DG^2: Data Augmentation Through Document Grounded Dialogue Generation

Anonymous ACL submission

Abstract

Collecting data for training dialog systems can be extremely expensive due to the involvement of human participants and the need for extensive annotation. Especially in document-grounded dialog systems, human experts need to carefully read the unstructured documents to answer the users' questions. As a result, existing document-grounded dialog datasets are relatively small-scale and obstruct the effective training of dialogue systems. In this paper, we propose an automatic data augmentation technique grounded on documents through a generative dialogue model. The dialogue model consists of a user bot and agent bot that can synthesize diverse dialogues given an input document, which are then used to train a downstream model. When supplementing the original dataset, our method achieves significant improvement over traditional data augmentation methods. We also achieve great performance in the low-resource setting.

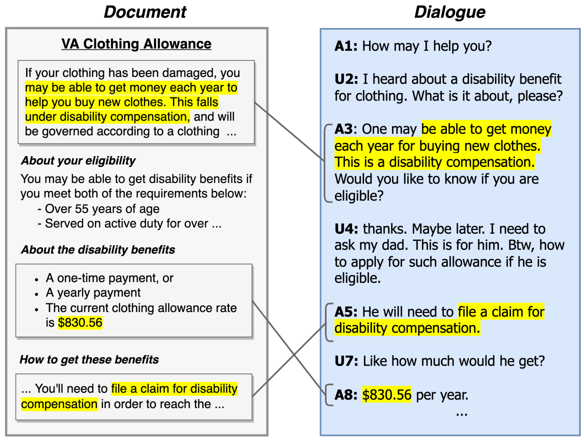

Figure 1: An example from Doc2Dial of dialogue conversation produced from grounding to an associated document. The agent must select the correct spans and engage in a fluent manner to generate a proper response.

1 Introduction

Most of human knowledge is stored in the form of documents, ranging from answering factoid questions (Reddy et al., 2019) to providing how-tos on millions of tasks (Zhang et al., 2020a). How to comprehend and retrieve relevant knowledge from documents given a user query is a challenging research problem. Inspired by real-world applications, there have been more works (Rajpurkar et al., 2016a, 2018; Kwiatkowski et al., 2019; Yang et al., 2015) that aims to tackle this challenge. In this work, we focus on the task of conversational information seeking based on the associated documents, which are often referred to as document-grounded dialogue systems (Ma et al., 2020).

Recent works have introduced various datasets for building document-grounded conversational question answering and dialogue systems. Some work such as QuAC (Choi et al., 2018) and CoQA (Reddy et al., 2019) first explored the direction of conversational question answering. Then, ShARC (Saeidi et al., 2018) added follow-up questions by agents. Later, Doc2Dial (Feng et al., 2020a) further expends the dialogue actions and domains, which aims to simulate more kinds of real-life scenarios. However, such dataset is typically hard to scale up and to new domains, as it requires carefully crafted dialogue flows and expensive human annotations.

As a consequence, one main obstacle for developing scalable and effective document grounded dialog systems is the lack of sufficient data. In chit-chat scenarios, recent works such as DialoGPT (Zhang et al., 2020b), Meena (Adiwardana et al., 2020), and Blender (Roller et al., 2021) have achieved human-like performance by taking the advantage of training on a large-scale corpus. Similarly, task-oriented dialog systems such as ARDM (Wu et al., 2021) and SimpleTOD (Hosseini-Asl et al., 2020) have also utilized large-scale corpora or pre-trained models to achieve good performance. The aforementioned models were trained with millions of samples, while the current document-grounded dialogue datasets like Doc2Dial (Feng et al., 2020a) only contains thousands of conversations. Training on such a small-scale dataset constrains the performance of neural network models. Therefore, augmenting existing datasets can help build a more effective document-grounded dialogue system.

One popular approach to augmenting datasets is to paraphrase existing seed data. The most straightforward form of paraphrasing is to directly use a model trained to generate paraphrase pairs (Gao et al., 2020). Back-translation serves as another type of paraphrasing, which first translates a sentence into another language and then back again (Chadha and Sood, 2019; Bornea et al., 2021). Back-translation ensures the quality and correctness of the augmented data and often shows improvement in downstream models. Both methods aim to provide variety to the training data without greatly altering the semantics of the original sentences. However, these methods only operate on the existing dialogue data and fail to take advantage of the available document for augmentation.

Another direction for data augmentation is to generate examples from scratch by grounding to auxiliary documentation. Lewis et al. (2021) generate question-answer pairs with a model pre-trained on available training data. This often requires additional filtering or denoising measures to ensure correctness of generated data. Also, these models are built for the purposes of single-turn question answering, rather than multi-turn dialogues.

