Joint Retrieval and Generation Training for Grounded Text Generation

Yizhe Zhang **Siqi Sun** **Xiang Gao** **Yuwei Fang**
Chris Brockett **Michel Galley** **Jianfeng Gao** **Bill Dolan**
Microsoft Corporation, Redmond, WA, USA
{yizzhang,siqi.sun,xiag,yuwfan,mgalley,chrisbkt,jfgao,billdol}@microsoft.com

Abstract

Recent advances in large-scale pre-training such as GPT-3 allow seemingly high quality text to be generated from a given prompt. However, such generation systems often suffer from problems of hallucinated facts, and are not inherently designed to incorporate useful external information. Grounded generation models appear to offer remedies, but their training typically relies on rarely-available parallel data where corresponding information-relevant documents are provided for context. We propose a framework that alleviates this data constraint by jointly training a grounded generator and document retriever on the language model signal. The model learns to reward retrieval of the documents with the highest utility in generation, and attentively combines them using a Mixture-of-Experts (MoE) ensemble to generate follow-on text. We demonstrate that both generator and retriever can take advantage of this joint training and work synergistically to produce more informative and relevant text in both prose and dialogue generation.[1]

1 Introduction

Recent large-scale pre-trained language models (LMs) such as BERT (Devlin et al., 2019), GPT-2 (Radford et al., 2019), GPT-3 (Brown et al., 2020), and T5 (Raffel et al., 2019) have brought numerous breakthroughs in natural language generation (NLG) across a variety of tasks. These models, however, are not designed to leverage external information to enhance or to verify the predicted text. Gao et al. (2020), for example, demonstrates that they fail to reliably generate responses grounded in real-world knowledge, and may fall short when generating goal-directed responses that are optimized for information-seeking task completion. These models pose several challenges in information-demanding scenarios: First, they are usually trained offline, rendering the model agnostic to the latest information (*e.g.*, asking a chatbot trained from 2011-2018 about COVID-19). Second, they are mostly trained on public data, rendering them less suitable in scenarios where customized or personalized information must be processed (*e.g.*, writing suggestions based on private user-data). Third, even in scenarios that call only for public information, generation from these LMs may be unfaithful to the facts (*e.g.*, hallucinations about birth dates), especially when the people or entities are less well known and the scenario demands a high degree of fidelity. As a practical matter, moreover, there remains a fundamental capacity issue in that large LMs cannot effectively represent all the information about every person or entity in the world.

A solution that would at first glance seem obvious is to ground the language model in real-world knowledge, which can be present in either structured data (*e.g.*, a knowledge-graph) or unstructured data (*e.g.*, documents such as Wikipedia, user documents or background stories) (Wu et al., 2020; Ghazvininejad et al., 2018; Dinan et al., 2019; Qin et al., 2019). The advantage of using unstructured grounding data over structured data is that the former provides richer information and it is typically more flexible when it comes to maintaining and updating the information base. However, training a grounded text generation model that takes additional unstructured documents as input typically demands that the training data contains pairs of context and corresponding oracle documents. These pairs are seldom available. Recent work, such as REALM (Guu et al., 2020) and RAG (Lewis et al., 2020b), attempts to leverage information retrieval machinery in real time to mitigate this data paucity in open-domain question answering systems. The approach taken in this paper is in similar vein, but

[1] The code for this project will be released at https://github.com/dreasysnail/RetGen.

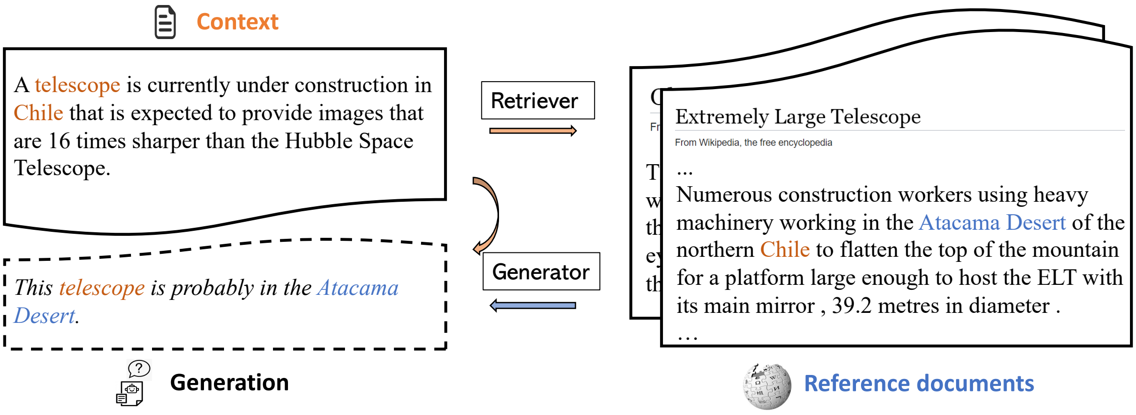

Context
A telescope is currently under construction in Chile that is expected to provide images that are 16 times sharper than the Hubble Space Telescope.
Retriever
Generator
This telescope is probably in the Atacama Desert.
Generation
Extremely Large Telescope
From Wikipedia, the free encyclopedia
...
Numerous construction workers using heavy machinery working in the Atacama Desert of the northern Chile to flatten the top of the mountain for a platform large enough to host the ELT with its main mirror , 39.2 metres in diameter .
...
Reference documents

Figure 1: An overview of retrieval-grounded text generation task.

is not confined to the specialized case of question answering, and seeks to present a mechanism to that addresses the broader problem of informational accuracy in text generation. More detailed discussion on what differentiates this work from previous work is provided in §6.

Below, we present a large-scale general purpose pre-training framework that jointly trains a document retriever and a multi-document grounded generator in end-to-end fashion and allows these to synergistically cooperate to optimize grounded text generation. Our method first selects and scores a collection of documents that are found to be most helpful to generation according to the language model signal. The multi-document generator then digests these documents and combines their information according to document-specific attention weights to generate a single prediction in a Mixture-of-Experts (MoE) manner.

The main contributions and advantages of this work are summarized as follows:

- We provide a *joint training framework* for grounded generation and document retrieval with a language model signal. Our method alleviates the need for oracle parallel data (prose-document pairs) with which to train a grounded model, enabling the use of massive non-parallel corpora.
- From the *retriever's perspective*, our approach can be viewed as using the language model signal to optimize the retriever, so that the documents with highest utility in generation are returned.
- From the *generator's perspective*, our approach learns to attend to and combine multiple retrieved documents to achieve a mixture of expert (MoE)-based generation. We apply mutual information maximization (MMI) and retriever correction to further enhance the model.
- The superiority of our methods is demonstrated by both crowd-sourced human judgments and automatic metrics from generation and information-retrieval domains.

2 Problem statement

We begin by formally defining our *retrieval-grounded text generation* (RGTG) task and laying out necessary notation. As shown in figure 1, RGTG aims to predict the upcoming text y that directly follows the existing source prompt x (x, y are from a corpus $\mathbf{D}$), while a document reference set $\mathbf{Z}$ is accessible and can be leveraged. In this task, $\mathbf{D}$ and $\mathbf{Z}$ are *non-parallel* to each other. In other words, in a given dataset, each x is paired with a y. However, the association between a document z in $\mathbf{Z}$ and the (*source, target*) tuple (x, y) is not necessarily known.

In response generation, (x, y) can be (*dialogue history, next utterance*). In prose generation, (x, y) can be (*preceding context, following text*). $\mathbf{Z}$ can be a Wikipedia dump, a set of user documents, or any other relevant textual material.

3 Methods

We propose a framework called RetGen to solve the retrieval-grounded text generation task. RetGen has two components: i) a dense document retriever and ii) a knowledge-grounded text generator.

3.1 Method overview

The objective of the retrieval-grounded text generation is to train a model to maximize the likelihood of y given x and $\mathbf{Z}$. Formally, the probability

$p(y|x)$ can be written as

$$p(y|x; \mathbf{Z}) = \sum_{z \in \mathbf{Z}} p(y|x, z)p(z|x), \quad (1)$$

In practice, $\mathbf{Z}$ often contains millions of documents, rendering enumeration over z impossible. Instead, we leverage a *dense document retriever* $r_\Phi(\cdot)$ to dramatically narrow down the search to a handful relevant documents, where Φ denotes the retriever parameters. r_Φ takes $\mathbf{Z}$ and x as input and yields relevance scores $\{s_1, \cdots, s_K\}$ of the top-K (K is a hyper-parameter) documents $\tilde{\mathbf{Z}} = \{z^{(1)}, \cdots, z^{(K)}\}$.

We further denote the *knowledge-grounded text generator* as $g_\Theta(\cdot)$, where Θ denotes the generator parameters. This generator module uses x and a single document z as input to produce a probability score for a given reference target y, *i.e.*, $g_\Theta(y|x, z) = p(y|x, z)$.

