A Controllable Model of Grounded Response Generation

Zeqiu Wu[♠] **Michel Galley**[◇] **Chris Brockett**[◇] **Yizhe Zhang**[◇] **Xiang Gao**[◇] **Chris Quirk**[◇]
Rik Koncel-Kedziorski[♠] **Jianfeng Gao**[◇] **Hannaneh Hajishirzi**[♠] **Mari Ostendorf**[♠] **Bill Dolan**[◇]

♠ University of Washington, Seattle, WA, USA
◇ Microsoft Research, Redmond, WA, USA

zeqiuwu1@washington.edu
mgalley@microsoft.com

Abstract

Current end-to-end neural conversation models inherently lack the flexibility to impose semantic control in the response generation process. This control is essential to ensure that users' semantic intents are satisfied and to impose a degree of specificity on generated outputs. Attempts to boost informativeness alone come at the expense of factual accuracy, as attested by GPT-2's propensity to "hallucinate" facts. While this may be mitigated by access to background knowledge, there is scant guarantee of relevance and informativeness in generated responses. We propose a framework that we call controllable grounded response generation (CGRG), in which lexical control phrases are either provided by an user or automatically extracted by a content planner from dialogue context and grounding knowledge. Quantitative and qualitative results show that, using this framework, a GPT-2 based model trained on a conversation-like Reddit dataset outperforms strong generation baselines.

1 Introduction

End-to-end neural models for open-domain response generation (Shang et al., 2015; Sordoni et al., 2015; Vinyals and Le, 2015; Gao et al., 2019a) are capable of generating conversational responses that are both fluent and contextually appropriate. Although the earliest neural generation models were characterized by bland and evasive responses (Li et al., 2016a), surprisingly human-like conversations can be generated using recent diversity-enhancing strategies (Holtzman et al., 2020; Gao et al., 2019b) and massive GPT-2 style models (Radford et al., 2019; Zhang et al., 2020).[1]

[1]For a related task (document creation), 72% of human judges found GPT-2 credible vs. 83% for New York Times articles: https://openai.com/blog/gpt-2-6-month-follow-up/

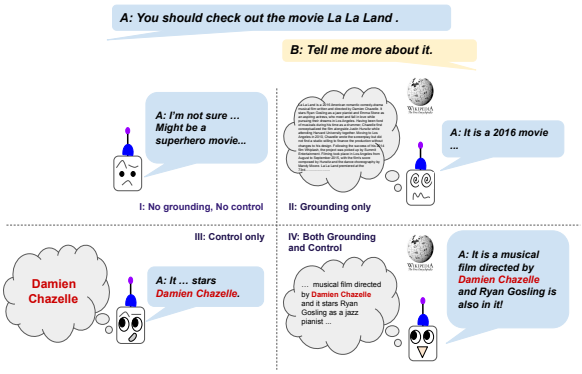

Figure 1: Generated responses tend to be generic or factually incorrect without grounding or control. Adding grounding improves information reliability but may lead to vague responses. Adding control boosts response specificity while it is hard to contextualize phrases without grounding. Adding both control and grounding leads to contentful and reliable responses.

While blandness may no longer present a challenge, the downside has been a propensity towards "hallucinated" or "fake" output (Zellers et al., 2019) of the kind illustrated in scenario I in Figure 1.

Grounded response generation (Ghazvininejad et al., 2018; Dinan et al., 2019; Qin et al., 2019) approaches can inhibit hallucination of facts. Yet grounding alone (e.g, the Wikipedia page about *La La Land* in scenario II of Figure 1) without control and semantic targeting may induce output that is accurate but vague or irrelevant. Controllable text generation (Hokamp and Liu, 2017; Keskar et al., 2019; Tang et al., 2019; See et al., 2019), on the other hand, provides a level of semantic control that can guide the decoder towards relevant output, but in the absence of grounding the model is prevented from associating control phrases with correct facts. We posit that both grounding knowledge and lexical control are essential to generating reliable information. We therefore introduce a gen-

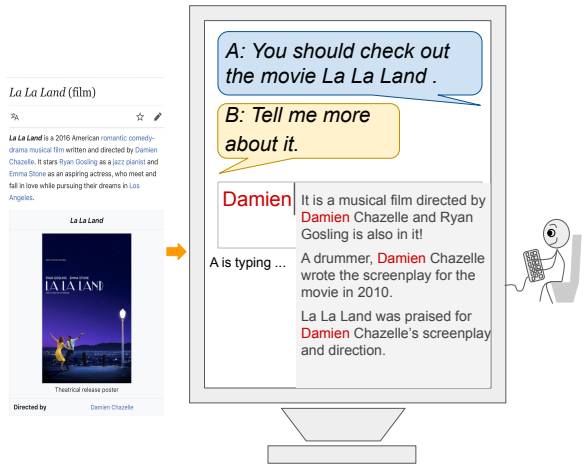

Figure 2: The machine acts as a response editorial assistant that suggests candidate responses for the user A according to the conversation history, the user's partial input (*Damien*) and grounding knowledge.

eration framework called controllable grounded response generation that incorporates both components. Lexical controls not only enforce response specificity, but filter lengthy, irrelevant or incoherent groundings.

Lexical control of conversational text generation has application in editorial assistants that help a person write a document, an email or message. Figure 2 depicts a person typing keywords to indicate their semantic intent, while the machine helps construct the response to be sent out.

