Simple Data Augmentation for Multilingual NLU in Task Oriented Dialogue Systems

Samuel Louvan
University of Trento
Fondazione Bruno Kessler
slouvan@fbk.eu

Bernardo Magnini
Fondazione Bruno Kessler
magnini@fbk.eu

Abstract

Data augmentation has shown potential in alleviating data scarcity for Natural Language Understanding (e.g. slot filling and intent classification) in task-oriented dialogue systems. As prior work has been mostly experimented on English datasets, we focus on five different languages, and consider a setting where limited data are available. We investigate the effectiveness of non-gradient based augmentation methods, involving simple text span substitutions and syntactic manipulations. Our experiments show that (i) augmentation is effective in all cases, particularly for slot filling; and (ii) it is beneficial for a joint intent-slot model based on multilingual BERT, both for limited data settings and when full training data is used.

1 Introduction

Natural Language Understanding (NLU) in task-oriented dialogue systems is responsible for parsing user utterances to extract the intent of the user and the arguments of the intent (i.e. *slots*) into a semantic representation, typically a semantic frame (Tur and De Mori, 2011). For example, the utterance "*Play Jeff Pilson on Youtube*" has the intent PLAYMUSIC and "*Youtube*" as value for the slot SERVICE. As more skills are added to the dialogue system, the NLU model frequently needs to be updated to scale to new domains and languages, a situation which typically becomes problematic when labeled data are limited (*data scarcity*).

One way to combat data scarcity is through data augmentation (DA) techniques performing *label preserving* operations to produce auxiliary training data. Recently, DA has shown potential in tasks such as machine translation (Fadaee et al., 2017), constituency and dependency parsing (Şahin and Steedman, 2018; Vania et al., 2019), and text classification (Wei and Zou, 2019; Kumar et al., 2020). As for slot filling (SF) and intent classification (IC), a number of DA methods have been proposed to generate synthetic utterances using sequence to sequence models (Hou et al., 2018; Zhao et al., 2019), Conditional Variational Auto Encoder (Yoo et al., 2019), or pre-trained NLG models (Peng et al., 2020). To date, most of the DA methods are evaluated on English and it is not clear whether the same finding apply to other languages.

In this paper, we study the effectiveness of DA on several non-English datasets for NLU in task-oriented dialogue systems. We experiment with existing lightweight, non-gradient based, DA methods from Louvan and Magnini (2020) that produces varying slot values through substitution and sentence structure manipulation by leveraging syntactic information from a dependency parser. We evaluate the DA methods on NLU datasets from five languages: Italian, Hindi, Turkish, Spanish, and Thai. The contributions of our paper are as follows:

1. We assess the applicability of DA methods for NLU in task-oriented dialogue systems in five languages.
2. We demonstrate that simple DA can improve performance on all languages despite different characteristic of the languages.
3. We show that a large pre-trained multilingual BERT (M-BERT) (Devlin et al., 2019) can still benefit from DA, in particular for slot filling.

2 Slot Filling and Intent Classification

The NLU component of a task-oriented dialogue system is responsible in a parsing user utterance into a semantic representation, such as semantic