With the above definitions, the loss can be approximated as:

$$\mathcal{L}(\Theta, \Phi) = \\ - \sum_{(x,y) \in \mathbf{D}} \log \sum_{k=1}^{K} p_\Theta(y|x, z^{(k)}) p_\Phi(z^{(k)}|x), \quad (2)$$

where $p(z^{(k)}|x) = \exp(s_k) / \sum_{i=1}^{K} \exp(s_i)$ is the normalized probability, and $\tilde{\mathbf{Z}} = \{z^{(1)}, \cdots, z^{(K)}\}$ are retrieved from $r_\Phi(\mathbf{Z}, x)$. An overview of the model is presented in Figure 2. We explain each module in the following sections.

3.2 Document Retriever

For the dense document retriever $p_\Phi(\cdot)$ in (2), we leverage a model similar to that of (Karpukhin et al., 2020; Xiong et al., 2020) to achieve efficient document retrieval with sublinear time. The documents Z and context queries x are mapped into the same dense embedding space. The relevance score $s(x, z)$ is computed as the vector inner product between document embedding $h_z = f_z(z)$ and query embedding $h_x = f_x(x)$, *i.e.*, $s(x, z) = h_x^T h_z$, where $f_z(\cdot)$ and $f_x(\cdot)$ represent learnable encoding networks for document and query respectively. $p(z^{(k)}|x)$ in (2) is finally given by *softmax*$^{(k)}(s(x, \tilde{\mathbf{Z}}))$.

To achieve sublinear searching time, Maximum Inner Product Search (MIPS) (Shrivastava and Li, 2014) is employed. The document embedding vectors are pre-computed and indexed according to locality using Locality Sensitivity Hashing (LSH) (Datar et al., 2004), so that the query vector can be hashed to a cluster of relatively relevant documents. This search strategy is approximate. However it yields good empirical search results when the document set is large. In practice, we use ANCE (Xiong et al., 2020) to initialize the retriever.

3.3 Knowledge-Grounded Text Generator

For the knowledge-grounded text generator (GTG) corresponding to $p_\Theta(\cdot)$ in (2), we employ a transformer-based architecture akin to GPT-2 (Radford et al., 2019). The GTG takes one document z and one context query x as the input, and the following text y as the target reference. Specifically, the z and x are first concatenated by a special separator token. The training objective is computed following a standard language model (LM) loss (Radford et al., 2019; Zhang et al., 2020):

$$p_\Theta(y|x, z) = \prod_{t=0}^{|y|} p(y_t|x, z, y_{0:t-1}), \quad (3)$$

where y_t represents the t-th token in y. $y_{i:j}$ denotes $\{y_i, \cdots, y_j\}$ and $|\cdot|$ denotes the cardinality. As opposed to GPT-2, we assign different token type embeddings to the tokens in z and x to help the model identify document and context.

We also employ a distinctive design for the position embedding in the grounded generation task. The document position id starts with M ($M = 400$ in our experiment) while the context position id starts with 0. The intent is to maximally separate z and x, thus reducing the chance that the model will be exposed to hints that z is part of the preceding context.[2] We found this facilitates the model in differentiating the document and the context, and in applying different strategies specific to each.

3.4 Joint Training

During the training time, Θ can be directly optimized from (2). We optimize the objective in (2) with respect to Φ by leveraging an unbiased estimation resembles the Actor-Critic (AC) algorithm.

[2] Our GTGs are initialized from GPT-2/DialoGPT, which were trained to recognize tokens with continuous position id as a piece of coherent text

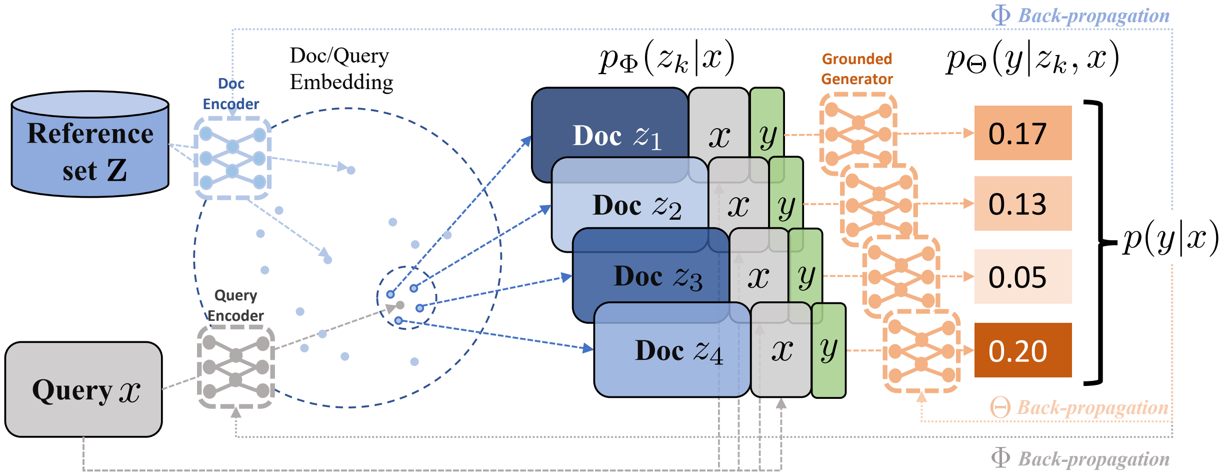


Figure 2: An overview of the joint training framework. A source context query x and documents from a reference database $\mathbf{Z}$ are first mapped to a joint embedding space via different encoders. A Maximum Inner Product Search (MIPS) is performed to retrieve top-K relevant documents ($K = 4$ in this figure) with their probability score $p(z_k|x)$. The retrieved documents are separately concatenated with query x and target upcoming text y and passed through a grounded text generator, to compute the document-dependent likelihood $p(y|z_k, x)$. The final objective $p(y|x)$ given by (2) is optimized to update the retriever parameters Φ and generator parameters Θ.

Following Guu et al. (2020),

$$\begin{aligned}\nabla_\Phi p(y|x) &= \sum_z p(y|z,x)\nabla_\Phi p(z|x)\\ &= \sum_z p(y|z,x)p(z|x)\nabla_\Phi \log p(z|x)\\ &= \sum_z [p(y|z,x) - C]p(z|x)\nabla_\Phi \log p(z|x), \quad (4)\end{aligned}$$

where the C is a constant baseline. The last step is because $\sum_z \nabla_\Phi p(z|x) \log p(z|x) = \nabla_\Phi \sum_z p(z|x) = 0$. C is commonly referred as a "control variate" (Williams, 1992; Nelson, 1990) and used to reduce the variance in Monte Carlo estimation as in (4). The $p(y|z, x)$ can be viewed as the "value" or "return" in the Actor-Critic algorithm. Document z will receive a positive update if it yields $p(y|z, x)$ larger than to the average performance of the retrieved documents. In our experiment, we set C as the expected reward, *i.e.*, $C = \sum_{z\in\tilde{\mathbf{Z}}} p(y|z, x)p(z|x)$. Finetuning the retriever model based on (4) needs good initializations from pretrained models to avoid cold-starting. In practice, we initialize the Θ from GPT-2 or DialoGPT, and initialize the Φ from ANCE to accelerate the join training.

Another practical challenge is that all the document embedding vectors need to be *refreshed* once the retriever is updated, which is expensive when the number of documents is large. Instead of encoding all the documents each time the Φ is updated to retrieve the top-K document set $\tilde{\mathbf{Z}}$, we asynchronously update the retrieved document for every M steps ($M = 200$ in our experiments). However, note that even if the $\tilde{\mathbf{Z}}$ is fixed for each K steps, the r_Φ and scores $\{s_1, \cdots, s_K\}$ are still updated at every step.

3.5 Multi-document Decoding

Mixture-of-Expert (MoE) Decoder During the inference time, the retriever first obtains the top-K documents as $\tilde{\mathbf{Z}}$, and their corresponding probabilities $p(z|x)$. The generator leverages all document in $\tilde{\mathbf{Z}}$ to generator a consensus prediction $\hat{y}$. One naive approach is to concatenate multiple documents into a "joint" document as the input for the generator. The problems for such an approach are that 1) the "joint" document may be too long to be efficiently processed;[3] 2) the order of the documents has impact on the generation; 3) the relevance information $p(z|x)$ will be ignored.

We therefore took a Mixture-of-Expert (MoE) approach following Cho et al. (2020) to decode the model in a document-specific fashion, and ensemble the output distributions at each time step. Specifically, we leverage K copies of the ground text generator $g_\Theta(\cdot)$ trained from (2). At time step t of the generation, we feed each copy of the generator with separate document z, the same context x, and the same current consensus gen-

[3] The time and memory footprint for a vanilla Transformer typically scale quadratically with the sequence length.

eration $\hat{y}_{0:t-1}$. We then harvest the individual output distribution from all generator copies, as $\{p(\hat{y}_t^{(1)}), \cdots, p(\hat{y}_t^{(K)})\}$. The assembled output distribution at step t is finally given by

$$p(\hat{y}_t|\tilde{\mathbf{Z}}, x, \hat{y}_{0:t-1}) \\ = \sum_{k=1}^{K} p(\hat{y}_t^{(k)}|z^{(k)}, x, \hat{y}_{0:t-1}) p(z^{(k)}|x). \quad (5)$$

Unlike recent FiD work (Izacard and Grave, 2020), which "fuses" the encoded representations from different documents, our "fusion" of information from different document occurs at output token distribution level. FiD requires training a grounded generation model by taking a fixed number of documents as input. However, our MoE approach can directly leverage a grounded generation model trained on single document as input, without additional training or fine-tuning. This yields convenience and flexibility in the number of documents K to leverage for inference.

