# Self-Training using Rules of Grammar for Few-Shot NLU

**Joonghyuk Hahn**[1*], **Hyunjoon Cheon**[1*], **Kyuyeol Han**[2], **Cheongjae Lee**[2],
**Junseok Kim**[2] and **Yo-Sub Han**[1]
[1]Department of Computer Science, Yonsei University, Seoul, Republic of Korea
[2]AIRS Company, Hyundai Motor Group, Seoul, Republic of Korea
{greghahn,hyunjooncheon,emmous}@yonsei.ac.kr,
{kyuyeol.han,lcj8004,junseok.kim}@hyundai.com

## Abstract

We tackle the problem of self-training networks for NLU in low-resource environment—few labeled data and lots of unlabeled data. The effectiveness of self-training is a result of increasing the amount of training data while training. Yet it becomes less effective in low-resource settings due to unreliable labels predicted by the teacher model on unlabeled data. Rules of grammar, which describe the grammatical structure of data, have been used in NLU for better explainability. We propose to use rules of grammar in self-training as a more reliable pseudo-labeling mechanism, especially when there are few labeled data. We design an effective algorithm that constructs and expands rules of grammar without human involvement. Then we integrate the constructed rules as a pseudo-labeling mechanism into self-training. There are two possible scenarios regarding data distribution: it is *unknown* or *known* in prior to training. We empirically demonstrate that our approach substantially outperforms the state-of-the-art methods in three benchmark datasets for both scenarios.

## 1 Introduction

Deep learning for natural language understanding (NLU) achieves satisfactory performance in various tasks such as intent detection (ID) or slot filling (SF) (Liu and Lane, 2016; Goo et al., 2018; E et al., 2019; Wang et al., 2020a). Given a sentence $S = w_1 w_2 \cdots w_n$, ID is a task to find an intent $I$ of $S$ among several possible intents, and SF is to identify keywords in $S$ and tag a correct slot sequence $s_1 s_2 \cdots s_n$ of $S$ in BIO format.

Recent studies have focused on the data scarcity problems for ID and SF tasks. In low-resource settings with few labeled data, the model performance often becomes poor if it is not properly trained to cope with the problem. Wang et al. (2020b) suggest several approaches for few-shot learning such as data augmentation and self-training (ST). Data augmentation has been popular in image processing and recently is being used in NLP applications. However, it is still an open issue if data augmentation is suitable for NLU since the quality of augmented data might not be reliable even if it improves the overall performance (Hedderich et al., 2020; Cengiz and Yuret, 2020). On the other hand, ST (Ratsaby and Venkatesh, 1995; Ye et al., 2020) shows promising performance for low-resource settings (Cho et al., 2019). Nevertheless, classic ST is often unreliable since the pseudo-labeling mechanism heavily depends on the teacher model of its own. This motivates researchers to add more reliable mechanisms to the teacher model and to make better ST models (He et al., 2020; Paul et al., 2019).

Rules of grammar are one of the oldest techniques to represent knowledge in NLP (Rizos et al., 2019; Jiang et al., 2020), and recently have become popular again for few-shot learning due to the reliability and explainability of rules (Luo et al., 2018; Abro et al., 2020). On the other hand, naive rule extraction is easy to overfit on the source corpus and is impossible to recognize data outside of the corpus. Generalizing the rules can solve this issue. We propose to use rules of grammar to enhance the teacher model in ST. Because we have formal grammars as rules, we can expand rules by grouping similar fragments and represent data more precisely with an easy grammar modification. We investigate the usefulness of rules of grammar from a quantitative perspective and study how good these rules for ID and SF, especially in low-resource settings. It is well-known that reflecting a real-world data distribution among different classes makes better models when such information is available in prior training

[*]The first two authors contributed equally to this work.

*Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 4576–4581
November 7–11, 2021.

