INTERSPEECH 2023
20-24 August 2023, Dublin, Ireland

Compositional Generalization in Spoken Language Understanding

Avik Ray[1], *Yilin Shen*[2], *Hongxia Jin*[3]

[1]Amazon Alexa, USA [2,3]Samsung Research America, USA

avik@utexas.edu, yilin.shen@samsung.com, hongxia.jin@samsung.com

Abstract

State-of-the-art spoken language understanding (SLU) models have shown tremendous success in benchmark SLU datasets, yet they still fail in many practical scenario due to the lack of model compositionality when trained on limited training data. In this paper, we study two types of compositionality: *novel slot combination*, and *length generalization*. We first conduct in-depth analysis, and find that state-of-the-art SLU models often learn spurious slot correlations during training, which leads to poor performance in both compositional cases. To mitigate these limitations, we create *the first compositional splits* of benchmark SLU datasets and we propose *the first compositional SLU model*, including compositional loss and paired training that tackle each compositional case respectively. On both benchmark and compositional splits in ATIS and SNIPS, we show that our compositional SLU model significantly outperforms (up to 5% F1 score) state-of-the-art BERT SLU model.

Index Terms: spoken language understanding, compositional generalization

1. Introduction

Spoken language understanding is an important component of task-oriented dialog systems powering today's voice controlled AI agents, and chat bots. Intent classification and slot tagging are two main sub-tasks in SLU [1, 2, 3, 4, 5]. Human language is inherently compositional [6], and humans possess the ability to understand infinite new utterances by focusing on relevant informative sub-parts of the utterance which were learned previously [7]. In this work, we consider informative sub-parts of an utterance containing slots. For example, humans can understand **slot value** "boston" is of the **slot type/label** *B-to-city* from the utterance *"show flights to boston"*. Similarly, from another utterance *"find flights from atlanta"*, they can learn "atlanta" has the slot type *B-from-city*. If presented with a new utterance *"show flights from atlanta to boston"*, humans can still infer the correct slot labels of "atlanta" and "boston", even though they have never seen these slots appear together before in the same utterance. Despite their success, current state-of-the-art SLU models [8], based on pre-trained language models [9, 10], struggle to perform such simple compositional generalization for slot tagging, as we demonstrate in this work.

In this paper, we investigate two main aspects of slot compositionality; (a) identifying a **novel combination of slot types** in an utterance which was never seen during training, and (b) identifying more number of slot types per utterance than any training utterance, which we refer to as **length generalization**. It is vitally important for SLU models to perform both these types of compositional generalization due to the following reasons. Firstly, in domains with a large number of slots (e.g. airline reservation), it can be both time-consuming and expensive to collect and annotate training utterances corresponding to each possible combination of slots. Secondly, for resource constrained cold-start skill developers [11], it is cheaper and easier to annotate a small number of short utterances (with just one or two slots) for training, than longer utterances with many slots which the SLU model may encounter after deployment. Building compositional SLU models which can generalize well under both these settings is vital for both scalable development, and reliability of future AI agents.

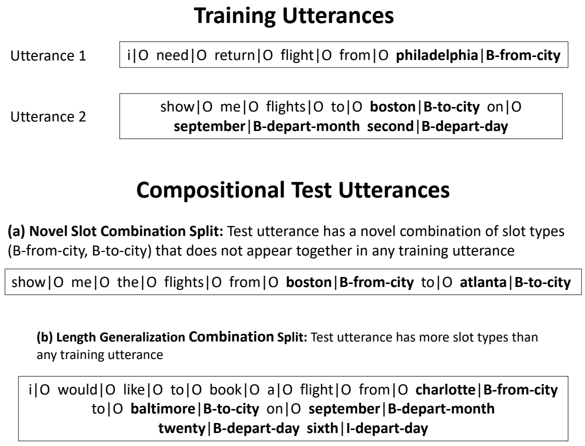
Training Utterances
Utterance 1
i|O need|O return|O flight|O from|O philadelphia|B-from-city
Utterance 2
show|O me|O flights|O to|O boston|B-to-city on|O september|B-depart-month second|B-depart-day
Compositional Test Utterances
(a) Novel Slot Combination Split: Test utterance has a novel combination of slot types (B-from-city, B-to-city) that does not appear together in any training utterance
show|O me|O the|O flights|O from|O boston|B-from-city to|O atlanta|B-to-city
(b) Length Generalization Combination Split: Test utterance has more slot types than any training utterance
i|O would|O like|O to|O book|O a|O flight|O from|O charlotte|B-from-city to|O baltimore|B-to-city on|O september|B-depart-month twenty|B-depart-day sixth|I-depart-day