Retriever Correction The fact that the model is trained to *autoregressively* generate y implies that the retriever score needs to be updated along the generation. To see this, we revisit (2) by expanding the $p(y|x)$ as $p(y_0|x) \prod_{t=1}^{|y|} p(y_t|x, y_{0:t-1})$. Each of the $p(y_t|x, y_{0:t-1})$ can be written as

$$p(y_t|x, y_{0:t-1}) \\ = \sum_{z} p(y_t|x, z, y_{0:t-1}) p(z|x, y_{0:t-1}). \quad (6)$$

where $p(y_t|x, z, y_{0:t-1})$ can be directly obtained from the generator model (3). The *updated document probability* $p(z|x, y_{0:t-1})$ can be further expanded as

$$p(z|x, y_{0:t-1}) = p(z|x) p(y_{0:t-1}|z, x)/p(y_{0:t-1}|x). \quad (7)$$

The term $p(y_{0:t-1}|z, x)/p(y_{0:t-1}|x) \triangleq F_t$ serves as a *correction factor* for updating the retriever's belief for document relevance with newly seen/generated text $y_{0:t-1}$. It can be computed by $F_t = p(y_{0:t-1}|z, x)/\sum_{z'} p(y_{0:t-1}|z', x)p(z'|x)$. When this factor is greater than 1 (*i.e.*, being grounded on z improves the probability of $y_{0:t-1}$), the corresponding document z will be assigned with a higher probability. The computation cost of the correction factor is negligible. The retriever correction simply multiplies this correction factor F_t to (5) at each time step.

MMI Following Zhang et al. (2020), we further implement a Maximum Mutual Information (MMI) scoring function (Li et al., 2016; Zhang et al., 2018) to enhance the "groundness" of the generation. MMI employs a pre-trained *backward* grounded text generation model to predict x and z from given prediction y, *i.e.*, $p(x, z|y)$.[4] We first generate a set of hypotheses using top-K sampling. Then we use the probability of $p(x, z|y)$. For multiple z we use the mean of these probabilities to rerank all hypotheses. Intuitively, maximizing backward model likelihood penalizes the bland hypotheses (Zhang et al., 2020) and encourages the generation y to tie better to the input document z and context x.

4 Experimental Setups

Datasets We use two datasets $\mathbf{D}$, Reddit and arXiv, which cover response generation and prose generation respectively, to evaluate the effectiveness of the proposed methods.

The ***Reddit*** dataset contains 2.56M/2K/2K training/validation/test instances [5]. Although our approach does not rely on parallel text-reference data, we collected data of this kind to build baselines and to evaluate the retriever performance. The training data is created using a pipeline similar to that in the DSTC-7 grounded response generation challenge (Galley et al., 2019): We first select the Reddit discussion threads that contain urls in the description, crawled from the Reddit with time span 2011-2017. Then, we restrict the url domain to Wikipedia, and extract the linked oracle passage by selecting the most relevant passage to the context according to ROUGE-F1. This yields about 2.56M data instances. We further select test examples from Reddit with time span 2018-2019 by requiring the context to have at least 6 different responses. This yields a 5-reference test set with 2,000 samples. For each instance, one of the 6 human responses is set aside to assess human performance.

The ***arXiv*** dataset is based on (Clement et al., 2019)[6], which provides a corpus of arXiv articles from 1991 to 2019. We construct the context and target pairs using the abstracts. For each sentence [7] in an abstract, we use the preceding sentences as the context for predicting the current sentence.

[4] This objective is designed to encourage y to incorporate information from both x and z.

[5] The script for creating this data will be released.

[6] https://github.com/mattbierbaum/arxiv-public-datasets

[7] we consider the title to be the first sentence

We processed the texts to replace citations, urls, and equations by special tokens. We apply a filtering step to select instances in the test set that are likely to benefit from external information by using Tagme (Ferragina and Scaiella, 2010), a Wikipedia name entity recognition (NER) tool. Tagme identifies the named entities that exist as Wikipedia entries and meanwhile occur in the target sentence. The Tagme threshold, which balances the precision and recall of NER, is set at 0.7. We only retain instances that pass this threshold. The final resulting train/validation/test contains 9.6M/57K/2K instances from 1.67M unique abstracts.

Reference dataset For the reference dataset **Z**, we extract about 5.7 million reference documents from Wikipedia dump of December 2018. For each Wikipedia entry, we only extract the first and second paragraph as these are typically most relevant and summarize the entire document. In addition, we truncate overlong sentences to 100 words, and remove the entry if it contains only one sentence.

Evaluation Metrics We performed automatic evaluation using standard machine translation metrics, including BLEU (Papineni et al., 2002), METEOR (Lavie and Agarwal, 2007), and NIST (Doddington, 2002). NIST is a variant of BLEU that weights n-gram matches by their information gain, i.e., it indirectly penalizes uninformative n-grams. We also use Entropy (Zhang et al., 2018) and Dist-n (Li et al., 2016) to evaluate lexical diversity. More details are provided in Zhang et al. (2020). For Reddit dataset, where 5 references are available for each instance, we compute all relevance metrics and aggregate all of them using max-pooling.

To evaluate how well the predicted text $\hat{y}$ can reflect the external information, we propose an evaluation score which we call a *Keyword Matching Ratio (KMR)*. KMR is defined as

$$\begin{aligned} &\text{K-words} = \text{set}(z) \setminus \text{set}(x), \\ &\text{KMR} = |\text{set}(\hat{y}) \cap \text{K-words}| / |\text{K-words}|, \end{aligned} \quad (8)$$

where $\cap, \setminus, |\cdot|$ denotes the set intersection, difference and cardinality, respectively. For each bag-of-word sets (*i.e.*, $\text{set}(\hat{y}), \text{set}(z), \text{set}(x)$), stop words based on the python NLTK module and frequency in the corpus are removed. Intuitively, K-words reflect important information (a set of keywords) in the reference documents z but not covered by context x. KMR calculates the percentage of these keywords covered by the predicted text y. Such a metric assesses the utilization of external information but not the relevance. If a model generates reasonable follow-up text, but fails to incorporate important external information in z, KMR will still be low.

Baselines & Model setups We compared RetGen with baseline non-grounded models (Dialogpt (345M) and GPT-2 (345M)) in two datasets. The Dialogpt and GPT-2 baselines are obtained by fine-tuning the original pre-trained models on the target training dataset to alleviate the dataset-shifting bias. For the Reddit dataset, since we have the oracle document for each conversation, it is possible to train a ground generation model as described in section §3.3 by directly grounding on the oracle documents. This model, called as *gDialogpt*, serves as an upperbound of the performance in our comparison.

Note that we perform fine-tuning rather than pre-training by leveraging existing popular pre-trained LMs and dense retrievers. All the grounded generators use the same transformer architectures, and are initialized with original DialoGPT/GPT-2 (345M) weights. The dense retrievers are all initialized from ANCE (Xiong et al., 2020). We set the learning rate to be 10^{-6} and batch size to be 128 for most of the experiments. For the retriever training, we save model checkpoints and index the documents for each 200 iterations. We observe that reducing the indexing frequency to 100 provides marginal performance gain, while yielding more computation. During training, we use $K = 4$ for RetGen. All generations except for MMI use greedy decoding.

All compared models are trained until no progress can be observed on validation loss. Most of the models are trained on workstations with 8 Nvidia V100 GPUs.

5 Results

5.1 Generation Evaluation

The automatic evaluation results are summarized in Table 1. For the Reddit dataset, we compared several variants of RetGen against DialoGPT. gDialoGPT (w/ oracle doc) denotes knowledge-grounded text generation model (described in §3.3) trained using the oracle document. gDialoGPT (w/ random doc) denotes knowledge-grounded text generation model trained using another random document in the same mini-batch. These two mod-

