SIDA: Slot-filling and Intent Detection Using Adapters

Zeinab Borhanifard
Romina Oji
Yadollah Yaghoobzadeh
Heshaam Faili
borhanifardz@ut.ac.ir
romina.oji@ut.ac.ir
y.yaghoobzadeh@ut.ac.ir
hfaili@ut.ac.ir

ABSTRACT

Task-oriented dialogue systems require developing a natural language understanding (NLU) component to extract semantic frames of user utterances. NLU is composed of two tasks: intent detection and slot filling. Learning these two tasks jointly has been shown to be effective in achieving the best performance. We propose a novel joint model called SIDA (slot filling and intent detection with adapters) to improve the performance of NLU. In SIDA, we include adapter layers in pre-trained language models and fine-tune only those during task learning. Experiments on five public datasets show that SIDA is more effective and efficient compared to the previous methods, especially for low-resource settings.

KEYWORDS

task-oriented dialogue system, natural language understanding, adapter, slot-filling, intent detection

ACM Reference Format:
Zeinab Borhanifard, Romina Oji, Yadollah Yaghoobzadeh, and Heshaam Faili. 2024. SIDA: Slot-filling and Intent Detection Using Adapters. In *Proceedings of Make sure to enter the correct conference title from your rights confirmation emai (Conference acronym 'XX).* ACM, New York, NY, USA, 5 pages. https://doi.org/XXXXXXX.XXXXXXX

1 INTRODUCTION

The task-oriented dialogue system is the basis of virtual assistants like Alexa, Siri, Cortana, and Portal has been increasingly used in modern society; users interact with them across different domains to complete diverse tasks and achieve their specific goals [14]. A key component of these task-oriented dialogue systems is Natural Language Understanding (NLU) which aims to derive the intent of users and fill the value for the slots of the utterance [22, 29]. For example, in the utterance *"Play a chant by Mj Cole"*, a dialogue system should correctly identify that the user's intention is to give a command to play a song, and that *Mj Cole* is the artist name that the user would like to listen.

Conference acronym 'XX, July 14–18, 2024, SIGIR, Washington D.C

ACM ISBN 978-1-4503-XXXX-X/24/06…$15.00
https://doi.org/XXXXXXX.XXXXXXX

In recent years, various neural-network based models have been experimented on slot filing (SF) and intent detection (ID) tasks separately. These proposed methods could handel several challenges, such as out-of-vocabulary words and long-distance dependency between words.

Recent research in this area has focused on joint model training [2, 21, 22, 29, 34] to better exploit the shared knowledge between these tasks and model the relationship between these two tasks simultaneously. Joint models based on pre-trained language models are the state of the art in NLU [2, 21].

Despite significant advances in joint models, achieving the best performance requires fine-tuning all pre-trained language model parameters, which is not optimal for low-resource datasets [7, 16]. Recently, adapters have been introduced as an alternative method in NLP applications when datasets are small [8, 19]. Adapter tuning consists of freezing pre-trained parameters of a model and adding lightweight modules between layers in transformers. Adapter-based models achieve the same levels of performance as fine-tuning but with an efficient number of parameters and lower training time.

The adapter architecture has not been applied in slot filling and intent detection problems yet. This paper proposes SIDA, a framework that jointly models slot filling and intent detection with training adapters in parallel. Our model achieves better slot filling performance on five public datasets.

Our contributions are twofold: (1) developing an adapter-based joint model for slot filling and intent detection using a variety of adapter combinations based on the stack, in parallel, and fusion architectures. To the best of our knowledge, we are the first to propose an adapter-based joint model with pre-trained language models for slot filling and intent detection. By incorporating this architecture promised a booster performance in terms of training time and model storage of SIDA; (2) demonstrating promising results of SIDA on five public datasets, and especially for low-resource settings.

2 RELATED WORK

Slot filling and intent detection. There exists various approaches, from traditional methods to deep learning methods, to solve this problem. Both traditional approaches [5] such as conditional random field (CRF) and deep learning methods [4, 11, 15] such as Recurrent Neural Networks (RNN) have been explored thoroughly and achieved comparable results. By improving upon RNN methods to overcome the challenges such as vanishing / exploding gradient, these new methods (LSTM, bidirectional methods, CNN) had developed and applied on solving slot filling [10, 26]. Other researchers combined LSTM with CRF to first complete the slot labeling task and included a regression model to capture label dependencies [31]. Also, [10] proposed leveraging sentence-level information with encoder LSTM for slot filling. [24] proposed encoder-decoder with jointly generate delexicalised sentences for slot filling. This approach is based on how different words that correspond to the same slot play a similar semantic and syntactic role in the sentence.