Figure 1: ***Example Utterances in Two Types of Slot Compositionalities***

Due to a lack of compositional objective during training, existing SLU models fail to learn the correct dependence of slot words on the relevant informative words that convey their meaning. Instead, they often rely on spurious slot correlations to make their decision. When these models encounter an utterance with a novel combination of slots unseen during training, they fail to exploit this learned correlation, hence do not generalize. When the models encounter longer utterances with many slots per utterance, than they have seen during training, they often fail due to poor quality slot representations under a longer sentence context. While some techniques have been proposed to improve compositionality of sequence-to-sequence models in small synthetic datasets [12, 13, 14, 15], these are computationally expensive to train, and hard to scale on real-world datasets.

Main contributions: In this work, we improve compositional generalization of SLU models by using explicit compositional objectives during training, and develop novel data augmentation

This work was completed when author 1 was at Samsung Research.

10.21437/Interspeech.2023-1419

technique that helps generalization to longer utterances. Our proposed methods are practical and enable SLU models to scale to large real-world datasets. Our main contributions are:

1. We create the first compositional splits of benchmark SLU datasets (ATIS, and SNIPS). These splits can be used as a new benchmark to evaluate compositional generalization properties of SLU models.
2. We explore two types of slot compositional generalization, *novel slot combination*, and *length generalization*, and conduct in-depth analysis to investigate why existing SLU models perform poorly in both cases.
3. We propose a new compositional loss that improves compositionality of SLU models to utterances with unseen slot combinations, and a new paired training technique that improves length generalization of SLU models.
4. We show that our new compositional SLU model can achieve significant (up to 5%) improvement in slot tagging F1 score on our new compositional splits.

2. Compositional Benchmark Datasets

In this section, we propose our method to create compositional splits of benchmark SLU datasets.

In order to systematically evaluate compositional generalization of SLU models, we start with two benchmark SLU datasets. The first benchmark dataset **ATIS** [16] contains utterances related to airline reservation. We consider the data split from [17, 1] containing 4,978 training , and 893 test utterances in the standard split $(T_{\text{train}}, T_{\text{test}})$. We also use the second benchmark dataset **SNIPS** [18] containing various utterances in entertainment, weather, and restaurant domains. In the standard split SNIPS has 13,784 training, and 700 test utterances. For each dataset, we create two compositional train/test splits by selecting a subset of utterances from their standard train/test splits.

A) Novel Slot Combination Split: Human can easily identify a slot type in isolation just by focusing on most informative words which are used to describe a slot, even when presented with an utterance having a new combination of two or more slots types that were not seen during learning (training) phase (also referred as *systematicity* in cognitive science [7]). To test this aspect of compositional generalization, we create a train/test split where none of the combination (or set) of slot types present in a test utterance appear during training. We generate this split (referred as *novel slot combination*) using the following steps: (a) We remove from standard training set all utterances which have a combination of slot types that appear in the standard test set. We do not remove utterances with a single slot since these are fundamental examples from which the model learns the true semantic meaning of such slots. (b) In order to better separate compositional generalization with OOV generalization [19, 20], we replace any OOV slot values with a randomly selected slot value (but of the same slot label) from the training set to generate the final test set. Figure 1 shows example utterances from our novel slot combination split of ATIS dataset.