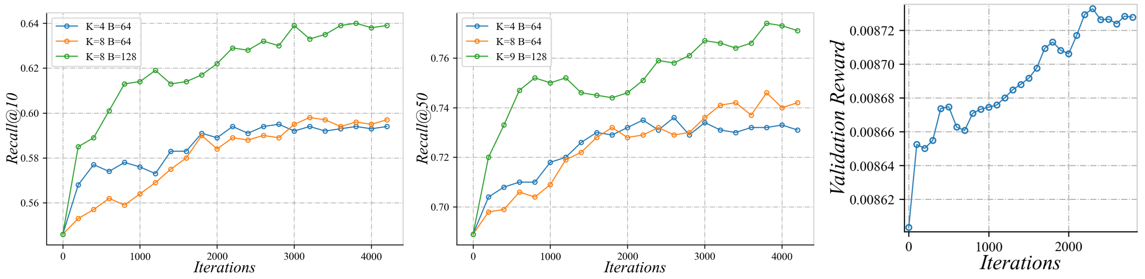
	NIST		BLEU		MET-	Entropy	Dist		Avg. Len.	KMR
Method	N-2	N-4	B-2	B-4	EOR	E-4	D-1	D-2		
***Reddit* dataset**										
DialoGPT	1.59	1.60	12.41%	2.34%	7.23%	8.34	13.2%	32.8%	12.0	-
gDPT (w/ oracle doc)	2.37	2.39	12.58%	2.57%	7.41%	9.04	13.0%	33.2%	15.1	4.8%
gDPT (w/ random doc)	2.03	2.05	10.14%	1.91%	7.12%	9.03	9.9%	27.2%	18.0	2.8%
RetGen ($K = 1$)	2.39	2.41	12.29%	2.32%	7.43%	9.33	14.1%	37.6%	15.6	4.9%
RetGen ($K = 4$)	2.40	2.42	**12.53%**	**2.52%**	7.47%	9.36	14.5%	38.7%	15.3	5.2%
-Φ Optim.	2.37	2.39	11.72%	2.31%	7.63%	9.21	12.9%	34.6%	16.9	4.3%
+MMI	**2.44**	**2.46**	10.98%	1.70%	**8.04%**	**10.30**	**18.6%**	**60.0%**	18.5	**6.3%**
Human oracle	2.13	2.15	13.39%	4.25%	7.34%	9.89	28.2%	77.1%	12.9	5.9%
***arXiv* dataset**										
GPT-2	1.04	1.07	9.85%	3.81%	8.59%	9.34	20.7%	51.3%	18.6	-
RetGen ($K = 1$)	1.81	1.84	11.75%	4.19%	9.04%	9.58	17.5%	46.1%	23.6	3.7%
RetGen ($K = 4$)	**1.82**	**1.86**	**11.85%**	**4.35%**	**9.04%**	9.57	17.5%	46.0%	23.7	3.8%
-Φ Optim.	1.78	1.81	11.79%	4.32%	9.01%	9.56	17.6%	46.4%	23.4	3.7%
+MMI	1.81	1.84	10.84%	3.32%	8.73%	**10.06**	**19.2%**	**59.0%**	28.2	**4.0%**
Human oracle	-	-	-	-	-	9.95	24.7%	71.7%	24.4	-

Table 1: Automatic evaluation results on the Reddit (upper) and arXiv (lower) datasets. gDialoGPT w/ oracle(random) doc denotes training a grounded generation model directly using oracle(random) document. -Φ Optim. denotes only fine-tuning generator parameters Θ while freezing the initial retriever parameters Φ in ANCE. +MMI represents the decoding results post-ranked using MMI.

	Reddit dataset	ArXiv dataset
Context	TIL: All arcade games imported into North America from 1989 to 2000 had the following FBI slogan included into their attract mode: **Winners Don't Use Drugs**.	(Title: It from Knot) **Knot physics** is the theory of the universe that not only unified all the fundamental interactions but also explores the underlying physics of quantum mechanics.
Ground truth target	The Scott Pilgrim vs the World game has one of these on the load screen, but it's the F. V. I. and says winners don't eat meat.	In knot physics, the most important physical result is the unification of everything (including matter, motion, interaction and space-time,...) into the entangled vortex-membranes (a knot).
DialoGPT /GPT	I'm pretty sure that's the slogan of the game in question.	The theory of the knot is a new approach to quantum mechanics.
RetGen	I have a feeling a major part of the reason was **Nixon** was in charge during that period of history.	A knot is a **finite** sequence of **discrete quantum states** that represent the **gravitational field** in a quantum theory.
Retrieved document(s)	**Winners Don't Use Drugs** is an anti-drug slogan that was included in arcade games ... **The slogan was part of a long-term effort** by the United States in its war on drugs started by President Richard **Nixon** in 1971 *(Winners Don't Use Drugs)*	In loop quantum gravity, ... , s-knots represent **the quantum states of the gravitational field** *(S-knot)* Knots have been used for ... Modern physics demonstrates that the **discrete** wavelengths depend on quantum energy levels. ... the Jones polynomial and its generalizations, called the **finite** type invariants... *(History of knot theory)*

Table 2: Generated examples for Reddit (left) and arXiv (right). The relevant parts are **highlighted**, and the title of the most relevant retrieved wikipedia entries are shown in *(article title)*.

els set up the upper and lower bounds of the performance of the grounded generator. For the arXiv dataset, oracle documents not available. Thus, we only compared model variants against GPT-2. For each dataset, we evaluate 4 variants of RetGen: i) **RetGen (K=1)** uses only top-1 retrieved document to generate text; ii) **RetGen (K=4)** uses all top-4 retrieved documents for generation; iii) **-Φ Optim.** is an ablation of RetGen (K=4) where the retriever parameters Φ are frozen during the training; iv) **+MMI** is a variant of RetGen (K=4) using MMI, (§3.5) (Li et al., 2016; Zhang et al., 2018). Following DialoGPT, we first generate 16 hypotheses using top-10 sampling, then rank all hypotheses using reverse model probability of $p(z, x|y)$. The reverse model is also a 345M model fine-tuned from DialoGPT/GPT-2 using the same dataset. The generation yield highest $p(z, x|y)$ are selected for



(a) Recall@10 (Reddit)

(b) Recall@50 (Reddit)

(c) Reward (arXiv)

Figure 3: Recall/Reward on validation set can improve during retriever-only training, where grounded generator Θ is fixed.

evaluation.

We observe that freezing the retriever to pretrained ANCE yield suboptimal evaluation metrics by comparing -Φ Optim. and RetGen (K=4). This implies that retriever fine-tuning is crucial to adapt the retriever to the generation task. Consistent with the observations in Zhang et al. (2020), the MMI re-ranking procedure produces more diverse text and achieves higher NIST and METEOR scores, albeit with a slight drop in BLEU. On the Reddit dataset, RetGen (K=4) achieves comparable performance to RetGen (w/ oracle doc), indicating the retrieved documents are of high quality.

We also compute KMR, which evaluates the ability of the knowledge-grounded text generator to utilize document z to generate text y, as described in §4. For the Reddit dataset, the KMR for gDialoGPT and the human oracle[8] is calculated against oracle document. Otherwise, KMR is calculated against the retrieved documents. If multiple documents are presented, the final KMR is the maximum over those.

We observe that, as expected, RetGen with MMI generally achieves the highest KMR, as it explicitly maximizes the mutual information between the documents and the generation output. For both datasets, RetGen with more documents and with trainable retriever achieves a higher KMR. Note that KMR only reflects how well the generation utilize grounding documents, which may not necessarily be associated with generation performance. However, except for MMI, a higher KMR presumably indicates the model is more effective in leveraging the external document to optimize the LM objectives.

Note that for some metrics the system generated responses can achieve higher score than human responses, which is consistent with the observation in Zhang et al. (2020). As discussed in Zhang et al. (2020), this observation does not imply that the machine generation achieves human parity, but is presumably an artifact of the randomness of human responses in the data.

Generated examples We provide generated examples for both datasets in Table 2. The generation of RetGen are from RetGen (K=4) with MMI. In general, RetGen demonstrates ability to integrate information from difference sources including context and multiple references, and sometimes generate text that reflects multi-hop cross-reference among all the sources. We empirically observe that the retrieved documents are usually relevant and may cover orthogonal aspects of the topics in the context.

Nevertheless, we observe several failure modes in our experiments with RetGen: i) the retrieved passage may not always be relevant and correct. We find that the RetGen can learn to be inclined to avoid using irrelevant documents, but we still see cases where poorly retrieved documents result in incorporation of hallucinations or irrelevant information in final generation; ii) the retrieved passage is relevant, but the grounded generation model may miss correct information and incorporate similar but incorrect/irrelevant information in the generated text (*e.g.*, when asked about who Barack Obama's grandfather was, the system offers his father's name which is also in the retrieved document). These issues do not dominate in our experiments, but resolving them is important and warrants further investigation. We provide exam-

[8] The human oracle is not generated, thus it only provides a reference baseline and may not be comparable with the compared systems. The human oracle KMR for arXiv cannot be computed as the oracle document is not available.

ples of above problems in Appendix B.

We also visualize how the document attention weights $p(z|x, y_{0:t-1})$ change during the generation process in Appendix C. We observed that the attention distribution over documents generally becomes more peaked over time.

Impact of number of documents It can be seen from Table 1 that for both datasets, RetGen with $K = 4$ consistently obtains higher NIST, BLEU and METEOR scores compared with $K = 1$, indicating that incorporating multiple retrieved documents may provide better coverage of the references. We also evaluate RetGen with K ranging from 1 to 4[9]. The results provided in the Appendix A demonstrate monotonic improvement of the automatic metrics when K increases.

5.2 Retrieval Evaluation

From Table 1, it can be seen that jointly training the retriever and generator leads to better automatic metrics, compared with training the generator only. This indicates that the retriever can also benefit from optimizing via the language model signal. However, evaluation using generation metrics in this manner is implicit, as the retriever evaluation and generation evaluation are coupled together. To explicitly assess how the retriever can benefit from joint training, we freeze the generator parameters Θ and only finetune the retriever parameters Φ, and monitor the training process of retriever using either ranking metrics or expected reward in (4).

