Frustratingly simple few-shot slot tagging

Jianqiang Ma[*]
Platform and Content Group, Tencent
alexanderma@tencent.com

Zeyu Yan[*]
AI Dept, Souche Inc.
yanzeyu@souche.com

Chang Li[*]
AI Dept, Souche Inc.
lichang@souche.com

Yang Zhang
AI Dept, Souche Inc.
zhangyang@souche.com

Abstract

We propose a simple and effective few-shot model for slot tagging. Recent work shows that it is promising to extend standard few-shot classification methods to sequence labeling with CRF-specific augmentations. Such methods show strengths in encoding slot name semantics and slot dependencies. However, we find these strengths can be obtained by a much simpler method, which casts slot tagging into machine reading comprehension (MRC). We fine-tune a standard BERT-based MRC model with a mixture of source domain and (few-shot) target domain data. Such simple method outperforms state-of-the-art methods by a large margin on the SNIPS dataset.

1 Introduction

This paper considers the task of few-shot slot tagging. Slot tagging (Zhang and Wang, 2016; Haihong et al., 2019) is a core component for task-oriented dialog systems (Papineni et al., 2001), where the goal is to provide a fine-grained, structured description of user request for a given intent. Example (1) shows the input-output of a slot tagging module for the `book restaurant` intent, where the module yields the semantic analysis for the input query in terms of slot label-value pairs such as `restaurant type` is ‘brasserie’ and `time range` is ‘15 minutes’, etc.

(1) **Query**: I want to book a far brasserie that serves minestrone in PA for a party of 9 in 15 minutes.
Tagged Slots: {`restaurant type`: ‘brasserie’, `time range`: ‘15 minutes’, `state`: ‘PA’, `party size number`: ‘9’, `served dish`: ‘minestrone’}

In real world systems, slot taggers are required to rapidly cover new domains to address increasing user needs. A key challenge here is that labeled data are often scarce in new domains and the high cost of manually annotating large-scale data becomes a major obstacle for domain adaptation. An attractive alternative is few-shot learning, which aims at achieving reasonable good results using only a few labeled instances in the new domain.

Although there are many successful few-shot *classification* methods, especially meta learning ones (Bapna et al., 2017; Luo et al., 2018; Fritzler et al., 2019), directly adapting them to slot tagging often yields unsatisfactory results (Hou et al., 2020). This is due to the *sentence-level, extractive nature* of slot tagging, where token dependencies within sentences are important but ignored by token-level classification models. Recent work (Hou et al., 2020) tackles this problem by extending meta learning methods to sequence transduction within the BERT-CRF framework, using several CRF-specific augmentations. Such augmentations show strengths in both encoding *slot name semantics* and modeling *slot dependencies*, which are key elements for effective few-shot slot tagging.

This paper shows that we can enjoy similar strengths with a *frustratingly simple* method. Our approach is based on the idea of transforming few-slot tagging into supervised machine reading comprehension (MRC), the detailed formulation of which is described in Section 3, with an example shown in Table 1. The implementation of our method is incredibly simple: we fine-tune an off-the-shelf BERT-based (Devlin et al., 2019) MRC model with data being merged from the source domain and (few shot) target domain data, without any meta learning or extra engineered components.

Our simple method works for good reasons. As the MRC-based approach extracts the full span of each slot value based on the complete sentence, such a model is aligned with the extractive nature of the task and implicitly considers slot dependencies. Moreover, the model can naturally encode label se-

[*]Equal contribution. Order decided by tossed coins.

Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021, pages 1028–1033
August 1–6, 2021. ©2021 Association for Computational Linguistics

mantics by mentioning the label names in the constructed questions. For our MRC model, the slot labels and the sentences being tagged from both source and target domains all reside in the same semantic space, where the training upon mixed data forces the model to generalize to a semantic space that is compatible with both domains. Through experiments, we also find that our model can better leverage linguistic and world knowledge in pre-trained language models, than previous BERT-based few-shot slot taggers.

The contributions of this paper are twofold: (1) We propose a simple and effective approach to few-shot slot tagging, which is based on training a supervised MRC model. (2) We empirically show the effectiveness of the proposed method, which outperforms previous state-of-the-art by 4+ points on the SNIPS benchmark.

2 Related Work

Slot Tagging Intent detection and slot tagging are two key modules in spoken language understanding. Slot tagging is often cast to sequence labeling problem (Zhang and Wang, 2016; Liu and Lane, 2016; Haihong et al., 2019; Qin et al., 2019). This paper adopts MRC formulation and focuses on the few shot learning setup.