This work makes the following contributions: (1) We propose a novel framework called controllable grounded response generation (CGRG) that generates a response from the dialogue context, lexical control phrases and groundings. To the best of our knowledge, this is the first work to integrate both control and grounding into response generation, and explore how they can be mutually beneficial. (2) We adapt the state-of-the-art generation model GPT-2 to this problem setting, and improve results by adding inductive attention to GPT-2. (3) We show through qualitative and quantitative evaluations that CGRG outperforms strong baselines where a) the control phrases are provided by a (simulated) user and b) automatically extracted by a content planner.

2 Controllable Models of Grounded Response Generation

We formalize the problem as follows: given dialogue context X, p lexical control phrases $C = (C_1, \cdots, C_p)$ and q sentences of grounding $G = (G_1, \cdots, G_q)$, generate a response $R = (r_1, \cdots, r_m)$ that contains semantic information guided by C. Control phrases can be either directly provided by a user or automatically derived from a content planner. To differentiate derived control phrases from gold or user-provided C, we denote these as $\tilde{C}$.

This new framework, called Controllable Grounded Response Generation (CGRG), assumes we have grounded conversational dataset, such as in (Qin et al., 2019). We assume that each data instance consists of a dialogue context, grounding knowledge and a reference response. To analyze this framework, we define a control mechanism that defines one or more control phrases for each instance. For more focus on grounding, our user controls are lexical phrases that are relevant to both target response and some part of grounding knowledge. Since it is costly and unscalable to have humans annotate the control phrases, we use lexical matching, defining control phrases to be informative n-grams that appear in both grounding and the reference response. Details of our dataset and its processing are presented in Section 3.

2.1 Extensions to GPT-2

GPT-2 is a transformer-based language model trained on large scale web data (Radford et al., 2019) and uses self-attention where each token attends to its left tokens. It is trained with the objective: predict the next word, given all of the previous words within a defined context window.

To apply GPT-2 within CGRG, we concatenate X, C (or $\tilde{C}$) and G_C to be our input sequence, as shown in Figure 3 (left). Then we have the model predict the next response word given the concatenated input sequence (denoted as S) and the previous response tokens in R. G_C is the subset of G that is relevant to C. For example, in this work, we denote the grounding sentences that contain any phrase in C as G_C. To differentiate the input elements, we insert an end-of-text token $\langle eos \rangle$ at the end of each dialogue utterance in X, a $\langle c \rangle$ token at the end of each control phrase in C and a $\langle s \rangle$ token at the end of each sentence in G_C.

We first concatenate the input sequence S and the response sequence R into a long text. We denote the source sequence as $S = (w_1, \cdots, w_n)$, which is used to generate target sentence R. The conditional probability of $P(R|S)$ can be written

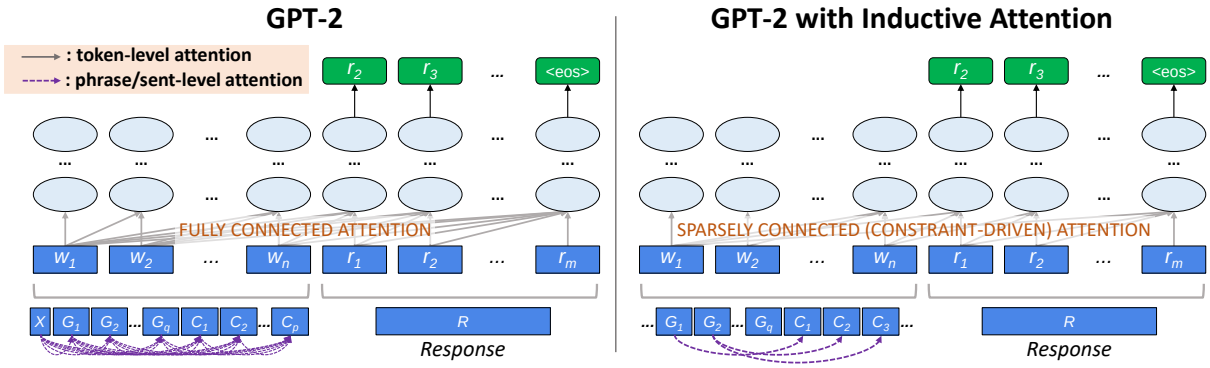

Figure 3: GPT-2 considers all possible forward attentions, which can overwhelm the model when the context contains context (X), grounding (G), and constraints (C). On the other hand, Inductive Attention helps focusing on attentions that are relevant to the constraints. Dashed arrows indicate which token-level attentions are preserved. Sparsely connected attention is implemented with a mask on all hidden layers.

as the product of conditional probabilities:

$$p(R|S) = \prod_{k=1}^{m+1} p(r_k|w_1, \cdots, w_n, r_1, \cdots, r_{k-1})$$

where r_{m+1} is the additional end-of-text token indicative of the end of generation.

2.2 GPT-2 with Inductive Attention

GPT-2 by default takes a consecutive text sequence as its input in order to train a language model. In our problem setting, we have each input element of X, C, G_C in a segmented format. Simply concatenating all these input elements into a GPT-2 model can induce noise, as some segments may not necessarily be strongly relevant, and we consider attention links between such segments to be uninformative.

We remove potentially uninformative attention links for each data example by injecting pre-established structural information between C and G_C. For example, in Figure 3 (right), say that C consists of C_1, C_2, C_3, and G_C consists of G_1 and G_2. If we know C_1 is only found in G_1, then we only want to keep the attention link between C_1 and G_1, and not between C_1 and any of the other grounded sentences. Since we think G_C is a set of segmented sentences from G, we remove all cross-sentence links within G_C tokens. Similarly, we remove all links between non-identical phrases. Thus, the attention links for each data example are pre-determined by structural information between C and G_C. To implement this, in each transformer layer, we apply attention masks where the removed attention links and links to future tokens have value 0 and the others have value 1. We refer to this pre-calculated attention as inductive attention. Each response token still attends to all input tokens and other response tokens on its left.

