AUGNLG: Few-shot Natural Language Generation using Self-trained Data Augmentation

Xinnuo Xu[1], **Guoyin Wang**[2], **Young-Bum Kim**[2], **Sungjin Lee**[2]
[1]The Interaction Lab, Heriot-Watt University, Edinburgh
[2]Amazon Alexa AI, Seattle, WA, USA
xx6@hw.ac.uk, guoyiwan, youngbum, sungjinl@amazon.com

Abstract

Natural Language Generation (NLG) is a key component in a task-oriented dialogue system, which converts the structured meaning representation (MR) to the natural language. For large-scale conversational systems, where it is common to have over hundreds of intents and thousands of slots, neither template-based approaches nor model-based approaches are scalable. Recently, neural NLGs started leveraging transfer learning and showed promising results in few-shot settings. This paper proposes AUGNLG, a novel data augmentation approach that combines a self-trained neural retrieval model with a few-shot learned NLU model, to automatically create MR-to-Text data from open-domain texts. The proposed system mostly outperforms the state-of-the-art methods on the FEWSHOTWOZ data in both BLEU and Slot Error Rate. We further confirm improved results on the FEWSHOTSGD data and provide comprehensive analysis results on key components of our system. Our code and data are available at https://github.com/XinnuoXu/AugNLG.

1 Introduction

Large-scale conversational systems provide a natural interface to achieve various daily-life tasks. Natural Language Generation (NLG) is a key component in such a system to convert the structured meaning representation (MR) to the natural language, as shown in Figure 1. In task-oriented dialogue systems, NLG is typically accomplished by filling out a basic set of developer-provided templates, leading to a conversational system generating unnatural, robotic responses. In order to make the system sound more human-like, model-based NLG approaches, in particular neural models, have recently been gaining an increasing traction (Gao et al., 2018; Wen et al., 2015). However, neither the template-based approaches nor the model-based approaches are sufficiently scalable for large-scale conversational systems, where it is common to have over hundreds of intents and thousands of slots.

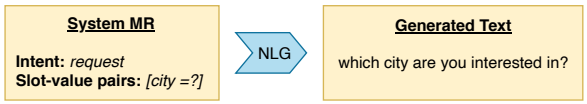

Figure 1: An example of NLG task. The model takes in the system MR, which consists of an intent with slot value pairs, and outputs text in natural language.

With the rise of neural transfer learning for NLP using pretrained LMs, recently, neural NLGs started to leverage transfer learning and showed some promising results (Radford et al., 2019; Brown et al., 2020; Dai et al., 2019; Edunov et al., 2019). In particular, Peng et al. (2020) proposed FEWSHOTWOZ, the first NLG benchmark test in few-shot learning settings, and achieved a SOTA performance by leveraging existing MR-to-Text data sets via task-specific continued pre-training. Despite the improved result, their approach leaves little room for further improvements as MR-to-Text data are expensive to obtain for new domains, practically circling back to the same scalability problem after exhausting the existing data.

In order to go beyond this restriction, this paper proposes AUGNLG, a novel data augmentation approach, that automatically creates MR-to-Text data from open-domain texts by combining a self-trained neural retrieval model with a few-shot learned NLU model. Since our data augmentation approach is orthogonal to the prior transfer learning approaches, one can use our approach in conjunction with other approaches. In experiments, we empirically show that AUGNLG mostly boosts the performance of both the fine-tuned GPT-2 (FT-GPT) (Radford et al., 2019) and SC-GPT (Peng et al., 2020), the continued pretraining approach with existing MR-to-Text data, on the FEWSHOT-

Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, pages 1183–1195
August 1–6, 2021.

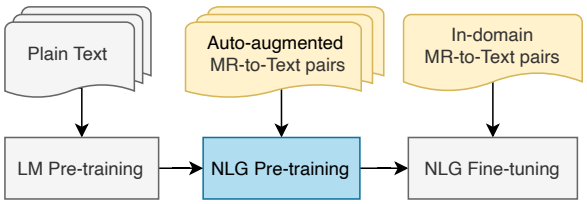

Figure 2: The training procedure for AUGNLG.

WOZ task. Furthermore, we construct another few-shot learning testbed, FEWSHOTSGD, out of the Schema-Guided Dialogue (SGD) corpus (Rastogi et al., 2020) and confirm improved results by applying AUGNLG to the FT-GPT. [1] Finally, we provide comprehensive analysis results on the key components of our system to gain detailed insights into the relationship between component-wise behavior and various parameters.

2 Related Work

NLG for Dialogue Response Generation There has been a body of work on neural NLG models, adopting various architectures, such as RNNs (Wen et al., 2015), attention RNNs (Dušek and Jurčíček, 2016), SC-LSTM (Wen et al., 2016), T2G2 (Kale and Rastogi, 2020), AdapterCL (Madotto et al., 2020) and associated variants (Tran and Le Nguyen, 2017; Tran et al., 2017). Despite the improved flexibility and naturalness over template-based methods, neural approaches require large amounts of annotated data to reach good performance.

Data Augmentation Data augmentation has been widely applied to a variety of NLP tasks, including sentence classification (Xie et al., 2020), natural language inference (Hu et al., 2019) and spoken language understanding (Li et al., 2019; Quan and Xiong, 2019; Zhao et al., 2019). Prior approaches for text data utilized back-translation (Sennrich et al., 2016; Edunov et al., 2018), c-BERT word replacement (Jiao et al., 2020), mixed labels and representations (Guo et al., 2019; Chen et al., 2020) and paraphrase data (Gao et al., 2020). However, the range of augmented data will be inherently limited, particularly in few-shot learning settings due to the nature of prior approaches, which only leverages in-domain data. In contrast, we take a rarely explored approach, tapping into a wealth of open-domain text that covers almost all topics. Recently, Du et al. (2021) proposed a self-training method to augment data for NLU tasks by retrieving sentences from data crawled on the web. However, their method cannot be directly applied to the NLG problem since it does not yield MR annotations. Our approach, in contrast, generates MR-to-Text data by jointly employing a self-trained neural retrieval model with a few-shot learned NLU model.

[1] Since SGD accounts for a large portion of the existing MR-to-Text data that SC-GPT utilized in training, we could not apply AUGNLG to SC-GPT for the FEWSHOTSGD task.

3 Few-shot Transfer Learning for NLG

The goal of NLG is to translate an MR $\mathcal{A}$ into its natural language response $x = [x^1, \ldots, x^T]$, where x^i is the ith token in the sequence x and T is the sequence length. $\mathcal{A}$ is defined as the combination of intent $\mathcal{I}$ and slot-value pairs $\{(s_i, v_i)\}_{i=1}^P$:

$$\mathcal{A} = \{\mathcal{I}, (s_1, v_1), \ldots, (s_P, v_P)\}, \quad (1)$$

where the intent stands for the illocutionary type of the system action while slot-value pairs indicate category names and their values to embed in the utterance. For example, in the MR, *inform (food = chinese ; price = cheap)*, *inform* is the intent, *food* and *price* are two slot keys and *chinese* and *cheap* are the corresponding slot values.