Few Shot Learning In NLP, few shot learning methods mostly focus on classification tasks (Sun et al., 2019; Geng et al., 2019), while efforts on sequence labeling like slot tagging are rarely (Luo et al., 2018; Fritzler et al., 2019). Hou et al. (2020) explored few shot slot tagging by considering both label dependency transfer and label name semantics. Our model enjoys similar strengths but is much simpler and more effective.

QA Format for NLP Tasks Question answering, in particular machine reading comprehension (MRC) models (Seo et al., 2017; Xiong et al., 2018), is typically trained to answer questions by extracting a text span from the given context. Recently, there is a trend to cast non-QA NLP tasks, such as information extraction (Levy et al., 2017; Li et al., 2019, 2020), coreference resolution (Hou, 2020; Wu et al., 2020) and more (McCann et al., 2018) into MRC, which can achieve comparable or improved results. Our work is inspired by these works, but tackles a new task of slot tagging with a focus on few-shot learning.

3 Method

3.1 Slot Tagging as MRC

MRC formulation Given a *question* $Q = q_1, q_2, ..., q_L$, and a *context* passage $C = c_1, c_2, ..., c_M$, where $|Q| = L$ and $|C| = M$ are their token numbers respectively, while question Q is used to extract required spans from context C. The task is to find the span between the start token C_{start} and end token C_{end} in the context, for the given question w.r.t. each slot type. Some example questions and their context are shown in Table 1. For all questions associated with the same sentence, we provide one *context* C, which consists of the original sentence, being concatenated by special tokens "NO ANSWER" , which should be extracted, when no answer (*span*) is available in the given sentence for that question (*slot type*).

Following the MRC setup as in BERT (Devlin et al., 2019), we resort to the standard question-answer usage of BERT to find span C_{start}-C_{end}, i.e. feeding the token sequence in the form of `[CLS]`, $q_1, q_2, .., q_L$, `[SEP]`, $c_1, c_2, ..., c_M$ as the input to the BERT model, where the special tokens `[CLS]` and `[SEP]` are inserted between Q and C to distinguish them. Then, the hidden states from the last layer of BERT are extracted as the representations of input tokens. Probabilities $P_{start}(i)$ and $P_{end}(i)$ of each token position being the start and end position of the answer span are computed through the following formulas (1), where $i = 1, 2, ..., M$. For both start and end index predictions, tokens between the highest $P_{start}(i)$ and $P_{end}(j)$ will be predicted as the slot content for the slot type being asked in the question Q.

$$\begin{aligned} H^Q; H^C &= \text{BERT}([Q; C]) \\ H^C &= [v_1; v_2; ...; v_M] \\ P_{start}(i) &= softmax(W_s v_i) \\ P_{end}(i) &= softmax(W_e v_i) \end{aligned} \tag{1}$$

To train the MRC model, we first convert the original dataset, pairs of split sentence tokens and slot types into a set of <*question*, *answer*, *context*> triples, similar to the *format* of SQuAD 1.1 dataset (Rajpurkar et al., 2016). Triples of different samples are shuffled into batches to feed into the MRC model to get predictions of start and end position indices. Every slot type in the corresponding domain will be asked sequentially to find corresponding spans, or 'NO ANSWER' will be extracted if that slot type doesn't appear. We use cross entropy loss between predictions and ground truths.

Question Generation For each slot type $y \in Y$ to be predicted, we use a unified template to generate a *question* Q by joining domain name with slot type, which are both split by upper case letters and underlines, since they have the necessary information and short enough to keep model focusing on context C. As shown in example (1), question about slot type restaurant_type in domain book restaurant is constructed as *Book Restaurant, restaurant type ?*.

<table>
<tr><td colspan="2"><b>Context</b>: <i>I want to book a far brasserie that serves minestrone in PA for a party of 9 in 15 minutes NO ANSWER</i></td></tr>
<tr><th>Question</th><th>Answer Set</th></tr>
<tr><td>Book Restaurant restaurant type ?</td><td>{brasserie}</td></tr>
<tr><td>Book Restaurant party size description ?</td><td>{NO ANSWER}</td></tr>
</table>

Table 1: Samples of Context, questions and answers in the MRC formulation for Example (1).