Inspired by Alberti et al. (2019), we propose an automatic document-grounded dialogue generation (DG^2) method that augments the amount of data available for training a dialogue system. The model consists of a user bot and an agent bot that alternately generates utterances to complete a conversation. The user bot includes a span extraction model that can first select a passage and then predict the rationale start and end positions inside a passage. The agent bot has a denoising mechanism to filter out generated rationales irrelevant to the conversation. The user bot begins by selecting a passage from the document that is most relevant to the current context. It then selects a rationale span from this passage and generates the user utterance. The agent bot takes the selected span from the user bot, and then checks if it can find the correct rationale span, and finally generates the agent response. This process repeats until an entire dialogue is generated.

We evaluate our model on a representative document-grounded dialog dataset Doc2Dial (Feng et al., 2020a). We test and generate additional dialogs with both the seen documents and unseen documents. We augment the original dataset and train it on a downstream model. The results show that our method improves the performance of the downstream model after augmentation. We also test scenarios of low-resource settings. We train and evaluate the generative models with only $25\%, 50\%, 75\%$ data. Experimental results show that our method perform well even when training data is scarce.

2 Related Work

2.1 Document Grounded Dialogue Systems

Document Grounded Dialogue System (DGDS) is the type of dialogue systems that the dialogues are grounded on the given documents. It helps humans to better retrieve information they want as most of human knowledge is stored in the form of documents. The study of DGDS can greatly impact the future way of interacting with knowledge.

Recently, there are many document grounded dialogue datasets proposed. Doc2Dial (Feng et al., 2020b) is a representative document grounded dialogue dataset which involved human-to-human conversations and focused on real scenarios under social welfare domains. Previous datasets such as CoQA (Reddy et al., 2019) and QuAC (Choi et al., 2018) focused on machine reading comprehensions. SharC (Saeidi et al., 2018) is close to Doc2Dial. Its conversations are grounded to short text snippets, and contains follow-up questions. ABCD (Chen et al., 2021) supports customer service interactions by providing Agent Guidelines as additional documentation to aid in task-oriented conversations.

An example of DGDS from Doc2Dial is shown in Figure 1. For each turn, the agent needs to look at the specific paragraph inside the document to be capable of answering the user's questions. Moreover, the agent can also ask follow-up questions. For A3, the agent asks "Would you like to know if you are eligible?". In this way, the agent guides the user to center more on the details in the document. Due to the complexity of Doc2Dial, simulating such dialogues is highly nontrivial.

2.2 Data Augmentation

Data augmentation for question answering and dialogue systems has been well-studied in the past.

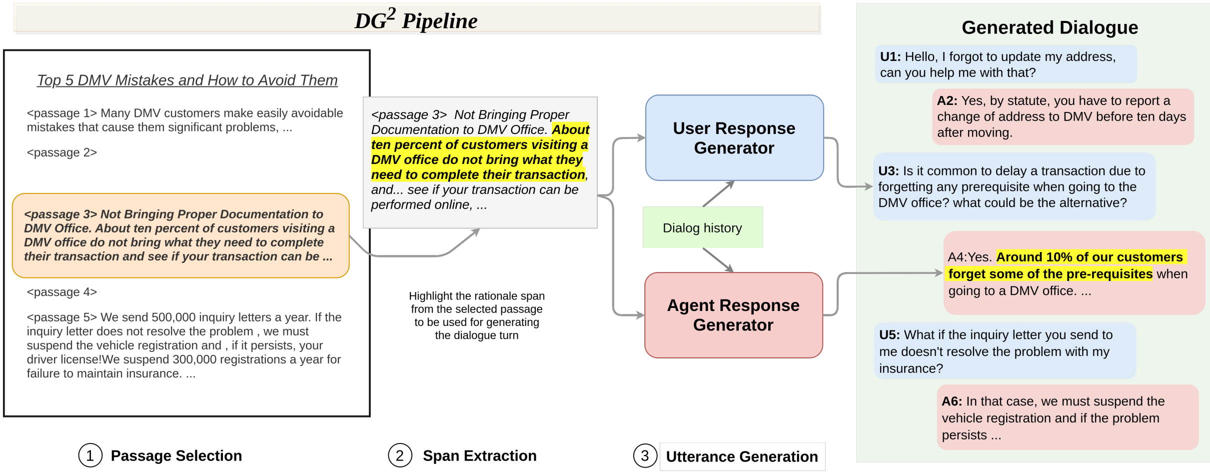

Figure 2: Overall pipeline of DG^2. Given a document and the dialogue history, DG^2 iteratively performs (1) passage selection, (2) rationale extraction, and (3) utterance generation to produce a completed dialogue.

There are two major directions: paraphrasing existing QA pairs from seed data or generating new QA pairs from scratch.

Paraphrasing is a simple and effective technique to augment natural language datasets. It has been widely used in many NLP tasks including natural language understanding, question answering, and task-oriented dialog systems (Gao et al., 2020) to improve the downstream models' performance. In question answering, paraphrasing with back-translation (Chadha and Sood, 2019; Bornea et al., 2021) is well-studied for datasets such as SQUAD (Rajpurkar et al., 2016b).