We denote the start and end positions of a control phrase $C_i \in C$ in S to be c_i^s and c_i^e and those of a grounding sentence $G_i \in G_C$ to be g_i^s and g_i^e. Then we calculate the attention mask M as follows:

$$M_{i,j} = \begin{cases} 0 & \text{if } i < j \\ 0 & \text{if } i \in [c_k^s, c_k^e], j \in [c_l^s, c_l^e], k \neq l \\ 0 & \text{if } i \in [g_k^s, g_k^e], j \in [g_l^s, g_l^e], k \neq l \\ 0 & \text{if } i \in [c_k^s, c_k^e], j \in [g_l^s, g_l^e], C_k \notin G_l \\ 1 & \text{otherwise} \end{cases}$$

Then for each transformer head, we have the stacked matrices Q, K and V to represent each example sequence (concatenated S and T) as in (Vaswani et al., 2017). We calculate the attention as follows (d is the model dimension):

$$\text{Attention}(Q, K, V) = \text{softmax}(\frac{M \circ QK^T}{\sqrt{d}})V$$

2.3 Content Planner

We experiment with two content planners in order to assess the effectiveness of our models when gold constraints are not provided by users. The first is a simple retrieval-based pipeline: for each test dialogue context, we (i) Rank the sentences in G by IDF-weighted word overlaps with X; (ii) Extract statistical phrases from the top 50 sentences; (iii) Obtain the 2 statistical phrases that appear most frequently in the 50 sentences as $\tilde{C}$. In order to reduce search space, we use noun phrases only. As there is no need to train such extraction pipeline, it is only applied during inference stage.

We also experiment with BERT QA as a content planner. We fine-tune a BERT QA model on our

training examples, with X as the query, G as the document and C as answers. Then we use the fine-tuned model to predict answers on test examples. We obtain the top 2 answers as predicted control phrases $\tilde{C}$ and drop the second if the string overlaps with the first.

3 Dataset

We use the grounded Reddit conversation dataset described in Qin et al. (2019), which features Reddit conversations about web pages such as news stories and Wikipedia articles, and covers diverse topics (178 subreddit topics ranging from news/technology to literature/music) and writing styles. As a social media aggregator, Reddit is akin to multiple datasets. In order to make this dataset support controllable text generation, we apply the following pipeline to extract control phrases: we match each n-gram ($n \leq 5$) in the reference response to each grounding sentence. In order to ensure certain informativeness of control phrases, we set an IDF threshold (8.5) for unigrams. When two n-grams are identical except for an added function word or punctuation, we use only the shorter version. In addition, we remove the matched n-grams that appear in dialogue context as we argue that new words are more informative. For each data instance, we have the remaining matched n-gram(s) as control phrases.

We use crowdsourced workers to annotate whether the extracted control phrases are central to the reference response, given the dialogue context. For each response, we had 3 judges to enter on a 1-6 scale and calculate the average score. In 2000 annotated examples, the median score was 4.33 and 67.4% of examples had a score over 4. Inter-rater agreement was "fair" with Krippendorff's alpha coefficient at 0.32. We keep only examples where at least one matched phrase can be found. Such strict lexical matching between target response and grounding assures that the kept examples have a high ratio of grounding utilization, which fits one focus of this work: leveraging grounding in response generation. After the processing, we reduce the number of utterances of train, dev and test from 2.36M, 0.12M and 0.34M to 390K, 6.7K and 21K respectively. And the average length of all reference responses increases from approximately 18.5 to 26.5. The average numbers of phrases in C for train, dev and test set are 1.32, 1.27 and 1.38 respectively. The average numbers of sentences in G_C for train, dev and test set are 4.37, 4.32 and 4.25 respectively. And we use up to 3 dialogue turns in experiments.

4 Experimental Setup

4.1 Training and Inference Setup

In our GPT-2 baseline and Inductive Attention (GPT2IA) models, we have both type and positional embedding for each input token. We treat X, each sentence in G_C, each phrase in C and response R as separate segments. We set the maximum number of sentences in G_C to be 20 and maximum number of phrases in C to be 10, then we have "0" for X; "1-20" for G_C; "21-30" for C and "31" for R tokens as type embedding. For each segment, we have the position embedding for each token as its position in that segment.

We use the small version of GPT-2 with 117M parameters, with the maximum length of the input or target response sequence to be 512. We use BPE tokenization, following GPT-2. We train our model and all other GPT-2-based baselines on top of DialoGPT (Zhang et al., 2020), which is a conversational response generation model trained on 147M Reddit comment chains on the basis of GPT-2. None of their Reddit training or validation examples overlap with our test examples. We use batch size 32. Learning rate and warmup steps are tuned on valid set.

We use greedy search as the decoding strategy for all GPT-2 and GPT2IA setups, except for a single experiment setting where grid beam search (GBS) (Hokamp and Liu, 2017) is applied for comparison with lexical constrained decoding. The goal of the comparison of our methods with GBS is to investigate whether it helps to encode the constraints into the hidden state during both training and inference, as GBS uses lexical constraints only during inference.

4.2 Evaluated Systems

We conduct experiments to draw insights from comparison of different response generation models and input settings. We evaluate our models according to the following settings:

X: This is the standard setting for non-controllable response generation, where only the dialogue context is given. We conduct experiments for the state-of-the-art generation model GPT-2.