3.2 Few Shot Learning

For a few-shot learning task, we have a *target domain* $D_1 = (x_i, y_i)$ with few labeled data, and n resource-rich source domains $D_2 ... D_n$. The task is to discover the optimal hypothesis h from x to y in domain D_1. To fit our MRC model into N-way k-shot settings, we follow the data construction procedures in Hou et al. (2020), where the support set $S = (x_i, y_i)$ is constructed by ensuring every slot in target domain appears approximately k times and each entity only appears once in a sentence.

We randomly generate 100 above support sets. For each set, we pair it with a query set having 20 excluded samples to form an episode. Together with D domains, our test set is made up of $D \times 100$ episodes. Note that our model is trained in one-go with the data being mixed up from the source domain and the support set of the target domain in each episode, unlike two-phase training in typical few shot learning methods. Despite the difference, the amount/split of the data for training and evaluation is exactly the same as in previous work.For zero-shot learning, we directly evaluate the source domain trained model on the full target domain.

4 Experiment

4.1 Setup

Dataset Our experiment is based on the SNIPS data set (Coucke et al., 2018) which is a benchmark dataset for slot tagging. It has data samples from 7 different domains, namely, Weather (We), Playlist (Pl), Book (Bo), Music (Mu), Restaurant (Re), Screening Event (Se) and Creative Work(Cr). Following few-shot setup in previous work, we split SNIPS data by domain. Each time, we leave one for testing, one for development and the others for training. Such procedure is repeated 7 times for cross validation.

Baselines **Bi-LSTM** (Schuster and Paliwal, 1997) is trained on the support set and tested on the query set using word embeddings of GloVe (Pennington et al., 2014). **Matching Network (MN) with BERT**(Vinyals et al., 2016) builds on top of BERT and labels the sequence in a token-level classification way. For each word, the most similar token in the support set is chosen and its label is assigned accordingly. **WarmProtoZero (WPZ)** (Fritzler et al., 2019) adopts similar strategy as MN, except replacing matching network with the prototypical network (Snell et al., 2017). **SimBERT** also classifies each token with the most similar word in the support set and assigns the corresponding label, where BERT embeddings are used without fine-tuning. **TransferBERT** is a domain transfer model based on vanilla BERT. It is pre-trained on source domain data, followed by fine-tuning on the support set of the target domain. **L-TapNet+CDT** (Hou et al., 2020) is a sequence labeling model based on BERT+CRF, where the Collapsed Dependency Transfer is used for transferring label-to-label dependencies and TapNet is used for transferring label semantics.

Implementation Details The pre-trained BERT-base uncased model is used for our method, where the batch size is set to 16, with max sequence length of 512. We use Adam optimizer (Kingma and Ba, 2014) with initial learning rate of 1×10^{-5} during training. We train the model with 30 epochs for each episode of evaluation, and get results according to develop domain.

4.2 Experimental Results

Main Results for 5-Shot Learning Table 2 shows the results for 5-shot learning. Each column indicates the per-domain results, where that domain

Model	5-shots Slot Tagging							
	We	**Mu**	**Pl**	**Bo**	**Se**	**Re**	**Cr**	**AVG**
WPZ	9.54	14.23	18.12	44.65	18.98	12.03	14.05	18.80
WPZ+GloVe	26.61	34.25	22.11	50.55	28.53	34.16	23.69	31.41
Bi-LSTM	25.17	39.80	46.13	74.60	53.47	40.35	25.10	43.52
TransferBERT	59.41	42.00	46.07	20.74	28.20	67.75	58.61	46.11
MN+BERT	36.67	33.67	52.60	69.09	38.42	33.28	72.10	47.98
SimBERT	53.46	54.13	42.81	75.54	57.10	55.30	32.38	52.96
WPZ+BERT	67.82	55.99	46.02	72.17	73.59	60.18	66.89	63.24
L-TapNet+CDT	71.64	67.16	75.88	**84.38**	**82.58**	70.05	**73.41**	75.01
Ours	**89.39**	**75.11**	**77.18**	84.16	73.53	**82.29**	72.51	**79.17**

Table 2: F1 scores results on 5-shot slot tagging. +CDT denotes collapsed dependency transfer. Scores below mid-line are from our methods, which achieve the best performance. AVG shows the averaged scores. Best results in bold.

is used as the target domain while others are used as source domains. In most domains, our model achieves improved results than all baseline, being based on BERT or not. In particular, our method outperforms the previous SOTA, L-TapNet+CDT, by 4.16% on average F1 score.

