Knowledge-Grounded Conversational Data Augmentation with Generative Conversational Networks

Yen-Ting Lin
ytl@ieee.org

Alexandros Papangelis   **Seokhwan Kim**   **Dilek Hakkani-Tur**
{papangea, seokhwk, hakkanit}@amazon.com

Abstract

While rich, open-domain textual data are generally available and may include interesting phenomena (humor, sarcasm, empathy, etc.) most are designed for language processing tasks, and are usually in a non-conversational format. In this work, we take a step towards automatically generating conversational data using Generative Conversational Networks, aiming to benefit from the breadth of available language and knowledge data, and train open domain social conversational agents. We evaluate our approach on conversations with and without knowledge on the Topical Chat dataset using automatic metrics and human evaluators. Our results show that for conversations without knowledge grounding, GCN can generalize from the seed data, producing novel conversations that are less relevant but more engaging and for knowledge-grounded conversations, it can produce more knowledge-focused, fluent, and engaging conversations. Specifically, we show that for open-domain conversations with 10% of seed data, our approach performs close to the baseline that uses 100% of the data, while for knowledge-grounded conversations, it achieves the same using only 1% of the data, on human ratings of engagingness, fluency, and relevance.

1   Introduction

Conversational Artificial Intelligence has progressed a lot in the recent past, partly due to advances in large pre-trained language models (PLM) and partly due to commercial conversational agents (Alexa, Siri, Cortana, Google Assistant, and others). It is evident, however, that many challenges still remain, such as handling idioms, humour, expressing empathy, processing unstructured knowledge, and so on. One big factor for this is the lack of large and rich conversational data that include these complex aspects of human communication. While the research community is making great efforts in collecting such data (e.g. empathetic dialogues (Rashkin et al., 2019), persuasion (Wang et al., 2019), and others), these are still small compared to the amount of data needed to train deep neural networks. Furthermore, these expensive data collections usually target a single phenomenon at a time, and hence do not necessarily scale to the richness of human conversations. Another challenge for real world applications is privacy, preventing the use of much of the publicly available conversational data.

In this work, we take a first step into automatically generating conversational data from unstructured textual knowledge (e.g. web sources) using Generative Conversational Networks (GCN) (Papangelis et al., 2021). GCN is a meta-learning method initially proposed for intent detection and slot tagging; we extend that approach and demonstrate that we can learn how to generate responses grounded in unstructured knowledge. Specifically, GCN learns how to generate labelled, diverse, and targeted data that are optimised with Reinforcement Learning (RL). This is achieved by using a generator model that produces new data which is used to train a separate learner model. The performance of the learner model is used as a reward signal to train the generator, so that over time the quality of the generated data increases. This reward signal can allow us to guide the data generation towards dimensions of interest, for example, knowledge-grounded, empathetic, or polite dialogues and can be derived from automatic metrics or human feedback if the system is deployed. In our case, the generator produces open-domain dialogues and the learner is a conversational agent that is trained on that data. Selecting an appropriate reward signal can be difficult, since we want to generate good quality dialogues that do not exist in the training data, but dialogue evaluation is a challenging open problem. We therefore investigate a combination of multiple metrics that capture different aspects: BLEU (Papineni et al., 2002)

and ROUGE (Lin, 2004) to ensure some similarity with the reference data, BERTScore (Zhang et al., 2020a)[1] to encourage good quality dialogues, and Knowledge F1[2] (Shuster et al., 2021) to encourage knowledge integration. It should be noted that while the focus in this work is knowledge grounding in open-domain response generation, our approach is extensible to other conversational phenomena with appropriate reward signals.

Our main contributions are: a) we generate knowledge-grounded conversational data from unstructured textual knowledge (e.g. the kind of knowledge available on the web); b) we improve response generation quality over a baseline that uses fine-tuning on seed data, eliminating the need for additional human-human data collection; and c) we demonstrate improved performance on knowledge-grounded response generation on Topical Chat, as measured by KF1 and human evaluations.

2 Related Work

Language Data Augmentation Approaches. There are a lot of recent works on data augmentation, but most of them are geared towards individual language processing tasks rather than training complete conversational agents. Due to lack of space we only mention the ones that are most relevant to our work.

PROTODA (Kumar et al., 2021) uses prototypical networks to augment data for intent classification while GenSF (Mehri and Eskenazi, 2021) uses DialoGPT (Zhang et al., 2020b) for zero-shot slot tagging; DINO (Schick and Schütze, 2021) uses PLM to generate data for semantic textual similarity; Campagna et al. (2020) focus on zero-shot dialogue state tracking and use an abstract dialogue model to generate data. SOLOIST (Peng et al., 2021) uses a PLM fine-tuned on large dialogue corpora and is designed for transactional (goal-oriented) dialogues. Mohapatra et al. (2020) use PLM to train user simulators from crowd-generated conversations and their instructions. Lin et al. (2021a) train domain-independent user simulators for transactional dialogues. Chang et al. (2021) augment data for Data-To-Text NLG by generating text in two steps: replacing values with alternatives and using GPT-2 to produce surface

text. They then do automatic labelling and enforce cycle-consistency (make sure text can be generated from data and vice versa). Stahlberg and Kumar (2021) focus on data generation for Grammatical Error Correction and propose a method that can generate an erroneous sentence given a correct sentence and an error tag. Chen and Yu (2021) use data augmentation to improve out of scope (OOS) detection models. Specifically, they extract utterances from a different dataset than the one they are targeting that can be labelled as OOS and then do some smart filtering to select good candidates. Kim et al. (2021) propose NeuralWOZ, a framework to generate dialogue state tracking data given goal descriptions and API calls. NeuralWOZ has a data generator and a data labeler that annotates the data. GCN does not need a separate labeler model and has the added option of being continually trained with RL. PromDA (Wang et al., 2022b) is a soft-prompt learning method for low-resource NLP tasks, that addresses the problem of overfitting (memorizing) when fine-tuning a PLM with a very small number of examples. The authors generate data for sequence classification and labelling. However, this approach is not tested on full dialogues which require significantly more context in the input. Bayer et al. (2022) propose a three step method, where they first fine-tune a PLM and then generate new data-points by adjusting the temperature of the generation. They then filter the generated data by putting a threshold on embedding similarity with respect to the target class centroid. GCN uses RL to guide the generation process, alleviating the need for explicit post-processing. Wang et al. (2022a) present a data augmentation approach for aspect-based sentiment analysis that can generate data along two dimensions: aspects and polarity. The resulting data are then used in a contrastive learning setting to train a sentiment classifier. Similarly to other approaches, it is not clear how it would perform in knowledge-grounded dialogue generation, with large inputs (context and available knowledge). For a more comprehensive review of data augmentation for language tasks, please see (Feng et al., 2021; Li et al., 2021; Sahin, 2022).