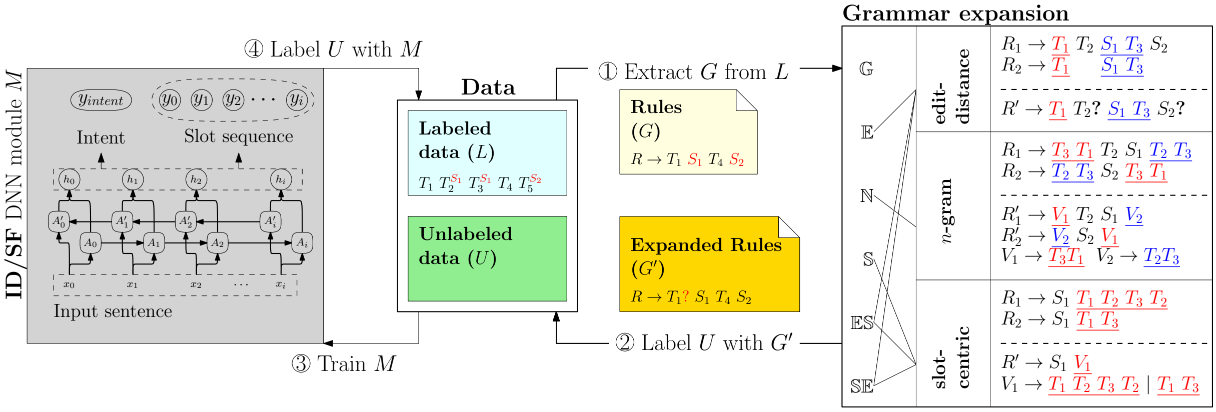
ID/SF DNN module $M$
$y_{intent}$
$y_0$ $y_1$ $y_2$ ⋯ $y_i$
Intent
Slot sequence
$h_0$ $h_1$ $h_2$ $h_i$
$A'_0$ $A'_1$ $A'_2$ $A'_i$
$A_0$ $A_1$ $A_2$ $A_i$
$x_0$ $x_1$ $x_2$ ⋯ $x_i$
Input sentence
④ Label $U$ with $M$
Data
Labeled data ($L$)
$T_1$ $T_2^{S_1}$ $T_3^{S_1}$ $T_4$ $T_5^{S_2}$
Unlabeled data ($U$)
③ Train $M$
① Extract $G$ from $L$
Rules ($G$)
$R \rightarrow T_1\ S_1\ T_4\ S_2$
Expanded Rules ($G'$)
$R \rightarrow T_1?\ S_1\ T_4\ S_2$
② Label $U$ with $G'$
Grammar expansion
$\mathbb{G}$
$\mathbb{E}$
$\mathbb{N}$
$\mathbb{S}$
$\mathbb{ES}$
$\mathbb{SE}$
edit-distance
$R_1 \rightarrow T_1\ T_2\ S_1\ T_3\ S_2$
$R_2 \rightarrow T_1\ S_1\ T_3$
$R' \rightarrow T_1\ T_2?\ S_1\ T_3\ S_2?$
$n$-gram
$R_1 \rightarrow T_3\ T_1\ T_2\ S_1\ T_2\ T_3$
$R_2 \rightarrow T_2\ T_3\ S_2\ T_3\ T_1$
$R'_1 \rightarrow V_1\ T_2\ S_1\ V_2$
$R'_2 \rightarrow V_2\ S_2\ V_1$
$V_1 \rightarrow T_3 T_1$ $V_2 \rightarrow T_2 T_3$
slot-centric
$R_1 \rightarrow S_1\ T_1\ T_2\ T_3\ T_2$
$R_2 \rightarrow S_1\ T_1\ T_3$
$R' \rightarrow S_1\ V_1$
$V_1 \rightarrow T_1\ T_2\ T_3\ T_2 \mid T_1\ T_3$

Figure 1: An overview of our self-training procedure using rules of grammar for accurate ID and SF when there are few labeled data. The rules of grammar are expanded by edit-distance, $n$-gram, and slot-centric methods.

the models (Yang et al., 2020). Thus, we consider two scenarios: when the distribution is unknown, or known in prior.

In summary, our research questions are **RQ1:** Are rules of grammar useful in ST for few-shot NLU? **RQ2:** Which rule expansion gives the best performance? **RQ3:** Does it make a difference to know the data distribution in prior for rules of grammar in ST?

## 2 Methodology

**Grammar extraction:** We substitute the slots in the initial corpus into a slot variable and construct initial rules of grammar. We denote a rule with annotated intent and slot information. For example, the following is a rule for an intent *ground_transport*.

$$R_{ground\_transport} \rightarrow \textit{transportation (in} \mid \textit{on)}\ \underline{\textit{\$day\_name}},$$

It parses a sentence *transportation on* $\underline{\textit{Monday}}$, and identifies its intent as *ground_transport*.