X+G: This is the standard setting for grounded response generation. We compare two models: CMR

(Qin et al., 2019) and GPT-2. CMR is the state-of-the-art grounded response generation model that combines a MRC model and a LSTM decoder. GPT-2 for this setting concatenates X and G as its input. Note that as both models have input sequence length limit, only a randomly chosen subset of grounding sentences are fed into each model.

X+C: This is the controllable response generation setting without grounding. We conduct experiments for GPT-2 by concatenating X and C.

X+G_C: This setting measures how the grounding only relevant to C can help with response generation, without explicitly providing C. We conduct experiments for GPT-2, by concatenating X and G_C to be the input.

X+C+G_C: This setting measures how grounded control can help with response generation. We conduct experiments for GPT-2 and GPT2IA, by concatenating X, G_C and C to be the input.

X+C+G: This setting is for comparison against existing constrained generation methods like grid beam search (GBS) introduced in Hokamp and Liu (2017), where lexical control phrases are added in decoding only without involving training. We conduct experiments for GPT-2 where X and G are the only encoded inputs and C is only applied in decoding with GBS.

To provide more insight into experiment scores, we also evaluate human responses as a 'system'. This is possible because we are using multi-reference test set (Qin et al., 2019) with 3.3k unique test dialogue contexts. For each test dialogue context, we retain up to 6 references and set aside one of these for evaluation, so the "human response" can be evaluated against the remaining references for automatic evaluation. To ensure comparability, all systems are evaluated against the same 5 references. For each evaluation metric, we report the highest score among the 5 references.

4.3 Automatic Evaluation

We experiment with both user-controllable and automatic response generation, with gold and predicted control phrases from a content planner respectively. As different reference responses incorporate different gold control phrases, we use single-reference evaluation for the user-controllable setting. Predicted control phrases are independent of reference responses, so we can use multi-reference evaluation in the automatic generation setting.

For automatic evaluation, we measure the overall relevance of the generated responses with metrics including BLEU-4 (Papineni et al., 2002) and NIST-4 (Doddington, 2002). NIST is a variant of BLEU that weights n-gram matches by their information gain, which penalizes uninformative n-grams. We measure the diversity of n-grams in generated responses with the ratio between the number of distinct n-grams and the total number of n-grams. Previous works (Li et al., 2016b; Simeng Sun, 2019) has shown that automatic metrics for generation can sometimes be unreliable, and response generation generally achieves low absolute metric scores. Accordingly our main conclusions are based on human evaluations (Section 4.4). Nevertheless, we find that our automatic evaluation results comport well with our human evaluations.

In order to provide a sense of how control phrases help enforce the specificity level for generation, in the user-controllable setting, we report control phrase inclusion rate, namely the percentage of gold control phrases included in the generated responses. However, lower inclusion rate does not necessarily indicate worse performance in satisfying the user's control request, as we treat the lexical control phrases as soft semantic guidance in generation, rather than as hard constraints.

4.4 Human Evaluation

Human evaluation was conducted using crowdsourced workers. Relevance and appropriateness to the preceding dialog and consistency with the background text (as a metric of factual correctness) were measured. Judges were presented with paired randomized outputs from each system. Document title, a short snippet of the document and up to two conversational turns were provided as context. Judgments were entered on a five-point Likert scale, and ties were permitted. Three to four judges evaluated each pair and metrics were imposed to block poorly performing judges. Inter-rater agreement, was "fair" with Krippendorff's alpha coefficient at 0.32.[2]

5 Results and Analysis

5.1 Controllable Response Generation

We focus here on analyzing the user-controllable grounded response generation framework, using

[2]Sample sizes vary. The number was reduced from an initial 1,000 when we automatically removed a number of instances where egregiously offensive content rendered them inappropriate to display to judges.

Setting	Model	NIST	BLEU	Div-2	Avg-L	Incl.
1) X	GPT-2	0.90	0.55%	4.9%	22.2	–
2) X+G	CMR	0.34	0.17%	11.3%	15.1	–
3) X+G	GPT-2	0.98	0.67%	7.5%	23.1	–
4) X+C	GPT-2	1.67	2.65%	10.7%	28.7	69.4%
5) X+G_C	GPT-2	1.34	1.58%	11.1%	26.6	34.8%
6) X+C+G	GPT-2+GBS[3]	1.60	2.38%	10.6%	26.8	98.0%
7) X+C+G_C	GPT-2	1.77	3.22%	11.3%	27.0	65.1%
8) X+C+G_C	GPT2IA	**1.80**	**3.26**%	**11.6**%	25.9	63.5%

Table 1: **Controllable Response Generation** automatic evaluation (with user constraints).

GPT2IA	Tied	GPT-2	
Relevance: *Which response is more relevant and appropriate to the preceding dialog?*			
X+C+G_C	**69.8**%	14.1%	16.1%	X+C+G+GBS
X+C+G_C	**42.1**%	23.5%	34.4%	X+C
X+C+G_C	**38.1**%	28.6%	33.3%	X+C+G_C
Consistency: *Which response is more consistent with the grounding text?*			
X+C+G_C	28.1%	44.3%	27.6%	X+C+G_C
X+C+G_C	**37.6**%	31.4%	31.0%	X+C

Table 2: **Controllable Response Generation** human evaluation for relevance and background consistency, showing preferences (%). A number in bold indicates the system is significantly better at $p \leq 10^{-5}$ computed using 10k bootstrap replications.