Regarding data augmentation for conversational agents, one of the most prominent methods is User Simulation (Schatzmann et al., 2007; Asri et al., 2016; Liu and Lane, 2018; Papangelis et al., 2019; Lin et al., 2021b; Shah et al., 2018, e.g.). These approaches, however, have been designed to work

---

[1]Data driven evaluation metrics tend to favor dialogues similar to the ones used during their training and we found that we cannot solely rely on such metrics.

[2]KF1 measures the token level F1 score between a knowledge piece and an utterance.
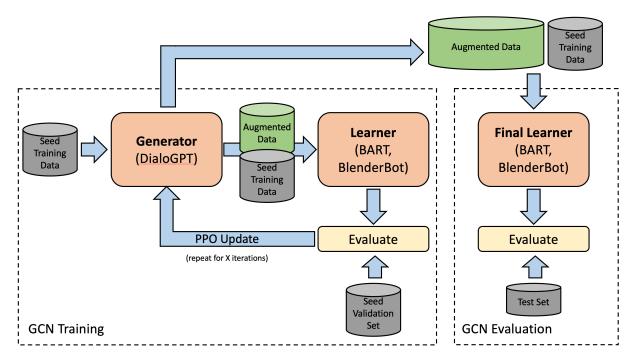
Figure 1: The architecture of our approach using Generative Conversational Networks for knowledge-grounded dialogues. The generator is first fine-tuned with seed data and produces an augmented dataset and those data are used to train a learner. The performance of the learner on a held-out validation set (along with auxiliary metrics) is used as a reward to update the generator.

with well-structured databases whereas we are concerned with grounding open-domain conversational responses in unstructured knowledge. DG^2 (Wu et al., 2021) focuses on data augmentation for document-grounded dialogues, using Doc2Dial (Feng et al., 2020). The authors use an agent bot and a user bot to conduct simulated conversations and generate data. However, unlike GCN, the bots are not continually updated and may not generalise well to produce novel content. The code was not available for a direct comparison on our dataset, however, in the few-shot learning experiments, they demonstrate good performance with as little as 25% of the data (869 Doc2Dial dialogues), whereas we demonstrate competitive performance by only using 1% of the training data (86 Topical Chat dialogues).

Few-Shot Approaches. Another line of related work is based on few-/zero-shot transfer learning for dialogue tasks. Again due to space we only mention the most relevant works. Earlier studies have focused on improving the generalizability of natural language understanding problems such as intent classification (Chen et al., 2016) and slot filling (Bapna et al., 2017; Shah et al., 2019) for unseen labels or domains. Then, focus was placed on other dialogue problems including dialogue state tracking (Wu et al., 2019; Rastogi et al., 2020),

next action prediction (Mosig et al., 2020), and natural language generation (NLG) (Peng et al., 2020). Bapna et al. (2017) and Shah et al. (2019) utilized slot descriptions for improving the zero-shot slot filling performance. Rastogi et al. (2020) used slot, intent, and task-specific API descriptions for schema-guided dialogue state tracking. Mosig et al. (2020) based on a structural schema in graph representations instead of textual descriptions for zero-shot action prediction and NLG. Peng et al. (2020) pre-trained on massive text data followed by dialog act labeled dialogue utterances. Madotto et al. (2020) used a large-scale pre-trained language model as a few-shot learner with task-specific prompting. All the methods presented above, however, are geared towards specific tasks and are not shown to generalize to open-domain social or knowledge-grounded conversation.

3 Notation

We conduct experiments under two settings: conversations without explicit knowledge-grounding (we call them *open-domain*) and knowledge-grounded conversations.

3.1 Open-domain conversations

We define a multi-turn conversation as a list of utterances: U_1, U_2, ..., U_N where U_i is the utterance at turn i, and N is the number of turns in the conversation. Each utterance is composed of words w_1, ..., w_M, where M is the number of words in the utterance. Conversational agents are given a subset of the dialog context, for example the t most recent turns U_{N-t-1}, ..., U_{N-1} and generate the response U_N.

3.2 Knowledge-grounded conversations

To formulate knowledge-grounded responses, conversational systems need two steps (sometimes taken jointly): knowledge selection and response generation (Dinan et al., 2019). The conversational agent should therefore first select relevant knowledge pieces from the sources provided with respect to the current dialog context and then generate a response that incorporates the selected knowledge. A knowledge piece in our case is defined as a fact consisting of one or more sentences (see Table 8 for some examples). To select a knowledge retrieval method, we conducted preliminary experiments comparing TF-IDF, BM25, and BERTScore and we saw that the more sophisticated parsing and dense retrieval methods did not outperform TF-IDF. We therefore represent conversation context and knowledge using TF-IDF vectors and utilize TF-IDF-based retrieval over documents as our knowledge selection mechanism. We select the most relevant knowledge using cosine similarity with the context $C = U_{N-t-1}, ..., U_{N-1}$:

$$k_N = \underset{k}{\operatorname{argmax}}\{\cos(t_C, t_k)\} \tag{1}$$

where t_C is the TF-IDF vector corresponding to the context and t_k is the vector corresponding to knowledge piece k. Knowledge-grounded conversational agents are given not only the dialog context C but also the selected knowledge k_N (or multiple pieces of knowledge as in our case) and are asked to generate a response U_N that incorporates k_N.