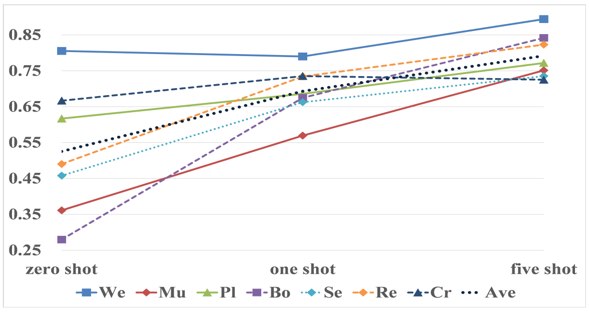

Figure 1: Avg F1 score for each domain for 1-shot and zero-shot learning, which have the same setup for training.

Analysis Figure 1 compares the performance of our model in zero/one/five-shot setup. Since the training of our method is just fine-tuning upon the mixture of source and target domain data, one-shot setup means using 0.01% extra (target domain) data over zero-shot setup. Yet, the boost is dramatic for most domains. We speculate that these tiny bit of target domain has a *catalyst effect*, which changes the optimization trajectory of model. It might be the case that such training forces our MRC model to generalize to a semantic space that is compatible with both source and target domains.

Note that our zero-shot model achieves an average F1-score of 52.5%, outperforming previous zero-shot SOTA, such as 40.6% of Shah et al. (2019) and 37.39% of Liu et al. (2020). As for one-shot setting, the average F1-score of our model is 69.3%, on par with 70.4% of the 1-shot SOTA (Hou, 2020).

Few shot learners typically rely on source domain data to arrive at a good hypothesis. Figure 2 shows the sensitivity of our model w.r.t. the scale of available *source domain* data. In each episode of evaluation, we select a subset (100, 1000, 2000) of sentences from source domains according to the rank of text similarity between them and the support set. While *the more the better* holds in general, we see that 1000 source domain sentences suffice for competitive results.

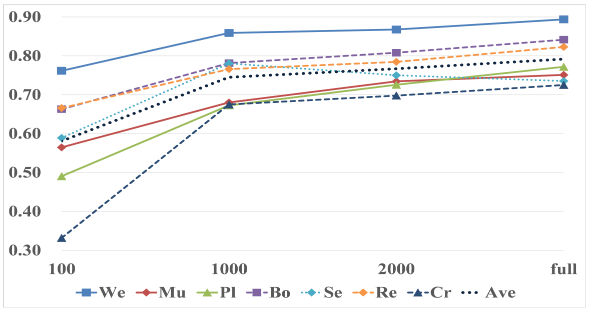

Figure 2: Avg F1 for each domain with different numbers of sentences from source domain. Where the full means all sentences from source domains and the number is near 12,000

5 Conclusion

In this paper, we propose a BERT-based MRC approach to few-shot slot tagging. By casting slot tagging into MRC problem, the learning consists of fine-tuning the MRC model with labeled sentences from a mixture of source domain and few-shot target domain data. Such an MRC-based method can naturally encode the label semantics in the form of questions, while the training forces the model to generalize to a semantic space that is compatible with both domains. Experiment results show the effectiveness of our simple method, as it outperforms previous SOTA on the SNIPS benchmark by a large margin. For future work, we plan to extend our approach to similar tasks, such as semantic role labeling and named entity recognition.

References

Ankur Bapna, G. Tür, Dilek Z. Hakkani-Tür, and Larry Heck. 2017. Towards zero-shot frame semantic parsing for domain scaling. *ArXiv*, abs/1707.02363.

Alice Coucke, Alaa Saade, Adrien Ball, Théodore Bluche, Alexandre Caulier, David Leroy, Clément Doumouro, Thibault Gisselbrecht, Francesco Caltagirone, Thibault Lavril, Maël Primet, and Joseph Dureau. 2018. Snips voice platform: an embedded spoken language understanding system for private-by-design voice interfaces. *CoRR*, abs/1805.10190.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Alexander Fritzler, Varvara Logacheva, and Maksim Kretov. 2019. Few-shot classification in named entity recognition task. In *Proceedings of the 34th ACM/SIGAPP Symposium on Applied Computing*, SAC '19, page 993–1000, New York, NY, USA. Association for Computing Machinery.

Ruiying Geng, Binhua Li, Yongbin Li, Xiaodan Zhu, Ping Jian, and Jian Sun. 2019. Induction networks for few-shot text classification. 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)*, pages 3904–3913, Hong Kong, China. Association for Computational Linguistics.

E. Haihong, Peiqing Niu, Zhongfu Chen, and Meina Song. 2019. A novel bi-directional interrelated model for joint intent detection and slot filling. In *ACL*.