**Grammar expansion:** The initial rules can parse only the sentences with the same structure and different keywords for slots. For example, a rule can parse a sentence *he leaves* but not *they leave*. This problem requires more general rules for parsing diverse sentences. Grammar expansion enables such diversity by relaxing substructures of rules. In particular, we use three grammar expansion methods—edit-distance, $n$-gram, slot-centric—and group similar substructures.

**Edit-distance expansion:** The edit-distance expansion groups the rules according to the structural similarity. The similarity is the edit-distance normalized by the maximum length of rules and a given threshold. Each group of rules with edit-distances lower than the threshold produces a single rule by merging every rule contained in the group. When computing the distance, we prevent edits involving a slot since slots are incompatible with words or different slots.

**$n$-gram expansion:** The $n$-gram expansion aims to maintain the semantics of rules and enables substitution between similar word sequences. The expansion first extracts every 4-word-gram of each rule and count its occurrences. These word-grams make groups based on semantic similarity of them and form a kind of slot represented by a variable. The same procedure repeats for 3- and 2-word-gram.

**Slot-centric expansion:** The slot-centric expansion focuses on slots. It aims to extract every word-only sequences in all rules with common slots. The expansion groups the rules with common slots and replaces the remaining word sequences by a symbol.

**Combining expansions:** A combination of grammar expansions often gives rise to better performance than the individuals. For example, a mix of edit-distance and slot-centric methods expands the expressive power both in syntactic and semantic perspectives. See Figure 2 for an example.

**Self-training:** ST mechanism uses a teacher model trained from a given labeled data and pseudo-labels additional data for successive training. Therefore the performance of the teacher model is crucial in ST. To make a better teacher model, we use rules of grammar, instead of a training model alone, as a pseudo-labeling mechanism

$R_{\text{abbr}} \rightarrow$ Describe fare code \$fare_basis_code
$R_{\text{abbr}} \rightarrow$ What's fare code \$fare_basis_code
$R_{\text{abbr}} \rightarrow$ What does code \$fare_basis_code mean
$R_{\text{abbr}} \rightarrow$ What does code \$fare_basis_code stand for
$R_{\text{flight}} \rightarrow$ What does flight code \$airline_code mean
$R_{\text{airline}} \rightarrow$ What does the airline code \$airline_code
stand for
⇓ (slot-centric expansion)
$R'_{\text{abbr}} \rightarrow V_1$ \$fare_basis_code $V_2$
$R'_{\text{flight}} \rightarrow V_3$ \$airline_code $V_4$
$R'_{\text{airline}} \rightarrow V_5$ \$airline_code $V_6$
$V_1 \rightarrow$ Describe the fare code | What's fare code
| What does code
$V_2 \rightarrow$ [NUL] | mean | stand for
...
⇓ (edit-distance expansion)
$R'_{\text{abbr}} \rightarrow V_1$ \$fare_basis_code $V_2$
$R'_{\text{flight}} \rightarrow V_3$ \$airline_code $V_2$
$R'_{\text{airline}} \rightarrow V_5$ \$airline_code $V_2$
$V_1 \rightarrow$ ( Describe | What's ) the? fare code
| What does code
$V_2 \rightarrow$ [NUL] | mean | stand for
...

Figure 2: An example of combined expansion by slot-centric and edit-distance. Here [NUL] denotes an empty word.

and improve the overall performance. Thus our teacher model for ST alternates between rules and a model as depicted in Figure 1. (A detailed pseudo-algorithm is described in Appendix A.)

## 3 Experiments and Analysis

We run experiments on two scenarios: 1) when the data distribution among classes is unknown and 2) when it is known in prior.

We first evaluate how well rules of grammar perform in low-resource settings for **RQ1**. We then verify the effectiveness of various grammar expansions for **RQ2**, and examine how the data distribution affects rules of grammar for **RQ3**.

### 3.1 Experiments

| Dataset | #Sentences | #Intents | #Slots |
|---|---|---|---|
| ATIS | 5871 | 26/6[†] | 44 |
| Snips | 14484 | 7 | 39 |
| Facebook | 31378 | 12 | 17 |

([†]For $n$-shot evaluations, there are only 6 intents with at least 30 sentences each)

Table 1: Benchmark datasets and their numbers of sentences, intents and slot types.