Setting	Model	Content Planner	NIST	BLEU	Div-2
X	GPT-2	-	1.42	**1.31**%	18.1%
X+$G_{\tilde{C}}$	GPT-2	Retrieval-based	1.61	1.26%	19.4%
X+$\tilde{C}$+$G_{\tilde{C}}$	GPT2IA	Retrieval-based	**1.67**	1.23%	**20.2**%
X+$\tilde{C}$+$G_{\tilde{C}}$	GPT2IA	BertQA	**1.67**	1.26%	19.6%
Human	-	-	2.04	2.56%	62.8%

Table 3: **Response Generation** automatic evaluation (multi-references) using constraints from content planner. Note that results of Tables 1 and 3, as user constraints give away significant information about the intended response.

single-reference evaluation. In Table 1, lines 1-3 are not controllable settings and do not have control phrases as input, while lines 4-8 have control phrases as input, either explicitly or implicitly. The huge performance gap between lines (1-3) and (4-8) demonstrates the value of adding control.

More importantly, we can draw the following conclusions by comparing rows in Table 1: (i) **1 vs. 3**: Simply adding groundings to the model input improves the performance to a limited extent; (ii) **2 vs. 3**: GPT-2 in general performs better than the state-of-the-art grounded model CMR, indicating that the combination of pre-training and having a transformer-based decoder helps improve generation; (iii) **4 vs. 7-8**: providing constraint-sensitive grounding boosts performance compared to having all the grounding (iv) **5 vs. 7-8**: providing control phrases in an explicit way is important; (v) **6 vs. 7-8**: applying control in hidden states helps the model generate better quality responses than applying control at decoding only; (vi) **7 vs. 8**: inductive attention helps reduce noise and improve the performance of GPT-2.

Although the comparison between line **6 vs. 7-8** shows that applying control in hidden states is more effective than strict constraints at decoding, it is possible that controls at the training and decoding stages could be complementary. We leave investigation of methods of combining these for future research.

Human evaluation results in Table 2 show X+C+G_C+GPT2IA outperforms other systems, except in the case of Consistency, where there is no statistical difference between X+C+G_C+GPT2IA and X+C+G_C+GPT2, both grounded systems.

5.2 Content-Planned Response Generation

In a fully automatic conversation scenario, we propose to have a content planner predict control phrases in order to leverage our proposed framework for automatic response generation. Table 3 compares settings where no control phrases and predicted control phrases ($\tilde{C}$) are provided to the model. We observe that both the retrieval-based or BERT QA based content planner achieve good results in terms of NIST and Div-2. (These are the methods presented previously in Section 2.3.) We also provide the evaluation results on the carved out human response in Table 3, which indicates the upper bounds for this task. As described in Section 4.3, we conduct multi-reference evaluation for the predicted control phrases setting.

As an intermediate assessment of the content planner, we report the Precision, Recall and F1 of tokens in $\tilde{C}$ and $G_{\tilde{C}}$, with respect to reference responses (counts for stop-words and punctuation

Content Planner	C-P	C-R	C-F	G-P	G-R	G-F
Retrieval-based	13.8%	5.6%	7.2%	5.5%	21.8%	7.7%
BertQA	14.7%	4.8%	6.5%	5.0%	21.3%	7.1%
Human	24.4%	6.1%	8.6%	6.6%	17.2%	8.0%

Table 4: Response coverage of control phrase $\tilde{C}$ and associated grounding $G_{\tilde{C}}$ tokens.

[3] X+C+G (GBS) only takes X+G as the encoder input while C is seen at decoding only.

Dialogue Context	With "nihonium", Japanese scientists become first from an Asian country to name atomic element.
Control Grounding X+C+G_C +GPT2IA	**periodic table** ... The **periodic table** is a great legacy in chemistry ... I'm not sure if this is a good thing or not, but I'm pretty sure the **periodic table** is a great legacy in chemistry.
Control Grounding X+C+G_C +GPT2IA	**artificially** ... The **artificially** synthesized element has 113 protons in its nucleus ... I wonder if they will be able to name a chemical that **artificially** produces atomic elements.

Table 5: For the same dialogue context, GPT2IA generates varied responses given different control phrases.

tokens are removed) in Table 4. For each test dialogue context, we calculate the values for the reference response that gives the highest F1 score and report the average among all test examples for each metric. We notice that the retrieved-based content planner predicts slightly better quality phrases than BERT QA, while still worse than the gold control phrases from the carved out human response.

5.3 Qualitative Analysis

To understand how grounding knowledge assists generation, we plot the token-level probability (Figure 4) for both X+C and X+C+G_C systems. We intentionally select an example about an uncommon entity to eliminate the possibility that the knowledge is captured in pre-training. This figure shows the token-level probability of a potential response, given the dialogue context *Do you know the education background of the new faculty, Sam?*, control phrases *University of Toronto* and *neural networks*, and grounding sentences *Sam got his bachelor degree in Physics at University of Science and Technology of China. He spent 6 months at University of Tokyo in Japan as a visiting student, when he was a master student in Computer Science at University of Hong Kong from 2010-2012. And he finished his PhD at University of Toronto in Canada with his research focused on interpretability of neural networks on text generation in 2017.* The grounded model assigns higher probabilities to contextual words from grounding such as *graduated* and *thesis* as well as to factually correct entity tokens like *2017*. It assigns lower probability to factually incorrect tokens such as *economics*. These facts suggest that grounding knowledge can potentially help controllable generation: (i) contextualize control phrases; (ii) distinguish correct and incorrect facts.