4 Generative Conversational Networks

GCN (Papangelis et al., 2021) (Figure 1) consist of two models in a meta-learning architecture: a data generator and a learner. The generator creates a labeled dataset that is used to train a new learner (a conversational agent in our case) in a supervised fashion. The learner is then evaluated on an external validation set and its performance is used as a proxy for the quality of the dataset. This quality measure is used as a reward in a RL setup that trains the generator. Over time, the generator learns to create data of better and better quality, with respect to the learner's task, leading the learner to perform well. To avoid overfitting the validation set, we can limit the number of meta-iterations or include domain-independent performance metrics, such as fluency, perplexity, or even human feedback. When deployed, the generator is directly optimized on the test set (i.e. real interactions). Both models can be pre-trained with seed data, if available, and paired with reward estimation, GCN can be used for continuous learning from user feedback. This approach has been proven to work well for intent detection and slot tagging in goal-oriented conversations (Papangelis et al., 2021) and we here apply it to train social conversational agents. Different from Generative Adversarial Networks (Goodfellow et al., 2014)[3] where the model tries to mimic the data, GCN models are guided by an external reward signal - that does not need to be differentiable - and can therefore generalize better. Depending on the optimization criteria, we can set the direction towards which the models will go, for example more polite conversations, more technical terminology, different dialect, knowledge grounding, and even directions that are not easily quantifiable (e.g. engagingness ratings from humans).

For open-domain conversations, as a proof of concept, we conduct few-shot experiments using 10% of the data and for knowledge-grounded conversations which is the main focus of this work, we use 1%, 5%, and 10% of the data; we call these the seed data (D_{seed}). At the beginning of training, we sample D_{seed} from the data D, fine-tune the generator on D_{seed} (see $G.train(D_{seed})$, line 4 in Algorithm 1), and then start the outer loop meta-iterations. Along with the training data, we sample the corresponding percentage of validation data D_{val}. Once the training is complete, we spawn a new learner, train it on the seed and synthetic data, and evaluate it on D_{test} which has been unseen so far. As described earlier, each meta-iteration has four phases: data generation, learner fine-tuning, learner evaluation, and generator update. Algorithm 1 summarizes the process.

---

[3] A direct comparison with GAN approaches is out of scope for this work and we leave it for the future.

4.1 Data generation

In the first phase of the process, the generator G is given some dialog context sampled from D_{seed} and, in the knowledge-grounded condition, top-m retrieved knowledge pieces k from the TFIDF retriever. Specifically, we give the last two turns as context and the top-3 matching knowledge pieces, and ask the generator to predict the next system response. At each turn i, the context C_i is used to retrieve relevant knowledge k_i that is then used as input to the generator which produces the next turn response U_i:

$$
\begin{aligned}
U_i &= G(C_i, k_i) \\
&= \bigcup_{w=0}^{n} \{sample(P_{LM}(w|w_{n-1}, ..., w_0, c_i, k_i))\}
\end{aligned} \quad (2)
$$

where P_{LM} is the probability of the underlying language model generating each word w of the response U_i, and *sample* is the method we use to sample from the PLM, (greedy, nucleus, etc). This way, the generator produces a synthetic dataset D_{synth} of size L, where each datapoint is a triplet of context C_i, knowledge k_i, and response U_i:

$$
D_{synth} = \{(C_i, k_i, U_i), i = 1, ..., L\} \quad (3)
$$

In essence, to create D_{synth}, instead of taking the human response from the data as a target, we use the generated response U as a target and feed that along with C and k to fine-tune the learner.

4.2 Learner fine-tuning and evaluation

Since the learner's task is knowledge-grounded dialogue, it does not have access to the TFIDF retriever and, as k may contain multiple relevant knowledge pieces, it will learn to perform its own implicit knowledge selection, not knowing what the exact knowledge piece used to produce U was.

At every iteration, we create a new learner (based on a pre-trained model) and fine-tune it on $D_{seed} \cup D_{synth}$ (see line 10 in Algorithm 1). The knowledge-grounded learners are trained using a combination of cross entropy loss and knowledge retrieval score, specifically, Knowledge F1 (KF1) (Shuster et al., 2021) which measures the F1 score between the produced utterance and the selected knowledge piece. The trained learner is then evaluated (see line 11 in Algorithm 1) and a numerical reward is computed by combining several metrics.

Algorithm 1 GCN training procedure.

1: **procedure** TRAIN($D_{seed}, D_{val}, D_{test}, \epsilon$)
2:      Initialize Generator G
3:      **if** D_{seed} **then**
4:          G.train(D_{seed})
5:      **end if**
6:      Performance$_{meta} \leftarrow 0$
7:      **while** Performance$_{meta} < 1 - \epsilon$ **do**
8:          $D_{synth} \leftarrow G$.generate()
9:          Sample and initialize new Learner l
10:         l.train($D_{seed} \cup D_{synth}$)
11:         Performance$_{meta} \leftarrow l$.evaluate(D_{val})
12:              $\triangleright$ Performance$_{meta} \in [0, 1]$
13:         G.update(Performance$_{meta}$)
14:      **end while**
15:      $D_{synth} \leftarrow G$.generate()
16:      Sample and initialize new final Learner L
17:      L.train($D_{seed} \cup D_{synth}$)
18:      L.evaluate(D_{test}) $\triangleright$ or other evaluator
19: **end procedure**

4.3 Generator update

Following (Ziegler et al., 2019) and (Papangelis et al., 2021), we use Proximal Policy Optimization (PPO) (Schulman et al., 2017) with the following modified reward R to train the generator using the learner's validation performance r:

$$
R(C, U) = r(C, U) - \beta \log \frac{G(U|C)}{G_{ref}(U|C)} \quad (4)
$$

where C represents the context including the knowledge if applicable, U represents the model's response, and β is a constant that prevents G from diverging too much from a reference generator G_{ref}.