**Dataset:** We use three benchmark datasets: ATIS (Hemphill et al., 1990), Snips (Coucke et al., 2018), and Facebook (Schuster et al., 2019). Each sentence in the datasets has an intent and a sequence of slot labels. We split a dataset into train, validation, and evaluation datasets with the ratio of 64%, 16%, and 20%, respectively.

For the first scenario when the data distribution is unknown among classes, we build an initial corpus of 10, 20 and 30 sentences for each intent from both the train and the validation sets at random. For the second scenario when the distribution is known in prior, we take 1%, 5% and 10% of our the train and the validation datasets according to the data distribution from the original corpus. We then erase labels in the remaining train dataset, which becomes the unlabeled dataset for ST.

**Baseline models:** In our empirical tests, we have observed that recent text classification models for few-shot settings such as UST (Mukherjee and Awadallah, 2020) or Delta-training (Jo and Cinarel, 2019) show competitive performance for ID but extremely poor performance for SF due to the nature of the SF task. Thus, we compare our model with the following two baselines, which are designed for the ID/SF tasks explicitly.

1. E et al. (2019) propose a state-of-the-art model (SF-ID network) for ID and SF tasks. The model consists of two subnets (ID subnet and SF subnet) connected bi-directionally. The baseline model $\text{B}_{\text{ST}}$ is the SF-ID network with the ST mechanism in few-shot settings.

2. Luo et al. (2018)[1] suggest another rule-assisted state-of-the-art model ($\text{B}_{\text{RE}}$) to solve ID and SF tasks in few-shot settings. $\text{B}_{\text{RE}}$ uses a bi-directional LSTM model and several components injecting external knowledge via regular expressions. We compare $\text{B}_{\text{RE}}$ to show the effectiveness of our rule construction methods.

**Experiment Setting:** We incorporate our rules of grammar into $\text{B}_{\text{ST}}$ and evaluate the effect of our method in few-shot settings. We follow the same parameters for $\text{B}_{\text{ST}}$ as in E et al. (2019). The initial threshold $\Theta$ for differentiating the correct labels is 0.8 for pseudo-labeling and the value is adjusted along the procedure. We also apply the same metrics as in E et al. (2019): accuracy (= #(correct intents) / #(sentences)) for ID, slot-unit $F_1$ score for SF and sentence accuracy (SA = #(correct intents & slots) / #(sentences)) for the overall accuracy of model predictions.

[1] As the source code was not provided, we implement it according to the paper.

### 3.2 Results and Analysis

We run experiments for five times and compute the average in each test. For the overall comparison, we report the average performance of different $n$-shots and $k\%$-samplings with respect to three benchmark datasets all together for ID and SF, respectively.All our methods achieve their best performance within 10 iterations, showing high convergence rate.

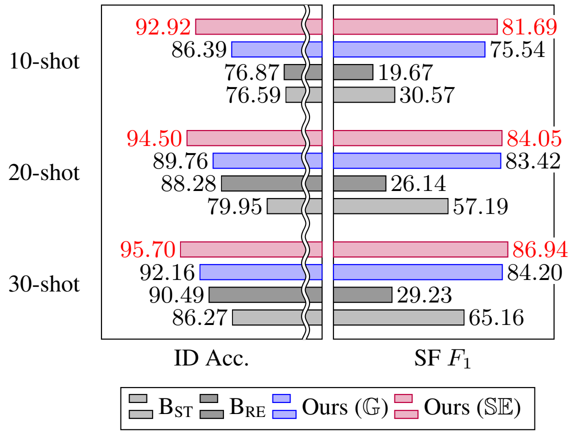

Figure 3: Average $n$-shot performance for the unknown data distribution. Red indicates the best score

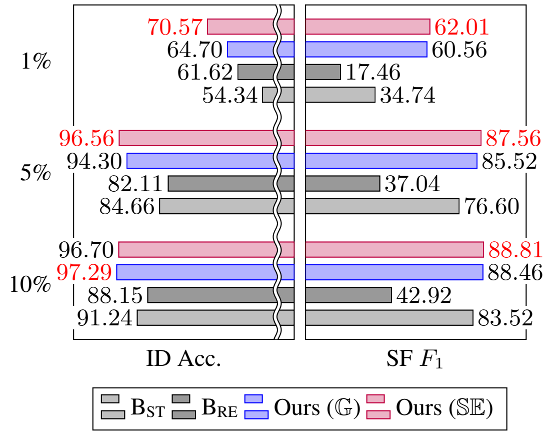

Figure 4: Average $k\%$-sampling performance for the known data distribution. Red indicates the best score