Prior work has shown that the attention mechanism helps RNNs deal with long-range dependencies. Therefore several methods applied attention to solving slot filing problem [9].

Recently, several joint models for intent detection and slot filling have proposed and achieved the state-of-the-art performance; This performance shows that there exist dependencies between the two tasks [12, 25, 27, 28, 33].

Adapters in NLP. Adapters have been applied in NLP tasks such as text classification, machine translation, transfer learning, and cross-lingual transfer [19]. To our best knowledge, slot filling and intent detection do not use adapters.

3 PROBLEM DEFINITION

The key component of dialog system is the NLU component responsible to the intent of the user in a certain part of the dialogue, associate it with a number of slot-value pairs that need to be filled to accomplish the intent. Table 1 shows an example of intent detection and slot filling for the utterance "Play a chant by Mj Cole". Slot filling can be taken as a sequence labeling task to identify a slot label sequence (e.g., O, B-music-item, B-artist, I-artist, etc.) to extract semantic concepts and detect intent as a classification task for determining utterance's user (e.g., Play Music). We formulate intent detection and slot filing as follows:

Intent Detection (ID). Based on an input utterance $X = (x_1, x_2, \dots, x_n)$, intent detection can be described as a classification task that outputs over utterances, where the system assigns a label y to each utterance for detection of the intended query.

Slot Filling (SF). This one can be viewed as a token-level tagging mechanism that maps an input utterance X into a slot output sequence $S = (s_1, s_2, \dots, s_n)$.

The goal is to learn a probabilistic model to estimate $p(y, S|X, \Theta)$ where Θ is the parameter of the model.

4 APPROACH

4.1 Adapter

Adapters are small modules for specific tasks that are added within layers in transformers [8]. Adapters are effective due to three additional factors: 1) The adapter parameters are updated during the training while the weights of the original pre-trained language model are frozen; 2) Fine-tuning the pre-trained parameters is not required; 3) Adapters do not need to adjust all pre-trained model parameters, and in fact, they may introduce a number of task-specific parameters. Figure 1 shows the structure of an adapter utilized in this work.

Table 1: An example of an utterance from the SNIPS dataset. Slot labels are in IOB format.

Sentence	Play	a	chant	by	Mj	Cole
Slots	O	O	B-music-item	O	B-artist	I-artist
Intent			PlayMusic			

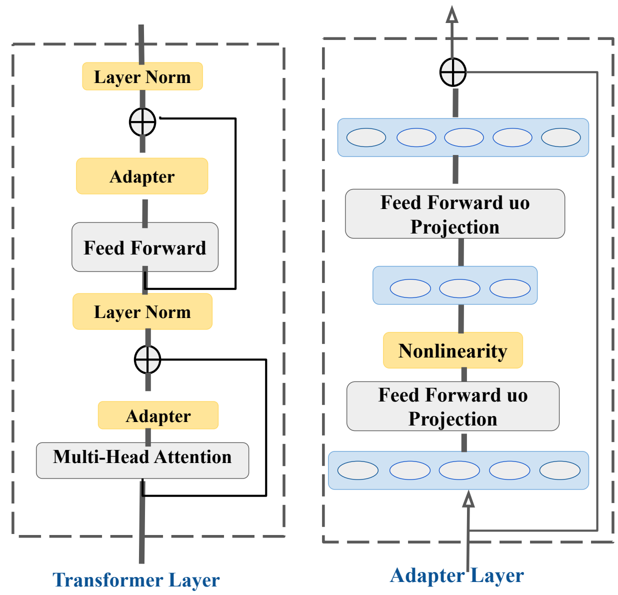


Figure 1: Adapter module architecture and integration with Transformer. Left: The adapter module is added twice to each Transformer layer: after the projection following multi-headed attention and after the two feed-forward layers. Right: The adapter consists of a bottleneck that contains only a few parameters compared to the attention and feed-forward layers in the original model [8].