Another approach is generating new question-answer pairs. Early question-answer generation models used rule-based methods (Rajpurkar et al., 2016b). More recently, neural network-based question-answer pair generation models have been studied. PAQ (Lewis et al., 2021) generates 65 million question-answer pairs based on Wikipedia and trained a retrieval based that.

However, existing approaches have not explored applications for conversational question answering yet, especially for document grounded dialog systems. Compared to single-turn question answering datasets like SQUAD (Rajpurkar et al., 2016b), it involves additional complexity of modeling dialog flow and interconnection naturalness. Also, instead of only providing an answer span, datasets like Doc2Dial (Feng et al., 2020b) have free-form agent responses. The agent needs to produce natural utterances conditional to the selected rationale.

Also, existing conversational question generation models (Gu et al., 2021) only focused on the quality of generations but did not address the improvement on downstream models. We design a specific dialog augmentation approach for document-grounded dialog systems. Our work can synthesize the entire conversation, and can be used to improve down-stream task's performance.

3 Document-Grounded Dialogue Setup

A dialogue can be thought of as a series of turns between two interlocutors. Within goal-oriented dialogues, we refer to the first speaker as the user, and the second speaker as the agent, whom we model as $d = [(u_1, a_1), (u_2, a_2), ...(u_t, a_t)]$. In a document-grounded setting, the conversation revolves around the topics and entities mentioned in the associated document. A document is composed of a series of text passages, which are themselves broken down further into spans.

Dialogue success is determined by following the typical success metrics for any given task, where the only difference is that the outcome of the conversation is likely to depend on the ability to reason about the contents of the document. While sophisticated architectures are certainly capable of improving document-grounding, we take a data-centric approach instead by generating new dialogues from the documents to serve as additional training data for the downstream model.

4 Data Augmentation via DG^2

We propose **D**ocument-**G**rounded **D**ialogue **G**eneration (DG^2) as a method of data augmentation. We aim to generate a complete and coherent dialogue given a document by building two bots talking to each other.

Given a document C, we can model a dialog d between the user and the agent with:

$$p(d|C) = \prod_{i=1}^{t} p(u_i, a_i|c_i \in C) \quad (1)$$

where u_i is the user turn utterance, a_i is the agent turn utterance, and c_i is the selected passage at i-th turn.

We further decompose the model into three parts: passage selection, rationale extraction, and utterance generation. We also apply a filtering model to ensure the quality of generated utterances.

4.1 Passage Selection

A document can often be very long, so it must be divided into smaller passages first. Then, we need to rank the passages, and select a relevant passage given the dialogue context. We can maximize the passage probability for c_t with contrastive loss where the positive passages are from ground truth, and the negative passages are from the same document.

$$p(c_t|\{u_i, a_i\}_{i<t}, C) \quad (2)$$

During generation, we sample from the probability distribution to select the passage. We choose to sample rather than perform greedy selection since this allows for choosing different passages given the same dialogue context, thereby increasing the diversity of the augmentation.

4.2 Rationale Extraction

Next, we further extract a rationale span from the selected passage.

$$p(r_t|\{u_i, a_i\}_{i<t}, c_t)$$

Span extraction systems typically model the start and end position of a span independently as $p(r_{\text{start}}|c) \times (r_{\text{end}}|c)$. This settings works well when the span is short, as is often the case for standard question answering tasks. However, the spans encountered in some document-grounded dialog datasets are much longer causing problems in traditional approaches. As an alternative, we propose an autoregressive method that samples the start and end position in sequentially with:

$$p(r_t) = p(r_{\text{start}}|c) \times p(r_{\text{end}}|r_{\text{start}}, c) \quad (3)$$

To ensure that the autoregressive property holds, we add the predicted start position's hidden state H_{start} and each position's hidden state H_i, and then we project the combined hidden state with a learnable function f_r to get the final predicted end position. Thus, the training objective becomes to maximize

$$r_{\text{end}} = \arg\max_i f_r(H_{\text{start}} + H_i) \quad (4)$$

When extracting a rationale, we first sample a start position from top-k options. Conditioned on this start index, we then sample the end position. This allows us to extract different rationales given the same context, which greatly improves the diversity of generated dialogues compared to using the same rationale.

4.3 Utterance Generation

Given the selected passage and the extracted rationale, we can now start to generate the user utterance and the agent utterance.

User Utterance As seen in Figure 2, user model generates a user utterance conditioned on the dialog history and the extracted rationale. Instead of only using the rationale to generate utterances, we provide the context passage along with the rationale for better performance. To tell the model where the rationale is in the passage, we highlight the rationale span by wrapping its text in the input with "[" and "]". The new passage with the rationale span information is defined as c'_t.