B) Length Generalization Split: Sequence based neural network models are inherently poor at generalization to longer sequences than what it observed during training [21, 22, 23]. Note that, the informativeness of an utterance directly depends on the number of slots present in the utterance, but it does not necessarily depend on actual length of the utterance. Hence, to test length generalization we consider the number of slots in the utterance to generate the split. Increasing the number of slots in an utterance also naturally increases its length. We create compositional train/test splits to test length generalization as follows: (a) From the standard training set, we only select utterances which have number of slots less than or equal to a fixed integer k. (we use $k = 2$ in our experiments) (b) We also remove from the test set utterances with slot combinations in the training set and substitute OOV slot values as before. We test if an SLU model has the ability to identify slots when number of slots in the utterance can be much larger than that observed during training. Figure 1 shows an example utterance from our ATIS length generalization split. Table 1 reports the split sizes.

3. Our Method

In this section, first we describe the baseline SLU model which we consider in our experiments, and explore why they have poor compositionality. Later, we propose new techniques to improve compositional generalization of SLU models. We use the following notations: Let $\mathbf{x} = (x_1, \ldots, x_n)$ denote an input utterance, where words/tokens $x_i \in \mathcal{V}$, the vocabulary. Each input token x_i is annotated by a slot label $y_i \in \mathcal{Y}$, the slot vocabulary. We consider slot labels in the standard IOB format, where label 'O' denotes the word/token that does not belong to any slot.

3.1. Baseline SLU Model Analysis

Large pre-trained language models have been shown to be successful in most natural language understanding tasks. Our baseline SLU model is based on one such model BERT [9], which also achieves state-of-the-art on benchmark SLU datasets. Our model is similar to the implementation in [8]. We train the model jointly on intent classification and slot tagging tasks using the objective: $\mathcal{L} = \mathcal{L}_{\text{intent}} + \lambda_1 \mathcal{L}_{\text{slot}}$, where $\mathcal{L}_{\text{intent}}$ is the intent classification loss, $\mathcal{L}_{\text{slot}}$ is the slot tagging loss, and λ_1 is a hyper-parameter to balance the losses.

Key issue: Although the BERT baseline model can achieve SOTA on standard splits of benchmark ATIS and SNIPS datasets, we observe that it often suffers significant drop in slot tagging performance on our compositional splits. Recall that in BERT, the self-attention layer of the transformer computes the attention distribution for each attention head h as follows:

$$P^h = \text{Softmax}\left(\frac{1}{\sqrt{d}} H W_h^Q (H W_h^K)^T\right) \tag{1}$$

where W_h^Q, W_h^K are the query, and key projection matrices of the h-th attention head, H is the output hidden layer vectors of previous layer, and d is the head dimension. By plotting this attention head distribution we can observe how much information each token contributes to the final slot label output logit. A human identifies a slot type by focusing on the surrounding most informative words in the utterance that help to convey the semantic meaning of the slot. One may expect that the final transformer layer in SLU model performs the same by providing higher attention weights to informative words corresponding to a slot value. However, we observe that this is not always the case. For example, as shown in Figure 2 in utterance *"play rock from the eighties"*, in order to identify the slot value *"rock"*, the BERT SLU model gives more attention to a different slot *"eighties"*, than informative context word "play". This could indicate that BERT SLU model often learns spurious correlations among context and slot words. Therefore, when such slots appear in an utterance with a new combination of slot not seen during training, the BERT model may fail to identify them. This results in poor slot tagging performance of BERT SLU model in our compositional splits (Section 4).

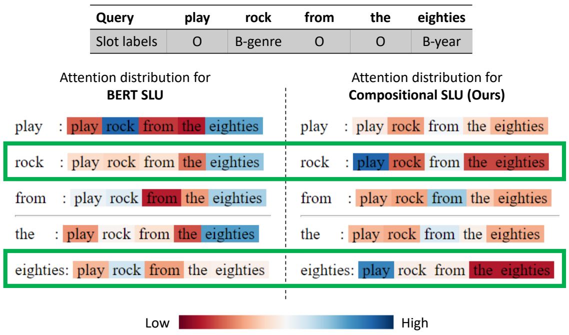
Query play rock from the eighties
Slot labels O B-genre O O B-year
Attention distribution for BERT SLU
Attention distribution for Compositional SLU (Ours)
play : play rock from the eighties
rock : play rock from the eighties
from : play rock from the eighties
the : play rock from the eighties
eighties: play rock from the eighties
Low
High

Figure 2: ***Visualization of the attention map (averaged over heads) for SNIPS utterance using baseline BERT SLU model (left), and our Compositional SLU model (right).*** **Blue** ***indicates high attention weights, and*** **Red** ***indicates low weights.*** **Green box** ***highlights the attention distribution corresponding to slot words. Compositional SLU model focus more on informative words to infer slot labels, and reduces spurious slot correlations, compared to BERT SLU model.***

3.2. Our Compositional SLU models

We now describe two main techniques that we show can improve compositional generalization of SLU models.