Yufang Hou. 2020. Bridging anaphora resolution as question answering. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1428–1438, Online. Association for Computational Linguistics.

Yutai Hou, W. Che, Y. Lai, Zhihan Zhou, Yijia Liu, H. Liu, and Ting Liu. 2020. Few-shot slot tagging with collapsed dependency transfer and label-enhanced task-adaptive projection network. In *ACL*.

Diederik P Kingma and Jimmy Ba. 2014. Adam: A method for stochastic optimization. *arXiv preprint arXiv:1412.6980*.

Omer Levy, Minjoon Seo, Eunsol Choi, and Luke Zettlemoyer. 2017. Zero-shot relation extraction via reading comprehension. pages 333–342.

Xiaoya Li, Jingrong Feng, Yuxian Meng, Qinghong Han, Fei Wu, and Jiwei Li. 2020. A unified MRC framework for named entity recognition. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, Online. Association for Computational Linguistics.

Xiaoya Li, Fan Yin, Zijun Sun, Xiayu Li, Arianna Yuan, Duo Chai, Mingxin Zhou, and Jiwei Li. 2019. Entity-relation extraction as multi-turn question answering. In *Proceedings of ACL*, pages 1340–1350, Florence, Italy. Association for Computational Linguistics.

Bing Liu and I. Lane. 2016. Attention-based recurrent neural network models for joint intent detection and slot filling. *ArXiv*, abs/1609.01454.

Zihan Liu, Genta Indra Winata, Peng Xu, and Pascale Fung. 2020. Coach: A coarse-to-fine approach for cross-domain slot filling. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 19–25, Online. Association for Computational Linguistics.

Bingfeng Luo, Yansong Feng, Zheng Wang, Songfang Huang, Rui Yan, and Dongyan Zhao. 2018. Marrying up regular expressions with neural networks: A case study for spoken language understanding. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 2083–2093, Melbourne, Australia. Association for Computational Linguistics.

B. McCann, N. Keskar, Caiming Xiong, and R. Socher. 2018. The natural language decathlon: Multitask learning as question answering. *ArXiv*, abs/1806.08730.

Kishore A Papineni, Salim Roukos, and Robert T Ward. 2001. Natural language task-oriented dialog manager and method. US Patent 6,246,981.

Jeffrey Pennington, Richard Socher, and Christopher D Manning. 2014. Glove: Global vectors for word representation. In *Proceedings of the 2014 conference on empirical methods in natural language processing (EMNLP)*, pages 1532–1543.

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.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*.

Mike Schuster and K. Paliwal. 1997. Bidirectional recurrent neural networks. *IEEE Trans. Signal Process.*, 45:2673–2681.

Minjoon Seo, Aniruddha Kembhavi, Ali Farhadi, and Hannaneh Hajishirzi. 2017. Bidirectional attention flow for machine comprehension. *ArXiv*, abs/1611.01603.

Darsh Shah, Raghav Gupta, Amir Fayazi, and Dilek Hakkani-Tur. 2019. Robust zero-shot cross-domain slot filling with example values. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5484–5490.

Jake Snell, Kevin Swersky, and Richard Zemel. 2017. Prototypical networks for few-shot learning. In I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, editors, *Advances in Neural Information Processing Systems 30*, pages 4077–4087. Curran Associates, Inc.

Shengli Sun, Qingfeng Sun, Kevin Zhou, and Tengchao Lv. 2019. Hierarchical attention prototypical networks for few-shot text classification. 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)*, pages 476–485, Hong Kong, China. Association for Computational Linguistics.

Oriol Vinyals, Charles Blundell, Timothy Lillicrap, koray kavukcuoglu, and Daan Wierstra. 2016. Matching networks for one shot learning. In D. D. Lee, M. Sugiyama, U. V. Luxburg, I. Guyon, and R. Garnett, editors, *Advances in Neural Information Processing Systems 29*, pages 3630–3638. Curran Associates, Inc.

Wei Wu, Fei Wang, Arianna Yuan, Fei Wu, and Jiwei Li. 2020. CorefQA: Coreference resolution as query-based span prediction. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 6953–6963, Online. Association for Computational Linguistics.

Caiming Xiong, Victor Zhong, and Richard Socher. 2018. Dcn+: Mixed objective and deep residual coattention for question answering. *ArXiv*, abs/1711.00106.

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*, IJCAI'16, page 2993–2999. AAAI Press.