We then model the user utterance with a encoder-decoder where the input is the dialogue history and the passage c'_t, and the output is the user utterance.

$$p(u_t) = p(u_t|\{u_i, a_i\}_{i<t}, c'_t) \quad (5)$$

Agent Utterance Similar to user utterance generation, we model the agent utterance with a encoder-decoder.

$$p(a_t) = p(a_t|\{a_i, u_i\}_{i<t}, c'_t) \quad (6)$$

The difference is that the dialogue history now includes the previous generated user utterance. The rationale position information in the passage is processed similarly as in user utterance generation. We can repeat the user utterance and agent utterance generation process to generate the entire dialogue.

4.4 Filtering the Augmented Data

Roundtrip consistency checking (Alberti et al., 2019; Zhong et al., 2020) has previously been used to improve the correctness of generated augmentation data. It utilizes a model to double-check whether the answer span is the same as the span used to generate the question. Based on this insight, rather than tuning a sampling temperature to trade-off against noise and diversity, we instead greedily pick the rationale span and use consistency checking to filter for quality. For our purposes, we expect the extracted rationale to be aligned with the dialogue context as well as the user utterance.

We build a new passage selector and rationale extraction model such that:

$$p(\hat{c}_t | \{u_i, a_i\}_{i<t}, , u_t, C) \quad (7)$$

$$p(\hat{r}_t | \{u_i, a_i\}_{i<t}, u_t, \hat{c}_t) \quad (8)$$

where $\hat{c}_t$ is the predicted passage from the document C with the dialogue context and the generated user utterance, and $\hat{r}_t$ is the prediction rationale within $\hat{c}_t$. When $\hat{r}_t$ is not aligned the previous r_t, we remove this utterance u_t. Because rationale spans can be very long, filtering based on exact match will be too strict, so we filter based on f1 word overlap.

4.5 Document Positional Information

When a document is divided into passages, it loses positional information between different passages. As a dialogue progresses, we can expect to focus more on the later part of a document, which involves more details of a topic. Therefore, it is important to incorporate the turn information and the passage position information into the model.

We use a simple yet effective method to combine the dialogue turn positional information and passage positional information. For the speaker positions we use a prompt "user{num}:" or "agent{num}:", where "num" is replaced with the number of turns so far. This allows the model to track how many turns have passed, leading to a more coherent dialog structure. For the passage positions, we embed a passage index to indicate the location of the passage within the document. Combining the two flows together, the model is able to have conversations focused on the beginning of the document at the first, and naturally shift towards the end of document later.

5 Experiments

We first introduce the datasets evaluated with our method, then the baselines for comparisons, and in the end our method's implementation details.

5.1 Datasets

	Dialogue Level				Document Level	
	#dial	#turns	#tok	span	#doc	#tok
train	3,474	11.8	15.0	26.5	415	834
valid	661	12.1	15.3	25.8	273	821
test	661	12.0	14.9	24.5	273	809
DG^2	3,474	12.0	14.2	42.2	415	834

Table 2: Doc2Dial dataset statistics. The following abbreviations are made: 'dial' is short for dialogue, 'tok' is short for tokens, and 'doc' is short for documents.

Doc2Dial consists of two subtasks around identifying relevant spans based on dialogue context and producing cohesive responses based on extracted rationales (Feng et al., 2020a). Formulated as a span selection task, user utterance understanding requires an agent to interpret user queries in the context of the dialogue history and then select the relevant span from the associated document. Predicted spans are graded based on Exact match (EM) and F1-score. Exact match is when the predicted span exactly lines up with the actual span. F1-score balances the recall and precision of the predicted uni-grams compared to the gold span.

The second subtask is agent response prediction, which requires an agent to generate a natural language response to the user query given the dialogue context and the document. Response quality is measured by SacreBLEU metric (Post, 2018) which aims to capture how closely the predicted response lines up with the gold response. Table 2 shows Doc2Dial's dialogue-level statistics and document-level statistics.

5.2 Baselines

We compare against a number of baselines typically used to augment natural language data. In contrast to our technique, these methods all operate on the existing dialogues, whereas our method generates new dialogues from scratch from the associated document.

Easy Data Augmentation Wei and Zou (2019) propose to augment data through a series of surface form alterations. In particular, Easy Data Augmentation (EDA) consists of inserting new tokens, deleting random tokens, swapping pairs of tokens, or replacing tokens with their synonyms.

Back-translation Back-translation is another strong augmentation method which first translates

Model	Validation			Test			Span Coverage
	EM	F1	BLEU	EM	F1	BLEU	
Original data	58.13	72.61	37.08	58.34	73.25	36.89	48.27
+ EDA	**60.40**	74.30	37.72	59.71	73.62	37.63	48.27*
+ Back-translation	60.15	73.74	36.68	60.17	73.35	37.32	48.27*
+ Paraphrase	59.97	73.92	37.76	57.98	72.71	38.40	48.27*
+ DG^2	60.30	**74.34**	**38.07**	**60.92**	**74.53**	**38.57**	**57.65**

Table 1: Experimental results on the Doc2Dial dataset. EM stands for Exact Match. DG^2 outperforms all other data augmentation methods on almost every metric. *EDA, Back-translation, and Paraphrase do not modify span information and thus are unable to increase span coverage in relation to the original data.

some text into a separate language and then back-translates to the original language. We follow BERT-QA (Chadha and Sood, 2019), in translating all user utterances to French and then back to English to augment the original dialogues.

