

Policy-Driven Neural Response Generation for Knowledge-Grounded Dialog Systems

Behnam Hedayatnia
Amazon Alexa AI
behnam@amazon.com

Karthik Gopalakrishnan
Amazon Alexa AI
karthgop@amazon.com

Seokhwan Kim
Amazon Alexa AI
seokhkw@amazon.com

Yang Liu
Amazon Alexa AI
yangliud@amazon.com

Mihail Eric
Amazon Alexa AI
mihaeric@amazon.com

Dilek Hakkani-Tür
Amazon Alexa AI
hakkanit@amazon.com

Abstract

Open-domain dialog systems aim to generate relevant, informative and engaging responses. In this paper, we propose using a dialog policy to plan the content and style of target, open-domain responses in the form of an action plan, which includes knowledge sentences related to the dialog context, targeted dialog acts, topic information, etc. For training, the attributes within the action plan are obtained by automatically annotating the publicly released Topical-Chat dataset. We condition neural response generators on the action plan which is then realized as target utterances at the turn and sentence levels. We also investigate different dialog policy models to predict an action plan given the dialog context. Through automated and human evaluation, we measure the appropriateness of the generated responses and check if the generation models indeed learn to realize the given action plans. We demonstrate that a basic dialog policy that operates at the sentence level generates better responses in comparison to turn level generation as well as baseline models with no action plan. Additionally the basic dialog policy has the added benefit of controllability.

1 Introduction

Open-domain dialog systems have typically been modeled using end-to-end approaches, more specifically encoder-decoder architectures (Sordoni et al., 2015; Serban et al., 2017, 2016; Vinyals and Le, 2015). These seq2seq models are commonly trained on a maximum likelihood objective, which leads to repetitive and uninformative responses (Wei et al., 2017). As seen in Figure 1, *candidate A* is a typical generic response given the dialog context. In order to deal with this problem, previous work proposed grounding generated responses on knowledge sentences related to the

...
Speaker 1: Right. Teams do all kinds of things to bother the competition. I've heard of teams having heated benches in the winter for themselves but not for the visitors.

Speaker 2: I would hate a cold bench. Then again, I wouldn't want to be some place that cold or watching football.

Speaker 1:

candidate A: yeah

knowledge:

The NFL has no official rule against female players.

candidate B:

I heard NFL has no official rule against female players.

candidate C:

Yeah. I would hate that too. Do you follow NFL? I heard they have no official rule against female players.

Figure 1: *candidate A* is an uninformative response. By grounding on knowledge we get more informative responses i.e., *candidates B* and *C*. *candidate B* contains only a statement, leading to an abrupt topic transition. *candidate C* smoothly transitions topics with dialog acts: feedback, statement, question, and statement.

dialog context (Ghazvininejad et al., 2018; Yavuz et al., 2019; Zhou et al., 2018; Dinan et al., 2018; Gopalakrishnan et al., 2019). To improve the diversity of generated responses, others proposed conditioning response generation on latent (Serban et al., 2016, 2017; Shen et al., 2017; Zhao et al., 2017; Xing et al., 2016) or discrete attributes (Sankar and Ravi, 2019; Li et al., 2016a; See et al., 2019; Serban et al., 2017). These discrete attributes are typically presented to the decoder at the turn level, and are not associated with a specific segment of the output.

Another issue with seq2seq approaches is that, due to the lack of explicit control mechanisms, the style of these responses does not always match with what would be suggested by user experience experts. For example, the generated response may not acknowledge what the user just said, or may

jump to a new topic without first introducing it. Figure 1 shows examples of two response candidates with similar content: *candidate C* acknowledges Speaker 2’s previous statement and follows up with a question introducing a new topic and statement, in contrast with *candidate B* which abruptly transitions into the new topic.

According to Schegloff (2007) human conversations are sequentially organized units. Turns and actions realized within them are related to what came before and affect what comes next. Inspired by the previous studies, we propose a policy-driven neural response generation (PD-NRG) approach for open-domain, knowledge-grounded dialog systems. Our motivation for this work is to have a mechanism for open domain conversational systems, i.e., a dialog policy, that can enable such higher-level control of generated responses. The dialog policy provides a sequential organization plan or *action plan*. The *action plan* specifies the order and relationship of sentences within a turn targeting engaging responses to users throughout the interaction.

We design a set of dialog policy models that adapt to the dialog context to appropriately control the responses at both the turn and sentence-levels. We extend the end-to-end-approach of (Dinan et al., 2018; Gopalakrishnan et al., 2019): we take in as input both the dialog context and an action plan to predict the next response. We train our PD-NRG model by fine-tuning on the Generative Pre-trained Transformer (GPT) (Radford et al., 2018) model in a TransferTransfo fashion (Wolf et al., 2019). Our approach differs from previous works that condition on discrete attributes independently by conditioning on these attributes jointly.

Our contributions include:

- i. an enriched version of the Topical-Chat dataset with annotations on multiple attributes (knowledge, topic, dialog act). These annotations were tagged automatically which reduces the cost and time of manual annotation while still obtaining strong results.¹
- ii. the design of a basic dialog policy to predict an action plan for controllable generation for neural response generators
- iii. a sentence-based generation approach that outperforms turn-level generation, and

- iv. investigation of simple hand-crafted policies as well as automatically learned policies that could be adapted to new applications.