1. New Compositional Loss to Improve Slot Combination Compositionality: We develop a new compositional loss which reduces spurious slot correlation and encourages the SLU model to focus its attention on informative context words. Intuitively, if two words have different slot labels, they should be identifiable based on a disjoint set of words. For example, for the utterance *"play rock from the eighties"* (Figure 2), to identify slot label for the word *"rock"* it is sufficient to focus on context words $S_{\text{rock}} = \{play, rock\}$. To identify slot label for word *"eighties"* it is sufficient to focus on words $S_{\text{eighties}} = \{play, rock, from, eighties\}$. Note that these two sets of words are different. Our compositional loss is a sum of two loss functions as follows:

$$\mathcal{L}_{\text{slot-pair}} = \frac{1}{N_1} \sum_h \sum_{i,j: y_i \neq y_j \neq O} \text{KL}(P_i^h, P_j^h) \tag{2}$$

$$\mathcal{L}_{\text{non-deg}} = \frac{1}{N_2} \sum_h \sum_{i: y_i \neq O} \text{KL}(P_i^h, \mathbf{1}_i) \tag{3}$$

where P_i^h is the attention probability distribution corresponding to the token x_i, head h of the final transformer layer, $\mathbf{1}_i$ is the indicator distribution over all input tokens with 1 at position i, and 0 elsewhere, and N_1, N_2 are normalizing constants. The slot pair loss $\mathcal{L}_{\text{slot-pair}}$ encourages the attention distribution for two slot words x_i, x_j with different slot labels y_i, y_j to focus on a disjoint set of context words. The second non-degenerate loss $\mathcal{L}_{\text{non-deg}}$ prevents the slot pair loss to converge to a degenerate solution where each token mainly focuses on itself. The final compositional loss for training our SLU model is given by:

$$\mathcal{L} = \mathcal{L}_{\text{intent}} + \lambda_1 \mathcal{L}_{\text{slot}} - \lambda_2 \mathcal{L}_{\text{slot-pair}} - \lambda_3 \mathcal{L}_{\text{non-deg}} \tag{4}$$

where $\lambda_1, \lambda_2, \lambda_3$ are hyper-parameters.

2. Paired Training to Improve Length Generalization: The compositional loss enables the SLU model to better generalize to utterances with new combination of slots not seen during training (Section 4). However, the model still perform poorly in *length generalization* splits. Length generalization has been shown to be particularly difficult for both sequence generation [21, 24], and multimodal tasks [25]. We hypothesize that the model fails to generate good hidden state representations when presented an utterance with many slots, greater than those learned at training. To mitigate this problem, we develop an effective data augmentation approach we refer as *paired training*. Previously, a data augmentation approach GECA has been used to improve compositional generalization in sequence-to-sequence models [21]. However, their performance on length generalization remained poor since it only replaces words/phrases in existing training sentences, and does not necessarily produce very long sentences. In our approach, we randomly select two distinct training utterances of the same intent but a disjoint combination of slots, and concatenate them with a period separator to form a new training sample. This exposes the model both to longer sequences, as well as new combination of slots not present in the original training set, resulting in better length generalization. Note that, neuro-symbolic approaches have been shown to perform effective length generalization in seq-to-seq models [14, 15]. However, these models are difficult to train, and they do not scale to real-world datasets which don't follow strict grammar rules. Model based data augmentation approach has also been explored for improving robustness of SLU models [26]. However, this requires additional NL template information for each intent/slot which is difficult to obtain for large domains (e.g. ATIS).