Given in-domain MR-to-Text data $\mathbb{D} = \{(\mathcal{A}_n, x_n)\}_{n=1}^N$ for training, where N is the number of examples, a statistical neural language model parameterized by θ is adopted to characterize the conditional probability $p_\theta(x|\mathcal{A})$. By adopting the chain rule on auto-regressive generation, the joint probability of x conditioned on $\mathcal{A}$ is decomposed as $\prod_{t=1}^T p_\theta(x^t|x^{<t}, \mathcal{A})$. The training process, i.e. the learning of θ, is then defined as maximizing the log-likelihood of the conditional probabilities over the entire training dataset:

$$\mathcal{L}_\theta(\mathbb{D}) = \sum_{n=1}^{|\mathbb{D}|} \log p_\theta(x_n|\mathcal{A}_n).$$

In the few-shot learning setup, the number of training examples N is extremely small (e.g. ≤ 50), which easily leads to non-fluent generated sentences with many grammar mistakes or missing pieces of information. In order to combat the data sparseness problem, inspired by prior transfer learning approaches, we introduce a three-step pipeline to gradually evolve a general large-scale language model to a domain-specific NLG model (shown in Figure 2): (1) pre-training a base language model with massive amounts of text, (2) NLG-specific continued pre-training with auto-augmented MR-to-Text data, and (3) final fine-tuning with the limited in-domain MR-to-Text ground-truth data.

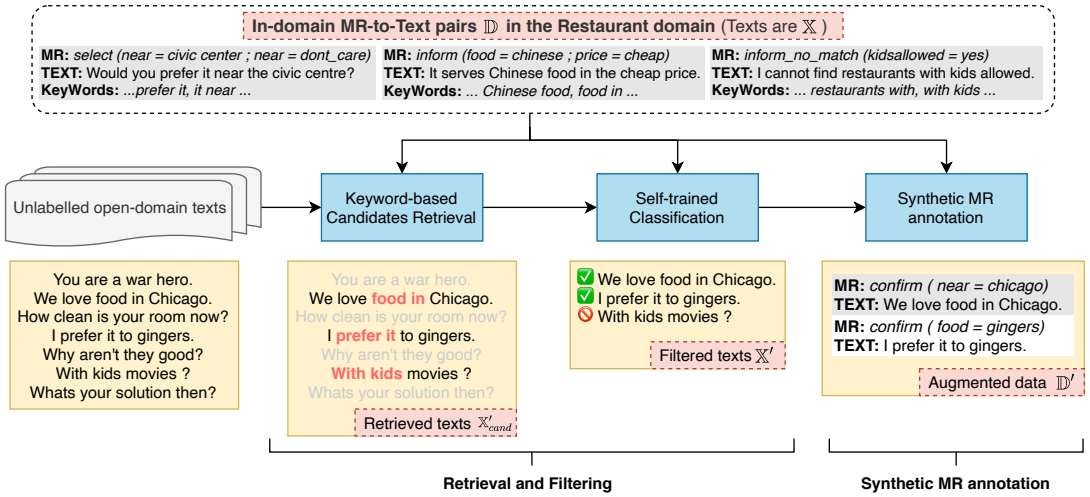

Figure 3: The overall pipeline for MR-to-Text data augmentation.

Specifically, in Step (1), we adopt GPT-2 (Radford et al., 2019) as our base language model since GPT-2 has demonstrated a remarkable performance on auto-regressive text generation tasks, which is close to MR-to-Text generation, in a variety of domains. However, GPT-2 is pre-trained on OpenWebText and the language style and topics thereof are quite different from those of daily conversations in a target domain. Furthermore, the generation task in NLG is conditioned on the input MR, as opposed to the unconditioned generation of the underlying GPT-2 pre-training task. Thus, to bring the model a step closer to the final NLG model in the target domain, in Step (2), we continuously pre-train the GPT-2 model on an automatically constructed set of augmented MR-to-Text pairs $\mathbb{D}' = \{(\mathcal{A}_m, x_m)\}_{m=1}^{M}$, where M is the number of augmented examples, which is much larger than the amount of in-domain ground-truth data. Data augmentation is achieved by retrieving a large amount of relevant text from Reddit (Henderson et al., 2019) with a self-trained neural retrieval model and then synthesizing MRs with a few-shot learned NLU model. The details of data augmentation is described in Section 4. Finally, in Step (3), we fine-tune the NLG model on a limited amount of in-domain ground-truth MR-to-Text pairs $\mathbb{D}$ for a final adaptation.

4 Data Augmentation

The data augmentation procedure aims to construct a large amount of MR-to-Text pairs $\mathbb{D}'$ from open-domain texts that are relevant to the in-domain ground-truth MR-to-Text pairs $\mathbb{D}$. The augmentation process consists of two stages: (1) retrieving keyword-matching utterances and filtering out domain-irrelevant instances, (2) generating synthetic MR annotations. Figure 3 illustrates the overall pipeline with some examples. For further analysis and studies, we release the data from all intermediate steps for each domain at https://github.com/XinnuoXu/AugNLG/tree/master/augmented_data.

4.1 Retrieval and Filtering

The utterance retrieval and filtering procedure consists of three steps: (1) keyword extraction that collects n-gram keywords from all in-domain utterances $\mathbb{X} = \{x_n\}_{n=1}^{N}$; (2) keyword-based retrieval that searches the open-domain texts for utterances that match any keywords extracted in the previous step, yielding a set of utterances $\mathbb{X}'_{cand}$; (3) self-trained neural classifier that filters out some retrieved utterances that are semantically irrelevant to the target domain. After the filtering, we form an augmented set of utterances $\mathbb{X}'$ with the unfiltered utterances.

Keywords Extraction. To efficiently extract keywords, we first gather all n-gram phrases that appear in $\mathbb{X}$. Since some phrases are too general to be effective, e.g. "*I cannot*", "*is your*", we use TF-IDF scores to measure the specificity of a phrase (see Appendix A for more detail). We first rank the collected n-grams according to their TF-IDF scores and filter out those n-gram phrases with relatively low TF-IDF score.

Keyword-based Retrieval. Having extracted the keywords, we retrieve utterances from the open-domain utterance pool that contains at least one

Algorithm 1 Self-trained Neural Filtering

Require: In-domain utterances $\mathbb{X}$ in the target domain; Retrieved utterances $\mathbb{X}'_{cand}$

$\mathcal{U}^+ \leftarrow$ Positive examples $\mathbb{X}$
$\mathcal{U}^- \leftarrow$ Randomly selected negative examples
$c^0 \leftarrow$ Train($\mathcal{U}^+, \mathcal{U}^-$)
$L \leftarrow$ Maximum number of iterations
$l = 1$; $\mathcal{E}_0^+ = \mathcal{U}^+$; $\mathcal{E}_0^- = \mathcal{U}^-$
while $l \leq L$ **do**
  $\mathcal{E}_l^+ \leftarrow \{x'$ if Predict($\mathbb{X}'_{cand}$, $c^{l-1}) \geq \sigma^+\}$
  $\mathcal{E}_l^- \leftarrow \{x'$ if Predict($\mathbb{X}'_{cand}$, $c^{l-1}) \leq \sigma^-\}$
  $\mathcal{E}_l^+ \leftarrow \mathcal{E}_l^+ + \mathcal{U}^+$
  if $\left|\mathcal{E}_l^+\right| - \left|\mathcal{E}_{l-1}^+\right| \leq \delta$ **then**
    Converged; Break
  end if
  $c^l \leftarrow$ Train($\mathcal{E}_l^+, \mathcal{E}_l^-$)
  $l \leftarrow l + 1$
end while
$\mathbb{X}' \leftarrow \{x'$ if Predict($\mathbb{X}'_{cand}$, $c^l) \geq \sigma\}$

extracted keyword in it. The aim of this step is to source a large amount of domain-relevant utterances $\mathbb{X}'_{cand}$ based on the surface-level overlap.