In the open-domain condition, the generator uses multiple losses to calculate r: BLEU (Papineni et al., 2002), ROUGE-L (Lin, 2004), and BERTScore (Zhang et al., 2020a) which measure the similarity of the learner-produced utterance and the utterance in the data (D_{seed} or D_{synth}). We evaluate each learner on the validation set D_{val} and compute the above metrics using the human responses in D_{val} as references. The weighted sum of the NLG metrics comprises the reward for the generator training. The weights were determined via grid search: 0.1, 0.01, 0.95, for BLEU, ROUGE-L and BERTScore, respectively. In the knowledge-grounded condition, we use a combination of BLEU-1 and KF1 (with weights 0.75 for

BLEU-1 and 0.25 for KF1) as we found via grid search that it produced better results.

After the meta-iterations are finished, we pick the best performing generator checkpoint (measured by the learners' performance on D_{val} at each meta-iteration) and create a final synthetic set D_{final_synth} that is 5 times the size of the seed. We then create a new learner as our final learner (i.e. the conversational agent) and fine-tune it on $D_{seed} \cup D_{final\_synth}$ (lines 15-18 in Algorithm 1). If D_{final_synth} is of good quality, we should expect the final learner to outperform the baseline, as it is trained with more data. The results presented next are all computed on the final learners, trained for 3 epochs, evaluated on D_{test}, and averaged over 3 runs (as are our baselines).

5 Experiments

To evaluate GCN as a data augmentation method for conversations with and without knowledge, we conduct few-shot experiments on Topical Chat (TC) (Gopalakrishnan et al., 2019). TC is a set of human-human conversations, without explicitly defined roles for each participant, collected over Amazon Mechanical Turk. Each participant had access to a set of facts or articles with some conversations being symmetric (participants had access to the same knowledge) and some being asymmetric. All experiments were conducted on 2 Tesla V100 GPUs with 32GB memory each.

5.1 Model ablations

To quantify the effect of data augmentation and RL in both conditions, we train BART (Lewis et al., 2020) or BlenderBot-small (BBs)[4] (Roller et al., 2021) models for no-knowledge and knowledge-grounded conversations respectively, under the following conditions:

- **Baseline (BART/BBs)**: In this condition, we train BART or BBs on the seed data. This will give us a lower bound on performance (if the augmented data is good, it should help performance).

- **Data augmentation without RL (GCN-RL)**: In this condition, we pre-train a DialoGPT-small[5] (Zhang et al., 2020b) generator with the seed data, and use that to generate 5x more data. We then use the seed and

---

generated data to train a final BART or BBs (learner) model depending on the task.

- **Data augmentation with RL (GCN+RL)**: In this condition, we take the GCN-RL generator and iteratively update it using RL, as described in section 4. This is the full GCN framework. At the end of the meta-iterations, we take the best-performing generator and use it to create 5x more data. We use the seed and generated data to train a final BART or BBs model.

- **Generator direct evaluation (G±RL generator)**: For the knowledge-grounded condition, in addition to the above three models, we evaluate the generator by having it directly interact with humans instead of generating data to train a learner.

5.2 Open-domain conversations

For the open-domain conversations, we sample 10% of TC as seed for GCN and use DialoGPT-small and BART as initial models for the generator and the learner, respectively. We compare the performance of the GCN learner and 3 baselines using automated metrics, and also conduct human evaluations. Our baselines are: BART trained with the same seed data (BART 10%), BART trained with the entire training set (BART 100%), and a GCN learner trained on seed and synthetic data but without updating the generator via RL (GCN-RL). Last, we also compare against the human responses that appear in the data ("Data" in Tables 1 and 3).

5.3 Knowledge-grounded conversations

For knowledge grounded conversations, we sample 1%, 5%, and 10% of TC as seed data for GCN. Again we use DialoGPT-small as a generator but we use BBs as our learner. We compare the performance of GCN against similar baselines to the open-domain condition: BBs trained on the seed or the entire data, GCN without RL, human responses from the data, and we also evaluate the generators themselves if we were to use them directly as conversational agents (G±RL generator). Even though KF1 is the metric of choice in related work on knowledge-grounded conversations, we did not find works that report KF1 for TC.

Model	BLEU	Rouge(1/2/L)	BScore	Engaging.	Fluency	Relevance	Overall
Data	-	-	-	3.85	4.55	3.77	4.06
BART (100%)	3.1	20.3/6.1/17.8	0.861	3.80	4.58	3.68	4.02
BART (10%)	**2.0**	**18.5/4.2/16.0**	**0.858**	3.63	4.50	**3.62**	3.92
GCN-RL	1.1	15.0/2.1/12.6	0.850	3.70	4.47	3.47	3.88
GCN+RL	1.3	15.8/2.7/13.6	0.851	**3.79**	4.49	3.58	**3.96**

Table 1: Automatic and human evaluation results. Human evaluators rate responses on a scale of 1 to 5. BScore stands for BERTScore. Bold indicates statistically significant difference (t-test assuming unequal variance). BART (100%) and BART (10%) are BART trained on 100% and 10% of the data, GCN-RL is GCN without RL, and GCN+RL is GCN with RL training.

Model	1% data			5% data			10% data		
	PPL	KF1	BL-4	PPL	KF1	BL-4	PPL	KF1	BL-4
BBs	23.39	0.10	0.07	23.52	0.17	0.09	21.69	0.17	0.09
GCN-RL	26.47	0.15	0.08	24.54	0.18	0.09	23.11	0.18	0.09
GCN+RL	27.11	**0.20**	0.08	24.60	**0.25**	0.14	23.67	**0.28**	0.10

Table 2: Results of automated evaluation on knowledge-grounded conversations. All models try to maximize KF1, and the baseline is the same model as the GCN learners (BBs: BlenderBot-small, 90M parameters).