Adapter layers can be described as follows: suppose that our adapter layer is represented by a function A. However, W^E projects the inputs to a smaller dimension, W^D projects them back to the original dimension, b_E and b_D are the corresponding biases, and f is a nonlinear function.

$$A(h_j) = W^D(f(W^E h_j + b_E)) + b_D \quad (1)$$

4.2 SIDA Method

Multiple adapters trained on different tasks can be combined in different ways when using adapters. In Pfeiffer et al. [19], a stack of adapters is applied for the cross-lingual transfer. The knowledge of multiple adapters can be combined into new downstream tasks using the fusion of adapters in non-destructively [17]. Using the parallel adapter block, parallel multi-task can be performed by different adapters. There are separate prediction heads for each adapter. Parallel adapters are used in inference by Rücklé et al. [23]. Training multiple adapters simultaneously and independently is possible by sharing all the parameters pre-trained.

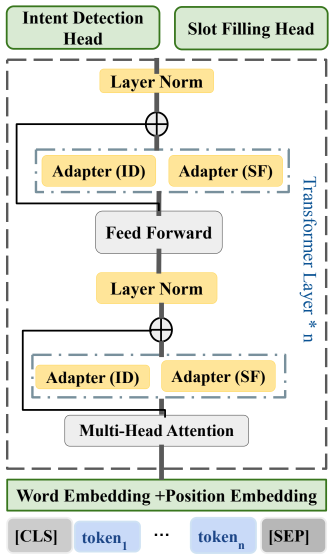


Figure 2: overall architecture of SIDA-P method

We propose the parallel adapter-based architecture for intent detection and slot filling. We have injected bottleneck adapters in BERT's transformer layers. Figure 2 shows how we insert four adapters in each transformer layer: two after the multi-head attention layer and two others after the feed-forward layer. Each two adapters for target tasks are trained simultaneously.

Parallel adapter block comprises two types of adapters: intent adapter (ID) and slot adapter (SF). These two adapters are formulated as follow, where A_i is the intent adapter and A_s is the slot adapter. Suppose A_p for the parallel adapter block that performs the operations in parallel before merging their output.

$$A_i(h_j) = W^D(f(W^E h_j + b_E)) + b_D \tag{2}$$

$$A_s(h_j) = W^D(f(W^E h_j + b_E)) + b_D \tag{3}$$

$$A_p(h_j) = A_i(h_j) + A_s(h_j) \tag{4}$$

5 EXPERIMENT SETUP

In this section, we introduce the datasets, metrics, and baselines.

Datasets. We show the effectiveness of our model in five public datasets: ATIS [13], SNIPS [3], MultiWoz [32], TaskMaster-1 [1], and CamRest [30]. A summary of the datasets and their key properties are shown in Table 2. ATIS and SNIPS are two commonly used benchmarks in slot filling and intent detection. The majority of research on natural language understanding has used these two datasets. However, these datasets are small, simple, and single-turn. Therefore, we consider assessing our model on more complex, more significant, and multi-turn datasets; these additional three datasets are related to task-oriented dialog systems. Multiwoz is one of the large, multi-domain, multi-turn datasets gathered by the Wizard of Oz method. Taskmaster-1 is another multi-domain dataset that includes six domains containing 13,215 dialogs, including 5,507 spoken and 7,708 written dialogs. Compared with the Multiwoz dataset, Taskmaster-1 is complicated to adapt due to its richness and divergence in language and unique words. To create TaskMaster-1, two procedures were utilized: Wizard of Oz and self-dialogs, and in this paper, we use self-dialogs conversations. CamRest is a single-domain (restaurant reservations) human-to-human conversation dataset (approximately 676 dialogues). This dataset is smaller than Multiwoz and Taskmaster-1 datasets, single-domain and multi-turn.

Table 2: Statistics and Properties of Datasets

DataSet	ATIS	SNIPS	CamRest	MultiWoz	TaskMaster-1
Train	4,478	13,084	3342	113,500	141,247
Val	500	700	1070	14,730	17,556
Test	893	700	1076	14,744	17,633
Intent types	21	15	1076	36	6
Slot types	120	72	57	120	88

Evaluation metrics. We evaluate the performance of slot filing by reporting the F1 score and intent prediction with accuracy. Also, we use a sentence-level semantic frame to evaluate the model's overall accuracy [29]. To calculate the overall accuracy, an utterance counts as correct if its intent and slots exactly match its ground truth.