2 Related Work

Controllability of generated output has been studied for multiple language generation tasks (such as poetry generation and summarization). Previous work on controlling the style and content of generated outputs focused on two main approaches, conditional generation and weighted decoding. Conditional generation modifies the input to the model to condition on control parameters. Previous works proposed conditioning response generators on latent (Serban et al., 2016, 2017; Shen et al., 2017; Zhao et al., 2017) or discrete attributes, including dialog acts (Sankar and Ravi, 2019), sentiment (Sankar and Ravi, 2019), speaker identifiers (Li et al., 2016a), lexical features (See et al., 2019) or topics (Serban et al., 2017).

Weighted decoding (See et al., 2019) instead uses token-level features that are controllable (Ghazvininejad et al., 2017; Baheti et al., 2018) and supplements the scores from the decoder model output with these features. Our work focuses on conditional generation methods with sentence-level control, as described in more detail in Section 4.

There is also previous work on controlling attributes such as question asking at the dialog level. See et al. (2019) initialized the generation of turns of a dialog with a fixed distribution that specified what percentage of generated turns should include questions during the dialog. However this does not allow for flexible control where the number of questions may need to vary depending on the course of the dialog. Therefore, we focus on learning a dialog policy model that automatically learns the style of the response based on the dialog context.

Similar to previous work for response generation we ground our generated responses on knowledge. Ghazvininejad et al. (2018), Yavuz et al. (2019), and Zhou et al. (2018) used end-to-end memory networks, copy mechanisms and static graph attention mechanisms respectively to incorporate knowledge. Dinan et al. (2018), Gopalakrishnan et al. (2019), and (Roller et al., 2020; Smith et al., 2020) used memory networks based on transformer architectures (Vaswani et al., 2017) to encode knowledge sentences and dialog history to decode a response.

¹<https://github.com/alexa/Topical-Chat/tree/master/TopicalChatEnriched>

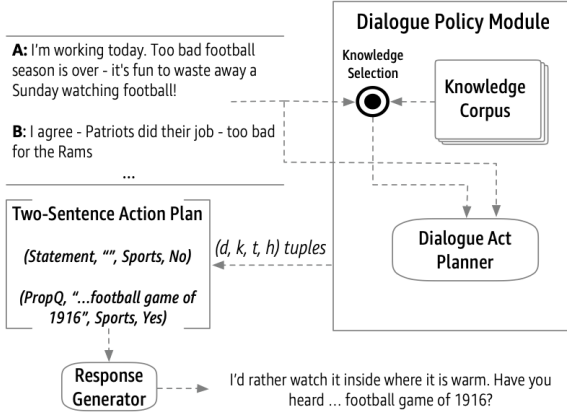


Figure 2: Policy-driven neural response generation.

There has been previous work on task-oriented systems that proposed explicit content and sentence planning (Walker et al., 2007) to further control the content and order of sentences within the generated response. Previous work for open-domain dialog systems also followed a similar method for content and sentence planning. Fang et al. (2018), Ahmadvand et al. (2018), Fulda et al. (2018), Pichl (2018), Cervone et al. (2017), Yu et al. (2019) and Bowden et al. (2019) extracted multiple features such as topic, intent, entities, and sentiment to send to a dialog policy model to plan the structure and content of the response. However, these previous works generated responses from a set of templates that are usually repetitive for open-domain conversations. Our work focuses on neural generative models for response generation in open-domain dialog systems.

The closest work to ours in terms of learning a dialog policy for open-domain dialog is (Xu et al., 2018) who designed a policy network to predict dialog acts and fed those acts into a response generation model to control responses. However, a key part of open-domain dialog is to introduce knowledge into a conversation. We design a policy that integrates knowledge with dialog acts at a sentence-level. In contrast to (Xu et al., 2018) that used a machine learning based approach, we show that a basic rule-based dialog policy can result in strong performance.

3 Dialog Policy

Our proposed PD-NRG approach has two parts: a *dialog policy* that determines the action plan based on the dialog context, and a *response generation* model that takes the action plan and the dialog context as input to generate a response. The

dialog policy has components that predict the individual elements of the action plan: *knowledge selection* and *dialog act planning*. Knowledge selection determines the knowledge to be integrated in the response by finding sentences from a knowledge document corpus that are relevant to the dialog context. Dialog act (DA) planning determines the style of the response in the form of DAs to be realized. We have two forms of DA planning methods: *Knowledge-dependent DA planning* and *Knowledge-independent DA planning*. Figure 2 depicts the architecture of PD-NRG.

Dialog Act	Definition
Apology	apology
ChoiceQ	Or-question
Commissive	Offer, Commit
Directive	Open-Option, Suggest
Feedback	Acknowledge
PropQ	Yes-no-question
Salutation	bye, greet
SetQ	Wh-question
Statement	Inform
Thanking	thanking, your-welcome

Table 1: The subset of ISO-Standard dialog acts proposed by (Mezza et al., 2018).

3.1 Action Plan (AP)

For the rest of this work, let $D_j = [x_1, \dots, x_j]$ denote a partial dialog containing a sequence of j turns. And let x_i represent a turn in a dialog where $1 \leq i \leq j$. Each x_i contains a sequence of n_i sentences, $x_i = [s_i^1, \dots, s_i^{n_i}]$.

Each x_i is generated according to an action plan that consists of one frame for each sentence $[f_i^1, \dots, f_i^{n_i}]$. The frames, formed of attributes and values, may include:

1. Dialog acts (d) at a sentence-level to help control the style of the generated response. Table 1 lists all the dialog acts used in this work.
2. Topics (t) at a turn-level to generate topically coherent responses. The complete list of topics are: fashion, politics, books, sports, general-entertainment, music, science & technology and movies.
3. Knowledge (k) at a turn or sentence-level to generate interesting and informative responses. The knowledge is represented as a sentence drawn from an unstructured knowledge corpus.
4. Use-knowledge flag (h) that signals whether or not to use the knowledge attribute (k) at the turn or sentence-level.