For the Reddit dataset, since oracle documents are available, we monitor the progress of recall@10 and recall@50 during this retriever-only training. The recall value is computed by averaging over 2,000 validation examples. The total number of passage candidates is 10k [10]. The results are provided in Figure 3 (a) and (b). With fluctuation, the recall generally improves as training progresses. We also observed that increasing the number of documents K from 4 to 8 brought only marginal gain in recall. However, increasing the number of examples in each batch led to more significant improvement of the recall.

For the arXiv dataset, since recall cannot be computed, we instead monitor the expected reward/return ($r = \sum_z p(y|z, x)p(z|x)$). Our reasoning here is that with fixed Θ, if the reward can be improved (*i.e.*, the target y is more likely given the current z), the only possible explanation is that the retrieved documents are more relevant and helpful in predicting the oracle target. We compute the expected reward over 2,000 validation examples. We observed that this reward metric can to some extent improve as training progresses. This verifies that the retriever is being optimized and benefits from language model signals.

5.3 Human Evaluation

We conducted a human evaluation on 500 examples taken from each of the datasets. The evaluation was conducted pairwise, each pair of system outputs being presented to 3 crowdsourced judges in random order. The judges ranked the pairs for coherence, informativeness and human-like properties using a 5-point Likert-like scale. Judges were vetted with a simple qualification test and were checked periodically for spamming.[11] Overall judge preferences in each of the 3 categories are shown in Table 3, where the 5-point scale has been collapsed to a 3-point preference for clarity. A moderate preference can be observed for the variant of RetGen with MMI over vanilla RetGen.

Table 3 suggests that the RetGen may begin to approximate human quality. As has been observed elsewhere, e.g., Zhang et al. (2020), we found that judges often prefer model generation over human responses. In the case of the Reddit dataset, we speculate that the original human responses may be more erratic and idiosyncratic than system outputs. Human evaluation of the arXiv dataset, meanwhile, is intrinsically difficult as responses typically involve domain knowledge: human judges may prefer system generated text that is potentially easier to understand.[12] How to evaluate generated text as systems improve remains a challenge, but further exploration of these issues falls beyond the scope of this work. Further details, including the human evaluation template used, are provided in the Appendix D.

[9] we only test the K up to 4 due to GPU memory constraint.

[10] This includes 2k oracle documents for each instance, and 8k documents that are close to these 2k oracle documents according to BM25.

[11] Held-out from the human text (for positive instances) and random generated text (for negative instances) were used to provide unambiguous cases for spam detection and training examples.

[12] These observations are consistent with the recent findings in Freitag et al. (2021) for Machine Translation to the effect that crowd-sourced human evaluation is error-prone and may not be as accurate as some automatic metrics.

Reddit					arXiv				
System A		Neutral	System B		System A		Neutral	System B	
Coherence: *A and B, which is more relevant to, and coherent with the context?*									
RetGen	**43.7**%	28.3%	28.0%	DialoGPT *	RetGen	**32.1**%	41.7%	26.3%	GPT-2
RetGen	33.3%	28.6%	**38.1**%	MMI	RetGen	29.9%	38.7%	**31.5**%	MMI
RetGen	**40.9**%	22.9%	36.3%	Human *	RetGen	**34.9**%	35.2%	29.9%	Human *
MMI	**45.9**%	23.1%	31.0%	Human *	MMI	**34.9**%	35.8%	29.3%	Human
Informativeness: *A and B, which is more informative (usually more specific content)?*									
RetGen	**44.5**%	27.8%	27.7%	DialoGPT	RetGen	**36.3**%	37.2%	26.5%	GPT-2
RetGen	32.7%	28.3%	**39.0**%	MMI	RetGen	28.9%	37.9%	**33.2**%	MMI
RetGen	**41.1**%	21.5%	37.5%	Human	RetGen	33.2%	32.4%	**34.4**%	Human
MMI	**47.5**%	21.1%	31.4%	Human *	MMI	**34.2**%	34.7%	31.1%	Human
Human-likeness: *A and B, which is more likely to be generated by human rather than a machine?*									
RetGen	**36.4**%	34.0%	29.6%	DialoGPT	RetGen	**29.7**%	43.6%	26.7%	GPT-2
RetGen	31.3%	33.9%	**34.9**%	MMI	RetGen	**28.6**%	42.9%	28.5%	MMI
RetGen	**40.1**%	28.5%	31.4%	Human *	RetGen	**33.7**%	38.9%	27.5%	Human *
MMI	**40.5**%	28.3%	31.1%	Human *	MMI	**33.1**%	38.3%	28.7%	Human

Table 3: Results of **Human Evaluation** for coherence, informativeness and human-text possibility, showing preferences (%) for our model (RetGen) vis-a-vis baselines and real human responses. **RetGen** denotes RetGen with $K = 4$, and **MMI** represents RetGen with MMI. Distributions skew towards RetGen, even when compared with human outputs. Numbers in bold indicate the preferred systems. Statistically significant results with p-value $\leq$ 0.05 are indicated by *.

6 Related Work

Retrieval-Augmented Language Modeling A series of previous work explores a retrieve-then-edit paradigm for text generation (Peng et al., 2019; Li et al., 2018; Cai et al., 2019; Hashimoto et al., 2018; Yang et al., 2019). This line of work either directly edits the retrieved text, or feeds the retrieved text to a fixed generator. REALM (Guu et al., 2020) has proposed a Retrieval-augmented encoder to extract salient text span for open-domain QA. The knowledge retriever is pre-trained by leveraging the masked language model signal. RAG (Lewis et al., 2020b) fine-tunes models that can leverage the Wikipedia documents to facilitate knowledge-intensive NLP tasks, and achieves strong performance on open-domain QA. Shuster et al. (2021) further extend RAG to open-domain response generation and demonstrate the effectiveness of this retrieval-then-generate scheme in hallucination reduction. Our approach differs in that we: 1) focus more on information-aware generation, rather than the information queries; 2) update the document encoder during training, whereas the document encoder in RAG is fixed. Lewis et al. (2020a) proposed a pre-training objective to reconstruct the original document from retrieved evidence documents, and employ the resulting model to improve translation and summarization results. The bulk of recent work has attempted to perform retrieval-augmented generation to either task-oriented (Thulke et al., 2021). However, as we understand it, their retrievers are not optimized during the training, an thus may be unable to learn from the rich language model signals.

Dense Retrieval Models Unlike standard information retrieval techniques such as BM25, Dense Retrieval (DR) models map documents and queries into an embedding space and match them according to semantic similarity. Representative works include (Lee et al., 2019; Karpukhin et al., 2020; Luan et al., 2020; Xiong et al., 2020), which achieve the state-of-the-art performance in tasks like open-domain QA and relevance ranking. Such dense retrieval models can be fine-tuned to accommodate customized needs, and have become a core component in many natural language systems (Khandelwal et al., 2019; Guu et al., 2020).

Grounded Generation Grounded generation based on external knowledge has been extensively studied. Some previous work leverages *structured* external sources like relational knowledge bases (Zhu et al., 2017; Liu et al., 2018) or knowledge graphs (Young et al., 2018) for conversation generation. More recently, Liu et al. (2019) have developed a hybrid graph-path-based method on knowledge graphs augmented with unstructured grounding information. Our work focuses on unstructured (raw text) grounding information and thus avoids the need of preconstructed knowledge graphs. Peng et al. (2020) grounds the task-oriented response generation on the retrieved database states.

Another line of research exclusively uses the *unstructured* grounding. Ghazvininejad et al. (2018) developed a memory network based model to leverage grounding information from Foursquare. Dinan et al. (2019) crowdsourced conversations where each utterance is grounded in no more than a single sentence. Zhou et al. (2018) collected a dataset for grounded conversation generation. Qin et al. (2019) employed a machine reading comprehension (MRC) model to extract salient grounding information to facilitate generation. Wu et al. (2020) used a controllable generation framework to generate dialogue responses by applying extracted lexical constraints. Annotated grounding in these works is often ad-hoc and not necessarily optimal for the task. Our work differs from these in that we jointly train a retriever and generator to optimize grounded text generation performance, and our proposed model does not rely on annotated text-reference parallel data, with the result that it can be trained on any target dataset without additional annotation.

7 Conclusion

We present a joint training framework to simultaneously optimize a dense passage retriever and a knowledge-grounded text generator in an end-to-end fashion. This approach enables leveraging the LM signal to optimize the information retrieval sub-component and thus permits the generation pipeline to output more informative text. The resulting algorithm leverages multiple retrieved documents during decoding time and generates text by selectively summarizing and combining information collected from all the references. We have demonstrated the effectiveness of this algorithm via crowd-sourced human evaluation and automatic evaluation that uses generation and retrieval metrics. In future work, this approach may be combined with contrastive learning and serve as a general pretraining pipeline for grounded generation scenarios. We plan also to leverage QA and cloze task objectives for factuality evaluation (Eyal et al., 2019; Huang et al., 2020).

Acknowledgments

We thank Chenyan Xiong, JJ Liu and Ryen White for their valuable feedback and discussion on this work.

References

Tom B. Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared Kaplan, and Prafulla Dhariwal et al. 2020. Language models are few-shot learners. *arXiv*.