**RQ1:** The experimental results in Figure 3 and 4 answer **RQ1**; using rules of grammar ($\mathbb{G}$) is viable for ST on few labeled data. In all cases of ID, SF and SA, we observe a large improvement from the baselines, proving that $\mathbb{G}$ is an effective method. Namely, our rule-based ST method without any expansions ($\mathbb{G}$) outperforms $\text{B}_{\text{ST}}$ and $\text{B}_{\text{RE}}$. It is also noticeable that the experimental results for both scenarios are very close to the performance of SOTA performance on ID and SF tasks.

One concern is whether $\mathbb{G}$ still works when treating unseen data. As $\mathbb{G}$ highly depends on the initial corpus, it is important to analyze how $\mathbb{G}$ performs with the data that is not in the initial corpus. We observe that, on average, about 60% of test dataset contains at least one slot value that is not in the initial train or the validation datasets. $\mathbb{G}$ predicts these unseen slots with 57% accuracy on average. This implies that $\mathbb{G}$ is inadequate for data outside the initial corpus.

**RQ2:** While $\mathbb{G}$ is more effective than the baselines, the resulting grammar is a straight conversion to CFGs and cannot cope with data with slight variations from the current rules. We resolve this problem by expanding grammars according to the data similarity. We test various grammar expansions: edit-distance with 0.3 threshold ($\mathbb{E}$), $n$-gram ($\mathbb{N}$), slot-centric ($\mathbb{S}$), and their combinationsAmong various combinations, $\mathbb{SE}$ shows the best performance. Since $\mathbb{S}$ uses slots which are keywords in sentences, $\mathbb{S}$ is appropriate for extracting semantic similarity. With rules grouped by their semantics, $\mathbb{E}$ then use rules' structure information for better expansion. This is why $\mathbb{SE}$ shows a substantial improvement.

Within unseen data, $\mathbb{SE}$ achieves 75% accuracy on average which is 18%p higher than the accuracy of $\mathbb{G}$. Our expansions aim to overcome the weakness of $\mathbb{G}$ which heavily depends on the initial corpus. The result shows that the expansion, $\mathbb{SE}$, develops the rules better than the naive extraction, $\mathbb{G}$.

**RQ3:** In practice, the distribution among classes may not be available in prior. Thus, for the few-shot problem, it is more realistic to have a uniform-sampled data for each class—$n$-shot test. The experimental result in Figure 3 shows that our method ($\mathbb{SE}$) is effective in $n$-shot test, and the performance gain is larger when $n$ is smaller.

For the other case when the data distribution is known in prior, one can sample labeled data according to the ratio. Figure 4 shows that our method still outperforms, but unlike Figure 3, the improvement of $\mathbb{SE}$ seems relatively small. This is because of the total amount of the labeled data. For instance, the 5% case has more data than the 30-shot case. In such cases, our grammar extraction $\mathbb{G}$ already works well and the extra gain from $\mathbb{SE}$ is marginal when there are a reasonable amount of small data. On the other hand, for the extreme few-shot settings (e.g., 10, 20-shots or 1%), $\mathbb{SE}$ shows an impressive performance among all possible combinations.

## 4 Conclusions

Recently, rules of grammar have become popular again in deep learning due to their reliability and explainability. We have adapted rules of grammar in ST for NLU in few-shot settings. A major problem when using rules of grammar is the fact that it might not be flexible to cope with exceptions or variant data from the current rules. We have resolved this problem by expanding rules based on the data/grammar similarity. This gives rise to more precise pseudo-labels in ST and better performance. We have demonstrated that the combination of slot-centric and edit-distance shows the best performance. For both scenarios when knowing data distribution in prior or not, our algorithm achieves a substantial improvement. Remark that rules of grammar do not depend on specific neural network models, and, therefore, learning with rules of grammar is a complementary solution for the few-shot problem in NLU.