Baseline and settings. Based on the existing baselines, we compared our model with:

- JointBert [2]: slot filling and intent detection are performed by fine-tuning a pre-trained BERT model. Several layers of transformer encoders are used to convert input into hidden state outputs to predict slot and intent labels. JointBert uses CLS for intent detection, and the other hidden states for slot filling.
- Stack-Propagation [20]: Through this method, token-level intent is predicted and used to fill in slots based on the intent detection output as the input.
- Co-Interactive [21]: A co-interactive module can consider cross-impacts between slot and intent by establishing a bidirectional connection between the two related tasks.

  Our different adapter-based methods are the followings:
- SIDA-S: A stack of adapters is trained in this method, and the output of a slot adapter feed as an input for an intent adapter. This method is similar to the sequential transfer learning method.
- SIDA-F: In this method block of adapters is combined in the fusion way.
- SIDA-P: Using this method, intent adapters and slot adapters train simultaneously, as illustrated in Figure 2.

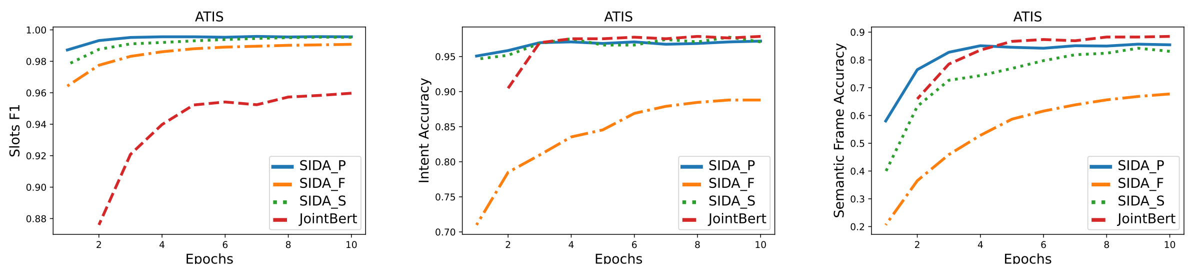
Training details. BERT-base-uncased is used as the pretrained language model in all experiments [6]. We implement adapters using AdapterHub libraries [18]. For all our experiments, we set the learning rate as 5e-5.

Empirical Results and Discussion. Table 3 summarizes the performance of different state-of-the-art models on the single turn and multiple turn datasets (30 epochs). The empirical results show that adapter-based methods for slot filling outperform all baselines in both single/multi turn by 2.4% to 13.2%. In multi turn datasets, the improvements for slot-filing are more significant. As a result, SIDA ensures intent is only detected with about ten percent of the

Table 3: Comparison of Methods in Multi-turn and Single-turn Datasets

	Single-turn Datasets					
	ATIS			SNIPS		
	Intent (Acc)	Slot (F1)	Overall	Intent (Acc)	Slot (F1)	Overall
JointBert[2]	97.87	95.97	88.46	98.71	96.88	**96.88**
Co-Interactive + BERT[21]	**98.00**	96.10	**88.80**	98.80	97.10	93.10
Stack-Propagation + BERT[20]	97.50	96.10	88.6	**99.00**	97.1	92.90
SIDA-F	88.80	99.07	67.74	94.14	98.01	52.85
SIDA-S	97.08	99.50	83.00	98.14	99.26	84.60
SIDA-P	97.20	**99.50**	85.40	97.80	**99.30**	84.10

	Multi-turn Datasets								
	CamRest			MutiWoz			TaskMaster-1		
	Intent (Acc)	Slot (F1)	Overall	Intent (Acc)	Slot (F1)	Overall	Intent (Acc)	Slot (F1)	Overall
JointBert[2]	**92.50**	97.75	**89.31**	**85.40**	87.60	**73.90**	**90.25**	88.66	**75.21**
SIDA-F	86.63	99.55	86.63	77.76	98.90	58.46	80.66	93.82	63.24
SIDA-S	91.42	99.69	83.45	82.20	98.01	66.01	87.45	95.73	69.02
SIDA-P	89.34	**99.79**	84.63	82.80	**99.20**	67.17	88.53	**96.33**	70.5