4. Experiments

4.1. Settings and baselines

We train SLU models jointly for intent classification, and slot tagging tasks. Our evaluation metrics are *slot tagging F1 score*, and *intent accuracy*. Incorporating dependency parse information is known to improve compositional generalization of neural networks [27, 28]. We test an advanced baseline model (**BERT SLU + parse tree**) which modifies the original attention scores in the final transformer layer with a weight inversely dependent on token distance on dependency tree. Intuitively, tokens which are further away in the dependency tree are assumed to be less informative, and given lower attention scores. We also test a third baseline (**BERT SLU + relative pos emb**) which incorporates relative position embedding in BERT's attention computation [29]. In seq-to-seq tasks, it has been shown that relative position embedding helps in length generalization [30].
Parameters: Our baseline and compositional SLU models are fine-tuned from BERT model *bert-base-uncased* [9]. We use hyper-parameters: batch size 32, learning rate $\in \{10, 5\} \times 10^{-5}$, number of training steps $N \in \{4K, 5K, 6K\}$, $\lambda_1 = 1$. For compositional models we use $\lambda_2 = 0.01, \lambda_3 = 0.1$.

4.2. Results

Results on novel slot combination split: First, we compare the performance of SLU models on the *novel slot combination* splits, where the test utterances have a distinct combination of slot types which doesn't appear together in training. Table 1 presents the results (averaged over 5 runs with different seeds). The first row corresponds to the performance of BERT SLU model when trained on the *full* standard training set T_{train}. This acts as an *upper bound* for the model performance. When the models are trained on the smaller compositional training set, the performance of the baseline models drop since they do not generalize well to the test sets. Observe that, in SNIPS compositional test set, the baseline performance drops about 3% F1 score. The intent accuracy also drop around 1%. BERT combined with dependency parse tree, fails to improve slot tagging performance, but it improves the intent accuracy. BERT with

Table 1: ***Performance on Our Compositional Splits. The (train, test) sizes of Novel Slot Combination Split are (1229, 496) in ATIS and (1939, 600) in SNIPS. Sizes in Length Generalization Split are (1494, 163) in ATIS and (7107, 253) in SNIPS.***

	ATIS				SNIPS			
	Novel Slot Combination Split		**Length Generalization Split**		**Novel Slot Combination Split**		**Length Generalization Split**	
Model \ Dataset	Slot (F1)	Intent (acc)	Slot (F1)	Intent (acc)	Slot (F1)	Intent (acc)	Slot (F1)	Intent (acc)
Full BERT SLU (upper bound)	98.83 ± 0.05	98.39 ± 0.02	97.97 ± 0.07	99.39 ± 0.03	97.17 ± 0.05	98.67 ± 0.02	97.51 ± 0.07	99.62 ± 0.02
BERT SLU	97.46 ± 0.05	97.72 ± 0.09	90.30 ± 0.25	96.32 ± 0.02	94.11 ± 0.27	97.17 ± 0.08	92.29 ± 0.69	99.13 ± 0.46
BERT SLU + relative pos emb	94.83 ± 0.06	95.16 ± 0.15	91.69 ± 0.18	93.87 ± 0.03	94.27 ± 0.15	96.00 ± 0.08	89.77 ± 0.51	98.85 ± 0.12
BERT SLU + parse tree	97.64 ± 0.06	96.57 ± 0.03	92.27 ± 0.12	96.07 ± 0.30	94.12 ± 0.17	**98.33** ± 0.02	92.36 ± 0.21	98.82 ± 0.03
Comp. SLU (Ours)	**98.10**† ± 0.12	**97.78** ± 0.41	**94.93**† ± 0.68	96.93† ± 0.30	**95.37**† ± 0.42	97.83† ± 0.59	**95.63**† ± 0.92	**99.64**† ± 0.11
- Comp. Loss	97.81 ± 0.14	96.97 ± 0.03	94.25 ± 0.12	96.87 ± 0.18	95.05 ± 0.10	97.33 ± 0.10	95.61 ± 0.85	99.60 ± 0.03
- Paired Training	97.78 ± 0.04	96.98 ± 0.06	91.82 ± 0.13	**97.79** ± 0.56	95.13 ± 0.13	96.17 ± 0.07	92.11 ± 0.32	98.81 ± 0.02

* Full BERT SLU is trained using the whole standard training dataset, which indicates the performance upper bound.