Paraphrase Paraphrasing can be achieved by training a sequence-to-sequence model on parallel paraphrase pairs corpora. In particular, we train a BART-base model (Lewis et al., 2020a) on the MRPC (Dolan and Brockett, 2005), QQP (Iyer et al., 2017) and PAWS (Zhang et al., 2019) datasets.

5.3 Coverage Metric

During inference, any section within the document is fair game for discussion. A model trained on dialogues that cover larger portions of the given documents should therefore perform better later on. Consequently, a strong data augmentation method should aim to generate dialogues that cover as much of the document as possible. We formalize this intuition with the span coverage metric, which we calculate as:

$$\text{Coverage} = \frac{\sum_{\text{span}} |\bigcup_{d \in \text{doc}_i} \bigcup_{s \in d} s|}{|\text{document}_i|}$$

where s refers to spans within a document and doc refers to the number of documents in the corpus.

5.4 Implementation Details

For passage ranker, and rationale extraction model, we fine-tuned RoBERTa-base (Liu et al., 2019) on the downstream training datasets. For utterance generators, we fine-tuned BART-base (Lewis et al., 2020b). We set total input length of 512-tokens which is 128 tokens for dialogue followed by 360 tokens for the document, with some room left over for special tokens. The augmented data is generated with beam size 4, top-p 0.9, and temperature 0.9. When utilizing the augmented data, we pretrained the downstream model on the augmented data for one epoch before fine-tuning (Alberti et al., 2019). The default f1 threshold is set to 0.9, which we determined by validating against the dev set. For fine-tuning, we train for five epochs, and use the same optimizer of AdamW (Loshchilov and Hutter, 2019) and learning rate of $3e-5$ for all experiments.

6 Results and Analysis

This section shows the results for the full dataset and low-resource settings. We also conduct human evaluation on the generated dialogues. Afterwards, we discuss the results by analyzing generated examples.

6.1 Main Results

As shown in Table 1, DG^2 achieves the overall best performance compared to other baselines that only augment the original human-annotated data. Other baselines all show some improvements over the downstream model only trained using the original data. EDA has very high EM and F1 scores for the rationale extraction task, but suffers at producing coherent dialogues as measured by BLEU. Paraphrase has relatively lower EM and F1 scores, but it achieves better BLEU scores than EDA and Back-translation. We suspect that this is because Paraphrase contains more diverse utterances as the inputs than other baselines.

When evaluating the augmented dialogues with the original training set's documents, we find that DG^2 achieves higher span coverage. Unlike the other methods, DG^2 is able to generate novel rationales to increase the diversity of the augmented data, which we believe plays a large factor in im-

Speaker	Utterance
User	I'm thinking about getting some retirement benefits.
Agent	Are you wondering what age to start your benefits? *Rationale: What Is The Best Age To Start Your Benefits?*
User	Yes, I want to know what age I should start my benefits.
Agent	Are you planning to apply for Medicare? *Rationale: Should I apply for Medicare?*
User	Yes, I am.
Agent	Then you should know that Medicare usually starts when you reach age 65. *Rationale: Remember, Medicare usually starts when you reach age 65. If you decide to delay starting your benefits, be sure to contact Social Security about 3 months before you turn age 65 to check about applying for Medicare. Even if you have health insurance through a ...*
User	What if I need to enroll but don't?
Agent	Your Medicare coverage may be delayed and cost more. *Rationale: your Medicare coverage may be delayed and cost more. Please read the general and special enrollment period information on page 12 of our Medicare booklet to find out what may happen if you delay.*
User	What age is the best age to begin benefits?
Agent	The answer is that there is no one " best age" for everyone and, ultimately, it is your choice. You should make an informed decision about when to apply for benefits based on your individual and family circumstances. *Rationale: The answer is that there is no one " best age " for everyone and, ultimately, it is your choice. You should make an informed decision about when to apply for benefits based on your individual and family circumstances.*

Table 4: Generated dialogue showing typical agent and user interaction. Unlike question answering datasets, the agent can also ask questions to the user to guide the direction of the conversation.

proving downstream metrics.

Filtering	#Spans	EM	F1
None	-	57.78	73.27
f1 < 0.5	top-1	57.73	73.01
f1 < 0.9	top-10	58.23	73.05
f1 < 0.9	top-1	**60.80**	**74.38**
f1 < 0.95	top-1	59.21	74.00
f1 < 0.98	top-1	59.26	73.84

Table 5: We test different quality thresholds to determine the optimal level of filtering. A higher F1-score means that more samples are filtered.