Each frame in the action plan corresponds to a sentence s_j^m and is denoted as a tuple containing

a set of the 4 attributes, $(d_j^m, t_j^m, k_j^m, h_j^m)$ where $1 \leq m \leq n_j$. In this work, we focus on these attributes for action plans, as they are the most basic and critical ways to control knowledge-grounded response generation.

3.2 Knowledge Selection

For the knowledge selection component of our dialog policy, referenced in Figure 2, we compute the following for each turn x_i at run time. Let c_i be defined as the dialog history x_1, \dots, x_{i-1} :

$$\hat{k} = \underset{k_m \in K}{\operatorname{argmax}} \left(\frac{\vec{c}_i \cdot \vec{k}_m}{\|\vec{c}_i\| \|\vec{k}_m\|} \right) \quad (1)$$

k_m is a knowledge sentence from an unstructured knowledge corpus, K , in the Topical-Chat dataset (Gopalakrishnan et al., 2019). We use the BM25 model (Robertson et al., 2009) to rank knowledge sentences and represent \vec{c}_i and \vec{k}_m as vectors for c_i and k_m . We compute cosine similarity between the vectors and *argmax* over the all k_m in our knowledge corpus. For c_i we are only using the most recent previous turn x_{i-1} for selection. We decide to use the knowledge sentence as input if the similarity score between the sentences is above a manually set threshold value of 0.2

3.3 Dialog Act Planning

For dialog act planning, we define a set of dialog act transitions from common examples in the Topical-Chat corpus. The set of dialog acts for the next response are determined by both dialog acts and the knowledge sentence selected, based on the dialog context. Figure 2 shows the output of the knowledge selection being fed as input into the dialog act planning component. We represent the transitions as a decision tree². In Figure 5, Speaker 2’s response is a *PropQ* act and from our decision tree we will predict the dialog acts for the next response, i.e. *Statement*, *PropQ*. Based on which set of dialog acts were outputted, we decide whether or not to include the knowledge sentence. Some dialog acts, such as *Feedback*, do not need to include knowledge by definition.

3.3.1 Knowledge-dependent DA Planning

We propose a Knowledge-dependent DA planning (KD-DA-P) where there are two inputs to predict the dialog acts for the next turn x_{j+1} :

²The full set of decision trees are presented here in the appendix <https://arxiv.org/pdf/2005.12529v4.pdf>

- the last dialog act associated with the previous sentence $s_j^{n_j}$
- the output of knowledge selection

The dialog act planner looks at the output of the knowledge selection model to see if the knowledge selected is the same or different as compared to the knowledge sentence selected for the previous turn x_j . Based on this information a certain subset of the transitions defined for dialog act planning are used to predict the dialog acts for the next response.

3.3.2 Knowledge-independent DA Planning

The prediction of the dialog acts is done independently of the selected knowledge in four ways:

1. *Simple DA planning*: We define a set of transitions that determine the set of DAs for the next response based solely on the previous dialog act.
2. *Seq2Seq Model for DA planning*: Using the OpenNMT library (Klein et al., 2017), we train a sequence-to-sequence model based on bi-directional LSTMs with Luong attention (Luong et al., 2015) to estimate the DAs of the current turn given the dialog context D_j . During training, each dialog act label is a separate token in the vocabulary and has its own embedding vector. Both the dialog act and word embeddings are initialized randomly and learned during training.
3. *PropQ DA planning*: For comparison to previous work we use the method in (See et al., 2019) which initializes the distribution of questions to be asked at the beginning of the conversation. The work finds that the best model generates questions 65.7% of the time. At each time-step the PropQ dialog act is picked 65.7% of the time thereby replicating this baseline. As shown in Table 1 PropQ corresponds to a Yes-No question, which is the most represented question dialog act in our dataset.
4. *AllQ DA planning*: We extend the PropQ DA Prediction baseline above by selecting the PropQ, ChoiceQ or SetQ questions each 21.9% of the time summing up to 65.7%. See et al. (2019) does not make a distinction as to what type of questions were asked.

4 Policy-driven Response Generation

As shown in Figure 2, at a given turn in the dialog context, the goal of the response generator is to

realize the action plan output by the dialog policy. Our proposed models generate the next turn based on the action plan at a sentence-level, in a sequential manner as opposed to at a turn-level. As shown in Figure 3a when decoding each sentence of the next turn, the dialog context D_j as well as the previous sentences generated for the next turn till that iteration are used as input. Algorithm 1 shows the process for sentence-level generation. As seen in the algorithm all the attributes within the AP are jointly taken in as input. To jointly condition on the action plan, each attribute is concatenated to the dialog history as shown in Figure 3c. In the training process each dialog act label is a separate token in the vocabulary and has its own embedding vector which is initialized randomly and learned during training. To train our model we represent the knowledge sentence and topic label with the pretrained embeddings from the GPT model whose vocabulary is BPE tokenized. Finally, the use-knowledge flag decides whether or not to include the knowledge embeddings as part of the input. In some of our experiments, we also include the dialog acts for the past turns by concatenating each turn in the dialog history with its respective acts.