† implies a significant improvement (p-value < 0.05) using t-test over baseline BERT SLU model.

Table 2: ***Slot Error Analysis in Length Generalization Split.*** L ***denotes the number of slots/utterance in test split***

	ATIS (F1 score)		SNIPS (F1 score)	
L	BERT SLU	Comp. SLU	BERT SLU	Comp. SLU
2	76.0	**88.46**	100.0	100.0
3	89.35	**94.01**	93.26	**95.43**
4	93.56	**95.58**	93.56	**95.05**
5	91.34	**92.19**	93.02	**97.78**
6	95.77	95.77	91.49	**96.91**
7	92.86	**100.0**	N/A	N/A

* In our SNIPS test set there are no utterances with more than 6 slots.

Table 3: ***Performance on Standard Splits***

	ATIS		SNIPS	
Model \ Dataset	Slot (F1)	Intent (acc)	Slot (F1)	Intent (acc)
BERT SLU	98.25 ± 0.02	97.8 ± 0.03	96.57 ± 0.16	98.97 ± 0.02
Comp. SLU	**98.42**† ± 0.09	**98.2**† ± 0.02	**96.85**† ± 0.09	**99.11**† ± 0.07

* For this experiment on SNIPS, we use the bert-base-cased model which has a slightly better performance. † implies a significant improvement (p-value < 0.05) using t-test over baseline BERT SLU model.

relative position embedding improves slot tagging slightly on SNIPS, but has poor intent accuracy. In contrast, our compositional SLU model improves slot tagging performance around 1% over baseline while maintaining similar intent accuracy. In ATIS, we observe 1% drop in F1 score of baseline model in test set. Our compositional model still improves performance over baseline. Note that, most models seem to perform better compositional generalization in ATIS split, than on SNIPS split. This is because, although both ATIS and SNIPS test sets have similar number of slot combinations (210 in ATIS, and 201 in SNIPS), in the compositional training set ATIS has much larger number of distinct combinations 661, versus only 406 for SNIPS. This helps models in ATIS learn better compositionality.

Results on length generalization split: Next, we investigate the ability of SLU models to perform length generalization. For this experiment we consider the split with maximum two slots per training utterance. Table 1 compares the model performance for SNIPS, and ATIS splits. Recall that, the top row indicates an upper bound on performance when the baseline model is trained on *full* training set T_{train}. We observe that when trained on compositional training set, the baseline model's slot tagging performance drops significantly; 5% for SNIPS, and 7% for ATIS. The intent accuracy suffer 3% degradation in ATIS, and 1% in SNIPS. BERT with dependency parse information show a small improvement. BERT with relative position embedding improves slot tagging in ATIS, but not in SNIPS indicating poor generalization across datasets. In contrast, our compositional SLU model significantly improves slot tagging performance, with about 5% F1 score in ATIS and 4% F1 score in SNIPS. The models also achieve similar intent accuracy as baseline. We further analyze the distribution of F1 scores w.r.t. the number of slots per test utterance in Table 2. We observe our compositional model consistently improves F1 score over baseline model, irrespective of the number of slots/ utterance.

Results on standard split: We also train our compositional model on full standard training set T_{train}, and evaluate on standard test set T_{test} for both ATIS and SNIPS. In Table 3, we compare the performance with baseline BERT SLU trained with same hyper-parameters. We observe that the compositional model achieves similar or better accuracy and F1 scores than baseline model in both datasets. Recall that, the standard train/test split in these benchmark datasets were generated randomly and hence they have a similar distribution. So it is expected that the BERT SLU model can have comparable performance to a compositional SLU model. However, as we discussed in Section 1, in real world cold-start settings this is often not the case which necessitates our compositional SLU model.