Model	Eng.	Flu.	Rel.	Avg
Data	3.74	3.98	3.57	3.76
BBs (100%)	3.69	3.99	3.57	3.75
BBs (1%)	3.64	3.86	3.42	3.64
G-RL generator	3.47	3.35	3.23	3.35
G-RL learner	3.58	3.85	**3.48**	3.64
G+RL generator	3.37	3.27	3.40	3.35
G+RL learner	**3.73**	**3.97**	**3.48**	**3.73**
	Wins Percentage			
Combinations	**Base**	**G-RL**	**G+RL**	**Tie**
BBs VS G-RL	40.0	**44.3**	-	15.7
BBs VS G+RL	44.7	-	**47.7**	7.6
All 3 models	29.3	25.7	**45.0**	-

Table 3: Human evaluation results (top) for knowledge-grounded conversations. Human evaluators rate responses with the same conversation context on a scale of 1 to 5. In a different evaluation (bottom), they were asked to choose the best response from two options. BBs: BlenderBot-small (90M), G-RL: GCN without RL, G+RL: GCN with RL.

6 Results

6.1 Automatic evaluation

We report perplexity (PPL), BLEU-4 (Papineni et al., 2002) with the "method 7" smoothing function from (Chen and Cherry, 2014) as it has higher correlation with human ratings, and KF1. We calculate these metrics on the TC "frequent" test set, (Tables 1 and 2). In the open-domain condition, we see that BART 10% outperforms GCN agents on all automated metrics. In knowledge-grounded conversations, we see that GCN+RL is able to incorporate more knowledge as evidenced by the higher KF1.

6.2 Human evaluation

Due to the intrinsic one-to-many property of conversation, reference-based metrics may not correlate with human ratings; our generated conversation may be appropriate for the dialogue context but different from the reference responses. For this reason, we also conduct human evaluation (following sub-section). Human evaluators rate the output of the GCN learner, the baselines, and the ground truth. Specifically, they rate how engaging, fluent, and relevant each response is, on a scale from 1 to 5. We generate 1,000 samples for each condition using the same context and make sure we have 3 ratings per sample per condition. Tables 1 (right) and 3 show the results of the evaluation, where we see that in the open-domain condition, the GCN learner produces engaging but less relevant conversations. This is likely because the model inserts facts or other output that is not entirely relevant, but is perceived as more engaging (e.g. information on a somewhat relevant subject, fun fact, etc.). Consistent with prior work, (Papangelis et al., 2021), this shows that GCN can generalize from the data. When it comes to knowledge-grounded conversations, where GCN is explicitly trained to optimize KF1 (among other metrics), then relevance is indeed higher than the baseline. Overall, averaging the three metrics, GCN+RL outperforms BART 10% and is close to BART 100%'s performance. All models are outperformed by the human responses, which may be due to the size of our models or the number of training iterations.

Iterations	PPL	KF1	BL-4
1	30.8	0.146	0.179
2	31.1	0.147	0.182
3	30.7	0.146	0.186
5	30.8	0.163	0.190
10	27.1	0.238	0.085

Table 4: Performance of GCN+RL for varying number of meta-iterations. Here, we generate 3x the seed data and use 1% of TC.

Data Mult.	PPL	KF1	BL-4
1	26.5	0.201	0.082
2	27.4	0.213	0.084
3	28.6	0.17	0.083
5	22.2	0.25	0.154
10	22.9	0.27	0.106

Table 5: Performance of GCN+RL for varying size of generated data (as a multiplier of the seed). Here, we do 5 meta-iterations and use 1% of TC.

For knowledge-grounded conversations (Table 3) we see that GCN+RL produces more engaging and fluent conversations and overall outperforms both baselines while again being close to BBs trained on all the data. In pair-wise comparisons, GCN+RL is generally preferred more than the other models. Overall, for the GCN conditions, given that we generate 5x the seed data, the total amount of data is about 6% of the size of TC and our results show that the generated data is indeed of high quality, since the same model (BlenderBot-small) using the generated data performs close to the one that uses 100% of the human-human data and close to the data itself. It should be noted that GCN achieves this performance using small models (in the order of 100M parameters each).

In Figure 2 in the appendix, we show the Amazon Mechanical Turk setup that we used during our human evaluations.

6.3 Generated data diversity

In this section we further analyze the performance of GCN, specifically its performance with respect to the number of meta-iterations (Table 4) and the amount of generated data (Table 5). In Table 4, we see that KF1 increases as we have more meta-iterations, meaning that the generator actually leads the learner to learn to produce more knowledgable responses. BLEU naturally drops as these more knowledgable responses may not appear in the data.

Data %	BBs	GCN-RL	GCN+RL
1%	8.1%	17.4%	25.1%
5%	8.5%	12.1%	24.5%
10%	5.9%	9.2%	13.6%

Table 6: Out-Of-Vocabulary (OOV) rates for various seed percentages.

We observe similar trends in Table 5, where we vary the amount of synthetic generated data (as a multiplier of the size of the seed data). Regarding data diversity, Table 6 presents out of vocabulary rates for all three conditions when using 1%, 5%, and 10% of the data as seed. Higher rates mean more diversity but may also mean that the generated data is farther from the seed data. Paired with the results in Tables 1-3, however, we can see that GCN+RL produces more diverse data that are still in-domain and useful.

6.4 Examples

In Table 7, we provide example responses for conversations without knowledge grounding, and in Table 8 we show example responses for knowledge-grounded conversations. We see that in both cases GCN+RL tries to insert knowledge or fun facts and that appear to be received well by the human judges. However, since there are no knowledge-grounding abilities, the model hallucinates and this is what likely drives relevance ratings down. In the knowledge-grounded example, we see that GCN+RL can use the knowledge pieces more effectively than the other models.

7 Conclusion

We presented a meta-learning method based on GCN to generate conversational data grounded on unstructured textual knowledge such as what can be found on the web. We show that given a small seed (1% of TC or 86 examples) our approach can generate high quality data that can be used to train a competitive conversational agent able to do knowledge selection and grounding. Lower reference-based metric scores (Table 1, left and Table 2) indicate that our method generates diverse data. Based on the good scores in the human evaluation (Table 1, right and Table 3), we see that the generated data is also useful for the task at hand, a result in line with the results on intent detection and slot tagging in (Papangelis et al., 2021). As future work, we are conducting thorough analyses and evaluations

Context

	Hello, do you watch football?
	Hi, yes. I love football! I did not realize the season lasted 17 weeks!
	Yeah pretty long. I guess with 32 teams playing it takes that much time
	Did you see they only rest one weekend during the season?
	That is crazy, so much work goes in there. Did you know U of Iowa painted their visitor locker room pink?
	Oh yes,that is so funny. they really want to intimidate the other team, but is more like a joke
	Wow, what an idea. Did you know bowlers used to make more than football players in the 60s

Responses

Baseline	that's a good point. i wonder how many of the top bowlers make twice as much as top football stars. do you have a favorite
GCN-RL	that's a good point. i'm not a big fan of football, but i do like to watch it. do you like football?
GCN+RL	i didn't know that, that's interesting. did you know there are more than 30 million players in the nfl? i wonder how many of those players are still playing?