Self-trained Neural Filtering. Although the keyword-based retrieval is efficient, the retrieved utterances $\mathbb{X}'_{cand}$ can be quite noisy since an n-gram keyword only matches some part of the utterance, failing to detect the existence of irrelevant pieces in other parts. For example, in Figure 3, even though the utterance "*With kids movies?*" contains the keyword "*with kids*", it is irrelevant to the target domain *Restaurant* given the word *movies*. Thus, we introduce a self-trained neural classifier to filter out domain-irrelevant utterances from $\mathbb{X}'_{cand}$ by considering the semantic representation of an entire utterance and yield a domain-relevant set $\mathbb{X}'$.

The algorithm of the self-training and filtering process is listed in Algorithm 1. We adopt a BERT (Devlin et al., 2019) model with a binary classification layer atop as the base model and then train the classifier with in-domain utterances $\mathbb{X}$ and randomly selected open-domain utterances[2] , serving as positive and negative examples ($\mathcal{U}^+$ and $\mathcal{U}^-$), respectively. After that, the self-training and filtering cycle starts. At each iteration, we make predictions on the utterances in $\mathbb{X}'_{cand}$ with the classifier trained in the previous iteration. All utterances with a score over the threshold σ^+, together with the in-domain utterances $\mathbb{X}$, are then taken as a new set of positive examples $\mathcal{E}^+$, whereas all utterances with a score less than the threshold σ^- are collected as a new set of negative examples $\mathcal{E}^-$.[3] The self-training loop terminates if either the increment of positive examples at the last iteration is less than the threshold δ or the iterations is over the pre-defined maximum number of iterations. Otherwise, a new classifier is trained on $\mathcal{E}^+$ and $\mathcal{E}^-$ and the algorithm keeps going on the loop. Once the loop terminated, we label all utterances in $\mathbb{X}'_{cand}$ with the classifier from the last iteration. Finally, we build a domain-relevant set of augmented utterances $\mathbb{X}'$ by taking all utterances with a score over the threshold σ.[4]

4.2 Synthetic MR Annotation

Having built the domain-relevant set of augmented utterances $\mathbb{X}'$, we now proceed to synthesize MR labels to produce a complete MR-to-Text dataset $\mathbb{D}'$. To this end, we build a few-shot NLU model by fine-tuning a BERT model with in-domain ground-truth data. To put the data in the right format for the NLU task, we take MRs and utterances as labels and model inputs, respectively. Each token is annotated with the slot name if it is a part of the associated slot value and the final hidden state of the special token [CLS] is used to predict the intent (see Figure 5 in Appendix B). Finally, we generate an MR-to-Text dataset $\mathbb{D}'$ by concatenating the utterances in $\mathbb{X}'$ with the synthetic MR labels predicted by the few-shot NLU model.

5 Experimental Setup

5.1 Dataset

Fewshot NLG Data FEWSHOTWOZ is a few-shot NLG benchmark, built upon RNNLG and MultiWOZ (Budzianowski et al., 2018). In each domain, MR-to-Text pairs are grouped according to their delexicalized MRs (i.e. slot values being masked) and a training set is created by taking a pair each from 50 random groups and then the rest are taken as the test set. We also construct a new dataset FEWSHOTSGD by applying the same

[2]All utterances in $\mathbb{X}'_{cand}$ are excluded from the open-domain utterance pool. To balance the precision and recall, we control the size of the initial negative set such that $\left|\mathcal{U}^-\right| = \lambda_1 \cdot \left|\mathcal{U}^+\right|$, where $\lambda_1 = 10$.

[3]To guarantee the precision of the positive examples, we use $\sigma^+ = 0.99$ and $\sigma^- = 0.5$. Also, we sub-sample negative examples such that $\left|\mathcal{E}^-\right| = \lambda_2 \cdot \left|\mathcal{E}^+\right|$, where $\lambda_2 = 5$.

[4]To harvest a large amount of utterances, we set the threshold σ to 0.5.

Statistics	-WOZ	-SGD
# Domains	7	16
Avg. # Intents	8.14	6.44
Avg. # Slots	16.2	11.3
Avg. # Delex MRs in Training	50	33
Avg. # Delex MRs in Testing	473	31
Avg. # Training Instances	50	35
Avg. # Test Instances	473	5618
Avg. # Test Instances per MR	1.14	472.9
Avg. # Test Novelty uni-gram (%)	12.97	23.90
Avg. # Test Novelty bi-gram(%)	44.42	65.29
Avg. # Test Novelty tri-gram(%)	68.20	84.44
Avg. # Test Novelty four-gram(%)	82.70	92.75
Avg. # Keywords (K)	0.20	0.12
Avg. # Retrieved Utterances (K)	854.8	731.3
Avg. # Augmented Pairs (K)	34.0	25.6
Avg. # Delex. MRs in Aug. Pairs (K)	2.12	0.57

Table 1: Comparison of FEWSHOTWOZ and FEWSHOTSGD. The bottom section shows the statistics for augmented data. The unit for all statistics in the bottom section is thousand(K).

preparation steps to the SGD corpus. The comparison of FEWSHOTWOZ and FEWSHOTSGD is presented in the top section in Table 1. Comparing to FEWSHOTWOZ, FEWSHOTSGD has (1) more domains, (2) less intents, slots and delexicalized MRs[5] (3) more testing examples for each delexicalized MR, (4) more novel n-grams[6] in test utterances.

Augmented Data Since Reddit has shown to provide natural conversational English data, we adopt Reddit (Henderson et al., 2019) as the open-domain utterance pool after filtering for utterances of length between 2 and 40, totalling about 0.7B utterances. The average number of extracted keywords, retrieved utterances, final augmented MR-to-Text pairs and delexicalized MRs over all domains in FEWSHOTWOZ and FEWSHOTSGD are shown in the bottom section of Table 1. The detailed breakdowns of each domain are listed in Table 9 and Table 10 in Appendix C.

5.2 Evaluation Metrics

Following Wen et al. (2015) and Peng et al. (2020), we use BLEU score and Slot Error Rate (ERR) for automatic evaluation. BLEU score measures the surface-level similarity between generated responses and human-authored references. Whereas, ERR measures the semantic alignment in terms of slot-value insertion and omission. Specifically, ERR $= (p + q)/M$, where M is the total number of slots in the MR and p, q are the number of missing and redundant slots in the surface realisation. Since the SGD dataset does not provide enough information to compute ERR, we report ERR only on FEWSHOTWOZ.

5.3 Systems

We apply our data augmentation approach AUGNLG to two baseline systems,

- **FT-GPT** GPT-2 is directly fine-tuned on the in-domain ground-truth MR-to-Text data. We introduce **AUGNLG-FT**, which further pre-trains GPT-2 on the augmented MR-to-Text data and performs a final fine-tuning on the in-domain data.

- **SC-GPT** (Peng et al., 2020) further pre-trains GPT-2 on existing MR-to-Text data borrowed from other NLG corpora and fine-tunes on the in-domain data. We introduce **AUGNLG-SC**, which pre-trains GPT-2 on both existing MR-to-Text data and automatically augmented data, and finally fine-tunes on the in-domain data.

6 Results

FEWSHOTWOZ Table 2 reports the results on FEWSHOTWOZ. AUGNLG-FT substantially outperforms FT-GPT across all domains in both BLEU and ERR. Similarly, AUGNLG-SC performs better than SC-GPT and achieves the state-of-the-art performance in most domains. Remarkably, AUGNLG-FT achieves a competitive performance with SC-GPT in many domains without leveraging any existing MR-to-Text data. It even outperforms SC-GPT in "*TV*" and "*Attraction*" domain in both BLEU and ERR.