6.2 Low Resource Setting

To further illustrate the performance of DG^2, we train all the models with only $25\%, 50\%, 75\%$ of the original training data. We generate the dialogues based on the documents in the knowledge base. In this limited data setting, our model generally outperformed Back-translation. However, compared to EDA, there is still some performance gap. We suspect that this is because when training with less data, the generative models' performance degenerates faster than the downstream model. We hope to overcome these issues with further improvements on data quality filtering.

6.3 Different Filtering Strategies

Prior works in data augmentation have shown that filtering the synthetically generated examples can provide a meaningful boost in the data quality (Chen and Yu, 2021). As a result, we tune against different F1-score thresholds and span counts on the validation set. When the generated dialogue produces a higher F1-score, then this example is more likely to also produce better results during testing. The span count determines how many examples we consider when calculating this score. While raising the F1-score threshold increases the potential quality of the data, it comes as the expense of keeping fewer of the generated examples. Based on Table 5, we observe a sweet spot at 0.9, where a stricter filtering process would remove too many examples while a looser filtering process would lower the quality too much.

Model	25%			50%			75%		
	EM	F1	BLEU	EM	F1	BLEU	EM	F1	BLEU
Baseline	43.08	64.01	32.76	41.61	62.25	34.35	58.03	72.61	36.48
+ EDA	46.68	64.68	33.97	**56.09**	70.51	35.84	**59.84**	**73.40**	36.24
+ Back-translation	**47.48**	65.18	33.00	54.44	69.52	35.30	58.66	72.75	36.08
+ DG^2	46.48	**65.58**	32.90	54.51	**71.40**	35.74	58.89	73.38	**37.01**

Table 6: Experimental results on low-resource settings.

	Consistency	Fluency	Naturalness	Overall
Human	3.80	3.96	3.56	3.70
DG^2	3.60	4.18	2.98	3.38

Table 7: Human evaluation results on the generated dialogues.

6.4 Human Evaluation

We conduct human evaluation on the human dialogues and the generated dialogues. We randomly sample 50 dialogues from each class. We shuffled the sampled dialogues and ask annotators to rate the dialogues with a score 1-5 in four different aspects: consistency, fluency, naturalness, and overall quality.

From the evaluation results, the generated dialogues show better fluency than original human dialogues with $p < 0.05$. We observed that some human dialogues contain typos and grammar errors, while the generated dialogues are more grammatically correct, which explains the score difference. In terms of other human evaluation metrics, the generated dialogues are still worse than the original human dialogues.

6.5 Qualitative Analysis

We now compare and contrast two examples generated by our procedure. Table 4 shows a good example from the document-grounded dialogue dataset. In the first four turns, the agent guides the user's focus by asking relevant questions. When the user wants to know more details, the agent then switches to provide the relevant knowledge retrieved from the rationale. This behavior is different from traditional question answering datasets where the agent simply reacts to user requests rather than exhibiting proactive behavior. On the flip side, one major problem of the current approach is repetition. The user continues to ask about forgetting to update their address despite attempts by the agent to answer their query. Although the surface form of the user utterances are different, the semantic meaning remains the same. This repetition confuses the agent who then extracts irrelevant rationales, further exacerbating the situation.

7 Ethical Consideration

The models and approaches introduced in our work involve using synthetic data as an enhancement to existing datasets for modeling document-grounded dialogue. For the existing datasets, they are often dialogue simulation data generated by human workers based on their understanding of the associated document content and dialogue context. There are potential biases or toxic content introduced in the existing simulation during data collection. We can address such concerns by making efforts to improve the quality of the generated data that has shown its effectiveness in the downstream task. Therefore, our method can add an extra layer of safety and privacy if we only use generated data for training downstream models. Future work can explore how data augmentation can help to build a more private and safe dataset.

8 Conclusion

To address the problem of limited data in document-grounded dialogue systems, we propose DG^2 to perform data augmentation via dialogue generation. Our technique generates diverse utterances grounded on the given document while filtering the utterances to ensure quality and correctness when training on the downstream model. We demonstrated the effectiveness of our pipeline by showing the improvement over the previous data augmentation methods. We additionally show competitive results in the low-resource setting when a limited amount of human annotated data is available for training. Future work will explore more techniques of filtering to improve data quality. We hope this spurs further research into document-grounded augmentation techniques for dialogue systems.

References

Daniel Adiwardana, Minh-Thang Luong, David R. So, Jamie Hall, Noah Fiedel, Romal Thoppilan, Zi Yang, Apoorv Kulshreshtha, Gaurav Nemade, Yifeng Lu, and Quoc V. Le. 2020. Towards a human-like open-domain chatbot. *CoRR*, abs/2001.09977.

Chris Alberti, Daniel Andor, Emily Pitler, Jacob Devlin, and Michael Collins. 2019. Synthetic QA corpora generation with roundtrip consistency. In *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 6168–6173. Association for Computational Linguistics.