Algorithm 1: Sentence-level generation

Result: x_{j+1}
 Given D_j
 $x_{j+1} = []$
 $ActionPlan = [f_{j+1}^1, \dots, f_{j+1}^{n_{j+1}}]$
 for f in $|ActionPlan|$
 $y = Model(D_j, f)$
 $x_{j+1} = x_{j+1} \oplus y$
 $D_j = D_j \oplus y$
 return x_{j+1}

For all our models, we use the GPT (Radford et al., 2018) model to finetune in a TransferTransfo (Wolf et al., 2019) fashion. The TransferTransfo model is a state-of-the-art neural open-domain dialog system that won 1st place in automated evaluation and 2nd place in human evaluation at the NeurIPS ConvAI2 conversational Intelligence Challenge (Dinan et al., 2020). We have two methods to generate responses from our models:

- **Model for turn-level generation:** As depicted in Figure 3b, our baseline (Wolf et al., 2019) is given the dialog context and knowledge sentence as input and predicts the response at the turn-level.

- **Models for sentence-level generation:** As depicted in Figure 3c, the PD-NRG models are given the AP and the dialog context as input to perform sentence-level prediction. Table 2 lists the versions of PD-NRG models we experimented with along with their corresponding APs. Baseline-Sent is similar to the Baseline-Turn model, except it generates responses sentence-by-sentence. The model generates as many sentences as in the human response.

PD-NRG Models	Action Plan (AP)
w/ DA	$\{d_{j+1}^m, k_{j+1}\}$
+knowledge flag	$\{d_{j+1}^m, k_{j+1}^m, h\}$
+knowledge flag +topic	$\{d_{j+1}^m, k_{j+1}^m, t_{j+1}, h\}$
Baseline Models	
Baseline-Turn	$\{k_{j+1}\}$
Baseline-Sent	$\{k_{j+1}\}$

Table 2: Models and their input AP for every timestep m where $1 \leq m \leq n_{j+1}$.

5 Experiments and Evaluation

5.1 Dataset

We use the publicly released Topical-Chat³ dataset, a large and diverse knowledge-grounded open-domain dialog dataset where the underlying knowledge spans 8 broad topics including fashion, books, and so on (Gopalakrishnan et al., 2019). Each dialog contains 20+ turns alternating between two crowd workers. For each dialog there is a reading set for each crowd worker. Each reading set has three entities and a set of corresponding knowledge sentences. When presenting the results, we use both test sets provided with the corpus, test frequent and test rare. Frequent and rare refer to the frequency of topics and entities being discussed in the training set.

5.2 Annotating Attributes in Topical-Chat

The dataset does not have annotations for some attributes such as dialog acts or fine-grained associations between knowledge sentences and dialog turns. Hence, we used out-of-the-box or simple models to automatically annotate our dataset with each attribute, as defined in Section 3.1. We assume these annotations are the ground-truth attributes for the ground-truth AP and use them for testing controllability without degrading the response appropriateness. By automatically annotating we reduce

³<https://github.com/alexa/Topical-Chat>

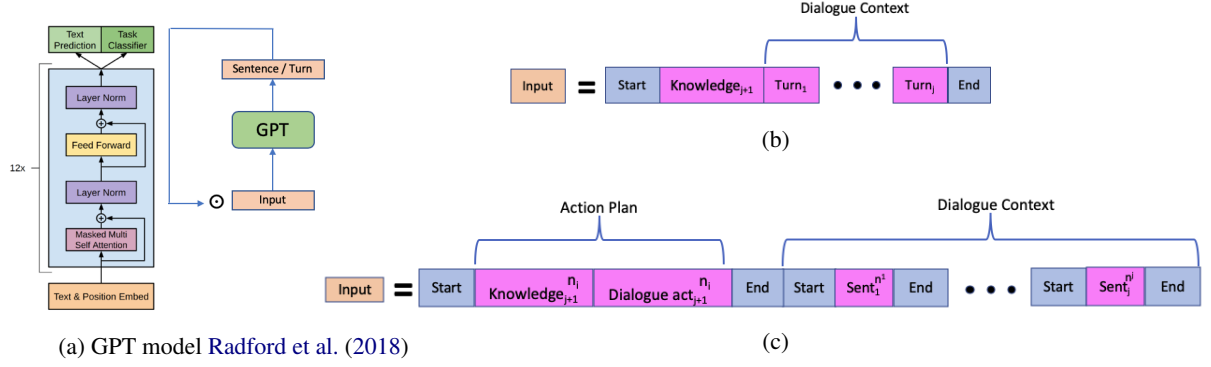


Figure 3: Figure 3a shows the generation process where the input is fed into the GPT model. The output is then concatenated back to the input. This process repeats until generation is complete. Figures 3b and 3c show the input for Baseline-Turn Model and PD-NRG model respectively.

the cost and time it takes to manually annotate our dataset along with getting strong results.

5.2.1 Annotating Knowledge Sentences

Each conversation in Topical-Chat has a pair of reading sets that were presented to crowdworkers before the conversation, to have a knowledgeable interaction. During their conversation crowd workers are asked to annotate which topics/entities were attributed to their turns in the conversation. However there is no fine-grained annotation of which knowledge sentence or sentences were used for a turn, hence we create ground-truth knowledge annotations as a corpus post-processing step. To obtain the knowledge annotation for each turn we use Equation 1 to compute similarity between x_{j+1} and k_m . To obtain the knowledge annotation for each sentence within a turn, we tokenize the turn into individual sentences. For each sentence we use the same equation to compute similarity between $s_{j+1}^{n_i}$ and k_m . For sentence-tokenization we use the NLTK library (Loper and Bird, 2002).