FEWSHOTSGD Table 3 shows the results in FEWSHOTSGD. Due to the higher novelty of the test examples and the smaller amount of training examples (see *Avg. # Test Novelty n-gram* and *# Training Instances* in Table 1), FT-GPT performs worse than on FEWSHOTWOZ. This indicates that the few-shot settings on FEWSHOTSGD are even more challenging. But AUGNLG-FT managed to outperform FT-GPT by a large margin via the continued pre-training on the augmented examples.

[5]Note that, the average number of delexicalized MRs in the training set is 33, which means the number of training examples in some domains are less than 50.

[6]The novelty is calculated by dividing the number of n-grams in the test set that does not appear in the training set by the number of n-grams in the test set.

[7]https://github.com/pengbaolin/SC-GPT.

Model	Restaurant		Laptop		Hotel		TV		Attraction		Train		Taxi	
	BLEU	ERR	BLEU	ERR	BLEU	ERR	BLEU	ERR	BLEU	ERR	BLEU	ERR	BLEU	ERR
FT-GPT	28.15	15.87	28.83	11.82	36.51	14.29	33.73	9.28	17.45	22.83	13.06	25.59	14.84	28.57
AUGNLG-FT	32.16	4.79	33.64	5.14	36.99	9.89	34.80	6.92	20.61	13.58	14.95	10.64	16.70	10.71
SC-GPT	30.48	6.89	33.51	5.38	**38.30**	8.24	33.82	7.32	22.24	16.62	**17.06**	8.82	**19.21**	4.76
AUGNLG-SC	**34.20**	**2.99**	**34.32**	**2.83**	34.96	**6.59**	**34.99**	**5.53**	**22.50**	**10.40**	16.35	**6.13**	17.81	**3.57**

Table 2: Evaluation results on FEWSHOTWOZ (BLEU↑, ERR↓). Note that, the SC-GPT model reported here was pre-trained and fine-tuned using the code and only the SGD data shared by the original authors [7].

Model	Restaurants	Hotels	Flights	Calendar	Banks	Weather	Buses	Events
FT-GPT	08.98	08.84	12.18	05.27	06.09	10.52	07.77	09.17
AUGNLG-FT	**17.83**	**17.23**	**17.58**	**10.45**	**08.94**	**13.75**	**14.26**	**18.68**
Model	**Homes**	**Media**	**Movies**	**Music**	**Rentalcars**	**Ridesharing**	**Services**	**Travel**
FT-GPT	03.75	03.17	10.05	05.79	06.79	13.87	09.79	02.08
AUGNLG-FT	**12.27**	**08.62**	**11.96**	**12.76**	**13.32**	**15.54**	**16.82**	**14.35**

Table 3: Evaluation results in BLEU on FEWSHOTSGD.

Qualitative Evaluation Table 4 compares some generated utterances by different models on FEWSHOTWOZ (examples in FEWSHOTSGD are shown in Table 16 in Appendix E). Both FT-GPT and SC-GPT are prone to omit important slots. Comparing to SC-GPT, FT-GPT tends to over-generate and introduces hallucinations. However, AUGNLG and AUGNLG-SC managed to generate fluent, natural text while precisely reflecting the the input MR. We further examined 70 randomly sampled utterances generated by AUGNLG-SC, whose BLEU scores are **lower** than those generated by SC-GPT, in the "*Hotel*", "*Train*" and "*Taxi*" domain to understand some potential factors causing the lower BLEU scores We found that the lower BLEU scores are mainly driven by BLEU penalizing semantically correct paraphrases due to the nature of BLEU only checking surface-level matches. Some examples of such penalization are provided in Table 15 in Appendix E. Only 7 out of the 70 manually checked examples generated by AUGNLG-SC are actually worse than SC-GPT.[8]

In sum, the results (1) verify the effectiveness of complementing existing transfer learning methods with our novel data augmentation approach; (2) reveal that automatically augmented MR-to-Text data alone can lead to a competitive performance, previously only achieved with existing MR-to-Text data. Since existing MR-to-Text data is not a scalable data source, our approach brings more practical values to real-world applications; (3) indicate that leveraging augmented MR-to-Text data on top of existing MR-to-Text data yields a new SOTA performance on the benchmark test.

7 In-depth Analysis

In this section, we provide comprehensive analysis results on the key components and parameters of our system to gain detailed insights: (1) intrinsic evaluation on augmented data, (2) influence of NLU quality, and (3) performance trends over varying amounts of augmented data.

7.1 Intrinsic Evaluation on Augmented Data

For intrinsic evaluation of augmented data, we first introduce four metrics:

• *MR coverage (MR Cov.)* evaluates the coverage of delexicalized MRs of the test set in the augmented set:

$$\text{MR Cov.} = \frac{\#\text{ delexicalized MRs} \in \mathbb{A}' \cap \mathbb{A}_{\text{test}}}{\#\text{ delexicalized MRs} \in \mathbb{A}_{\text{test}}},$$

where $\mathbb{A}'$ and $\mathbb{A}_{\text{test}}$ denote delexicalized MRs in the augmented set and the test set, respectively. Higher *MR Cov.* values indicate that more delexicalized MRs of the test set appear in the augmented set.

• *Slot coverage (SL Cov.)* evaluates the coverage of slot keys of the test set in the augmented set.

• *Language model perplexity (PPL)* is the perplexity of augmented utterances calculated by a GPT-2 language model fine-tuned on the test set. Lower *PPL* values indicate that the distribution of augmented utterances is close to that of the test utterances.

• *Average n-gram novelty (Nvt.)* N-gram novelty measures the fraction of the n-grams in the test set

[8] We also examined 70 randomly sampled utterances generated by AUGNLG-SC, whose BLEU scores are **equal/higher** than those generated by SC-GPT. Among these examples, 35 examples are actually better and 7 examples are worse than the SC-GPT generations.

Domain: *Restaurant*	
Input MR	*inform(name=marlowe; goodformeal=dinner; area=mission bay)*
Reference	marlowe serves dinner in the mission bay area.
FT-GPT	there is a restaurant marlowe in the mission bay area called dinner, it is good for dinner, a good area.
SC-GPT	marlowe is a good restaurant in mission bay that serves dinner.
AUGNLG-FT	marlowe is a good restaurant that serves good dinner in mission bay.
AUGNLG-SC	marlowe is good for dinner in mission bay.
Domain: *Laptop*	
Input MR	*inform(name=tecra proteus 23;type=laptop;battery=9 hour;memory=4 gb;isforbusiness=false)*
Reference	the tecra proteus 23 laptop has a 9 hour battery life with 4 gb memory but is not for business computing.
FT-GPT	tecra proteus 23 laptop with 9 hour battery and 4 gb memory. **% miss** {***isforbusiness=false***}
SC-GPT	tecra proteus 23 laptop with 4 gb memory and 9 hour battery. **% miss** {***isforbusiness=false***}
AUGNLG-FT	the tecra proteus 23 laptop has a 9 hour battery, 4 gb of memory, and is not for business computing.
AUGNLG-SC	tecra proteus 23 laptop - 9 hour battery - 4 gb memory - not used for business computing.
Domain: *TV*	
Input MR	*inform(name=pontus 45; type=tv;ecorating=a+; screensizerange=medium; hdmiport=2)*
Reference	the pontus 45 tv has a a+ eco rating and a medium screen size and 2 hdmi ports.
FT-GPT	the pontus 45 tv has 2 hdmi ports, a medium screen size , and 2 screensize. **% miss** {***ecorating=a+***}
SC-GPT	the pontus 45 tv has a medium screen size and 2 hdmi ports. **% miss** {***ecorating=a+***}
AUGNLG-FT	the pontus 45 tv has a+ eco rating, 2 hdmi ports, and a medium screen size.
AUGNLG-SC	the pontus 45 tv has a medium screen size of 2 hdmi ports and a+ eco rating.
Domain: *Attraction*	
Input MR	*inform(area=north;price=no entrance fee; type=park; phone=01223)*
Reference	yes, it's a park located in the north area of town and has no entrance fee. the phone number is 01223.
FT-GPT	yes, there is a free entrance fee of £1. phone 01223 for more information. **% miss** {***area=north***}
SC-GPT	no problem. 01223 is the phone number and the park is north. the entrance fee is free.
AUGNLG-FT	yes, the entrance fee is no entrance fee. the park is in the north. phone is 01223.
AUGNLG-SC	yes, the park is in the north. no entrance fee. phone number 01223.