Table 7: Example responses for open-domain conversations. The models here are **not** trained to optimize knowledgable responses and therefore may hallucinate. This may be the reason for lower relevance scores for GCN+RL (Table 1); however, GCN+RL's tendency to provide fun facts is what may be leading to more engaging responses.

Context

	Did you know that the University of Iowa's locker room is painted pink? I wonder why?
	I think I did hear something about that. I imagine it is an attempt to psych the other team out.
	So, it would be in the visiting team's locker room but not their own?

Retrieved Knowledge (subset)

	the University of Iowa's locker room for visiting football teams is completely painted pink in the 1960's top bowlers made twice as much as top football stars
	Football is a family of team sports that involve, to varying degrees, kicking a ball [...]
	in 39 out of 50 states, the highest paid state employee is either a college football or basketball coach.

Responses

Baseline	i think it would be in the visiting team's locker room. i don't know why they would want to psych the other team out
GCN-RL	that's interesting. i've never heard that before. do you have a favorite team?
GCN+RL	i'm not too familiar with iowa's locker room for visiting football teams, but i do know that they're the highest paid employees in the state of iowa. do you know why they are paid so much?

Table 8: Example responses for knowledge-grounded conversations. In this setting, models are trained to optimize knowledgable responses which explains in part why GCN+RL is preferred in human judgements (Table 3).

of the generated data, exploring more model architectures, and are working on more conversational phenomena.

References

Layla El Asri, Jing He, and Kaheer Suleman. 2016. A sequence-to-sequence model for user simulation in spoken dialogue systems. In *Interspeech 2016, 17th Annual Conference of the International Speech Communication Association, San Francisco, CA, USA, September 8-12, 2016*, pages 1151–1155. ISCA.

Ankur Bapna, Gökhan Tür, Dilek Hakkani-Tür, and Larry P. Heck. 2017. Towards zero-shot frame semantic parsing for domain scaling. In *Interspeech 2017, 18th Annual Conference of the International Speech Communication Association, Stockholm, Sweden, August 20-24, 2017*, pages 2476–2480. ISCA.

Markus Bayer, Marc-André Kaufhold, Björn Buchhold, Marcel Keller, Jörg Dallmeyer, and Christian Reuter. 2022. Data augmentation in natural language processing: a novel text generation approach for long and short text classifiers. *International Journal of Machine Learning and Cybernetics*, pages 1–16.

Giovanni Campagna, Agata Foryciarz, Mehrad Moradshahi, and Monica Lam. 2020. Zero-shot transfer learning with synthesized data for multi-domain dialogue state tracking. In *Proceedings of the 58th Association for Computational Linguistics*.

Ernie Chang, Xiaoyu Shen, Dawei Zhu, Vera Demberg, and Hui Su. 2021. Neural data-to-text generation

with lm-based text augmentation. 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 758–768. Association for Computational Linguistics.

Boxing Chen and Colin Cherry. 2014. A systematic comparison of smoothing techniques for sentence-level bleu. In *Proceedings of the Ninth Workshop on Statistical Machine Translation*, pages 362–367.

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.

Yun-Nung Chen, Dilek Hakkani-Tür, and Xiaodong He. 2016. Zero-shot learning of intent embeddings for expansion by convolutional deep structured semantic models. In *2016 IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 6045–6049. IEEE.

Emily Dinan, Stephen Roller, Kurt Shuster, Angela Fan, Michael Auli, and Jason Weston. 2019. Wizard of wikipedia: Knowledge-powered conversational agents. In *7th International Conference on Learning Representations, ICLR 2019, New Orleans, LA, USA, May 6-9, 2019*. OpenReview.net.

Song Feng, Kshitij Fadnis, Q Vera Liao, and Luis A Lastras. 2020. Doc2dial: a framework for dialogue composition grounded in documents. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 13604–13605.

Steven Y. Feng, Varun Gangal, Jason Wei, Sarath Chandar, Soroush Vosoughi, Teruko Mitamura, and Eduard H. Hovy. 2021. A survey of data augmentation approaches for NLP. In *Findings of the Association for Computational Linguistics: ACL/IJCNLP 2021, Online Event, August 1-6, 2021*, volume ACL/IJCNLP 2021 of *Findings of ACL*, pages 968–988. Association for Computational Linguistics.

Ian Goodfellow, Jean Pouget-Abadie, Mehdi Mirza, Bing Xu, David Warde-Farley, Sherjil Ozair, Aaron Courville, and Yoshua Bengio. 2014. Generative adversarial nets. *Advances in neural information processing systems*, 27.

Karthik Gopalakrishnan, Behnam Hedayatnia, Qinglang Chen, Anna Gottardi, Sanjeev Kwatra, Anu Venkatesh, Raefer Gabriel, Dilek Hakkani-Tür, and Amazon Alexa AI. 2019. Topical-chat: Towards knowledge-grounded open-domain conversations. In *INTERSPEECH*, pages 1891–1895.

Sungdong Kim, Minsuk Chang, and Sang-Woo Lee. 2021. NeuralWOZ: Learning to collect task-oriented dialogue via model-based simulation. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*, pages 3704–3717, Online. Association for Computational Linguistics.

Manoj Kumar, Varun Kumar, Hadrien Glaude, Cyprien de Lichy, Aman Alok, and Rahul Gupta. 2021. Protoda: Efficient transfer learning for few-shot intent classification. In *2021 IEEE Spoken Language Technology Workshop (SLT)*, pages 966–972. IEEE.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. 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.

