y.-N. Chen, "Slot-gated modeling for joint slot filling and intent prediction," in *NAACL-HLT*, 2018.

[5] E. Simonnet, S. Ghannay, N. Camelin, and Y. Estève, "Simulating ASR errors for training SLU systems," in *LREC*, 2018.

[6] D. Hakkani-Tür, F. Béchet, G. Riccardi, and G. Tur, "Beyond asr 1-best: Using word confusion networks in spoken language understanding," *Computer Speech & Language*, vol. 20, no. 4, pp. 495–514, 2006.

[7] G. Tür, A. Deoras, and D. Z. Hakkani-Tür, "Semantic parsing using word confusion networks with conditional random fields," in *INTERSPEECH*, 2013.

[8] F. Ladhak, A. Gandhe, M. Dreyer, L. Mathias, A. Rastrow, and B. Hoffmeister, "Latticernn: Recurrent neural networks over lattices," *INTERSPEECH*, 2016.

[9] P. G. Shivakumar and P. Georgiou, "Confusion2vec: Towards enriching vector space word representations with representational ambiguities," *arXiv preprint arXiv:1811.03199*, 2018.

[10] P. G. Shivakumar, M. Yang, and P. Georgiou, "Spoken language intent detection using confusion2vec," *arXiv preprint arXiv:1904.03576*, 2019.

[11] C.-W. Huang and Y.-N. Chen, "Adapting pretrained transformer to lattices for spoken language understanding," in *ASRU*, 2019.

[12] S. Zhu, O. Lan, and K. Yu, "Robust spoken language understanding with unsupervised asr-error adaptation," in *ICASSP*, 2018.

[13] M. E. Peters, M. Neumann, M. Iyyer, M. Gardner, C. Clark, K. Lee, and L. Zettlemoyer, "Deep contextualized word representations," in *NAACL-HLT*, 2018.

[14] J. Devlin, M.-W. Chang, K. Lee, and K. Toutanova, "Bert: Pre-training of deep bidirectional transformers for language understanding," *arXiv preprint arXiv:1810.04805*, 2018.

[15] A. Siddhant, A. Goyal, and A. Metallinou, "Unsupervised transfer learning for spoken language understanding in intelligent agents," *arXiv preprint arXiv:1811.05370*, 2018.

[16] S. Hochreiter and J. Schmidhuber, "Long short-term memory," *Neural computation*, 1997.

[17] J. Howard and S. Ruder, "Universal language model fine-tuning for text classification," in *ACL*, 2018.

[18] A. Chronopoulou, C. Baziotis, and A. Potamianos, "An embarrassingly simple approach for transfer learning from pretrained language models," *arXiv preprint arXiv:1902.10547*, 2019.

[19] C. T. Hemphill, J. J. Godfrey, and G. R. Doddington, "The ATIS spoken language systems pilot corpus," in *Speech and Natural Language: Proceedings of a Workshop Held at Hidden Valley, Pennsylvania, June 24-27,1990*, 1990.

[20] D. A. Dahl, M. Bates, M. Brown, W. Fisher, K. Hunicke-Smith, D. Pallett, C. Pao, A. Rudnicky, and E. Shriberg, "Expanding the scope of the atis task: The atis-3 corpus," in *HLT*, 1994.

[21] G. Tur, D. Hakkani-Tür, and L. Heck, "What is left to be understood in ATIS?," in *SLT*, 2010.

[22] A. Saade, A. Coucke, A. Caulier, J. Dureau, A. Ball, T. Bluche, D. Leroy, C. Doumouro, T. Gisselbrecht, F. Caltagirone, et al., "Spoken language understanding on the edge," *arXiv preprint arXiv:1810.12735*, 2018.

[23] A. Coucke, A. Saade, A. Ball, T. Bluche, A. Caulier, D. Leroy, C. Doumouro, T. Gisselbrecht, F. Caltagirone, T. Lavril, et al., "Snips voice platform: an embedded spoken language understanding system for private-by-design voice interfaces," *arXiv preprint arXiv:1805.10190*, 2018.

[24] D. B. Paul and J. M. Baker, "The design for the wall street journal-based csr corpus," in *Proc. of the Workshop on Speech and Natural Language*, 1992, HLT '91.

[25] D. Povey, A. Ghoshal, G. Boulianne, L. Burget, O. Glembek, N. Goel, M. Hannemann, P. Motlicek, Y. Qian, P. Schwarz, et al., "The kaldi speech recognition toolkit," Tech. Rep., IEEE Signal Processing Society, 2011.

[26] D. P. Kingma and J. Ba, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.