Figure 5 illustrates another example to analyze the functions of control and grounding for generation. We list top 6 tokens after a partial response given the same dialogue context and grounding, and control phrase *Canada*. The ungrounded and non-controllable model gives equally distributed probabilities to commonly known American state names after *University of*. Adding grounding helps the model rank locations based on the background knowledge. Further adding controls helps the model locate the correct or intended answer.

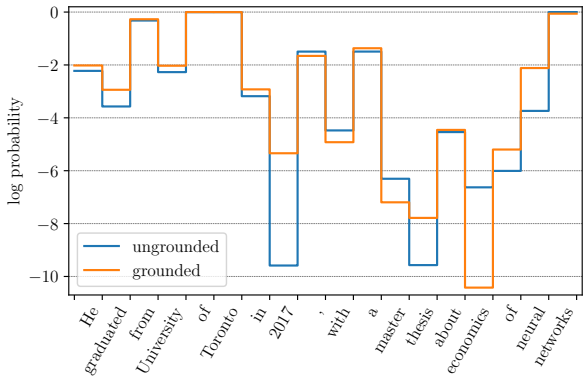

Figure 4: Sample showing our grounded model (X+C+G_C+GPT2IA) offers better discrimination against an ungrounded model (X+C+GPT2), given a document about a person's education background (constraint: *University of Toronto*; *neural networks*).

In order to quantify the observations in Figure 4 and Figure 5, we sample 100 test examples and randomly pick an entity from each reference response to calculate the entity's probability from each model. We restrict the entity to be not in control phrases. Then we calculate the average probability ratio for X+C/X+C+G_C and X+G/X+C+G_C, to be 0.773 and 0.886 respectively. Both of them are smaller than 1.0, which indicates having both grounding and control phrases gives higher probability to correct entities than having grounding or control phrases alone.

Explicit control phrases can be leveraged to dissect the generation process. Table 5 shows how

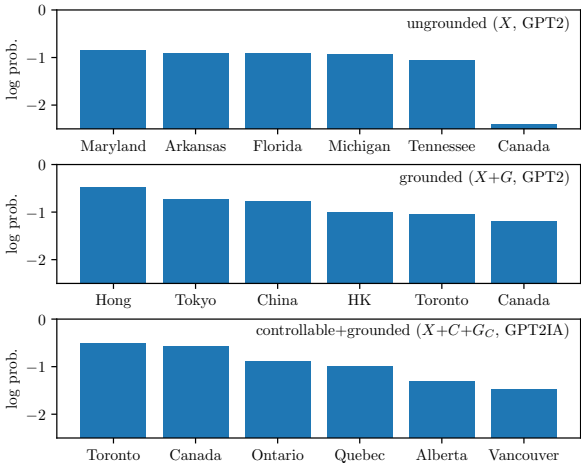

Figure 5: The top 5 tokens (plus *Canada*) to generated after the partial response *Sam just graduated from University of*. While the ungrounded model makes mostly generic predictions, the grounded model provides more topically relevant ones and the constraint further positively influences the hidden state.

controls may guide or perturb the GPT2IA model to produce responses with diverging semantics. And we provide more sample outputs of different systems in Table 6.

6 Related Work

6.1 Grounded Response Generation

Although some relevant work draws on external knowledge sources, none incorporates user control. Ghazvininejad et al. (2018) develop a memory network based model that leverages grounding information from Foursquare tips. Moghe et al. (2018), while Zhou et al. (2018) collect movie discussion datasets via crowdsourcing. These are limited to specific domains. Dinan et al. (2019) crowdsource conversations where each utterance is grounded in up to one single selected Wikipedia sentence. We focus on a more realistic, scalable setting, in which a response may constitute a blend of multiple grounding information pieces, rather than a single factual sentence rephrasing. Other researchers propose a copy mechanism to import tokens from both dialogue context and grounding (Yavuz et al., 2018) or leverage a reading comprehension model to co-encode dialogue context and grounding knowledge (Qin et al., 2019).

Other work incorporates relational knowledge bases (Zhu et al., 2017; Liu et al., 2018) or commonsense knowledge graphs (Young et al., 2018) to conversational models. More recently, Liu et al. (2019) develop a graph-path-based method on knowledge graphs augmented with unstructured grounding. Our present work focuses on text based grounding knowledge and does not require preconstructed knowledge graphs.

6.2 Controlled Generation

Prior work on machine translation and language generation has sought enforce user-specified constraints, primarily in the form of lexical constraints (Hokamp and Liu, 2017; Hu et al., 2019b,a; Miao et al., 2019). These approaches exploit constraints at inference time only; in our case, constraints are applied during training, with the option of also being applied at inference. Application during training enables the constraints to be incorporated into the latent space for better predictions.

Other related work (See et al., 2019; Keskar et al., 2019; Tang et al., 2019) have explored non-lexical constraints, but do not examine how these could facilitate use of grounding and external knowledge. We see this line of research as complementary to ours.[4]

Controllable text generation has also been employed in text style transfer (Hu et al., 2017) and other tasks (Ficler and Goldberg, 2017; Dong et al., 2017; Gao et al., 2019c), to disentangle high-level style information from contextual information such that the style information can be independently manipulated. (Zhao et al., 2018) uses discrete latent actions to learn an interpretable representation for task-oriented dialogue systems. While these works use "style" labels (e.g. positive/negative, formal/informal) as controlling signal, our framework controls generation with specific lexical constraints, allowing for fine-grained semantic control.