(a) Slot Filing (F1 Score) (b) Intent (Accuracy) (c) Overall (Semantic frame Accuracy)

Figure 3: Metrics vs. Epoch for ATIS Dataset

Table 4: Comparison of Methods in Few-Shot Samples of ATIS and SNIPS Datasets

	ATIS Dataset								
	Sample=100			Sample=500			Sample=1000		
	Intent (Acc)	Slot (F1)	Overall	Intent (Acc)	Slot (F1)	Overall	Intent (Acc)	Slot (F1)	Overall
JointBert[2]	**71.1**	68.5	2.9	**71.1**	68.7	2.9	86.3	79.7	50.6
SIDA-F	70.7	92.2	2.2	70.7	92.8	4.1	73.5	96.4	20.9
SIDA-S	70.7	93.2	5.1	70.7	92.5	2.5	89.1	98.1	48.1
SIDA-P	70.8	**93.8**	**5.6**	70.8	**93.4**	**4.1**	**91.5**	**98.7**	**60.2**
	SNIPS Dataset								
JointBert[2]	**88.7**	9.4	0.1	89.7	19.4	3.1	95.4	62.2	22.0
SIDA-F	73.4	30.4	0.1	73.5	89.3	0.1	81.4	91.6	1.6
SIDA-S	84.2	89.1	0.1	83.4	93.4	5.2	96.1	96.2	21.1
SIDA-P	84.3	**89.7**	**0.2**	**92.1**	**94.8**	**7.4**	**96.0**	**96.9**	**34.0**

parameters as compared to other state-of-the-artmethods. The results of Table 4 for small samples also show that the adapter-based method is more efficient than joint methods in all metrics when the dataset is small. Figure 3 illustrates the noticeable performance advantage of adapter-based models when compared to JointBERT for epochs less than four. SIDA-F has lower performance than SIDA-P and SIDA-S. Compared to other methods, the adapter-based methods achieve high performance with low training epochs, have a shorter training time, and the capacity of the model is low. In limited resources and time for model training, adapter-based methods can perform very well in lower epochs in comparison with other methods.

6 CONCLUSION

We proposed a novel joint model of slot filling and intent detection based on adapters (SIDA). To demonstrate the potential of our approach, we conducted experiments on five datasets with different complexity and compared with standard fine-tuning baselines. The performance of SIDA model in slot filling tasks was consistently significant and ensured that intents were detected with an efficient parameter set. We also analyzed the performance of SIDA in low-resource settings, with small sample size, which showed how paramater-efficient modeling using adapters is important.

REFERENCES

[1] Bill Byrne, Karthik Krishnamoorthi, Chinnadhurai Sankar, Arvind Neelakantan, Ben Goodrich, Daniel Duckworth, Semih Yavuz, Amit Dubey, Kyu-Young Kim, and Andy Cedilnik. 2019. Taskmaster-1: Toward a Realistic and Diverse Dialog Dataset. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Association for Computational Linguistics, Hong Kong, China, 4516–4525. https://doi.org/10.18653/v1/D19-1459

[2] Qian Chen, Zhu Zhuo, and Wen Wang. 2019. BERT for Joint Intent Classification and Slot Filling. arXiv:1902.10909 [cs.CL]

[3] Alice Coucke, Alaa Saade, Adrien Ball, Théodore Bluche, Alexandre Caulier, David Leroy, Clément Doumouro, Thibault Gisselbrecht, Francesco Caltagirone, Thibaut Lavril, Maël Primet, and Joseph Dureau. 2018. Snips Voice Platform: an embedded Spoken Language Understanding system for private-by-design voice interfaces. arXiv:1805.10190 [cs.CL]

[4] Li Deng, Xiaodeng He, Gokhan Tur, and Dilek Hakkani-Tur. 2015. Kernel deep convex networks and end-to-end learning. US Patent 9,099,083.

[5] Anoop Deoras and Ruhi Sarikaya. 2013. Deep belief network based semantic taggers for spoken language understanding.. In *Interspeech*. 2713–2717.