## Acknowledgments

This research was supported by the Hyundai Motor AIRS company, the NRF grant funded by the Korea government (MSIT) (NRF-2020R1A4A3079947), and the AI Graduate School Program (2020-0-01361). Han is a corresponding author.

## References

Waheed Ahmed Abro, Guilin Qi, Zafar Ali, Yansong Feng, and Muhammad Aamir. 2020. Multi-turn intent determination and slot filling with neural networks and regular expressions. *Knowledge-Based Systems*, 208:106428.

Cemil Cengiz and Deniz Yuret. 2020. Joint training with semantic role labeling for better generalization in natural language inference. In *Proceedings of the 5th Workshop on Representation Learning for NLP*, pages 78–88.

Eunah Cho, He Xie, and William M. Campbell. 2019. Paraphrase generation for semi-supervised learning in NLU. In *Proceedings of the Workshop on Methods for Optimizing and Evaluating Neural Language Generation*, pages 45–54.

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. *CoRR*, abs/1805.10190.

Haihong E, Peiqing Niu, Zhongfu Chen, and Meina Song. 2019. A novel bi-directional interrelated model for joint intent detection and slot filling. In *Proceedings of the 57th Conference of the Association for Computational Linguistics*, pages 5467–5471.

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*, pages 753–757.

Junxian He, Jiatao Gu, Jiajun Shen, and Marc'Aurelio Ranzato. 2020. Revisiting self-training for neural sequence generation. In *8th International Conference on Learning Representations*.

Michael A. Hedderich, Lukas Lange, Heike Adel, Jannik Strötgen, and Dietrich Klakow. 2020. A survey on recent approaches for natural language processing in low-resource scenarios. *CoRR*, abs/2010.12309.

Charles T. Hemphill, John J. Godfrey, and George R. Doddington. 1990. The ATIS spoken language systems pilot corpus. In *Speech and Natural Language*, pages 96–101.

Chengyue Jiang, Yinggong Zhao, Shanbo Chu, Libin Shen, and Kewei Tu. 2020. Cold-start and interpretability: Turning regular expressions into trainable recurrent neural networks. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing*, pages 3193–3207.

Hwiyeol Jo and Ceyda Cinarel. 2019. Delta-training: Simple semi-supervised text classification using pretrained word embeddings. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, pages 3456–3461.

Bing Liu and Ian Lane. 2016. Attention-based recurrent neural network models for joint intent detection and slot filling. In *Interspeech 2016, 17th Annual Conference of the International Speech Communication Association*, pages 685–689.

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.

Subhabrata Mukherjee and Ahmed Hassan Awadallah. 2020. Uncertainty-aware self-training for few-shot text classification. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020*.

Debjit Paul, Mittul Singh, Michael A. Hedderich, and Dietrich Klakow. 2019. Handling noisy labels for robustly learning from self-training data for low-resource sequence labeling. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 29–34.

Joel Ratsaby and Santosh S. Venkatesh. 1995. Learning from a mixture of labeled and unlabeled examples with parametric side information. In *Proceedings of the Eighth Annual Conference on Computational Learning Theory*, pages 412–417.

Georgios Rizos, Konstantin Hemker, and Björn Schuller. 2019. Augment to prevent: Short-text data augmentation in deep learning for hate-speech classification. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, pages 991–1000.

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.

Congrui Wang, Zhen Huang, and Minghao Hu. 2020a. 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*, pages 29–33.

Yaqing Wang, Quanming Yao, James T. Kwok, and Lionel M. Ni. 2020b. Generalizing from a few examples: A survey on few-shot learning. *ACM Computing Surveys*, 53(3):63:1–63:34.

Kaiyu Yang, Klint Qinami, Li Fei-Fei, Jia Deng, and Olga Russakovsky. 2020. Towards fairer datasets: Filtering and balancing the distribution of the people subtree in the imagenet hierarchy. In *Proceedings of the 2020 Conference on Fairness, Accountability, and Transparency*, pages 547–558.

Zhiquan Ye, Yuxia Geng, Jiaoyan Chen, Jingmin Chen, Xiaoxiao Xu, Suhang Zheng, Feng Wang, Jun Zhang, and Huajun Chen. 2020. Zero-shot text classification via reinforced self-training. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 3014–3024.