Ablation study: Finally, we perform an ablation study to better understand the contribution of each component of our compositional SLU model. The two bottom rows in Table 1 show the performance of our model when individual components (a) compositional loss, and (b) paired training are removed. We observe that for novel slot combination split, the model suffers similar drop in F1 score in test set, when the above two components are removed. This indicates they have a similar effect on the model for this split. However, in length generalization split of both datasets, the compositional model suffer significant drop (around 3%) in F1 score when paired training is removed, but suffer smaller degradation by removing the compositional objective. This supports our hypothesis that paired training plays a significant effect in length generalization.

5. Conclusion

In this work, we demonstrate that SOTA SLU models based on pre-trained language models have poor generalization when: (1) an utterance has a novel combination of slots unseen during training, and (2) when an utterance has more slots than in any training utterances; scenarios which the models often encounter in practice. We develop a new compositional SLU model to tackle these issues. First showing that, by adding a new compositional loss, the model's attention distribution can better focus on informative words, thereby improving model's generalization to novel slot combination. We further propose a new paired training data augmentation technique which greatly improves length generalization. In our future work, we want to further explore the impact of OOV slot values on compositionality.

6. References

[1] D. Hakkani-Tür, G. Tür, A. Celikyilmaz, Y.-N. Chen, J. Gao, L. Deng, and Y.-Y. Wang, "Multi-domain joint semantic frame parsing using bi-directional rnn-lstm." in *INTERSPEECH*, 2016, pp. 715–719.

[2] B. Liu and I. Lane, "Attention-based recurrent neural network models for joint intent detection and slot filling," in *Interspeech 2016*, 2016, pp. 685–689.

[3] Y. Kim, S. Lee, and K. Stratos, "ONENET: joint domain, intent, slot prediction for spoken language understanding," in *2017 IEEE Automatic Speech Recognition and Understanding Workshop*, 2017, pp. 547–553.

[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 *Proc. of the 16th NAACL-HLT*, 2018.

[5] Y. Wang, Y. Shen, and H. Jin, "A bi-model based RNN semantic frame parsing model for intent detection and slot filling," in *Proc. of the 2018 NAACL-HLT, Volume 2 (Short Papers)*, 2018, pp. 309–314.

[6] N. Chomsky, "Syntactic structures," *Mouton*, 1957.

[7] J. A. Fodor and Z. W. Pylyshyn, "Connectionism and cognitive architecture: A critical analysis," *Cognition*, vol. 28, no. 1-2, pp. 3–71, 1988.

[8] Q. Chen, Z. Zhuo, and W. Wang, "BERT for joint intent classification and slot filling," *CoRR*, vol. abs/1902.10909, 2019. [Online]. Available: http://arxiv.org/abs/1902.10909

[9] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "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, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*. Association for Computational Linguistics, 2019, pp. 4171–4186.

[10] Y. Liu, M. Ott, N. Goyal, J. Du, M. Joshi, D. Chen, O. Levy, M. Lewis, L. Zettlemoyer, and V. Stoyanov, "Roberta: A robustly optimized BERT pretraining approach," *CoRR*, vol. abs/1907.11692, 2019. [Online]. Available: http://arxiv.org/abs/1907.11692

[11] Y. Shen, A. Ray, A. Patel, and H. Jin, "CRUISE: cold-start new skill development via iterative utterance generation," in *Proceedings of ACL 2018, System Demonstrations*. Association for Computational Linguistics, 2018, pp. 105–110.

[12] B. M. Lake, "Compositional generalization through meta sequence-to-sequence learning," in *Advances in Neural Information Processing Systems 32: Annual Conference on Neural Information Processing Systems 2019, NeurIPS 2019, December 8-14, 2019, Vancouver, BC, Canada*, 2019, pp. 9788–9798.

[13] J. Gordon, D. Lopez-Paz, M. Baroni, and D. Bouchacourt, "Permutation equivariant models for compositional generalization in language," in *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020.

[14] M. I. Nye, A. Solar-Lezama, J. Tenenbaum, and B. M. Lake, "Learning compositional rules via neural program synthesis," in *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.

[15] X. Chen, C. Liang, A. W. Yu, D. Song, and D. Zhou, "Compositional generalization via neural-symbolic stack machines," in *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.