[6] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding. arXiv:1810.04805 [cs.CL]

[7] Jesse Dodge, Gabriel Ilharco, Roy Schwartz, Ali Farhadi, Hannaneh Hajishirzi, and Noah Smith. 2020. Fine-Tuning Pretrained Language Models: Weight Initializations, Data Orders, and Early Stopping. arXiv:2002.06305 [cs.CL]

[8] Neil Houlsby, Andrei Giurgiu, Stanislaw Jastrzebski, Bruna Morrone, Quentin De Laroussilhe, Andrea Gesmundo, Mona Attariyan, and Sylvain Gelly. 2019. Parameter-Efficient Transfer Learning for NLP. In *Proceedings of the 36th International Conference on Machine Learning (Proceedings of Machine Learning Research, Vol. 97)*, Kamalika Chaudhuri and Ruslan Salakhutdinov (Eds.). PMLR, 2790–2799. https://proceedings.mlr.press/v97/houlsby19a.html

[9] Mandy Korpusik, Zoe Liu, and James Glass. 2019. A Comparison of Deep Learning Methods for Language Understanding. In *Proc. Interspeech 2019*. ISCA, 849–853. https://doi.org/10.21437/Interspeech.2019-1262

[10] Gakuto Kurata, Bing Xiang, Bowen Zhou, and Mo Yu. 2016. Leveraging Sentence-level Information with Encoder LSTM for Semantic Slot Filling. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. 2077–2083.

[11] Bing Liu and Ian Lane. 2015. Recurrent neural network structured output prediction for spoken language understanding. In *Proc. NIPS Workshop on Machine Learning for Spoken Language Understanding and Interactions*.

[12] Bing Liu and Ian Lane. 2016. Attention-Based Recurrent Neural Network Models for Joint Intent Detection and Slot Filling. In *Interspeech 2016*. 685–689. https://doi.org/10.21437/Interspeech.2016-1352

[13] Xingkun Liu, Arash Eshghi, Pawel Swietojanski, and Verena Rieser. 2021. Benchmarking natural language understanding services for building conversational agents. In *Increasing Naturalness and Flexibility in Spoken Dialogue Interaction*. Springer, 165–183.

[14] Andrea Madotto, Zhaojiang Lin, Zhenpeng Zhou, Seungwhan Moon, Paul Crook, Bing Liu, Zhou Yu, Eunjoon Cho, Pascale Fung, and Zhiguang Wang. 2021. Continual Learning in Task-Oriented Dialogue Systems. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Online and Punta Cana, Dominican Republic, 7452–7467. https://doi.org/10.18653/v1/2021.emnlp-main.590

[15] Grégoire Mesnil, Xiaodong He, Li Deng, and Yoshua Bengio. 2013. Investigation of recurrent-neural-network architectures and learning methods for spoken language understanding.. In *Interspeech*. 3771–3775.

[16] Matthew E. Peters, Sebastian Ruder, and Noah A. Smith. 2019. To Tune or Not to Tune? Adapting Pretrained Representations to Diverse Tasks. arXiv:1903.05987 [cs.CL]

[17] Jonas Pfeiffer, Aishwarya Kamath, Andreas Rücklé, Kyunghyun Cho, and Iryna Gurevych. 2021. AdapterFusion: Non-Destructive Task Composition for Transfer Learning. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume*. Association for Computational Linguistics, Online, 487–503. https://doi.org/10.18653/v1/2021.eacl-main.39

[18] Jonas Pfeiffer, Andreas Rücklé, Clifton Poth, Aishwarya Kamath, Ivan Vulić, Sebastian Ruder, Kyunghyun Cho, and Iryna Gurevych. 2020. AdapterHub: A Framework for Adapting Transformers. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*. Association for Computational Linguistics, Online, 46–54. https://doi.org/10.18653/v1/2020.emnlp-demos.7

[19] Jonas Pfeiffer, Ivan Vulić, Iryna Gurevych, and Sebastian Ruder. 2020. MAD-X: An Adapter-based Framework for Multi-task Cross-lingual Transfer. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*. Association for Computational Linguistics, Online, 7654–7673. https://doi.org/10.18653/v1/2020.emnlp-main.617

[20] Libo Qin, Wanxiang Che, Yangming Li, Haoyang Wen, and Ting Liu. 2019. A Stack-Propagation Framework with Token-Level Intent Detection for Spoken Language Understanding. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. 2078–2087.