We decide whether or not the turn or sentences within a turn should be linked to a knowledge sentence by manually setting a threshold value on the similarity score between the knowledge and turn or sentences within a turn. We use the same threshold, 0.2, as described in Section 3.2.

5.2.2 Annotating Dialog Acts

We obtain the dialog acts for each sentence by running an off-the-shelf SVM dialog act tagger⁴ (Mezza et al., 2018) which takes in as input the current sentence to predict one of 11 dialog acts listed in Table 1. We also experimented with using

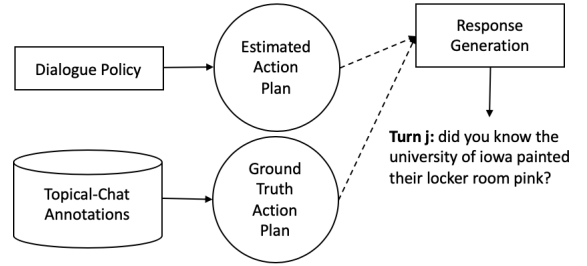


Figure 4: We calculate automated metrics with both a ground truth and an estimated AP

past dialog acts predicted from the tagger as additional input; however, this did not change the result. If the confidence score from the SVM tagger is not above a threshold of 0.5, the tagger would output no dialog act which we denote with a special dialog act token *NoDialogAct*. 2.1% of sentences within the Topical-Chat dataset were labeled as *NoDialogAct*. Of the 11 dialog acts the most represented ones were *Statement*, *PropQ* and *Feedback* where each act had 80%, 6% and 5% sentences tagged respectively. We assume these are the ground-truth dialog acts in our dataset. To view the performance of the model we ask two crowd workers to segment and annotate a small set of 100 turns into individual sentences along with their respective dialog act. The dialog act tagger obtained an F1 of 0.54, precision of 0.77 and a recall of 0.59 on consolidated test set.

5.2.3 Annotating Topic Labels

For the topic label, we use the topic annotations by the Turkers from the original Topical-Chat data collection. For each turn there are multiple topic annotations; however, unlike the dialog acts and knowledge sentence, topic annotations are at the turn level and are not linked to individual sentences.

⁴<https://github.com/ColingPaper2018/dialogAct-Tagger>

Models	Past DA	PPL	BLEU-4@1	ROUGE-L	Avg #	
					words	sentences
Human		- / -	- / -	- / -	24.3 / 25.0	2.10 / 2.15
Baseline-Turn		12.92 / 13.53	0.024 / 0.028	0.134 / 0.130	20.7 / 21.7	1.87 / 1.93
Baseline-Sent		13.85 / 14.36	0.016 / 0.021	0.107 / 0.103	13.7 / 13.9	2.09 / 2.15
PD-NRG w/ DA		12.72 / 13.01	0.024 / 0.027	0.121 / 0.118	18.5 / 19.3	2.05 / 2.10
PD-NRG w/ DA	✓	12.39 / 12.80	0.021 / 0.021	0.115 / 0.111	16.0 / 15.8	1.77 / 1.77
+knowledge flag		12.66 / 12.99	0.025 / 0.027	0.122 / 0.118	17.3 / 18.1	2.03 / 2.08
+knowledge flag	✓	12.25 / 12.62	0.019 / 0.020	0.113 / 0.108	15.2 / 15.3	1.68 / 1.76
+knowledge flag +topic		12.76 / 13.07	0.023 / 0.026	0.123 / 0.117	16.8 / 18.2	2.10 / 2.14
+knowledge flag +topic	✓	12.28 / 12.65	0.019 / 0.020	0.115 / 0.109	16.3 / 16.7	1.82 / 1.85
Corpus Diversity						
		F1	Precision	Recall	n=1	n=2
Human		- / -	- / -	- / -	0.037 / 0.050	0.266 / 0.326
Baseline-Turn		0.249 / 0.253	0.275 / 0.272	0.229 / 0.236	0.018 / 0.027	0.118 / 0.165
Baseline-Sent		0.220 / 0.220	0.258 / 0.252	0.191 / 0.195	0.018 / 0.026	0.115 / 0.156
PD-NRG w/ DA		0.241 / 0.240	0.281 / 0.279	0.210 / 0.212	0.018 / 0.027	0.123 / 0.165
PD-NRG w/ DA	✓	0.230 / 0.227	0.291 / 0.287	0.185 / 0.185	0.021 / 0.032	0.133 / 0.180
+knowledge flag		0.240 / 0.242	0.280 / 0.280	0.209 / 0.213	0.018 / 0.027	0.122 / 0.164
+knowledge flag	✓	0.222 / 0.223	0.287 / 0.281	0.180 / 0.180	0.032 / 0.022	0.137 / 0.181
+knowledge flag + topic		0.244 / 0.245	0.276 / 0.274	0.210 / 0.213	0.018 / 0.027	0.118 / 0.159
+knowledge flag + topic	✓	0.224 / 0.221	0.272 / 0.271	0.187 / 0.186	0.020 / 0.029	0.136 / 0.177

Table 3: Automated metrics with ground-truth Action Plan on test freq / rare

5.3 Evaluation Measures

For automatic evaluation we compute a set of metrics between our generated and ground truth response: perplexity, BLEU-1, ROUGE-L, unigram F1-score. We also compute n-gram diversity as defined in (Ghazvininejad et al., 2018).