Mihaela A. Bornea, Lin Pan, Sara Rosenthal, Radu Florian, and Avirup Sil. 2021. Multilingual transfer learning for QA using translation as data augmentation. In *Thirty-Fifth AAAI Conference on Artificial Intelligence, AAAI 2021, Thirty-Third Conference on Innovative Applications of Artificial Intelligence, IAAI 2021, The Eleventh Symposium on Educational Advances in Artificial Intelligence, EAAI 2021, Virtual Event, February 2-9, 2021*, pages 12583–12591. AAAI Press.

Ankit Chadha and Rewa Sood. 2019. BERTQA - attention on steroids. *CoRR*, abs/1912.10435.

Derek Chen, Howard Chen, Yi Yang, Alexander Lin, and Zhou Yu. 2021. Action-based conversations dataset: A corpus for building more in-depth task-oriented dialogue systems. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2021, Online, June 6-11, 2021*, pages 3002–3017. Association for Computational Linguistics.

Derek Chen and Zhou Yu. 2021. GOLD: improving out-of-scope detection in dialogues using data augmentation. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 429–442. Association for Computational Linguistics.

Eunsol Choi, He He, Mohit Iyyer, Mark Yatskar, Wen-tau Yih, Yejin Choi, Percy Liang, and Luke Zettlemoyer. 2018. Quac: Question answering in context. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 2174–2184. Association for Computational Linguistics.

William B. Dolan and Chris Brockett. 2005. Automatically constructing a corpus of sentential paraphrases. In *Proceedings of the Third International Workshop on Paraphrasing, IWP@IJCNLP 2005, Jeju Island, Korea, October 2005, 2005*. Asian Federation of Natural Language Processing.

Song Feng, Kshitij P. Fadnis, Q. Vera Liao, and Luis A. Lastras. 2020a. Doc2dial: A framework for dialogue composition grounded in documents. In *The Thirty-Fourth AAAI Conference on Artificial Intelligence, AAAI 2020, The Thirty-Second Innovative Applications of Artificial Intelligence Conference, IAAI 2020, The Tenth AAAI Symposium on Educational Advances in Artificial Intelligence, EAAI 2020, New York, NY, USA, February 7-12, 2020*, pages 13604–13605. AAAI Press.

Song Feng, Hui Wan, R. Chulaka Gunasekara, Siva Sankalp Patel, Sachindra Joshi, and Luis A. Lastras. 2020b. doc2dial: A goal-oriented document-grounded dialogue dataset. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 8118–8128. 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, ACL 2020, Online, July 5-10, 2020*, pages 639–649. Association for Computational Linguistics.

Jing Gu, Mostafa Mirshekari, Zhou Yu, and Aaron Sisto. 2021. Chaincqg: Flow-aware conversational question generation. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, EACL 2021, Online, April 19 - 23, 2021*, pages 2061–2070. Association for Computational Linguistics.

Ehsan Hosseini-Asl, Bryan McCann, Chien-Sheng Wu, Semih Yavuz, and Richard Socher. 2020. A simple language model for task-oriented dialogue. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*.

Shankar Iyer, Nikhil Dandekar, and Kornel Csernai. 2017. First quora dataset release: Question pairs. *Kaggle Competition*.

Tom Kwiatkowski, Jennimaria Palomaki, Olivia Redfield, Michael Collins, Ankur Parikh, Chris Alberti, Danielle Epstein, Illia Polosukhin, Matthew Kelcey, Jacob Devlin, Kenton Lee, Kristina N. Toutanova, Llion Jones, Ming-Wei Chang, Andrew Dai, Jakob Uszkoreit, Quoc Le, and Slav Petrov. 2019. Natural questions: a benchmark for question answering research. *Transactions of the Association of Computational Linguistics*.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020a. BART: denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 7871–7880. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020b. BART: denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 7871–7880. Association for Computational Linguistics.

Patrick S. H. Lewis, Yuxiang Wu, Linqing Liu, Pasquale Minervini, Heinrich Küttler, Aleksandra Piktus, Pontus Stenetorp, and Sebastian Riedel. 2021. PAQ: 65 million probably-asked questions and what you can do with them. *CoRR*, abs/2102.07033.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized BERT pretraining approach. *CoRR*, abs/1907.11692.

Ilya Loshchilov and Frank Hutter. 2019. Decoupled weight decay regularization. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Longxuan Ma, Wei-Nan Zhang, Mingda Li, and Ting Liu. 2020. A survey of document grounded dialogue systems (DGDS). *CoRR*, abs/2004.13818.

Matt Post. 2018. A call for clarity in reporting BLEU scores. In *Proceedings of the Third Conference on Machine Translation: Research Papers, WMT 2018, Belgium, Brussels, October 31 - November 1, 2018*, pages 186–191. Association for Computational Linguistics.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for squad. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics, ACL 2018, Melbourne, Australia, July 15-20, 2018, Volume 2: Short Papers*, pages 784–789. Association for Computational Linguistics.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016a. Squad: 100, 000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 2383–2392. The Association for Computational Linguistics.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016b. Squad: 100, 000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing, EMNLP 2016, Austin, Texas, USA, November 1-4, 2016*, pages 2383–2392. The Association for Computational Linguistics.