Bohan Li, Yutai Hou, and Wanxiang Che. 2021. Data augmentation approaches in natural language processing: A survey. *CoRR*, abs/2110.01852.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Hsien-chin Lin, Nurul Lubis, Songbo Hu, Carel van Niekerk, Christian Geishauser, Michael Heck, Shutong Feng, and Milica Gašić. 2021a. Domain-independent user simulation with transformers for task-oriented dialogue systems. *arXiv preprint arXiv:2106.08838*.

Hsien-Chin Lin, Nurul Lubis, Songbo Hu, Carel van Niekerk, Christian Geishauser, Michael Heck, Shutong Feng, and Milica Gasic. 2021b. Domain-independent user simulation with transformers for task-oriented dialogue systems. In *Proceedings of the 22nd Annual Meeting of the Special Interest Group on Discourse and Dialogue, SIGdial 2021, Singapore and Online, July 29-31, 2021*, pages 445–456. Association for Computational Linguistics.

Bing Liu and Ian Lane. 2018. End-to-end learning of task-oriented dialogs. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Student Research Workshop*, pages 67–73.

Andrea Madotto, Zihan Liu, Zhaojiang Lin, and Pascale Fung. 2020. Language models as few-shot learner for task-oriented dialogue systems. *arXiv preprint arXiv:2008.06239*.

Shikib Mehri and Maxine Eskenazi. 2021. Gensf: Simultaneous adaptation of generative pre-trained models and slot filling. *SIGdial*.

Biswesh Mohapatra, Gaurav Pandey, Danish Contractor, and Sachindra Joshi. 2020. Simulated chats for task-oriented dialog: Learning to generate conversations from instructions. *arXiv e-prints*, pages arXiv–2010.

Johannes EM Mosig, Shikib Mehri, and Thomas Kober. 2020. Star: A schema-guided dialog dataset for transfer learning. *arXiv preprint arXiv:2010.11853*.

Alexandros Papangelis, Karthik Gopalakrishnan, Aishwarya Padmakumar, Seokhwan Kim, Gokhan Tur, and Dilek Hakkani-Tur. 2021. Generative conversational networks. In *SIGDial*.

Alexandros Papangelis, Yi-Chia Wang, Piero Molino, Gokhan Tur, and AI Uber. 2019. Collaborative multi-agent dialogue model training via reinforcement learning. In *20th Annual Meeting of the Special Interest Group on Discourse and Dialogue*, page 92.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318.

Baolin Peng, Chunyuan Li, Jinchao Li, Shahin Shayandeh, Lars Liden, and Jianfeng Gao. 2021. SOLOIST: building task bots at scale with transfer learning and machine teaching. *Trans. Assoc. Comput. Linguistics*, 9:907–824.

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 *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: Findings*, pages 172–182.

Hannah Rashkin, Eric Michael Smith, Margaret Li, and Y-Lan Boureau. 2019. Towards empathetic open-domain conversation models: A new benchmark and dataset. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5370–5381.

Abhinav Rastogi, Xiaoxue Zang, Srinivas Sunkara, Raghav Gupta, and Pranav Khaitan. 2020. Towards scalable multi-domain conversational agents: The schema-guided dialogue dataset. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 34, pages 8689–8696.

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.

Gözde Gül Sahin. 2022. To augment or not to augment? A comparative study on text augmentation techniques for low-resource NLP. *Comput. Linguistics*, 48(1):5–42.

Jost Schatzmann, Blaise Thomson, Karl Weilhammer, Hui Ye, and Steve Young. 2007. Agenda-based user simulation for bootstrapping a pomdp dialogue system. In *Human Language Technologies 2007: The Conference of the North American Chapter of the Association for Computational Linguistics; Companion Volume, Short Papers*, pages 149–152.

Timo Schick and Hinrich Schütze. 2021. Generating datasets with pretrained language models. 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 6943–6951. Association for Computational Linguistics.

John Schulman, Filip Wolski, Prafulla Dhariwal, Alec Radford, and Oleg Klimov. 2017. Proximal policy optimization algorithms. *arXiv preprint arXiv:1707.06347*.

Darsh Shah, Raghav Gupta, Amir Fayazi, and Dilek Hakkani-Tur. 2019. Robust zero-shot cross-domain slot filling with example values. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5484–5490.

Pararth Shah, Dilek Hakkani-Tür, Gokhan Tür, Abhinav Rastogi, Ankur Bapna, Neha Nayak, and Larry Heck. 2018. Building a conversational agent overnight with dialogue self-play. *arXiv preprint arXiv:1801.04871*.

Kurt Shuster, Spencer Poff, Moya Chen, Douwe Kiela, and Jason Weston. 2021. Retrieval augmentation reduces hallucination in conversation. In *Findings of the Association for Computational Linguistics: EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 16-20 November, 2021*, pages 3784–3803. Association for Computational Linguistics.

Felix Stahlberg and Shankar Kumar. 2021. Synthetic data generation for grammatical error correction with tagged corruption models. In *Proceedings of the 16th Workshop on Innovative Use of NLP for Building Educational Applications*, pages 37–47.

Bing Wang, Liang Ding, Qihuang Zhong, Ximing Li, and Dacheng Tao. 2022a. A contrastive cross-channel data augmentation framework for aspect-based sentiment analysis. *arXiv preprint arXiv:2204.07832*.

Xuewei Wang, Weiyan Shi, Richard Kim, Yoojung Oh, Sijia Yang, Jingwen Zhang, and Zhou Yu. 2019. Persuasion for good: Towards a personalized persuasive dialogue system for social good. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 5635–5649, Florence, Italy. Association for Computational Linguistics.

Yufei Wang, Can Xu, Qingfeng Sun, Huang Hu, Chongyang Tao, Xiubo Geng, and Daxin Jiang. 2022b. Promda: Prompt-based data augmentation for low-resource NLU tasks. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), ACL 2022, Dublin, Ireland, May 22-27, 2022*, pages 4242–4255. Association for Computational Linguistics.