Table 4: Example utterances generated by different models on FEWSHOTWOZ (Better viewed in color). Errors are shown in three colors. The red text starting with "%" denotes omission. The blue text indicates hallucination. The green text means non-fluent generation.

Metrics	**Re**	**La**	**Ho**	**TV**	**At**	**Tr**	**Ta**
MR Cov. ↑	.70	.21	.71	.40	.66	.44	.59
SL Cov. ↑	1.0	.95	1.0	.94	.89	.92	.86

Table 5: Augmented data evaluation of *MR Cov.* and *SL Cov.* on FEWSHOTWOZ. The domain names are represented by the first two letters.

Metrics	**Re**	**Ho**	**Fl**	**Ca**	**Ba**	**We**	**Bu**	**Ev**
MR Cov ↑	.80	.72	.66	.65	.43	.70	.58	.80
SL Cov ↑	.92	.85	.89	.75	.57	.86	.88	.93
Metrics	**Ho**	**Me**	**Mo**	**Mu**	**Re**	**Ri**	**Se**	**Tr**
MR Cov ↑	.59	.58	.74	.67	.81	.77	.88	.55
SL Cov ↑	.75	.67	.80	.75	.80	.78	.93	.71

Table 6: Augmented data evaluation of *MR Cov.* and *SL Cov.* on FEWSHOTSGD. The domain names are represented by the first two letters.

that do not appear in the augmented set:

$$\text{N-gram novelty} = 1 - \frac{\#\text{ n-grams} \in \mathbb{X}' \cap \mathbb{X}_{\text{test}}}{\#\text{ n-grams} \in \mathbb{X}_{\text{test}}},$$

where $\mathbb{X}'$ and $\mathbb{X}_{\text{test}}$ denote utterances in the augmented set and test set, respectively. Lower *Nvt.* values indicate that more n-grams of the test set appear in the augmented set. We consider from 1-grams to 4-grams and report the average value.

The results of *MR Cov.* / *SL Cov.* on FEWSHOTWOZ and FEWSHOTSGD are shown in Table 5 and Table 6, respectively. *SL Cov.* achieves 70% in most domains on both datasets while *MR Cov.* has a wide range of values across domains. Noteworthily, Table 6 strongly correlates with Table 3 – "Banks" and "Media" domains are worse than other domains in both coverage metrics and NLG performance. On the other hand, "Restaurants" and "Events" domains are better than the others in both aspects. Although we do not see the same pattern on FEWSHOTWOZ, it could be attributed to the large variance in the number of delexicalized MRs in each domain (see Table 2 in (Peng et al., 2020)).

The results of *PPL* and *Nvt.* on FEWSHOTWOZ are shown in Table 7. We compare the augmented data (*AUG*) with the existing MR-to-Text data (*EXIST*). The top section shows that *AUG* achieves lower *PPL* values in all seven domains compared to *EXIST*. The bottom section again demonstrates that *AUG* achieves lower *Nvt.* values in most domains. However, in the "*Train*" and "*Taxi*" domains *EXIST* attains lower novelty values, which matches the results in Table 2, SC-GPT outperforming AUGNLG-SC in these two domains.[9]

[9]Detailed breakdowns of novelty scores from 1-grams to 4-grams are provided in Table 11 in Appendix C. The *Nvt.* re-

Metrics	Data	Restaurant	Laptop	Hotel	TV	Attraction	Train	Taxi
PPL ↓	EXIST	04.14	22.92	04.09	19.53	08.28	09.04	06.74
	AUG	**03.48**	**08.46**	**02.89**	**05.77**	**04.73**	**06.77**	**06.72**
Nvt. (%) ↓	EXIST	57.36	71.11	55.21	72.34	55.37	**53.45**	**46.94**
	AUG	**54.50**	**50.73**	**48.39**	**44.93**	**39.83**	56.24	55.38

Table 7: Language Model perplexity (PPL) and average n-gram novelty (Nvt.) on augmented data.

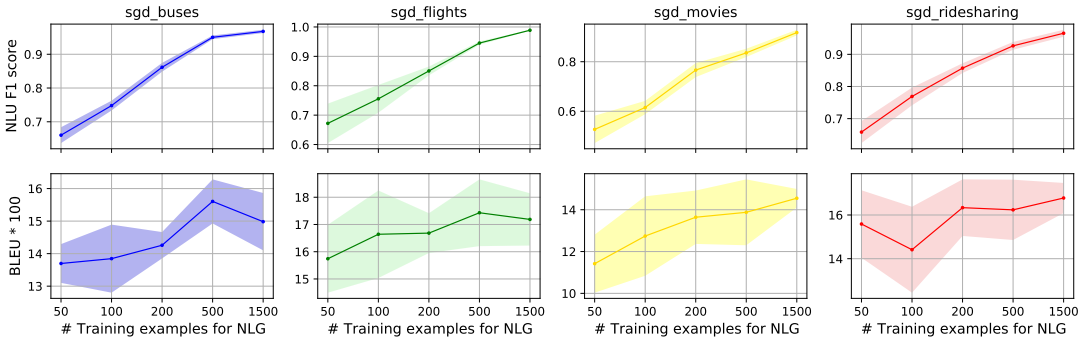

Figure 4: The influence of NLU on four domains in FEWSHOTSGD. The top row shows NLU F1 scores with *50, 100, 200, 500, 1500* training examples. The bottom row shows the BLEU scores of AUGNLG-FT pre-trained using these NLU models. All experiments are repeated for 5 times with different samples.

7.2 Influence of NLU

Few-shot NLU performance Since few-shot NLU models are a key component of our system, we report their performance in F1 score. For each domain, we evaluate the few-shot NLU model on the Text-to-MR test set, prepared in Section 4.2. The average F1 over all domains on FEWSHOTWOZ and FEWSHOTSGD are 0.77 and 0.68, respectively. A further breakdown over the domains are provided in Table 13 and Table 14 in Appendix D.

Influence of NLU Quality The mediocre NLU performance on FEWSHOTSGD leads to the following research question: *can better NLU models boost NLG performance?* To answer this question, we select four domains from FEWSHOTSGD with relatively low NLU performance: "*Buses* (0.63)", "*Flights* (0.74)", "*Movies* (0.44)", and *Ridesharing* (0.63). In each domain, we construct a new test set by randomly sampling 500 MR-to-Text pairs from the original test set, and take the rest as the NLU training pool. To obtain NLU models of varying quality, we train a set of models while varying the amount of training data with stratified sampling. The top row in Figure 4 clearly shows that F1 score increases in proportion to the training size, reaching 0.95 in F1 in all four domains. We then annotate the augmented utterances with different

sults on FEWSHOTSGD are shown in Table 12 in Appendix C, demonstrating similar trends.