For human evaluation, we followed a similar setup as (Li et al., 2016b) and generated 200 snippets which contain a dialog context of 5 turns. We generated responses from 2 models to compare against. We asked a set of 3 crowd workers “Which final response is more appropriate for the given conversation?”.

5.4 Results Using the Ground-Truth Action Plan

We first check whether the PD-NRG approach results in better responses when we use the ground truth AP. As seen in Figure 4 instead of using a dialog policy, we form ground truth APs from the annotations described in Section 5.2. We then use them to generate a response for that turn. Table 3 presents automated evaluation results for Baseline-Turn, Baseline-Sent and variations of the PD-NRG models. As seen in the results table, adding dialog acts increases diversity for all the proposed models. This aligns with previous work that using dialog acts leads to more diverse responses (Sankar and Ravi, 2019). The F1, BLEU, and ROUGE scores of the PD-NRG w/ DA model are lower than the Baseline-Turn model due to the PD-NRG model decoding shorter sentences resulting in lower recall. The PD-NRG w/ DA model with the addition of

previous dialog acts as input results in the lowest perplexity for both frequent and rare test sets.

5.4.1 Do the Models Follow the Action Plan?

By jointly conditioning on the attributes in the AP, we aim to control multiple aspects of the response, such as content and style. The dialog acts determine if the response should be a question, statement or should give feedback. The knowledge determines what content should be present in the response. To see if the model responses follow the AP, we manually annotated if the model’s responses realize the dialog acts and their respective knowledge sentence (focusing on the cases where the AP included a knowledge sentence) in their input. Turns with no dialog acts, i.e., marked as *NoDialogAct*, were ignored. The results from the manual evaluation are presented in Table 4. The PD-NRG w/ DA + knowledge flag model has the highest accuracy in realizing the input AP, achieving 80.6% accuracy on the dialog acts of the generated responses, and 52.1% accuracy in correctly integrating the provided knowledge sentences. Figure 5 presents an example from this model.

5.5 Results Using an Estimated Action Plan

Using our dialog policy models, we estimate an AP for each turn. Given the dialog context and the AP we then generate responses using the PD-NRG w/ DA + knowledge flag model + Past DA model. We evaluate the responses using both automated and human evaluation. We present our automatic metrics in Table 5. The KD-DA-P and KI-DA-P(Simple) produced more *Feedback* and *PropQ* di-

Models	Past DA	% DA	%K
Baseline-Turn (Wolf et al., 2019)		26.7	-
Baseline-Sent		59.4	30.8
PD-NRG w/ DA		69.1	47.0
PD-NRG w/ DA	✓	69.7	39.3
+knowledge flag		80.6	52.1
+knowledge flag	✓	68.1	47.8
+knowledge flag +topic		77.8	47.4
+knowledge flag +topic	✓	69.0	45.3

Table 4: % of Dialog Acts (DA) and Knowledge (K) Realized for PD-NRG Models to showcase controllability.

...
Speaker 1: Free with you, they should have had Snoop Dogg make a theme song for the game like he did for his son’s high school football team LOL
Speaker 2: Interesting, do you play golf?
Speaker 1:
<i>Baseline-Turn:</i>
no, i don’t play golf, but i hear it has been a lot of years since the last time.
<i>PD-NRG model:</i>
Statement → not really, i’m not a huge fan of golf.
PropQ → have you ever played?

Figure 5: Baseline-Turn model versus PD-NRG model

alog acts than the actual distribution of dialog acts in the dataset, where over 80% of the dialog acts were *Statements*. For example KD-DA-P produced 41% *Feedback* dialog acts whereas the actual distribution contains only 5% *Feedback* dialog acts. We believe this change in the distribution resulted in our models generating responses with fewer words and as a result these models have lower F1-scores.

The limitation of n-gram overlap measures is that they do not capture the diverse set of responses that can be generated in an open-domain setting. For a more realistic comparison of our dialog policy models to our baselines, we ran human evaluation. We provided a set of crowd workers outputs from two models along with the dialog context, and asked them “Which final response is more appropriate for the given conversation?”. Crowd workers were provided with 3 options: first response, second response and not sure (limited to those cases when the two responses are equally good/bad). Table 6 presents results from the manual evaluations. As seen, the KD-DA-P responses were chosen over the B-Turn model by a large margin. This result is also seen in KD-DA-P responses versus the KI-DA-P (PropQ/AllQ) responses, proving that its is better to have a dialog policy adapting to the course of the dialog versus using a fixed distribution (See et al., 2019) to predict the dialog acts. However, the KI-DA-P (Seq2Seq) results in worse responses than the

Policy	F1	Avg # words	Avg # sentences
Ground truth	0.22 / 0.22	15.2 / 15.3	1.68 / 1.76
Baseline-Turn	0.18 / 0.17	19.8 / 19.7	1.86 / 1.87
KI-DA-P (Simple)	0.14 / 0.14	12.9 / 12.2	1.89 / 1.89
KD-DA-P	0.14 / 0.14	12.3 / 11.5	1.91 / 1.91
KI-DA-P(Seq2Seq)	0.14 / 0.17	13.1 / 13.4	1.46 / 1.56

Table 5: Automated metrics with estimated Action Plan. Baseline-Turn (Wolf et al., 2019)

Policy	%W	%T	%L	IAA
KD-DA-P vs. Baseline*	40.8	30.3	28.9	0.43
KI-DA-P(Seq2Seq) vs. B-Turn*	25.1	35.7	39.2	0.47
KD-DA-P vs. See et.al(PropQ)**	54.2	5.5	40.2	0.46
KD-DA-P vs. See et al. (2019)**	54.1	7.4	38.3	0.48
KD-DA-P vs. Human response**	16.7	35.3	48.0	0.53