7 Conclusion

The CGRG framework allows users to inject soft semantic control into the generation process. It incorporates grounding to contextualize users' semantic intents as well as to boost information reliability. We introduce an inductive attention mechanism for self-attention-based generation models like GPT-2 in order to boost its performance. We also demonstrate that this framework can benefit standard automatic response generation when integrated with a content planner. Some interesting

[4]These papers also make the assumption that (gold) constraints can always given to the system, which limits the potential to demonstrate broader benefits of the approaches. To address this concern, we also evaluate our models in settings where *gold* constraints are unavailable (e.g., based on predicted constraints produced by a content planner).

Dialogue Context	More cops have died of illness linked to 9/11 than had perished in it on that day.
Control	**people died**; **elevators**
Grounding	... 1,402 **people died** at or above the floors of impact in the north **tower**. ... many **elevators** did not plunge, but were destroyed due to the crash and subsequent fires, or were stranded in the shafts. ...
X	I'm not sure if this is true, but I'm pretty sure it's true.
XC	I'm pretty sure the **elevators** were not meant to be elevated above the **elevators**.
XCG_C	I'm surprised that the number of **people died** in the **elevators** is low.
XCG_C +IA	I'm pretty sure the number of **people died** in the **elevators** is higher than the number of **people died** in the **towers**.
$X\tilde{C}G_{\tilde{C}}$ +IA	I'm pretty sure the 9/11 attacks were the most horrific of the world trade center attacks.
Dialogue Context	In 1997 a killer whale held a great white shark upside down for 15 minutes, causing it to suffocate, after which it ate only its liver
Control	**orca**
Grounding	... The **orca** held the **shark** upside down to induce tonic immobility and kept the **shark** still for fifteen minutes , causing it to suffocate ... The **orca** then proceeded to eat the dead shark 's liver ...
X	I think the article is about a great white **shark**.
XC	I'm not sure if this is the **orca** that killed the killer whale, but i think it's the killer whale that killed the killer whale.
XCG_C	I think the **orca** is the one who's the bad guy.
XCG_C +IA	I'm pretty sure the **orca** is the one who killed the **shark**.
$X\tilde{C}G_{\tilde{C}}$ +IA	I'm pretty sure that's a great white **shark**.
Dialogue Context	Millions of Puerto Ricans just lost power again after a line repaired by Whitefish Energy failed.
Control	**restored**
Grounding	... just 18% of the territory now has **power**, and officials did not have a clear timeline on when the **power** will be **restored**. ...
X	I'm not sure if this is true, but I'm pretty sure it's true.
XC	I'm not surprised. I'm surprised that the government is still trying to get the rest of the population **restored**.
XCG_C	I'm surprised they didn't just repair the line. I mean, it's not like they were going to be able to get back to the island they were supposed to be.
XCG_C +IA	I'm surprised they didn't just **restore power**.
$X\tilde{C}G_{\tilde{C}}$ +IA	I'm from San Juan and I'm pretty sure I'm not the only one that noticed this.
Dialogue Context	76 % of all known serial killers in the 20th century were from the United States.
Control	**law enforcement**
Grounding	... and may include more stringent policies for military personnel in **law enforcement** or security ... Should the cases cross multiple jurisdictions, the **law enforcement** system in **the US** is fragmented and thus not configured to detect multiple similar murders across a large geographic area ...
X	I'm pretty sure **the US** had a police force of around 100,000 people.
XC	I'm pretty sure the USA had a large number of serial killers in **the US**. I'm sure the USA had a large number of **law enforcement** officers in **the US**.
XCG_C	I'm not sure if this is true, but I'm pretty sure that **the US** has a lot of **law enforcement** officers that are from **the US**.
XCG_C +IA	I'm not sure if this is true, but I'm pretty sure that the **law enforcement** in **the US** is not very good at **detecting** serial killers.
$X\tilde{C}G_{\tilde{C}}$ +IA	I'm pretty sure that the USA has a large population of female serial killers.

Table 6: Sample outputs of the systems, with baseline outputs for comparison.

future directions include exploring various types of user desired control and extending the controllable grounded generation concept to broader generation tasks like document writing assistance.

8 Acknowledgements

We thank members of Microsoft Research and University of Washington's NLP groups who provided feedback and insights to this work.

References

Emily Dinan, Stephen Roller, Kurt Shuster, Angela Fan, Michael Auli, and Jason Weston. 2019. Wizard of Wikipedia: Knowledge-powered conversational agents. In *ICLR*.

George Doddington. 2002. Automatic evaluation of machine translation quality using n-gram cooccurrence statistics. In *Proc. of HLT*.

Li Dong, Shaohan Huang, Furu Wei, Mirella Lapata, Ming Zhou, and Ke Xu. 2017. Learning to generate product reviews from attributes. In *Proc. of EACL*.

Jessica Ficler and Yoav Goldberg. 2017. Controlling linguistic style aspects in neural language generation. *Proc. of EMNLP*.

Jianfeng Gao, Michel Galley, and Lihong Li. 2019a. Neural approaches to conversational AI. *Foundations and Trends in Information Retrieval*, 13(2-3):127âĂŞ298.

Xiang Gao, Sungjin Lee, Yizhe Zhang, Chris Brockett, Michel Galley, Jianfeng Gao, and Bill Dolan. 2019b. Jointly optimizing diversity and relevance in neural response generation. In *Proc. of NAACL*.

Xiang Gao, Yizhe Zhang, Sungjin Lee, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2019c. Structuring latent spaces for stylized response generation. In *Proc. of EMNLP*.

Marjan Ghazvininejad, Chris Brockett, Ming-Wei Chang, Bill Dolan, Jianfeng Gao, Wen tau Yih, and Michel Galley. 2018. A knowledge-grounded neural conversation model. In *Proc. of AAAI*.