Do	Mod	50	100	200	500	1500
Bu	**FT**	7.87	10.38	15.21	21.83	24.91
	AUG	14.37	15.36	17.06	22.18	24.98
Fl	**FT**	10.40	12.93	19.91	25.97	29.18
	AUG	14.07	15.50	21.55	25.38	26.62
Mo	**FT**	13.30	16.13	21.99	29.76	34.04
	AUG	17.13	17.55	23.68	29.14	33.55
Ri	**FT**	12.32	16.99	23.25	27.99	29.02
	AUG	17.18	22.06	24.76	26.87	28.60

Table 8: BLEU scores for FT-GPT (FT) and AUGNLG-FT (AUG) with different training sizes (50, 100, 200, 500, 1500). "Bu", "Fl", "Mo" and "Ri" are short for the domain names "Buses", "Flights", "Movies", "Ridesharing". All experiments are repeated for 5 times with different samples.

NLU models and pre-train the NLG models with the augmented MR-to-Text data updated with new MR labels. Finally, we fine-tune the NLG models on the in-domain training set $\mathbb{D}$ and perform evaluation on the newly constructed 500 test set. The bottom row in Figure 4 confirms that there is a general proportional relationship between the performances of NLU and NLG.

7.3 Varying Amounts of Augmentation

Lastly, we investigate the relationship between the amount of in-domain ground-truth data and the effect of augmentation. As in the previous section, we build new test sets by randomly taking 500 examples and vary the size of training set to train both NLU and NLG models. Table 8 shows that, in all four domains, the performance difference

between AUGNLG-FT and FT-GPT culminates at the smallest training set and gradually diminishes as more training data become available.

8 Conclusion

In this paper, we proposed AUGNLG, a novel data augmentation approach that combines a self-trained retrieval model with a few-shot learned NLU, to automatically create MR-to-Text data from open-domain texts. Experimental results verify the effectiveness of our approach by establishing new SOTA performances on two benchmark tests. More importantly, we showed how our approach complements the previous SOTA approach, which hinges on unscalable data sources, with unlimited open-domain data. Future work includes (1) technical innovations on each component of our system for further performance improvements, (2) exploring self-training on the NLU side too to evolve both the NLU and NLG model at the same time.

Acknowledgments

We would like to thank the first author of Peng et al. (2020), Baolin Peng, for his generous help. We also thank the anonymous reviewers for their helpful comments.

References

Tom B Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, et al. 2020. Language models are few-shot learners. *arXiv preprint arXiv:2005.14165*.

Paweł Budzianowski, Tsung-Hsien Wen, Bo-Hsiang Tseng, Inigo Casanueva, Stefan Ultes, Osman Ramadan, and Milica Gašić. 2018. Multiwoz-a large-scale multi-domain wizard-of-oz dataset for task-oriented dialogue modelling. *arXiv preprint arXiv:1810.00278*.

Jiaao Chen, Zichao Yang, and Diyi Yang. 2020. Mixtext: Linguistically-informed interpolation of hidden space for semi-supervised text classification. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 2147–2157.

Zihang Dai, Zhilin Yang, Yiming Yang, Jaime G Carbonell, Quoc Le, and Ruslan Salakhutdinov. 2019. Transformer-xl: Attentive language models beyond a fixed-length context. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 2978–2988.

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.

Jingfei Du, Edouard Grave, Beliz Gunel, Vishrav Chaudhary, Onur Celebi, Michael Auli, Veselin Stoyanov, and Alexis Conneau. 2021. Self-training improves pre-training for natural language understanding. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 5408–5418, Online. Association for Computational Linguistics.

Ondřej Dušek and Filip Jurčíček. 2016. Sequence-to-sequence generation for spoken dialogue via deep syntax trees and strings. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 45–51, Berlin, Germany. Association for Computational Linguistics.

Sergey Edunov, Alexei Baevski, and Michael Auli. 2019. Pre-trained language model representations for language generation. 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 4052–4059.

Sergey Edunov, Myle Ott, Michael Auli, and David Grangier. 2018. Understanding back-translation at scale. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*, pages 489–500.

Jianfeng Gao, Michel Galley, and Lihong Li. 2018. Neural approaches to conversational AI. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics: Tutorial Abstracts*, pages 2–7, Melbourne, Australia. Association for Computational Linguistics.

Silin Gao, Yichi Zhang, Zhijian Ou, and Zhou Yu. 2020. Paraphrase augmented task-oriented dialog generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 639–649.

Hongyu Guo, Yongyi Mao, and Richong Zhang. 2019. Augmenting data with mixup for sentence classification: An empirical study. *arXiv preprint arXiv:1905.08941*.

Matthew Henderson, Paweł Budzianowski, Iñigo Casanueva, Sam Coope, Daniela Gerz, Girish Kumar, Nikola Mrkšić, Georgios Spithourakis, Pei-Hao Su, Ivan Vulic, and Tsung-Hsien Wen. 2019.

A repository of conversational datasets. In *Proceedings of the Workshop on NLP for Conversational AI*. Data available at github.com/PolyAI-LDN/conversational-datasets.

Zhiting Hu, Bowen Tan, Russ R Salakhutdinov, Tom M Mitchell, and Eric P Xing. 2019. Learning data manipulation for augmentation and weighting. In *Advances in Neural Information Processing Systems*, volume 32, pages 15764–15775. Curran Associates, Inc.

Xiaoqi Jiao, Yichun Yin, Lifeng Shang, Xin Jiang, Xiao Chen, Linlin Li, Fang Wang, and Qun Liu. 2020. TinyBERT: Distilling BERT for natural language understanding. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 4163–4174, Online. Association for Computational Linguistics.

Mihir Kale and Abhinav Rastogi. 2020. Template guided text generation for task-oriented dialogue. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6505–6520, Online. Association for Computational Linguistics.

Juntao Li, Lisong Qiu, Bo Tang, Dongmin Chen, Dongyan Zhao, and Rui Yan. 2019. Insufficient data can also rock! learning to converse using smaller data with augmentation. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 33, pages 6698–6705.

Andrea Madotto, Zhaojiang Lin, Zhenpeng Zhou, Seungwhan Moon, Paul Crook, Bing Liu, Zhou Yu, Eunjoon Cho, and Zhiguang Wang. 2020. Continual learning in task-oriented dialogue systems. *arXiv preprint arXiv:2012.15504*.

Baolin Peng, Chenguang Zhu, Chunyuan Li, Xiujun Li, Jinchao Li, Michael Zeng, and Jianfeng Gao. 2020. Few-shot natural language generation for task-oriented dialog. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 172–182, Online. Association for Computational Linguistics.

Jun Quan and Deyi Xiong. 2019. Effective data augmentation approaches to end-to-end task-oriented dialogue. In *2019 International Conference on Asian Language Processing (IALP)*, pages 47–52. IEEE.

Alec Radford, Jeffrey Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. 2019. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9.

Abhinav Rastogi, Xiaoxue Zang, Srinivas Sunkara, Raghav Gupta, and Pranav Khaitan. 2020. Schema-guided dialogue state tracking task at dstc8. *arXiv preprint arXiv:2002.01359*.

Rico Sennrich, Barry Haddow, and Alexandra Birch. 2016. Improving neural machine translation models with monolingual data. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 86–96.

Van-Khanh Tran and Minh Le Nguyen. 2017. Natural language generation for spoken dialogue system using rnn encoder-decoder networks. In *Proceedings of the 21st Conference on Computational Natural Language Learning (CoNLL 2017)*, pages 442–451.

Van-Khanh Tran, Minh Le Nguyen, and Satoshi Tojo. 2017. Neural-based natural language generation in dialogue using rnn encoder-decoder with semantic aggregation. In *Proceedings of the 18th Annual SIGdial Meeting on Discourse and Dialogue*, pages 231–240.