Chien-Sheng Wu, Andrea Madotto, Ehsan Hosseini-Asl, Caiming Xiong, Richard Socher, and Pascale Fung. 2019. Transferable multi-domain state generator for task-oriented dialogue systems. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 808–819.

Qingyang Wu, Song Feng, Derek Chen, Sachindra Joshi, Luis A Lastras, and Zhou Yu. 2021. Dg2: Data augmentation through document grounded dialogue generation. *arXiv preprint arXiv:2112.08342*.

Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020a. Bertscore: Evaluating text generation with bert. In *International Conference on Learning Representations*.

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

Daniel M Ziegler, Nisan Stiennon, Jeffrey Wu, Tom B Brown, Alec Radford, Dario Amodei, Paul Christiano, and Geoffrey Irving. 2019. Fine-tuning language models from human preferences. *arXiv preprint arXiv:1909.08593*.
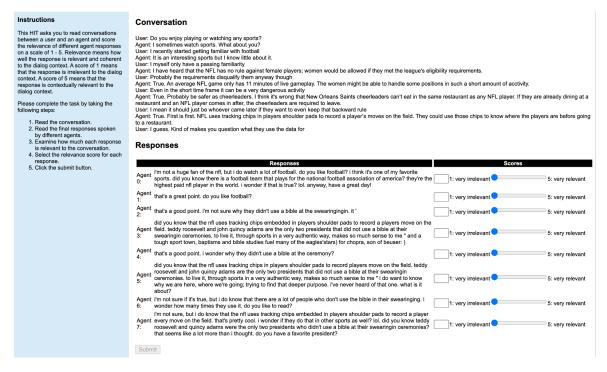
Instructions

This HIT asks you to read conversations between a user and an agent and score the relevance of different agent responses on a scale of 1 - 5. Relevance means how well the response is relevant and coherent to the dialog context. A score of 1 means that the response is irrelevant to the dialog context. A score of 5 means that the response is contextually relevant to the dialog context.

Please complete the task by taking the following steps:

1. Read the conversation.
2. Read the final responses spoken by different agents.
3. Examine how much each response is relevant to the conversation.
4. Select the relevance score for each response.
5. Click the submit button.

Conversation

User: Do you enjoy playing or watching any sports?
Agent: I sometimes watch sports. What about you?
User: I recently started getting familiar with football
Agent: It is an interesting sports but I know little about it.
User: I myself only have a passing familiarity
Agent: I have heard that the NFL has no rule against female players; women would be allowed if they met the league's eligibility requirements.
User: Probably the requirements disqualify them anyway though
Agent: True. An average NFL game only has 11 minutes of live gameplay. The women might be able to handle some positions in such a short amount of acctivity.
User: Even in the short time frame it can be a very dangerous activity
Agent: True. Probably be safer as cheerleaders. I think it's wrong that New Orleans Saints cheerleaders can't eat in the same restaurant as any NFL player. If they are already dining at a restaurant and an NFL player comes in after, the cheerleaders are required to leave.
User: I mean it should just be whoever came later if they want to even keep that backward rule
Agent: True. First is first. NFL uses tracking chips in players shoulder pads to record a player's moves on the field. They could use those chips to know where the players are before going to a restaurant.
User: I guess. Kind of makes you question what they use the data for

Responses

Responses	Scores
Agent 0: i'm not a huge fan of the nfl, but i do watch a lot of football. do you like football? i think it's one of my favorite sports. did you know there is a football team that plays for the national football association of america? they're the highest paid nfl player in the world. i wonder if that is true? lol. anyway, have a great day!	1: very irrelevant ——————— 5: very relevant
Agent 1: that's a great point. do you like football?	1: very irrelevant ——————— 5: very relevant
Agent 2: that's a good point. i'm not sure why they didn't use a bible at the swearingingin. it '	1: very irrelevant ——————— 5: very relevant
Agent 3: did you know that the nfl uses tracking chips embedded in players shoulder pads to record a players move on the field. teddy roosevelt and john quincy adams are the only two presidents that did not use a bible at their swearingin ceremonies. to live it, through sports in a very authentic way, makes so much sense to me " and a tough sport town, baptisms and bible studies fuel many of the eagles'stars) for chopra, son of beuser:)	1: very irrelevant ——————— 5: very relevant
Agent 4: that's a good point. i wonder why they didn't use a bible at the ceremony?	1: very irrelevant ——————— 5: very relevant
Agent 5: did you know that the nfl uses tracking chips in players shoulder pads to record players move on the field. teddy roosevelt and john quincy adams are the only two presidents that did not use a bible at their swearingin ceremonies. to live it, through sports in a very authentic way, makes so much sense to me " i do want to know why we are here, where we're going; trying to find that deeper purpose. i've never heard of that one. what is it about?	1: very irrelevant ——————— 5: very relevant
Agent 6: i'm not sure if it's true, but i do know that there are a lot of people who don't use the bible in their swearinging. i wonder how many times they use it. do you like to read?	1: very irrelevant ——————— 5: very relevant
Agent 7: i'm not sure, but i do know that the nfl uses tracking chips embedded in players shoulder pads to record a player every move on the field. that's pretty cool. i wonder if they do that in other sports as well? lol. did you know teddy roosevelt and quincy adams were the only two presidents who didn't use a bible at their swearingin ceremonies? that seems like a lot more than i thought. do you have a favorite president?	1: very irrelevant ——————— 5: very relevant

Submit

Figure 2: Our human evaluation setup.

A Limitations and Risks

Given that GCN relies on Reinforcement Learning, it may be difficult to tune for more complex applications. Also, due to its meta-learning nature it can be computationally expensive, although we show it here working with small versions of each model.

As with any machine learning approach - especially ones that rely on PLMs - there are inherent risks as to the language that will be produced, which may be biased in many ways. Our approach works with any machine learning model and therefore inherits the risks of the models used. Reinforcement Learning can also play a role in learning biased models if we are not careful.

B Amazon Mechanical Turk Setup

In Figure 2 we show a screenshot of our Amazon Mechanical Turk setup for human evaluation.