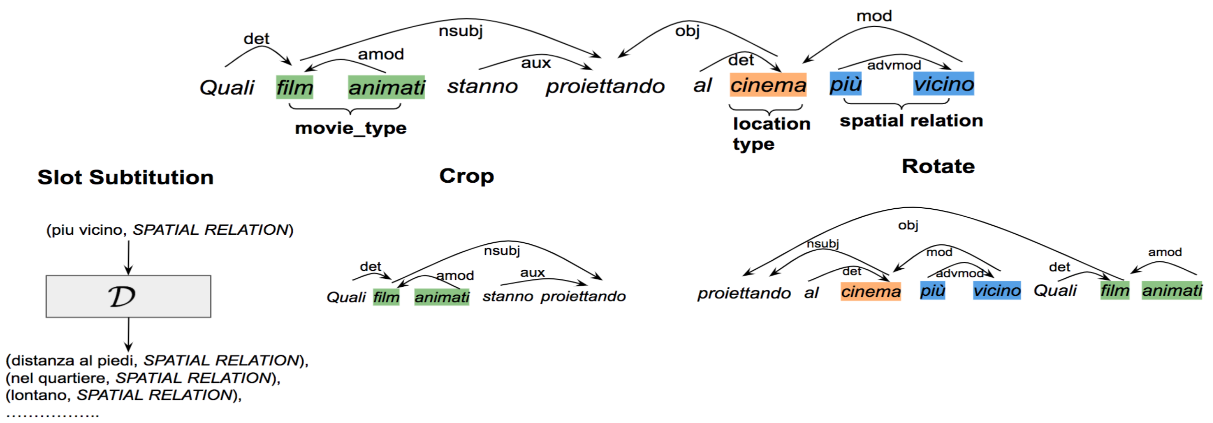
det
nsubj
amod
aux
obj
det
mod
advmod
Quali film animati stanno proiettando al cinema più vicino
movie_type
location type
spatial relation
Slot Subtitution
Crop
Rotate
(piu vicino, SPATIAL RELATION)
$\mathcal{D}$
(distanza al piedi, SPATIAL RELATION),
(nel quartiere, SPATIAL RELATION),
(lontano, SPATIAL RELATION),
....
Quali film animati stanno proiettando
proiettando al cinema più vicino Quali film animati

Figure 1: Augmentation operations performed on an utterance, "*Quali film animati stanno proiettando al cinema piu vicino*" ("*Which animated films are showing at the nearest cinema*"). The utterance is taken from the Italian SNIPS dataset.

frame. The semantic frame conveys information, namely the user intent and the corresponding arguments of the intent. Extracting such information involves slot filling (SF) and intent classification (IC) tasks.

Given an input utterance of n tokens, $x = (x_1, x_2, .., x_n)$, the system needs to assign a particular intent y^{intent} for the whole utterance x and the corresponding slots that are mentioned in the utterance $y^{slot} = (y_1^{slot}, y_2^{slot}, .., y_n^{slot})$. In practice, IC is typically modeled as text classification and SF as a sequence tagging problem. As an example, for the utterance "*Play Jeff Pilson on Youtube*", y^{intent} is PlayMusic, as the intent of the user is to ask the system to play a song from a musician and y^{slot} = (O, B-ARTIST, I-ARTIST, O, B-SERVICE) in which the artist is "*Jeff Pilson*" and the service is "*Youtube*"". Slot labels are in BIO format: B indicates the start of a slot span, I the inside of a span while O denotes that the word does not belong to any slot. Recent approaches for SF and IC are based on neural network methods that models SF and IC jointly (Goo et al., 2018; Chen et al., 2019) by sharing model parameter among both tasks.

3 Data Augmentation (DA) Methods

DA aims to perform *semantically preserving* transformations on the training data $\mathcal{D}$ to produce auxiliary data $\mathcal{D}'$. The union of $\mathcal{D}$ and $\mathcal{D}'$ is then used to train a particular NLU model. For each utterance in $\mathcal{D}$, we produce N augmented utterances by applying a specific augmentation operation. We adopt a subset of existing augmentation methods from Louvan and Magnini (2020), that has shown promising results on English datasets. We describe the augmentation operations in the following sections.

3.1 Slot Substitution (Slot-Sub)

Slot-Sub (Figure 1 left) performs augmentation by substituting a particular text span (*slot-value pair*) in an utterance with a different text span that is semantically consistent i.e., the slot label is the same. For example, in the utterance "*Quali film animati stanno proiettando al cinema più vicino*", one of the spans that can be substituted is the slot value pair (*più vicino*, SPATIAL RELATION). Then, we collect other spans in $\mathcal{D}$ in which the slot values are different, but the slot label is the same. For instance, we found the substitute candidates $SP' = \{($"*distanza a piedi*", SPATIAL RELATION), ("*lontano*", SPATIAL RELATION), ("*nel quartiere*", SPATIAL RELATION), $\ldots\}$, and then we sample one span to replace the original span in the utterance.