Tsung-Hsien Wen, Milica Gašić, Nikola Mrkšić, Lina M. Rojas-Barahona, Pei-Hao Su, David Vandyke, and Steve Young. 2016. Multi-domain neural network language generation for spoken dialogue systems. In *Proceedings of the 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 120–129, San Diego, California. Association for Computational Linguistics.

Tsung-Hsien Wen, Milica Gašić, Nikola Mrkšić, Pei-Hao Su, David Vandyke, and Steve Young. 2015. Semantically conditioned LSTM-based natural language generation for spoken dialogue systems. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing*, pages 1711–1721, Lisbon, Portugal. Association for Computational Linguistics.

Qizhe Xie, Zihang Dai, Eduard Hovy, Thang Luong, and Quoc Le. 2020. Unsupervised data augmentation for consistency training. *Advances in Neural Information Processing Systems*, 33.

Zijian Zhao, Su Zhu, and Kai Yu. 2019. 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)*, pages 3628–3634.

A The calculation of TF-IDF

To calculate the TF-IDF score for a n-gram phrase, we take all in-domain texts $\mathbb{X}$ as one document d to calculate its TF (Term Frequency) score, and randomly selected open-domain texts as the set of documents D to calculate the IDF (Inverse Document Frequency) score[10]. Thus, we formulate the TF-IDF score for n-gram phrase ph_i as:

$$\text{TF-IDF}\left(ph_i, d, D\right) = \text{tf}\left(ph_i, d\right) \cdot \text{idf}\left(ph_i, D\right),$$

where,

$$\text{tf}\left(ph_i, d\right) = \log\left(1 + \text{freq}\left(ph_i, d\right)\right)$$
$$\text{idf}\left(ph_i, D\right) = \log\left(\frac{|D|}{|\{ph_i \in d\}|}\right),$$

in which, $\text{freq}\left(ph_i, d\right)$ denotes the raw count of the phrase ph_i appears in the document d.

B The structure of the BERT-based NLU annotation

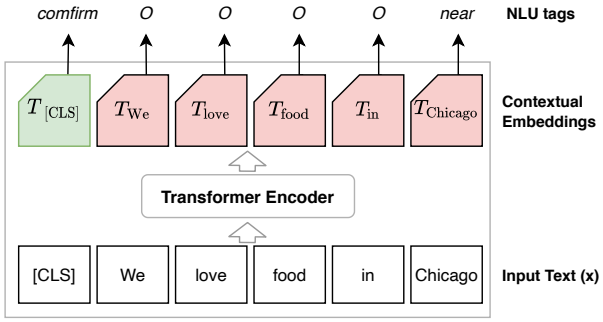

Figure 5: The structure of the BERT-based NLU annotation. The MR for the text "We love food in Chicago" is "*confirm (near = Chicago)*". Each slot-value token is annotated with the slot-name. The rest tokens are annotated with "*O*".

C Statistics for the Augmented Data

Domains	Restaurant	Laptop	Hotel	TV	Attraction	Train	Taxi
# InD Pairs	51	51	51	51	50	50	40
# Keywords (K)	0.23	0.06	0.22	0.28	0.27	0.18	0.17
# Rtv Texts (K)	885.46	1000.13	760.61	850.00	1262.69	650.53	573.93
# Aug Pairs (K)	30.97	36.62	40.46	49.76	65.48	9.60	4.95
# Delex MRs (K)	0.78	5.84	0.91	6.39	0.33	0.54	0.05

Table 9: FEWSHOTWOZ statistics of the augmented pairs over 7 different domains. InD is short for in-domain.

Domains	Restaurants	Hotels	Flights	Calendar	Banks	Weather	Buses	Events
# InD Pairs	50	50	50	25	23	11	50	50
# Keywords (K)	0.23	0.15	0.25	0.09	0.05	0.04	0.17	0.17
# Rtv Texts (K)	1021.64	1068.43	1195.41	582.22	112.78	387.90	749.46	1305.41
# Aug Pairs (K)	61.51	20.64	39.59	56.87	1.27	6.39	11.15	56.55
# Delex MRs (K)	1.15	0.77	1.64	0.19	0.03	0.04	1.31	1.05
Model	**Homes**	**Media**	**Movies**	**Music**	**Rentalcars**	**Ridesharing**	**Services**	**Travel**
# InD Pairs	21	14	30	21	50	48	50	14
# Keywords (K)	0.07	0.04	0.08	0.08	0.12	0.18	0.22	0.04
# Rtv Texts (K)	403.91	335.42	538.68	1033.63	469.95	1180.02	953.51	362.45
# Aug Pairs (K)	8.04	3.90	5.90	29.69	6.41	27.02	60.09	14.80
# Delex MRs (K)	0.15	0.05	0.06	0.13	0.20	0.23	2.00	0.05

Table 10: FEWSHOTSGD statistics of the augmented pairs over 16 domains. InD is short for in-domain.

[10]Here, each open-domain text represents a document.

Nvt. (%) ↓	Data	Restaurant	Laptop	Hotel	TV	Attraction	Train	Taxi
Nvt. uni	EXIST	12.46	28.93	11.36	27.55	08.84	12.19	09.22
Nvt. bi		48.70	69.82	46.68	72.53	46.66	46.13	**35.40**
Nvt. tri		77.21	88.68	74.33	91.33	75.48	**70.75**	**62.69**
Nvt. four		**91.07**	97.02	**88.46**	97.94	90.49	**84.74**	**80.46**
Nvt. uni	AUG	**11.20**	**06.33**	**06.13**	**04.51**	**04.09**	**09.80**	**07.56**
Nvt. bi		**39.45**	**37.86**	**31.37**	**25.72**	**21.60**	**38.68**	39.00
Nvt. tri		**73.63**	**69.10**	**66.98**	**61.10**	**53.72**	80.21	79.10
Nvt. four		93.73	**89.63**	89.10	**88.39**	**79.92**	96.28	95.87

Table 11: N-gram novelty (↓) breakdowns in FEWSHOTWOZ.

Nvt. (%) ↓	Data	Restaurants	Hotels	Flights	Calendar	Banks	Weather	Buses	Events
Nvt. uni	InD pairs	18.41	14.16	14.23	25.41	16.06	25.60	18.11	21.51
Nvt. bi		58.08	53.84	59.32	69.45	49.91	72.73	66.16	62.22
Nvt. tri		79.82	74.84	84.08	86.67	72.36	88.47	87.07	85.09
Nvt. four		91.62	**85.00**	93.75	94.49	**84.25**	96.43	94.14	93.10
Nvt. uni	AUG	**02.97**	**03.64**	**01.78**	**02.30**	**08.45**	**04.22**	**02.17**	**02.38**
Nvt. bi		**18.94**	**21.63**	**19.54**	**17.22**	**37.91**	**34.61**	**28.09**	**19.88**
Nvt. tri		**51.72**	**62.72**	**61.20**	**49.51**	**68.73**	**75.77**	**72.05**	**53.82**
Nvt. four		**81.62**	88.40	**89.62**	**79.57**	87.10	**93.30**	**93.15**	**82.53**

Nvt. (%) ↓	Data	Homes	Media	Movies	Music	Rentalcars	Ridesharing	Services	Travel
Nvt. uni	InD pairs	30.73	30.85	32.06	35.91	19.81	15.71	21.59	42.11
Nvt. bi		77.08	73.92	73.53	77.84	57.08	51.60	60.56	83.76
Nvt. tri		90.78	**88.90**	87.20	91.35	76.91	78.27	81.44	93.68
Nvt. four		95.22	**93.68**	**92.77**	96.55	**87.09**	**89.74**	92.37	97.59
Nvt. uni	AUG	**08.22**	**10.57**	**08.86**	**03.59**	**01.92**	**03.68**	**03.76**	**06.47**
Nvt. bi		**37.85**	**60.10**	**47.53**	**32.38**	**26.68**	**25.48**	**24.82**	**38.29**
Nvt. tri		**75.64**	93.48	**81.17**	**73.03**	**68.75**	**68.60**	**55.26**	**73.45**
Nvt. four		**92.74**	98.75	93.12	**93.37**	91.65	92.50	**80.79**	**90.90**

Table 12: *N*vt. (↓) breakdowns in FEWSHOTSGD. *E*XIST are from the SGD, we compare with in-domain pairs.