Chris Hokamp and Qun Liu. 2017. Lexically constrained decoding for sequence generation using grid beam search. In *Proc. of ACL*.

Ari Holtzman, Jan Buys, Maxwell Forbes, and Yejin Choi. 2020. The curious case of neural text degeneration. In *ICLR*.

J. Edward Hu, Huda Khayrallah, Ryan Culkin, Patrick Xia, Tongfei Chen, Matt Post, and Benjamin Van Durme. 2019a. Improved lexically constrained decoding for translation and monolingual rewriting. In *Proc. of NAACL*.

J. Edward Hu, Rachel Rudinger, Matt Post, and Benjamin Van Durme. 2019b. ParaBank: Monolingual bitext generation and sentential paraphrasing via lexically-constrained neural machine translation. In *Proc. of AAAI*.

Zhiting Hu, Zichao Yang, Xiaodan Liang, Ruslan Salakhutdinov, and Eric P. Xing. 2017. Toward controlled generation of text. In *Proc. of ICML*.

Nitish Shirish Keskar, Bryan McCann, Lav R. Varshney, Caiming Xiong, and Richard Socher. 2019. CTRL: A conditional transformer language model for controllable generation. *Computing Research Repository*, arXiv:1909.05858. Version 2.

Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2016a. A diversity-promoting objective function for neural conversation models. In *Proc. of NAACL*, pages 110–119.

Jiwei Li, Michel Galley, Chris Brockett, Georgios Spithourakis, Jianfeng Gao, and Bill Dolan. 2016b. A persona-based neural conversation model. In *Proc. of ACL*.

Shuman Liu, Hongshen Chen, Zhaochun Ren, Yang Feng, Qun Liu, and Dawei Yin. 2018. Knowledge diffusion for neural dialogue generation. In *Proc. of ACL*.

Zhibin Liu, Zheng-Yu Niu, Hua Wu, and Haifeng Wang. 2019. Knowledge aware conversation generation with explainable reasoning over augmented graphs. In *Proc. of EMNLP*.

Ning Miao, Hao Zhou, Lili Mou, Rui Yan, and Lei Li. 2019. CGMH: constrained sentence generation by metropolis-hastings sampling. In *Proc. of AAAI*.

Nikita Moghe, Siddhartha Arora, Suman Banerjee, and Mitesh M. Khapra. 2018. Towards exploiting background knowledge for building conversation systems. In *Proc. of EMNLP*.

Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: a method for automatic evaluation of machine translation. In *Proc. of ACL*.

Lianhui Qin, Michel Galley, Chris Brockett, Xiaodong Liu, Xiang Gao, Bill Dolan, Yejin Choi, and Jianfeng Gao. 2019. Conversing by reading: Contentful neural conversation with on-demand machine reading. In *Proc. of ACL*.

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

Abigail See, Stephen Roller, Douwe Kiela, and Jason Weston. 2019. What makes a good conversation? how controllable attributes affect human judgments. In *Proc. of NAACL*.

Lifeng Shang, Zhengdong Lu, and Hang Li. 2015. Neural responding machine for short-text conversation. In *Proc. of ACL-IJCNLP*.

Ani Nenkova Simeng Sun. 2019. The feasibility of embedding based automatic evaluation for single document summarization. In *Proc. of EMNLP*.

Alessandro Sordoni, Michel Galley, Michael Auli, Chris Brockett, Yangfeng Ji, Margaret Mitchell, Jian-Yun Nie, Jianfeng Gao, and Bill Dolan. 2015. A neural network approach to context-sensitive generation of conversational responses. In *Proc. of NAACL-HLT*.

Jianheng Tang, Tiancheng Zhao, Chenyan Xiong, Xiaodan Liang, Eric P. Xing, and Zhiting Hu. 2019. Target-guided open-domain conversation. In *Proc. of ACL*.

Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Ł ukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in Neural Information Processing Systems 30*.

Oriol Vinyals and Quoc Le. 2015. A neural conversational model. In *Proc. of ICML Deep Learning Workshop*.

Semih Yavuz, Abhinav Rastogi, Guan-Lin Chao, and Dilek Hakkani-Tur. 2018. DeepCopy: Grounded response generation with hierarchical pointer networks. In *Proc. of NeurIPS Conversational AI Workshop*.

Tom Young, Erik Cambria, Iti Chaturvedi, Minlie Huang, Hao Zhou, and Subham Biswas. 2018. Augmenting end-to-end dialogue systems with commonsense knowledge. In *Proc. of AAAI*.

Rowan Zellers, Ari Holtzman, Hannah Rashkin, Yonatan Bisk, Ali Farhadi, Franziska Roesner, and Yejin Choi. 2019. Defending against neural fake news. In *NeurIPS*.

Yizhe Zhang, Siqi Sun, Michel Galley, Yen-Chun Chen, Chris Brockett, Xiang Gao, Jianfeng Gao, Jingjing Liu, and Bill Dolan. 2020. DialoGPT: Large-scale generative pre-training for conversational response generation. In *ACL demo paper*.

Tiancheng Zhao, Kyusong Lee, and Maxine Eskenazi. 2018. Unsupervised discrete sentence representation learning for interpretable neural dialog generation. In *Proc. of ACL*.

Kangyan Zhou, Shrimai Prabhumoye, and Alan W. Black. 2018. A dataset for document grounded conversations. In *Proc. of EMNLP*.

Wenya Zhu, Kaixiang Mo, Yu Zhang, Zhangbin Zhu, Xuezheng Peng, and Qiang Yang. 2017. Flexible end-to-end dialogue system for knowledge grounded conversation. *arXiv:1709.04264*.