Deng Cai, Yan Wang, Wei Bi, Zhaopeng Tu, Xiaojiang Liu, and Shuming Shi. 2019. Retrieval-guided dialogue response generation via a matching-to-generation framework. In *EMNLP*.

Woon Sang Cho, Yizhe Zhang, Sudha Rao, Asli Celikyilmaz, Chenyan Xiong, Jianfeng Gao, Mengdi Wang, and Bill Dolan. 2020. Contrastive multi-document question generation. In *EACL*.

Colin B. Clement, Matthew Bierbaum, Kevin P. O'Keeffe, and Alexander A. Alemi. 2019. On the use of arxiv as a dataset. *arXiv*.

Mayur Datar, Nicole Immorlica, Piotr Indyk, and Vahab S Mirrokni. 2004. Locality-sensitive hashing scheme based on p-stable distributions. In *Proceedings of the twentieth annual symposium on Computational geometry*, pages 253–262.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *NAACL*.

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 co-occurrence statistics. In *ICHLTR*.

Matan Eyal, Tal Baumel, and Michael Elhadad. 2019. Question answering as an automatic evaluation metric for news article summarization. In *NAACL*, Minneapolis, Minnesota.

Paolo Ferragina and Ugo Scaiella. 2010. TAGME: On-the-fly annotation of short text fragments (by wikipedia entities). In *Proceedings of the 19th ACM International Conference on Information and Knowledge Management*, CIKM '10, page 1625–1628, New York, NY, USA. Association for Computing Machinery.

Markus Freitag, George Foster, David Grangier, Viresh Ratnakar, Qijun Tan, and Wolfgang Macherey. 2021. Experts, errors, and context: A large-scale study of human evaluation for machine translation. *arXiv*.

Michel Galley, Chris Brockett, Xiang Gao, Jianfeng Gao, and Bill Dolan. 2019. Grounded response generation task at dstc7. In *AAAI Dialog System Technology Challenges Workshop*.

Jianfeng Gao, Baolin Peng, Chunyuan Li, Jinchao Li, Shahin Shayandeh, Lars Liden, and Heung-Yeung Shum. 2020. Robust conversational ai with grounded text generation. *arXiv*.

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 *AAAI*.

Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. 2020. Realm: Retrieval-augmented language model pre-training. *arXiv*.

Tatsunori B. Hashimoto, Kelvin Guu, Yonatan Oren, and Percy Liang. 2018. A retrieve-and-edit framework for predicting structured outputs. In *NeurIPS*.

Luyang Huang, Lingfei Wu, and Lu Wang. 2020. Knowledge graph-augmented abstractive summarization with semantic-driven cloze reward. In *ACL*, Online. Association for Computational Linguistics.

Gautier Izacard and Edouard Grave. 2020. Leveraging passage retrieval with generative models for open domain question answering. *arXiv*.

Vladimir Karpukhin, Barlas Oğuz, Sewon Min, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. 2020. Dense passage retrieval for open-domain question answering. *arXiv*.

Urvashi Khandelwal, Omer Levy, Dan Jurafsky, Luke Zettlemoyer, and Mike Lewis. 2019. Generalization through memorization: Nearest neighbor language models. In *ICLR*.

Alon Lavie and Abhaya Agarwal. 2007. Meteor: An automatic metric for mt evaluation with high levels of correlation with human judgments. In *Proceedings of the Second Workshop on Statistical Machine Translation*, pages 228–231. Association for Computational Linguistics.

Kenton Lee, Ming-Wei Chang, and Kristina Toutanova. 2019. Latent retrieval for weakly supervised open domain question answering. In *ACL*.

Mike Lewis, Marjan Ghazvininejad, Gargi Ghosh, Armen Aghajanyan, Sida Wang, and Luke Zettlemoyer. 2020a. Pre-training via paraphrasing. *NeurIPS*.

Patrick Lewis, Ethan Perez, Aleksandara Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, et al. 2020b. Retrieval-augmented generation for knowledge-intensive nlp tasks. *arXiv*.

Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2016. A diversity-promoting objective function for neural conversation models. *NAACL*.

Juncen Li, Robin Jia, He He, and Percy Liang. 2018. Delete, retrieve, generate: A simple approach to sentiment and style transfer. In *NAACL*.

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

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

Yi Luan, Jacob Eisenstein, Kristina Toutanova, and Michael Collins. 2020. Sparse, dense, and attentional representations for text retrieval. *arXiv*.

Barry L Nelson. 1990. Control variate remedies. *Operations Research*, 38(6):974–992.

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

Baolin Peng, Chunyuan Li, Jinchao Li, Shahin Shayandeh, Lars Liden, and Jianfeng Gao. 2020. Soloist: Few-shot task-oriented dialog with a single pretrained auto-regressive model. *arXiv*.

Hao Peng, Ankur P. Parikh, Manaal Faruqui, Bhuwan Dhingra, and Dipanjan Das. 2019. Text generation with exemplar-based adaptive decoding. In *NAACL*.

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 *ACL*.

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

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv*.

Anshumali Shrivastava and Ping Li. 2014. Asymmetric lsh (alsh) for sublinear time maximum inner product search (mips). *NeurIPS*.

Kurt Shuster, Spencer Poff, Moya Chen, Douwe Kiela, and Jason Weston. 2021. Retrieval augmentation reduces hallucination in conversation. *arXiv*.

David Thulke, Nico Daheim, Christian Dugast, and Hermann Ney. 2021. Efficient retrieval augmented generation from unstructured knowledge for task-oriented dialog. *arXiv*.

Ronald J Williams. 1992. Simple statistical gradient-following algorithms for connectionist reinforcement learning. *Machine learning*, 8(3-4):229–256.

Zeqiu Wu, Michel Galley, Chris Brockett, Yizhe Zhang, Xiang Gao, Chris Quirk, Rik Koncel-Kedziorski, Jianfeng Gao, Hannaneh Hajishirzi, Mari Ostendorf, and Bill Dolan. 2020. A controllable model of grounded response generation. *arXiv*.

Lee Xiong, Chenyan Xiong, Ye Li, Kwok-Fung Tang, Jialin Liu, Paul Bennett, Junaid Ahmed, and Arnold Overwijk. 2020. Approximate nearest neighbor negative contrastive learning for dense text retrieval. *arXiv*.

Liu Yang, Junjie Hu, Minghui Qiu, Chen Qu, Jianfeng Gao, W Bruce Croft, Xiaodong Liu, Yelong Shen, and Jingjing Liu. 2019. A hybrid retrieval-generation neural conversation model. In *Proceedings of the 28th ACM International Conference on Information and Knowledge Management*, pages 1341–1350. ACM.

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

Yizhe Zhang, Michel Galley, Jianfeng Gao, Zhe Gan, Xiujun Li, Chris Brockett, and Bill Dolan. 2018. Generating informative and diverse conversational responses via adversarial information maximization. *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, system demonstration*.

Kangyan Zhou, Shrimai Prabhumoye, and Alan W Black. 2018. A dataset for document grounded conversations. In *EMNLP*, pages 708–713.

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*.

Appendix for Joint Retrieval and Generation Training for Grounded Text Generation

A Impact of number of document K

Below are the automatic evaluation results using different K during the decoding time in Table 4, for both Reddit and arXiv datasets. It can be seen that, in general, incorporating more documents yields better automatic metrics.

	NIST		BLEU		MET-	Entropy	Dist		Avg. Len.	KMR
Method	N-2	N-4	B-2	B-4	EOR	E-4	D-1	D-2		
Reddit dataset										
DialoGPT	1.59	1.60	12.41%	2.34%	7.23%	8.34	13.2%	32.8%	12.0	-
gDPT (w/ oracle doc)	2.37	2.39	12.58%	2.57%	7.41%	9.04	13.0%	33.2%	15.1	4.8%
gDPT (w/ random doc)	2.03	2.05	10.14%	1.91%	7.12%	9.03	9.9%	27.2%	18.0	2.8%
RetGen ($K = 1$)	2.39	2.41	12.29%	2.32%	7.43%	9.33	14.1%	37.6%	15.6	4.9%
RetGen ($K = 4$)	2.40	2.42	**12.53%**	**2.52%**	7.47%	9.36	14.5%	38.7%	15.3	5.2%
-Φ Optim.	2.37	2.39	11.72%	2.31%	7.63%	9.21	12.9%	34.6%	16.9	4.3%
+MMI	**2.44**	**2.46**	10.98%	1.70%	**8.04%**	**10.30**	**18.6%**	**60.0%**	18.5	**6.3%**
Human oracle	2.13	2.15	13.39%	4.25%	7.34%	9.89	28.2%	77.1%	12.9	5.9%
arXiv dataset										
GPT-2	1.04	1.07	9.85%	3.81%	8.59%	9.34	20.7%	51.3%	18.6	-
RetGen ($K = 1$)	1.81	1.84	11.75%	4.19%	9.04%	9.58	17.5%	46.1%	23.6	3.7%
RetGen ($K = 4$)	**1.82**	**1.86**	**11.85%**	**4.35%**	**9.04%**	9.57	17.5%	46.0%	23.7	3.8%
-Φ Optim.	1.78	1.81	11.79%	4.32%	9.01%	9.56	17.6%	46.4%	23.4	3.7%
+MMI	1.81	1.84	10.84%	3.32%	8.73%	**10.06**	**19.2%**	**59.0%**	28.2	**4.0%**
Human oracle	-	-	-	-	-	9.95	24.7%	71.7%	24.4	-

Table 4: Automatic evaluation results on the Reddit (upper) and arXiv (lower) datasets with different numbers of retrieved document for decoding time

B Issues with generation

Below, we provide examples where the RetGen can fail to generate text that are factually correct in either retriever or grounded generator in Table 5. Even if these generation outputs are not very common, they still pose important challenges for improving the models.