[16] D. A. Dahl, M. Bates, M. Brown, W. M. Fisher, K. Hunicke-Smith, D. S. Pallett, C. Pao, A. I. Rudnicky, and E. Shriberg, "Expanding the scope of the ATIS task: The ATIS-3 corpus," in *Human Language Technology, Proceedings of a Workshop held at Plainsboro, New Jerey, USA*. Morgan Kaufmann, 1994.

[17] Y. He and S. J. Young, "Semantic processing using the hidden vector state model," *Comput. Speech Lang.*, vol. 19, no. 1, pp. 85–106, 2005.

[18] A. Coucke, A. Saade, A. Ball, T. Bluche, A. Caulier, D. Leroy, C. Doumouro, T. Gisselbrecht, F. Caltagirone, T. Lavril, M. Primet, and J. Dureau, "Snips voice platform: an embedded spoken language understanding system for private-by-design voice interfaces," *CoRR*, vol. abs/1805.10190, 2018.

[19] A. Ray, Y. Shen, and H. Jin, "Iterative delexicalization for improved spoken language understanding," in *Interspeech 2019, 20th Annual Conference of the International Speech Communication Association, Graz, Austria, 15-19 September 2019*. ISCA, 2019, pp. 1183–1187.

[20] Y. Yan, K. He, H. Xu, S. Liu, F. Meng, M. Hu, and W. Xu, "Adversarial semantic decoupling for recognizing open-vocabulary slots," in *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*. Association for Computational Linguistics, 2020, pp. 6070–6075.

[21] B. M. Lake and M. Baroni, "Generalization without systematicity: On the compositional skills of sequence-to-sequence recurrent networks," in *Proceedings of the 35th International Conference on Machine Learning, ICML 2018, Stockholmsmässan, Stockholm, Sweden, July 10-15, 2018*, ser. Proceedings of Machine Learning Research, vol. 80, 2018, pp. 2879–2888.

[22] L. Ruis, J. Andreas, M. Baroni, D. Bouchacourt, and B. M. Lake, "A benchmark for systematic generalization in grounded language understanding," in *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020*, 2020.

[23] D. Hupkes, V. Dankers, M. Mul, and E. Bruni, "Compositionality decomposed: How do neural networks generalise?" *J. Artif. Intell. Res.*, vol. 67, pp. 757–795, 2020.

[24] E. Akyürek, A. F. Akyürek, and J. Andreas, "Learning to recombine and resample data for compositional generalization," in *9th International Conference on Learning Representations, ICLR 2021*. OpenReview.net, 2021.

[25] T. Gao, Q. Huang, and R. J. Mooney, "Systematic generalization on gscan with language conditioned embedding," in *Proceedings of AACL/IJCNLP 2020*. Association for Computational Linguistics, 2020, pp. 491–503.

[26] Z. Zhao, S. Zhu, and K. Yu, "Data augmentation with atomic templates 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 2019, Hong Kong, China, November 3-7, 2019*. Association for Computational Linguistics, 2019, pp. 3635–3641.

[27] V. Cirik, T. Berg-Kirkpatrick, and L. Morency, "Using syntax to ground referring expressions in natural images," in *Proc. of the 32nd AAAI Conference on Artificial Intelligence, (AAAI-18), the 30th innovative Applications of Artificial Intelligence (IAAI-18), and the 8th AAAI Symposium on Educational Advances in Artificial Intelligence (EAAI-18)*. AAAI Press, 2018, pp. 6756–6764.

[28] Y. Kuo, B. Katz, and A. Barbu, "Compositional networks enable systematic generalization for grounded language understanding," in *Findings of the Association for Computational Linguistics: EMNLP 2021*. Association for Computational Linguistics, 2021, pp. 216–226.

[29] Z. Huang, D. Liang, P. Xu, and B. Xiang, "Improve transformer models with better relative position embeddings," in *Findings of the Association for Computational Linguistics: EMNLP 2020, Online Event, 16-20 November 2020*, ser. Findings of ACL, vol. EMNLP 2020, 2020, pp. 3327–3335.

[30] S. Ontañón, J. Ainslie, Z. Fisher, and V. Cvicek, "Making transformers solve compositional tasks," in *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL*. Association for Computational Linguistics, 2022, pp. 3591–3607.