3.2 Crop and Rotate

In order to produce sentence variations, we apply the crop and rotate operations proposed in Şahin and Steedman (2018), which manipulate the sentence structure through its dependency parse tree. The goal of Crop (Figure 1 middle) is to simplify the sentence so that it focuses on a particular *fragment* (e.g. subject/object) by removing other fragments in the sentence. Crop uses the dependency tree to identify the fragment and then remove it and its children from the dependency tree.

		#Label		**#Utterances** ($\mathcal{D}$)			**#Augmented Utterances** ($\mathcal{D}'$)		
Dataset	**Language**	**#slot**	**#intent**	**#train**	**#dev**	**#test**	**#Slot-Sub**	**#Crop**	**#Rotate**
SNIPS-IT	Italian	39	7	574	700	698	5,404	1,431	1,889
ATIS-HI	Hindi	73	17	176	440	893	1,286	460	472
ATIS-TR	Turkish	70	17	99	248	715	144	161	194
FB-ES	Spanish	11	12	361	1,983	3,043	1,455	769	1,028
FB-TH	Thai	8	10	215	1,235	1,692	781	-	-

Table 1: Statistics on the datasets. #train indicates our limited training data setup (*10% of full training data*). $\mathcal{D}'$ is produced by tuning the number of augmentations per utterance (N) on the dev set.

Model	**DA**	**SNIPS-IT**		**ATIS-HI**		**ATIS-TR**		**FB-ES**		**FB-TH**	
		Slot	Intent	Slot	Intent	Slot	Intent	Slot	Intent	Slot	Intent
M-Bert	None	78.25	94.99	69.57	86.57	64.36	78.98	84.13	97.68	56.06	89.80
	Slot-Sub	**81.97**†	94.93	**72.44**†	87.29	66.60†	79.85	**84.27**	97.72	**59.68**†	**91.42**†
	Crop	80.12†	94.60	70.04	86.92	65.11	79.48	83.85	98.08†	-	-
	Rotate	79.24†	**95.37**	70.69	**87.60**†	65.20	80.06	83.28	**98.20**†	-	-
	Combine	81.27†	95.00	72.13†	86.93	**66.68**†	**81.12**†	83.67	97.94	-	-

Table 2: Performance comparison of the baseline and augmentation methods on the test set. F1 score is used for slot filling and accuracy for intent classification. Scores are the average of 10 different runs. $\dagger$ indicates statistically significant improvement over the baseline (p-value < 0.05 according to Wilcoxon signed rank test).

The Rotate (Figure 1 right) operation is performed by moving a particular fragment (including subject/object) around the root of the tree, typically the verb in the sentence. For each operation, all possible combinations are generated, and one of them is picked randomly as the augmented sentence. Both Crop and Rotate rely on the universal dependency labels (Nivre et al., 2017) to identify relevant fragments, such as NSUBJ (nominal subject), DOBJ (direct object), OBJ (object), IOBJ (indirect object).

4 Experiments

Our primary goal is to verify the effectiveness of data augmentation on Italian, Hindi, Turkish, Spanish and Thai NLU datasets with limited labeled data. To this end, we compare the performance of a baseline NLU model trained on the original training data ($\mathcal{D}$) with a NLU model that incorporates the augmented data as additional training instances ($\mathcal{D} + \mathcal{D}'$). To simulate the limited labeled data situation we randomly sample 10% of the training data for each dataset.

Baseline and Data Augmentation (DA) Methods. We use the state of the art BERT-based joint intent slot filling model (Chen et al., 2019) as the baseline model. We leverage the pre-trained *multilingual* BERT (M-Bert), which is trained on 104 languages. During training, M-Bert is fine tuned on the slot filling and intent classification tasks. Given a sentence representation $x = ([CLS]\, t_1\, t_2 \dots t_L)$, we use the hidden state $h_{[CLS]}$ to predict the intent, and $h_{t_i}$to predict the slot label. As for DA methods, in addition to the methods described in Section 2, we add one configuration Combine, which combines the result of Slot-Sub and Rotate, as Rotate obtains better results than Crop on the development set.