C Attention Visualization of multi-document MoE decoder

To better understand how the document attention weights $p(z|x, y_{0:t-1})$ are influenced by the documents and existing generated text over the progress of generation, we visualize the attention weights in Figure 4. The context sentence x for the given example is *TIL the 3 in iron man 3 is a reference to the movies iron man and iron man 2. The usage of the number 3 implies that the movies are in chronological order*, and the retrieved top-4 documents are provided in Table 6.

It can be seen that at the initial stage of the generation, the MoE decoder refers to all retrieved documents with relatively even probability. However as the generation become more specific (*e.g.*, mentioning "Shane Black"), the MoE decoder starts to focus more on the first two documents and assigns negligible attention to the documents #3 and #4. We observe that it is typical that during generation, the MoE decoder gradually reinforces the attention to one or two documents by looking at its own existing generation, and the attention distribution becomes more peaked. This typically reduces the likelihood that irrelevant documents (like document #4 in Table 6) will have large impact on generation.

D Additional Details of Human Evaluation

Judges were vetted with a simple qualification test and were checked periodically for spamming. Held-out from the human text (for positive instances) and random generated text (for negative instances) were used

	Retrieved passage is less relevant	Passage is correct, however the model fails to incorporate the right information
Context	TIL Chess Grandmaster Bobby Fischer died at the Age of 64. There are 64 squares on a chess board.	TIL the oldest naval ship still in commission is HMS Victory. She was launched in 1765.
RetGen	And if you ever play chess, you will realize that a board is **a board of 64 pieces**, with one king, one queen, two knights. It is very difficult to play with a board that has so many pieces	It's not in commission anymore. The Royal Navy has retired her **since 1824 because the crew is dead**. It's now a museum.
Retrieved document	The following outline is provided as an overview of and topical guide to chess: Chess is a two-player board game played on a chessboard (a square-checkered board with 64 squares arranged in an eight-by-eight grid). In a chess game, each player begins with sixteen pieces: one king, one queen, two rooks, two knights, two bishops, and eight pawns. The object of the game is to checkmate the opponent's king, whereby the king is under immediate attack (in "check") and there is no way to remove or defend it from attack, or force the opposing player to forfeit *(Outline of Chess)*	HMS "Victory" is a 104-gun first-rate ship of the line of the Royal Navy, ordered in 1758, laid down in 1759 and launched in 1765. She is best known for her role as Lord Nelson's flagship at the Battle of Trafalgar on 21 October 1805. She additionally served as Keppel's flagship at Ushant, Howe's flagship at Cape Spartel and Jervis's flagship at Cape St Vincent. After **1824**, she was relegated to the role of harbour ship. In 1922, she was moved to a dry dock at Portsmouth, England, and preserved as a museum ship *(HMS Victory)*
Issue	The Wiki entry for Bobby Fischer is not the top-1. The generated text deviates from the major topic which is Bobby Fischer. The highlighted text contains hallucination	The highlighted generation either use wrong part of the retrieved document, or hallucinates facts.

Table 5: Failure modes for RetGen.

Document #1	Document #2	Document #3	Document #4
Studios and distributed by Walt Disney Studios Motion Pictures. It is the sequel to "Iron Man" (2008) and "**Iron Man 2**" (2010), and the seventh film in the Marvel Cinematic Universe (MCU). The film was directed by **Shane Black** from a screenplay he co-wrote with Drew Pearce, and stars Robert Downey Jr. as Tony Stark / Iron Man alongside Gwyneth Paltrow, Don Cheadle, Guy Pearce, Rebecca Hall, Stphanie Szostak, James Badge Dale, Jon Favreau, and Ben Kingsley	**Iron Man 2** is a 2010 American superhero film based on the Marvel Comics character Iron Man. Produced by Marvel Studios and distributed by Paramount Pictures, it is the sequel to "Iron Man" (2008) and the third film in the Marvel Cinematic Universe (MCU). Directed by Jon Favreau and written by Justin Theroux, the film stars Robert Downey Jr. as Tony Stark / Iron Man alongside Gwyneth Paltrow, Don Cheadle, Scarlett Johansson, Sam Rockwell, Mickey Rourke, and Samuel L. Jackson	**Iron Man 3** (Original Motion Picture Soundtrack) is the film score for the Marvel Studios film, "Iron Man 3" by Brian Tyler, released on April 30, 2013. A separate soundtrack and concept album titled, Iron Man 3: Heroes Fall (Music Inspired by the Motion Picture) by various artists was released on the same date by Hollywood Records and Marvel Music. Composer Brian Tyler acknowledged that the film's score needed to be darker and more melodic than Ramin Djawadi and John Debney's previous scores, citing the change in Tony Stark's life following the events of "The Avengers" as the catalyst	Men in Black 3 (alternatively Men in Black III, and stylized as "MIB)" is a 2012 American science fiction action comedy film directed by Barry Sonnenfeld and starring Will Smith, Tommy Lee Jones and Josh Brolin. It is the third installment in the "Men in Black" film series which in turn is loosely based on the comic book series "The Men in Black" by Lowell Cunningham. It was released fifteen years after the original "Men in Black" (1997) and ten years after the first sequel, "Men in Black II" (2002)

Table 6: Retrieved documents for the context *TIL the 3 in iron man 3 is a reference to the movies iron man and iron man 2. The usage of the number 3 implies that the movies are in chronological order.*

to provide unambiguous cases for spam detection and training examples. Judges were paid $0.15 per HIT and averaged 99 HITS per hour. This is more than prevailing local minimum wage. In total we paid $5,400 to the crowd-sourced workers. They were told not to attempt the task if they did not wish to be exposed to offensive material.

The instructions and template for human evaluation are provided in Figure5 and Figure 6 below.

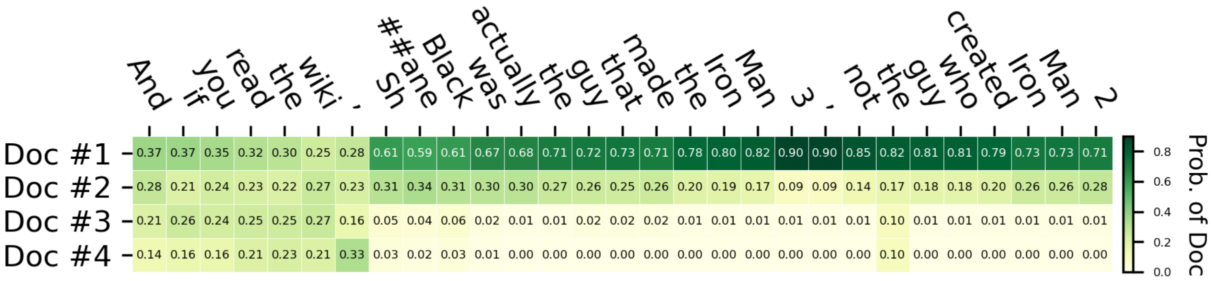
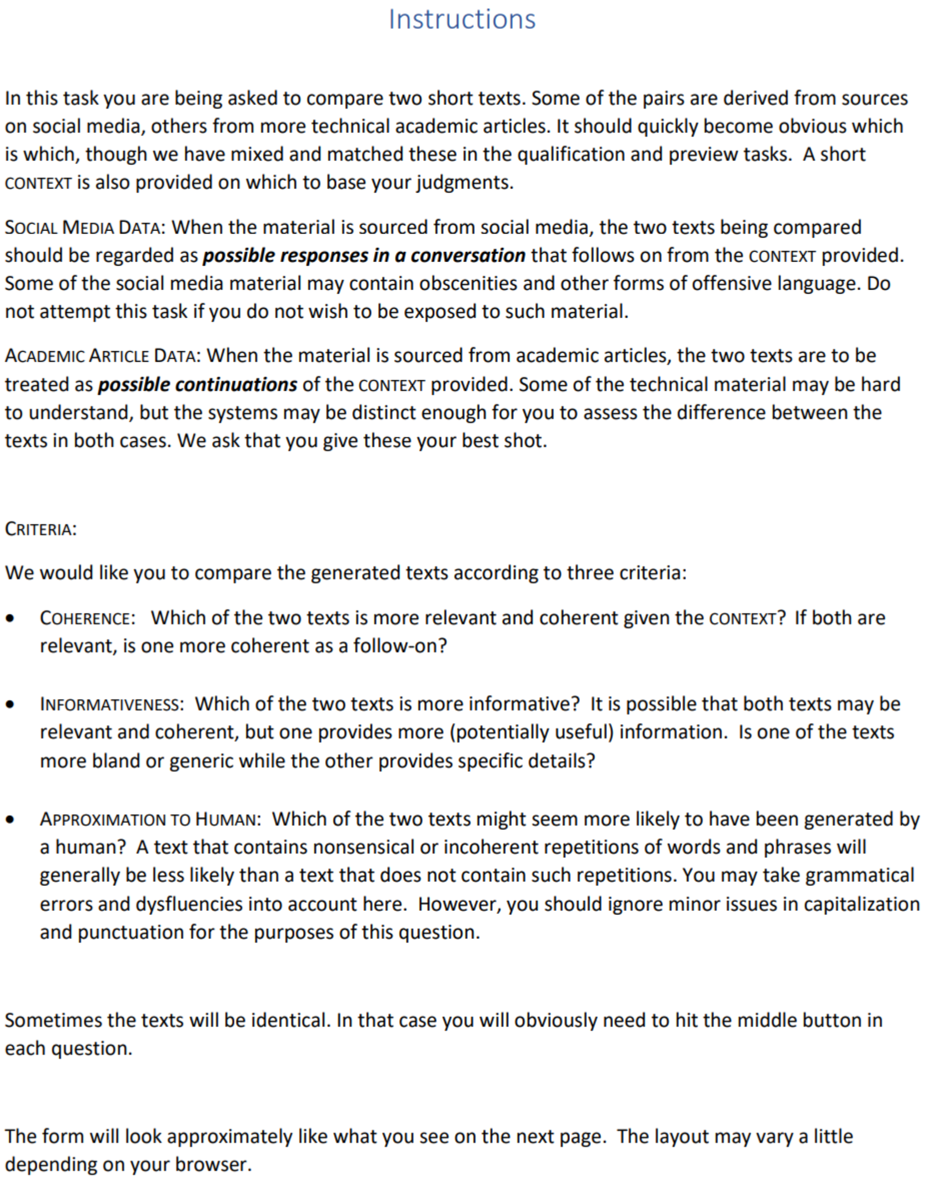