[21] Libo Qin, Tailu Liu, Wanxiang Che, Bingbing Kang, Sendong Zhao, and Ting Liu. 2021. A Co-Interactive Transformer for Joint Slot Filling and Intent Detection. In *ICASSP 2021 - 2021 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. 8193–8197. https://doi.org/10.1109/ICASSP39728.2021.9414110

[22] Libo Qin, Tianbao Xie, Wanxiang Che, and Ting Liu. 2021. A Survey on Spoken Language Understanding: Recent Advances and New Frontiers. In *IJCAI*.

[23] Andreas Rücklé, Gregor Geigle, Max Glockner, Tilman Beck, Jonas Pfeiffer, Nils Reimers, and Iryna Gurevych. 2021. AdapterDrop: On the Efficiency of Adapters in Transformers. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, Online and Punta Cana, Dominican Republic, 7930–7946. https://doi.org/10.18653/v1/2021.emnlp-main.626

[24] Youhyun Shin, Kang Min Yoo, and Sang-goo Lee. 2018. Slot Filling with Delexicalized Sentence Generation.. In *INTERSPEECH*. 2082–2086.

[25] Aditya Siddhant, Anuj Goyal, and Angeliki Metallinou. 2019. Unsupervised transfer learning for spoken language understanding in intelligent agents. In *Proceedings of the AAAI conference on artificial intelligence*, Vol. 33. 4959–4966.

[26] Ngoc Thang Vu. 2016. Sequential Convolutional Neural Networks for Slot Filling in Spoken Language Understanding. arXiv:1606.07783 [cs.CL]

[27] Congrui Wang, Zhen Huang, and Minghao Hu. 2020. SASGBC: Improving Sequence Labeling Performance for Joint Learning of Slot Filling and Intent Detection. In *Proceedings of 2020 the 6th International Conference on Computing and Data Engineering* (Sanya, China) *(ICCDE 2020)*. Association for Computing Machinery, New York, NY, USA, 29–33. https://doi.org/10.1145/3379247.3379266

[28] Yu Wang, Yilin Shen, and Hongxia Jin. 2018. A Bi-Model Based RNN Semantic Frame Parsing Model for Intent Detection and Slot Filling. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*. Association for Computational Linguistics, New Orleans, Louisiana, 309–314. https://doi.org/10.18653/v1/N18-2050

[29] H. Weld, X. Huang, S. Long, J. Poon, and S. C. Han. 2021. A survey of joint intent detection and slot-filling models in natural language understanding. arXiv:2101.08091 [cs.CL]

[30] Tsung-Hsien Wen, Yishu Miao, Phil Blunsom, and Steve Young. 2017. Latent Intention Dialogue Models. In *Proceedings of the 34th International Conference on Machine Learning*, Doina Precup and Yee Whye Teh (Eds.), Vol. 70. PMLR, 3732–3741. https://proceedings.mlr.press/v70/wen17a.html

[31] Kaisheng Yao, Baolin Peng, Geoffrey Zweig, Dong Yu, Xiaolong Li, and Feng Gao. 2014. Recurrent conditional random field for language understanding. In *2014 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*. IEEE, 4077–4081. https://doi.org/10.1109/ICASSP.2014.6854368

[32] Xiaoxue Zang, Abhinav Rastogi, Srinivas Sunkara, Raghav Gupta, Jianguo Zhang, and Jindong Chen. 2020. MultiWOZ 2.2 : A Dialogue Dataset with Additional Annotation Corrections and State Tracking Baselines. arXiv:2007.12720 [cs.CL]

[33] Linhao Zhang, Dehong Ma, Xiaodong Zhang, Xiaohui Yan, and Houfeng Wang. 2020. Graph LSTM with Context-Gated Mechanism for Spoken Language Understanding. *Proceedings of the AAAI Conference on Artificial Intelligence* 34, 05 (Apr. 2020), 9539–9546. https://doi.org/10.1609/aaai.v34i05.6499

[34] Xiaodong Zhang and Houfeng Wang. 2016. A joint model of intent determination and slot filling for spoken language understanding.. In *IJCAI*, Vol. 16. 2993–2999.