Settings. The model is trained with the BertAdam optimizer for 30 epochs with early stopping. The learning rate is set to 10^{-5} and batch size is 16. All the hyperparameters are listed in Appendix A. For Slot-Sub the number of augmentation per sentence N is tuned on the development set. To produce the dependency tree, we parse the sentence using Stanza (Qi et al., 2020). For both Crop and Rotate we follow the default hyperparameters from Şahin and Steedman (2018). We did not experiment with Thai for Crop and Rotate as Thai is not supported by Stanza. The number of augmented sentences ($\mathcal{D}'$) for each method is listed in Table 1. For evaluation metric, we use the standard CoNLL script to compute F1 score for slot filling and accuracy for intent classification.

Datasets. For the Italian language, we use the data from Bellomaria et al. (2019), translated from the English SNIPS dataset (Coucke et al., 2018). SNIPS has been widely used for evaluating NLU models and consists of utterances in multiple domains. As for Hindi and Turkish, we use the ATIS dataset from Upadhyay et al. (2018), derived from Hemphill et al. (1990). ATIS is a well known NLU dataset on flight domain. As for Spanish and Thai we use the FB dataset from Schuster et al. (2019) that contains utterances in alarm, weather, and reminder domains. The overall statistics of the datasets are shown in Table 1.

5 Results

The overall results reported in Table 2 show that applying DA improves performance on slot filling and intent classification across all languages. In particular, for SF, the Slot-Sub method yields the best result, while for IC, Rotate obtains better performance compared to Crop in most cases. These results are consistent with the finding from Louvan and Magnini (2020) on the English dataset, where Slot-Sub improves SF and Crop or Rotate improve IC. In general, Rotate is better than Crop for most cases on IC, and we think this is because Crop may change the intent of the original sentence. Intents typically depend on the occurrence of specific slots, so when the cropped part is a slot-value, it may change the sentence's overall semantics.

We can see that languages with different typological features (e.g. subject/verb/object ordering)[1] benefit from Rotate operation for IC. This result suggests that augmentation can produce useful noise (regularization) for the model to alleviate overfitting when labeled data is limited. When we use Combine, it still helps the performance of both SF and IC, although the improvements are not as high as when only one of the augmentation method is applied. The only language that gets the benefits the most from Combine is Turkish. We hypothesize that as Turkish has a more flexible word order than the other languages it benefits the most when Rotate is performed.

Performance on varying data size. To better understand the effectiveness of Slot-Sub, we perform further analysis on different training data size (see Figure 2). Overall, we observe that as we increase the training size, the benefit of Slot-Sub is decreasing for all datasets. For some datasets, namely ATIS-HI and FB-ES, Slot-Sub can cause performance drop for larger data size, although it is reasonably small (less than 1 F1 point). FB-TH consistently benefits from Slot-Sub even when full training data is used. Until which training data size the improvement is significant vary across datasets[2]. For SNIPS-IT, improvement is clear for all training data size and they are statistically significant up until the training data size is 80%. For ATIS-HI improvements are significant until data size of 40%. As for FB datasets, improvements are significant only until the training data size is 10%. Overall, we can see that Slot-Sub is effective for cases where data is scarce (5%, 10%), while it is still relatively robust for larger data size on all datasets.

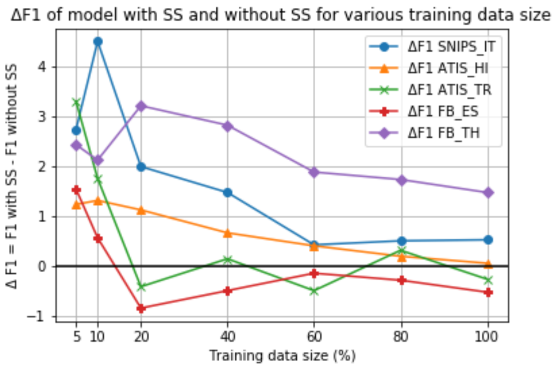

Figure 2: Improvement ($\Delta F1$) obtained by Slot-Sub (SS) on different training data size. Positive numbers mean that the model with SS yields gain.