Table 6: % of Wins(W), Ties (T) and Losses(L) for the baseline models vs PD-NRG model on appropriateness. The KD-DA-P policy is statistically significant compared to the B-Turn(Baseline-Turn) (Wolf et al., 2019) as well as the KI-DA-P(PropQ) and KI-DA-P(PropQ) baselines (See et al., 2019). We compute Krippendorff’s alpha for Inter-annotator agreement(IAA). We computed the p-value using a two-tailed binomial test. * refers to a p-value < 0.05 and ** refers to a p-value < 0.01.

baseline. We believe this is because the *Statement* dialog act is a large portion of the dataset, making learning other acts harder for the model. For future work, we will investigate machine learning approaches to learn better models for the dialog policy. The proposed KD-DA-P results in responses that are better than or similar to human responses in 52% of the cases.

6 Conclusions

In this work, we propose a policy-driven neural response generation approach for knowledge grounded open-domain dialog systems. We estimate an action plan that consists of a set of attributes that control the content and style of the generated responses at the turn and sentence levels. We investigate both manual and machine learning based policies. Through human evaluation, we empirically demonstrate that a basic dialog policy that does sentence level generation outperforms turn level generation, as well as knowledge-grounded response generation baselines. Furthermore, the generated responses realize their respective action plans. This allows builders of dialog systems control over the model’s responses allowing for more consistent user experiences. Our future work includes investigation of better approaches for learning such dialog policy models along with adding other attributes such as sentiment.

References

- Ali Ahmadvand, Ingyu Jason Choi, Harshita Sahijwani, Justus Schmidt, Mingyang Sun, Sergey Volokhin, Zihao Wang, and Eugene Agichtein. 2018. Emory iris-bot: An open-domain conversational bot for personalized information access. *Alexa Prize Proceedings*.
- Ashutosh Baheti, Alan Ritter, Jiwei Li, and Bill Dolan. 2018. [Generating more interesting responses in neural conversation models with distributional constraints](#). In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*.
- Kevin K Bowden, Jiaqi Wu, Wen Cui, Juraj Juraska, Vrindavan Harrison, Brian Schwarzmann, Nick Santer, and Marilyn Walker. 2019. Slugbot: Developing a computational model and framework of a novel dialogue genre. *arXiv preprint arXiv:1907.10658*.
- Alessandra Cervone, Giuliano Tortoreto, Stefano Mezza, Enrico Gambi, Giuseppe Riccardi, et al. 2017. Roving mind: a balancing act between open-domain and engaging dialogue systems.
- Emily Dinan, Varvara Logacheva, Valentin Malykh, Alexander Miller, Kurt Shuster, Jack Urbanek, Douwe Kiela, Arthur Szlam, Iulian Serban, Ryan Lowe, et al. 2020. The second conversational intelligence challenge (convai2). In *The NeurIPS'18 Competition*, pages 187–208. Springer.
- Emily Dinan, Stephen Roller, Kurt Shuster, Angela Fan, Michael Auli, and Jason Weston. 2018. Wizard of wikipedia: Knowledge-powered conversational agents. *arXiv preprint arXiv:1811.01241*.
- Hao Fang, Hao Cheng, Maarten Sap, Elizabeth Clark, Ari Holtzman, Yejin Choi, Noah A Smith, and Mari Ostendorf. 2018. Sounding board: A user-centric and content-driven social chatbot. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Demonstrations*, pages 96–100.
- Nancy Fulda, Tyler Etchart, William Myers, Daniel Ricks, Zachary Brown, Joseph Szendre, Ben Murdoch, Andrew Carr, and David Wingate. 2018. Byueve: Mixed initiative dialog via structured knowledge graph traversal and conversational scaffolding.
- 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 *Thirty-Second AAAI Conference on Artificial Intelligence*.
- Marjan Ghazvininejad, Xing Shi, Jay Priyadarshi, and Kevin Knight. 2017. Hafez: an interactive poetry generation system. In *Proceedings of ACL 2017, System Demonstrations*, pages 43–48.
- Karthik Gopalakrishnan, Behnam Hedayatnia, Qinlang Chen, Anna Gottardi, Sanjeev Kwatra, Anu Venkatesh, Raefer Gabriel, and Dilek Hakkani-Tür. 2019. Topical-chat: Towards knowledge-grounded open-domain conversations. *Proc. Interspeech 2019*, pages 1891–1895.
- Guillaume Klein, Yoon Kim, Yuntian Deng, Jean Senellart, and Alexander M. Rush. 2017. [OpenNMT: Open-source toolkit for neural machine translation](#). In *Proc. ACL*.
- Jiwei Li, Michel Galley, Chris Brockett, Georgios P Spithourakis, Jianfeng Gao, and Bill Dolan. 2016a. A persona-based neural conversation model. *arXiv preprint arXiv:1603.06155*.
- Jiwei Li, Will Monroe, Alan Ritter, Michel Galley, Jianfeng Gao, and Dan Jurafsky. 2016b. Deep reinforcement learning for dialogue generation. *arXiv preprint arXiv:1606.01541*.
- Edward Loper and Steven Bird. 2002. Nltk: the natural language toolkit. *arXiv preprint cs/0205028*.
- Minh-Thang Luong, Hieu Pham, and Christopher D Manning. 2015. Effective approaches to attention-based neural machine translation. *arXiv preprint arXiv:1508.04025*.
- Stefano Mezza, Alessandra Cervone, Giuliano Tortoreto, Evgeny A Stepanov, and Giuseppe Riccardi. 2018. Iso-standard domain-independent dialogue act tagging for conversational agents. *arXiv preprint arXiv:1806.04327*.
- Jan Pichl. 2018. Alquist 2.0: Alexa prize socialbot based on sub-dialogue models.
- Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training. [URL https://s3-us-west-2.amazonaws.com/openai-assets/researchcovers/languageunsupervised/language_understanding_paper.pdf](https://s3-us-west-2.amazonaws.com/openai-assets/researchcovers/languageunsupervised/language_understanding_paper.pdf).
- Stephen Robertson, Hugo Zaragoza, et al. 2009. The probabilistic relevance framework: Bm25 and beyond. *Foundations and Trends® in Information Retrieval*, 3(4):333–389.
- Stephen Roller, Emily Dinan, Naman Goyal, Da Ju, Mary Williamson, Yinhan Liu, Jing Xu, Myle Ott, Kurt Shuster, Eric M Smith, et al. 2020. Recipes for building an open-domain chatbot. *arXiv preprint arXiv:2004.13637*.
- Chinnadhurai Sankar and Sujith Ravi. 2019. Deep reinforcement learning for modeling chit-chat dialog with discrete attributes. *arXiv preprint arXiv:1907.02848*.
- Emanuel A Schegloff. 2007. *Sequence organization in interaction: A primer in conversation analysis I*, volume 1. Cambridge University Press.