D Few-shot NLU Performance

Metrics	Restaurant	Laptop	Hotel	TV	Attraction	Train	Taxi
Precision	1.000	.8229	.7500	.7904	.6050	.6552	.6178
Recall	1.000	.8490	.7500	.8382	.6904	.6706	.7239
F1 score	1.000	.8357	.7500	.8136	.6449	.6628	.6667

Table 13: NLU evaluation for FEWSHOTWOZ (Precision↑), (Recall↑), (F1 score↑)

Metrics	Restaurants	Hotels	Flights	Calendar	Banks	Weather	Buses	Events
Precision	.6346	.6516	.7229	.8332	.8971	.7177	.6289	.5333
Recall	.6635	.6866	.7560	.8897	.8684	.7183	.6372	.5870
F1 score	.6487	.6686	.7391	.8605	.8825	.7180	.6330	.5589

Metrics	Homes	Media	Movies	Music	Rentalcars	Ridesharing	Services	Travel
Precision	.8201	.6404	.4787	.8011	.5183	.6145	.5470	.8132
Recall	.8166	.5060	.3996	.7889	.5288	.6546	.6146	.8713
F1 score	.8183	.5653	.4356	.7950	.5235	.6340	.5788	.8412

Table 14: NLU evaluation for FEWSHOTSGD (Precision↑), (Recall↑), (F1 score↑).

E Generation Examples

	Domain: *Hotel*
Input MR	*inform(name=parker guest house; area=mastro; address=520 church street; phone=4156213222)*
Reference	parker guest house is near mastro at 520 church street. their phone number is 4156213222.
SC-GPT	the parker guest house is located in mastro and is at 520 church street. the phone number is 4156213222.
AUGNLG-SC	the parker guest house in mastro's address is 520 church street. the phone number is 4156213222.
	Domain: *Train*
Input MR	*inform(day=sunday; id=tr5413; arriveby=00:04) @ book (none)*
Reference	train tr5413 arrives at 24:44 or 00:04 on sunday. would you like to book it now ?
SC-GPT	tr5413 arrives at 00:04. would you like me to book it for you ?
AUGNLG-SC	tr5413 arrives at 00:04 on sunday. would you like to book ?
	Domain: *Taxi*
Input MR	*inform(arriveby=16:00)*
Reference	yes , the taxi is due to arrive at 16:00.
SC-GPT	yes , the taxi will arrive by 16:00.
AUGNLG-SC	i 'm on it , the taxi will arrive by 16:00.

Table 15: Generation examples in "*Hotel*", "*Train*" and "*Taxi*" domain on FEWSHOTWOZ. In the same example, the BLEU scores for the SC-GPT generations are higher than the AUGNLG-SC generations.

Domain: *Banks*	
Input MR	*OFFER(balance=2,890.54;account_type=checking)*
Reference	your checking balance is 2,890.54 .
FT-GPT	now you have 2,890.54 to spend on goods and services .
AUGNLG-FT	balance of checking account is 2,890.54
Domain: *Buses*	
Input MR	*OFFER(departure_time=10:50 am;price=42;fare_type=economy)*
Reference	there is a bus leaving at 10:50 am . its economy ticket costs 42
FT-GPT	please confirm that you want to leave from 10:50 am to 10:50 pm .
AUGNLG-FT	sure , when do you want the bus to leave at ? 10:50 am , economy ticket or 42 ?
Domain:*Calendar*	
Input MR	*OFFER(available_start_time=8 am;available_end_time=7:30 pm) @ INF_COUNT (count)*
Reference	yes , you have 2 empty slots from 8 am to 7:30 pm .
FT-GPT	you have 1 available seat .
AUGNLG-FT	is your event scheduled to start at 8 am and end at 7:30 pm ?
Domain: *Events*	
Input MR	*CONFIRM(number_of_tickets=1;city=new york)*
Reference	no problem , i 'll make it 1 ticket for new york .
FT-GPT	please confirm your number of tickets .
AUGNLG-FT	1 ticket to the event in new york .
Domain: *Flights*	
Input MR	*CONFIRM(airlines=southwest airlines;return_date=march 12th)*
Reference	so you want a southwest airlines flight that returns on march 12th ?
FT-GPT	it arrives on march 12th at southwest airlines .
AUGNLG-FT	it's not an upgrade , it's a southwest airlines flight that leaves on march 12th .
Domain: *Homes*	
Input MR	*INFORM(phone_number=650-961-2975;furnished=false)*
Reference	the phone number is 650-961-2975 . and the apartment is not furnished .
FT-GPT	you are willing to call 650-961-2975 to schedule an appointment .
AUGNLG-FT	650-971-2975 is the phone number for the furnished apartment .
Domain: *Movies*	
Input MR	*OFFER(movie_name=hackers, high life, sunset)*
Reference	i found hackers , high life , and sunset for you
FT-GPT	what movie would you like ?
AUGNLG-FT	what movies are you interested in seeing then ?
Domain: *Music*	
Input MR	*CONFIRM(song_name=mile high;playback_device=bedroom speaker)*
Reference	just to make sure , you want to listen to mile high on the bedroom speaker .
FT-GPT	you are listening to the song mile high by mile high .
AUGNLG-FT	what is your favorite song from the mile high club ? (or any song for that matter)
Domain: *Rentalcars*	
Input MR	*REQUEST(pickup_date=march 5th;dropoff_date=march 9th)*
Reference	would you like to rent it on march 5th and return it on march 9th ?
FT-GPT	what date would you like the car to arrive from ?
AUGNLG-FT	will you be picking it up on march 5th and returning it on march 9th ?
Domain: *Ridesharing*	
Input MR	*INFORM(wait_time=2;ride_fare=36.01)*
Reference	the cab will arrive in 2 minutes and it's going to cost 36.01 .
FT-GPT	what is the fare for 2 ?
AUGNLG-FT	please wait 2 hours and 36.01 will be added to your cart .
Domain: *Services*	
Input MR	*INFORM(is_unisex=true;phone_number=925-446-4144) @ NOTIFY_SUCCESS (none)*
Reference	it is uniesex and you have an appointment set . phone them at 925-446-4144 .
FT-GPT	your appointment has been scheduled . the salon is unisex and phone is 925-446-4144 .
AUGNLG-FT	your appointment is confirmed . the unisex bathroom is located at 925-446-4144 .
Domain: *Weather*	
Input MR	*INFORM(humidity=43)*
Reference	it will be around 43 % .
FT-GPT	the humidity will be 43 percent .
AUGNLG-FT	the humidity is 43 percent .
Domain: *Media*	
Input MR	*INFORM(directed_by=roxann dawson) @ NOTIFY_SUCCESS (none)*
Reference	okay , i started the movie . the director is roxann dawson .
FT-GPT	the movie has started .
AUGNLG-FT	it's directed by roxann dawson and it's calledn'tify success .

Table 16: Randomly sampled generation examples from FEWSHOTSGD.