Figure 4: Attention map for multi-document MoE decoder. The context sentence is *TIL the 3 in iron man 3 is a reference to the movies iron man and iron man 2. The usage of the number 3 implies that the movies are in chronological order*, and the resulting MoE generation over 4 documents is *And if you read the wiki, Shane Black was actually the guy that made the Iron Man 3, not the guy who created Iron Man 2*. The retrieved documents are shown in Table 6.



Instructions

In this task you are being asked to compare two short texts. Some of the pairs are derived from sources on social media, others from more technical academic articles. It should quickly become obvious which is which, though we have mixed and matched these in the qualification and preview tasks. A short CONTEXT is also provided on which to base your judgments.

SOCIAL MEDIA DATA: When the material is sourced from social media, the two texts being compared should be regarded as ***possible responses in a conversation*** that follows on from the CONTEXT provided. Some of the social media material may contain obscenities and other forms of offensive language. Do not attempt this task if you do not wish to be exposed to such material.

ACADEMIC ARTICLE DATA: When the material is sourced from academic articles, the two texts are to be treated as ***possible continuations*** of the CONTEXT provided. Some of the technical material may be hard to understand, but the systems may be distinct enough for you to assess the difference between the texts in both cases. We ask that you give these your best shot.

CRITERIA:

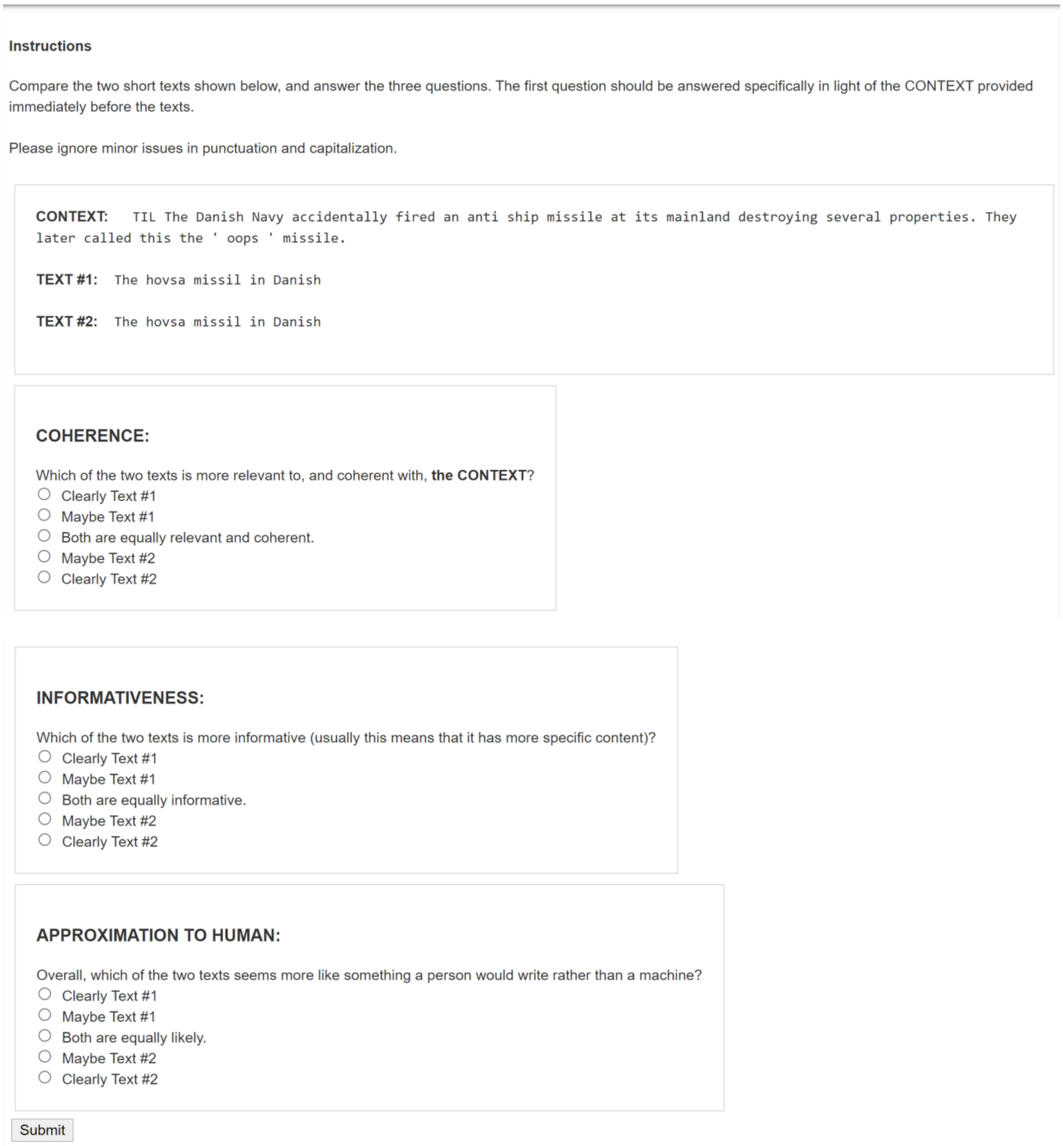
We would like you to compare the generated texts according to three criteria:

- COHERENCE: Which of the two texts is more relevant and coherent given the CONTEXT? If both are relevant, is one more coherent as a follow-on?
- INFORMATIVENESS: Which of the two texts is more informative? It is possible that both texts may be relevant and coherent, but one provides more (potentially useful) information. Is one of the texts more bland or generic while the other provides specific details?
- APPROXIMATION TO HUMAN: Which of the two texts might seem more likely to have been generated by a human? A text that contains nonsensical or incoherent repetitions of words and phrases will generally be less likely than a text that does not contain such repetitions. You may take grammatical errors and dysfluencies into account here. However, you should ignore minor issues in capitalization and punctuation for the purposes of this question.

Sometimes the texts will be identical. In that case you will obviously need to hit the middle button in each question.

The form will look approximately like what you see on the next page. The layout may vary a little depending on your browser.

Figure 5: Human evaluation instructions.



Instructions

Compare the two short texts shown below, and answer the three questions. The first question should be answered specifically in light of the CONTEXT provided immediately before the texts.

Please ignore minor issues in punctuation and capitalization.

CONTEXT: `TIL The Danish Navy accidentally fired an anti ship missile at its mainland destroying several properties. They later called this the ' oops ' missile.`

TEXT #1: `The hovsa missil in Danish`

TEXT #2: `The hovsa missil in Danish`

COHERENCE:

Which of the two texts is more relevant to, and coherent with, **the CONTEXT**?

○ Clearly Text #1
○ Maybe Text #1
○ Both are equally relevant and coherent.
○ Maybe Text #2
○ Clearly Text #2

INFORMATIVENESS:

Which of the two texts is more informative (usually this means that it has more specific content)?

○ Clearly Text #1
○ Maybe Text #1
○ Both are equally informative.
○ Maybe Text #2
○ Clearly Text #2

APPROXIMATION TO HUMAN:

Overall, which of the two texts seems more like something a person would write rather than a machine?

○ Clearly Text #1
○ Maybe Text #1
○ Both are equally likely.
○ Maybe Text #2
○ Clearly Text #2

Figure 6: Human evaluation template.