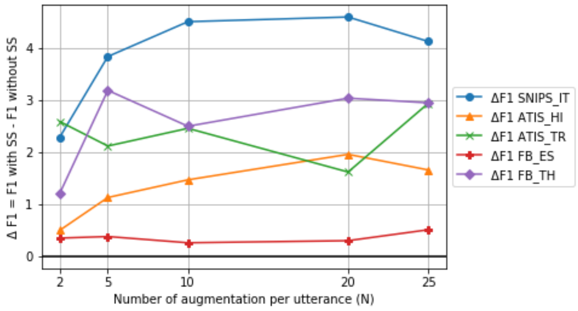

Figure 3: Gain ($\Delta F1$) obtained by Slot-Sub (SS) on various number of augmented sentence (N). Positive numbers mean that the model with SS yields gain.

[1] Italian, Spanish, and Thai are SVO languages while Hindi and Turkish are SOV languages.

[2] For more details of the p-value of the statistical tests please refer to Appendix B

Performance on different numbers of augmentation per utterance (N). We examine the effect of a larger number of augmentations per utterance (N) to the model performance, specifically for SF (see Figure 3). For FB-ES, similarly to the results in Table 2, increasing N does not affect the performance. For the other datasets, increasing N brings performance improvement. For ATIS-HI, SNIPS-IT, and FB-TH the trend is that, as we increase N, performance goes up and plateau. For ATIS-TR, changing N does not really affect the gain of the performance as the performance trend is quite steady across number of augmentations. For most combinations of N in each dataset (except FB-ES), the difference between the performance of model that using SLOT-SUB and the model that does not use SLOT-SUB is significant [3].

6 Related Work

Data augmentation methods that has been proposed in NLP aims to automatically produce additional training data through different kinds of methods ranging from simple word substitution (Wei and Zou, 2019) to more complex methods that aims to produce semantically preserving sentence generation (Hou et al., 2018; Gao et al., 2020). In the context of slot filling and intent classification, recent augmentation methods typically apply deep learning models to produce augmented utterances.

Hou et al. (2018) proposes a two-stages methods to produce the delexicalized utterances generation and slot values realization. Their method is based on a sequence to sequence based model (Sutskever et al., 2014) to produce a paraphrase of an utterance with its slot values placeholder (delexicalized) for a given intent. For the slot values lexicalization, they use the slot values in the training data that occur in similar contexts. Zhao et al. (2019) trains a sequence to sequence model with training instances that consist of a pair of atomic templates of dialogue acts and its sentence realization. Yoo et al. (2019) proposes a solution by extending Variational Auto Encoder (VAE) (Kingma and Welling, 2014) into a Conditional VAE (CVAE) to generate synthetic utterances. The CVAE controls the utterance generation by conditioning on the intent and slot labels during model training. Recent work from Peng et al. (2020) make use of Transformer (Vaswani et al., 2017) based pre-trained NLG namely GPT-2 (Radford et al., 2019), and fine-tune it to slot filling dataset to produce synthetic utterances. We consider these deep learning based approaches as *heavyweight* as they often require several stages in the augmentation process namely generating augmentation candidates, ranking and filtering the candidates before producing the final augmented data. Consequently, the computation time of these approaches is generally more expensive as separate training is required to train the augmentation and joint SF-IC models. Recent work from Louvan and Magnini (2020) apply a set of *lightweight* methods in which most of the augmentation methods do not require model training. The augmentation methods focus on varying the slot values through substitution mechanisms and varying sentence structure through dependency tree manipulation. While the methods are relatively simple it obtains competitive results with deep learning based approaches on the standard English slot filling benchmark datasets namely ATIS (Hemphill et al., 1990), SNIPS (Coucke et al., 2018), and FB (Schuster et al., 2019) datasets.

Existing methods mostly evaluate their approaches on English datasets, and little work has been done on other languages. Our work focuses on investigating the effect of data augmentation on five non-English languages. We apply a subset of *lightweight* augmentation methods from Louvan and Magnini (2020) that do not require separate model training to produce augmentation data.

7 Conclusion

We evaluate the effectiveness of data augmentation for slot filling and intent classification tasks in five typologically diverse languages. Our results show that by applying simple augmentation, namely slot values substitutions and dependency tree manipulations, we can obtain substantial improvement in most cases when only small amount of training data is available. We also show that a large pre-trained multilingual BERT benefits from data augmentation.