Siva Reddy, Danqi Chen, and Christopher D. Manning. 2019. CoQA: A conversational question answering challenge. *Transactions of the Association for Computational Linguistics*, 7:249–266.

Stephen Roller, Emily Dinan, Naman Goyal, Da Ju, Mary Williamson, Yinhan Liu, Jing Xu, Myle Ott, Eric Michael Smith, Y-Lan Boureau, and Jason Weston. 2021. Recipes for building an open-domain chatbot. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, EACL 2021, Online, April 19 - 23, 2021*, pages 300–325. Association for Computational Linguistics.

Marzieh Saeidi, Max Bartolo, Patrick S. H. Lewis, Sameer Singh, Tim Rocktäschel, Mike Sheldon, Guillaume Bouchard, and Sebastian Riedel. 2018. Interpretation of natural language rules in conversational machine reading. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing, Brussels, Belgium, October 31 - November 4, 2018*, pages 2087–2097. Association for Computational Linguistics.

Jason W. Wei and Kai Zou. 2019. EDA: easy data augmentation techniques for boosting performance on text classification tasks. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 6381–6387. Association for Computational Linguistics.

Qingyang Wu, Yichi Zhang, Yu Li, and Zhou Yu. 2021. Alternating recurrent dialog model with large-scale pre-trained language models. In *Proceedings of the 16th Conference of the European Chapter of the Association for Computational Linguistics: Main Volume, EACL 2021, Online, April 19 - 23, 2021*, pages 1292–1301. Association for Computational Linguistics.

Yi Yang, Wen-tau Yih, and Christopher Meek. 2015. Wikiqa: A challenge dataset for open-domain question answering. In *Proceedings of the 2015 Conference on Empirical Methods in Natural Language Processing, EMNLP 2015, Lisbon, Portugal, September 17-21, 2015*, pages 2013–2018. The Association for Computational Linguistics.

Li Zhang, Qing Lyu, and Chris Callison-Burch. 2020a. Intent detection with WikiHow. In *Proceedings of the 1st Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics and the 10th International Joint Conference on Natural Language Processing*, pages 328–333, Suzhou, China. Association for Computational Linguistics.

Yizhe Zhang, Siqi Sun, Michel Galley, Yen-Chun Chen, Chris Brockett, Xiang Gao, Jianfeng Gao, Jingjing Liu, and Bill Dolan. 2020b. DIALOGPT : Large-scale generative pre-training for conversational response generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations, ACL 2020, Online, July 5-10, 2020*, pages 270–278. Association for Computational Linguistics.

Yuan Zhang, Jason Baldridge, and Luheng He. 2019. PAWS: paraphrase adversaries from word scrambling. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 1298–1308. Association for Computational Linguistics.

Victor Zhong, Mike Lewis, Sida I. Wang, and Luke Zettlemoyer. 2020. Grounded adaptation for zero-shot executable semantic parsing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 6869–6882. Association for Computational Linguistics.

A Training Details

	Hyper-parameters
batch size	16
warm-up steps	1,000
learning rate	3e-5
dropout	0.1
weight decay	0.01
max gradient norm	1.0
training epochs	5

Table 8: Training Details

We trained our model on and A4000 and A6000 GPUs. The detailed hyper-parameters can be found in the table above.

B Bad Generation Examples

We show a bad generation example here to illustrate one of the generation problem.

Speaker	Utterance
User	I was planning to move and forgot to update my address. What do I do?
Agent	By statute, you must report a change of address to DMV within ten days of moving. That is the case for the address associated with your license, as well as all the addresses associated with each registered vehicle, which may differ *Rationale: By statute, you must report a change of address to DMV within ten days of moving. That is the case for the address associated with your license, as well as all the addresses associated with each registered vehicle, which may differ.*
User	Am I able to do anything if I forget to update my address?
Agent	It is not sufficient to only: write your new address on the back of your old license; tell the United States Postal Service; or inform the police officer writing you a ticket. *Rationale: It is not sufficient to only: write your new address on the back of your old license; tell the United States Postal Service; or inform the police officer writing you a ticket.*
User	What happens if I forget to update my address?
Agent	You will miss a suspension order and may be charged with operating an unregistered vehicle and/or aggravated unlicensed operation, both misdemeanors. This really happens, but the good news is this is a problem that is easily avoidable. *Rationale: you will miss a suspension order and may be charged with operating an unregistered vehicle and/or aggravated unlicensed operation, both misdemeanors. This really happens, but the good news is this is a problem that is easily avoidable. Learn more ...*

Table 9: A bad generated dialogue here shows repetition. Although the user utterances' surface form are different, their semantic meaning is the same.