- Abigail See, Stephen Roller, Douwe Kiela, and Jason Weston. 2019. What makes a good conversation? how controllable attributes affect human judgments. *arXiv preprint arXiv:1902.08654*.
- Iulian V Serban, Alessandro Sordoni, Yoshua Bengio, Aaron Courville, and Joelle Pineau. 2016. Building end-to-end dialogue systems using generative hierarchical neural network models. In *Thirtieth AAAI Conference on Artificial Intelligence*.
- Iulian Vlad Serban, Alessandro Sordoni, Ryan Lowe, Laurent Charlin, Joelle Pineau, Aaron Courville, and Yoshua Bengio. 2017. A hierarchical latent variable encoder-decoder model for generating dialogues. In *Thirty-First AAAI Conference on Artificial Intelligence*.
- Xiaoyu Shen, Hui Su, Yanran Li, Wenjie Li, Shuzi Niu, Yang Zhao, Akiko Aizawa, and Guoping Long. 2017. A conditional variational framework for dialog generation. *arXiv preprint arXiv:1705.00316*.
- Eric Michael Smith, Mary Williamson, Kurt Shuster, Jason Weston, and Y-Lan Boureau. 2020. Can you put it all together: Evaluating conversational agents’ ability to blend skills. *arXiv preprint arXiv:2004.08449*.
- 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. *arXiv preprint arXiv:1506.06714*.
- Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N Gomez, Łukasz Kaiser, and Illia Polosukhin. 2017. Attention is all you need. In *Advances in neural information processing systems*, pages 5998–6008.
- Oriol Vinyals and Quoc Le. 2015. A neural conversational model. *arXiv preprint arXiv:1506.05869*.
- Marilyn A Walker, Amanda Stent, François Mairesse, and Rashmi Prasad. 2007. Individual and domain adaptation in sentence planning for dialogue. *Journal of Artificial Intelligence Research*, 30:413–456.
- Bolin Wei, Shuai Lu, Lili Mou, Hao Zhou, Pascal Poupart, Ge Li, and Zhi Jin. 2017. Why do neural dialog systems generate short and meaningless replies? a comparison between dialog and translation. *arXiv preprint arXiv:1712.02250*.
- Thomas Wolf, Victor Sanh, Julien Chaumond, and Clement Delangue. 2019. Transfertransfo: A transfer learning approach for neural network based conversational agents. *arXiv preprint arXiv:1901.08149*.
- Chen Xing, Wei Wu, Yu Wu, Jie Liu, Yalou Huang, Ming Zhou, and Wei-Ying Ma. 2016. Topic aware neural response generation. *arXiv preprint arXiv:1606.08340*.
- Can Xu, Wei Wu, and Yu Wu. 2018. Towards explainable and controllable open domain dialogue generation with dialogue acts. *arXiv preprint arXiv:1807.07255*.
- Semih Yavuz, Abhinav Rastogi, Guan-Lin Chao, and Dilek Hakkani-Tur. 2019. Deepcopy: Grounded response generation with hierarchical pointer networks. *arXiv preprint arXiv:1908.10731*.
- Dian Yu, Michelle Cohn, Yi Mang Yang, Chun Yen Chen, Weiming Wen, Jiaping Zhang, Mingyang Zhou, Kevin Jesse, Austin Chau, Antara Bhowmick, et al. 2019. Gunrock: A social bot for complex and engaging long conversations. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP): System Demonstrations*, pages 79–84.
- Tiancheng Zhao, Ran Zhao, and Maxine Eskenazi. 2017. Learning discourse-level diversity for neural dialog models using conditional variational autoencoders. *arXiv preprint arXiv:1703.10960*.
- Hao Zhou, Tom Young, Minlie Huang, Haizhou Zhao, Jingfang Xu, and Xiaoyan Zhu. 2018. Commonsense knowledge aware conversation generation with graph attention. In *IJCAI*, pages 4623–4629.