Acknowledgments

We thank Valentina Bellomaria for providing the Italian SNIPS dataset. We thank Clara Vania for the feedback on the early draft of the paper.

[3] For more details of the p-value of the statistical tests please refer to Appendix B

References

Valentina Bellomaria, Giuseppe Castellucci, Andrea Favalli, and Raniero Romagnoli. 2019. Almawave-slu: A new dataset for SLU in italian. In Raffaella Bernardi, Roberto Navigli, and Giovanni Semeraro, editors, *Proceedings of the Sixth Italian Conference on Computational Linguistics, Bari, Italy, November 13-15, 2019*, volume 2481 of *CEUR Workshop Proceedings*. CEUR-WS.org.

Qian Chen, Zhu Zhuo, and Wen Wang. 2019. Bert for joint intent classification and slot filling. *arXiv preprint arXiv:1902.10909*.

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*, 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, June. Association for Computational Linguistics.

Marzieh Fadaee, Arianna Bisazza, and Christof Monz. 2017. Data augmentation for low-resource neural machine translation. In Regina Barzilay and Min-Yen Kan, editors, *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 2: Short Papers*, pages 567–573. Association for Computational Linguistics.

Silin Gao, Yichi Zhang, Zhijian Ou, and Zhou Yu. 2020. Paraphrase augmented task-oriented dialog generation. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel R. Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 639–649. Association for Computational Linguistics.

Chih-Wen Goo, Guang Gao, Yun-Kai Hsu, Chih-Li Huo, Tsung-Chieh Chen, Keng-Wei Hsu, and Yun-Nung Chen. 2018. Slot-gated modeling for joint slot filling and intent prediction. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 2 (Short Papers)*, pages 753–757.

Charles T. Hemphill, John J. Godfrey, and George R. Doddington. 1990. The ATIS spoken language systems pilot corpus. In *Speech and Natural Language: Proceedings of a Workshop Held at Hidden Valley, Pennsylvania, USA, June 24-27, 1990*. Morgan Kaufmann.

Yutai Hou, Yijia Liu, Wanxiang Che, and Ting Liu. 2018. Sequence-to-sequence data augmentation for dialogue language understanding. In *Proceedings of the 27th International Conference on Computational Linguistics*, pages 1234–1245, Santa Fe, New Mexico, USA, August. Association for Computational Linguistics.

Diederik P. Kingma and Max Welling. 2014. Auto-encoding variational bayes. In Yoshua Bengio and Yann LeCun, editors, *2nd International Conference on Learning Representations, ICLR 2014, Banff, AB, Canada, April 14-16, 2014, Conference Track Proceedings*.

Varun Kumar, Ashutosh Choudhary, and Eunah Cho. 2020. Data augmentation using pre-trained transformer models. *arXiv preprint arXiv:2003.02245*.

Samuel Louvan and Bernardo Magnini. 2020. Simple is better! lightweight data augmentation for low resource slot filling and intent classification. *arXiv preprint https://arxiv.org/abs/2009.03695*. PACLIC 2020 - The 34th Pacific Asia Conference on Language, Information and Computation.

Joakim Nivre, Željko Agić, Lars Ahrenberg, Lene Antonsen, Maria Jesus Aranzabe, Masayuki Asahara, Luma Ateyah, Mohammed Attia, Aitziber Atutxa, Liesbeth Augustinus, et al. 2017. Universal dependencies 2.1.

Baolin Peng, Chenguang Zhu, Michael Zeng, and Jianfeng Gao. 2020. Data augmentation for spoken language understanding via pretrained models. *CoRR*, abs/2004.13952.

Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. 2020. Stanza: A python natural language processing toolkit for many human languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 101–108, Online, July. Association for Computational Linguistics.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners.

Gözde Gül Şahin and Mark Steedman. 2018. Data augmentation via dependency tree morphing for low-resource languages. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 5004–5009, Brussels, Belgium, October-November. Association for Computational Linguistics.

Sebastian Schuster, Sonal Gupta, Rushin Shah, and Mike Lewis. 2019. Cross-lingual transfer learning for multilingual task oriented dialog. 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 3795–3805, Minneapolis, Minnesota, June. Association for Computational Linguistics.

Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In Zoubin Ghahramani, Max Welling, Corinna Cortes, Neil D. Lawrence, and Kilian Q. Weinberger, editors, *Advances in Neural Information Processing Systems 27: Annual Conference on Neural Information Processing Systems 2014, December 8-13 2014, Montreal, Quebec, Canada*, pages 3104–3112.

Gokhan Tur and Renato De Mori. 2011. *Spoken language understanding: Systems for extracting semantic information from speech*. John Wiley & Sons.

Shyam Upadhyay, Manaal Faruqui, Gokhan Tür, Hakkani-Tür Dilek, and Larry Heck. 2018. (almost) zero-shot cross-lingual spoken language understanding. In *2018 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6034–6038. IEEE.

Clara Vania, Yova Kementchedjhieva, Anders Søgaard, and Adam Lopez. 2019. A systematic comparison of methods for low-resource dependency parsing on genuinely low-resource languages. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, editors, *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*, pages 1105–1116. Association for Computational Linguistics.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, 4-9 December 2017, Long Beach, CA, USA*, pages 5998–6008.

Jason W. Wei and Kai Zou. 2019. EDA: easy data augmentation techniques for boosting performance on text classification tasks. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, editors, *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*, pages 6381–6387. Association for Computational Linguistics.

Kang Min Yoo, Youhyun Shin, and Sang-goo Lee. 2019. Data augmentation for spoken language understanding via joint variational generation. In *The Thirty-Third AAAI Conference on Artificial Intelligence, AAAI 2019, The Thirty-First Innovative Applications of Artificial Intelligence Conference, IAAI 2019, The Ninth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2019, Honolulu, Hawaii, USA, January 27 - February 1, 2019*, pages 7402–7409. AAAI Press.

Zijian Zhao, Su Zhu, and Kai Yu. 2019. Data augmentation with atomic templates for spoken language understanding. In Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan, editors, *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*, pages 3635–3641. Association for Computational Linguistics.

Appendix A. Hyperparameters

Hyperparameter	Value
Learning rate	10^{-5}
Dropout	0.1
Mini-batch size	16
Optimizer	BertAdam
Number of epoch	30
Early stopping	10
N	Tuned on $\{2, 5, 10\}$
Max rotation	3
Max crop	3

Table 3: List of hyperparameters used for the BERT model and data augmentation methods

Appendix B. Statistical Significance

Dataset	Nb Aug	p-value
ATIS-TR	2	0.005062032126
	5	0.01251531869
	10	0.006910429808
	20	0.5001842571
	25	0.07961580146
ATIS-HI	2	0.1097446387
	5	0.005062032126
	10	0.005062032126
	20	0.04311444678
	25	0.04311444678
SNIPS-IT	2	0.005062032126
	5	0.005062032126
	10	0.005062032126
	20	0.04311444678
	25	0.04311444678
FB-ES	2	0.0663160313
	5	0.02831405495
	10	0.09260069782
	20	0.3452310718
	25	0.07961580146
FB-TH	2	0.03665792867
	5	0.005062032126
	10	0.005062032126
	20	0.04311444678
	25	0.04311444678

Table 5: The p-values of statistical tests on the experiments on Figure 3

Dataset	Training Size (%)	p-value
ATIS-HI	5	0.04311444678
	10	0.005062032126
	20	0.04311444678
	40	0.04311444678
	80	0.1380107376
	100	0.2733216783
ATIS-TR	5	0.224915884
	10	0.005062032126
	20	0.7150006547
	40	0.1797124949
	80	0.1797124949
	100	0.1797124949
SNIPS-IT	5	0.04311444678
	10	0.005062032126
	20	0.04311444678
	40	0.04311444678
	80	0.04311444678
	100	0.04311444678
FB-ES	5	0.04311444678
	10	0.02831405495
	20	0.1797124949
	40	0.1755543028
	80	0.1380107376
	100	0.1797124949
FB-TH	5	0.04311444678
	10	0.005062032126
	20	0.1797124949
	40	0.1797124949
	80	0.1797124949
	100	0.10880943

Table 4: The p-values of statistical tests on the experiments on Figure 2.