Knowledge-Consistent Dialogue Generation with Knowledge Graphs

Anonymous Author(s)
Affiliation
Address
email

Abstract

Pre-trained generative language models have achieved impressive performances on dialogue generation tasks. However, when generating responses for a conversation that requires complicated factual knowledge, they are far from perfect, due to the lack of mechanisms to retrieve, encode, and reflect the knowledge in the generated responses. Unlike the methods working with unstructured text that are inefficient in retrieving and encoding the knowledge, some of the knowledge-grounded dialogue generation methods tackle this problem by leveraging the structured knowledge from the Knowledge Graphs (KGs). However, existing methods do not guarantee that the language model utilizes a relevant piece of knowledge for the given dialogue, and that the model generates dialogues which are consistent with the knowledge, from the KG. To overcome this limitation, we propose **SU**bgraph **R**etrieval-augmented **GE**neration (**SURGE**), a framework for generating knowledge-consistent, context-relevant dialogues with a KG. Specifically, our method first retrieves the relevant subgraph from the given KG, and then enforces consistency across the facts by perturbing their word embeddings conditioned on the retrieved subgraph. Then, it learns the latent representation space using graph-text multi-modal contrastive learning which ensures that the generated texts have high similarity to the retrieved subgraphs. We validate the performance of our SURGE framework on the OpendialKG dataset and show that our method does generate high-quality dialogues that faithfully reflect the knowledge from the KG.

1 Introduction

Dialogue systems aim at conversing with humans by generating human-like responses, considering the context and history of the dialogue. Recently, thanks to the development of pre-trained language models (PLMs) for text generation [32, 34], neural dialogue agents are able to generate fluent responses. However, despite their satisfactory fluency, they often generate factually incorrect responses due to a lack of explicit knowledge. The problem can become worse, when the conversation requires accurate knowledge about certain subjects. Thus, to overcome such limitations, some of the recent methods access the external knowledge sources, for example, Wikipedia [7] or Web [21], and then retrieve the documents containing the relevant knowledge for ongoing conversations.

While retrieving the relevant documents from a large-scale text corpus with information retrieval techniques significantly boosts the performance of dialogue agents [18, 24], the computational burden of searching for the relevant documents and embedding them on the fly could be high, which may compromise the responsiveness of the conversation agent. Thus, we instead consider the approach that utilizes the pre-compiled Knowledge Graph (KG) [2, 43] consisting of symbolic facts, which represent the entities as nodes and their relations as edges, in the form of a triplet, e.g., *(Pride & Prejudice, written by, Jane Austen)*. Such KG-augmented dialogue generation models are highly efficient compared to retrieving from and augmenting with unstructured texts. This is because we

Submitted to 36th Conference on Neural Information Processing Systems (NeurIPS 2022). Do not distribute.

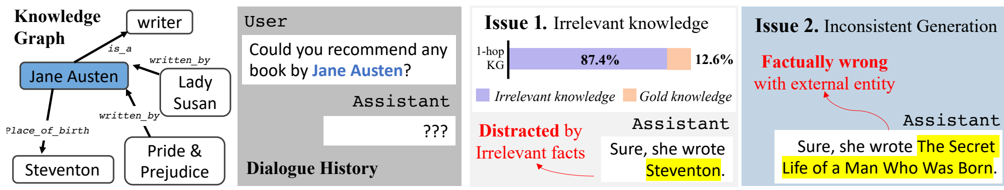

Figure 1: **Motivation.** Existing knowledge-grounded dialogue generation models with KG utilize the multi-hop subgraph for entities in the dialogue context (**Jane Austen**). However, they suffer from the following two problems: **(1) irrelevant knowledge** where only 12.6% of facts from 1-hop KG are useful to generate the target responses given a dialogue context, and **(2) inconsistent generation** including the factually wrong statement.

can directly retrieve entities from the context without searching for all candidate documents from a large text corpus (e.g. Wikipedia), and the retrieved facts succinctly encode the required knowledge in the most compact and effective form. Figure 1 shows an example which illustrates how the KG represents facts as relations among entities, that help generate a knowledge-grounded response.

Few recent works [41, 9, 50] use the KG to provide facts associated with the entities in the dialogue context to the conversation agents. However, they utilize all the triplets associated to the given entity, whose facts are mostly irrelevant to the dialogue context (e.g., Jane Austen was born in Steventon in Figure 1), which could mislead the model into generating factually incorrect responses. We found that about 87% of facts from 1-hop KG are irrelevant to the context in the OpendialKG dataset [29]. Moreover, encoding all the facts including the unnecessary ones is computationally inefficient [9].

Even after correctly retrieving the relevant facts from the KG, it is not straightforward to combine the representations from two heterogeneous modalities: the dialogue context is represented as a text, meanwhile, the knowledge is represented as a graph. Moreover, since the PLMs already have tons of pre-trained parameters trained on the unstructured texts, properly conditioning the structured graph to the PLM is highly important. If not done so, PLMs may generate inconsistent responses with regard to the knowledge from the retrieved subgraph, whose phenomenon is known as hallucination [37, 8] where PLMs generate responses with their own memorized yet incorrect knowledge.

In this work, we tackle such challenging and fundamental issues of knowledge-consistent dialogue generation with KG[1]. In particular, at the first step, we propose a context-relevant subgraph retrieval that retrieves only the relevant triplets from a large KG to prevent the model from generating context-irrelevant responses. Notably, our subgraph retrieval method is end-to-end trainable jointly with the generation objective by marginalizing the likelihood of the generated sentences over the latent variable of the retrieved subgraph [24]. Then, to encode the retrieved subgraph along with the input text sequence, we propose a graph encoding that is permutation and relation inversion invariant yet efficient. Furthermore, to ensure that the model does make use of the encoded knowledge when generating responses, we propose a multi-modal contrastive learning objective between the two different graph-text modalities to enforce the consistency across the retrieved facts and the generated texts. We refer to our framework as **SU**bgraph **R**etrieval-augmented **GE**neration (**SURGE**).

We validate our SURGE framework on the OpendialKG [29] dataset against relevant baselines, with the T5-small [34] model as the base PLM. When evaluating the generated responses from the dialogue agents, the conventional metrics (e.g. BLEU [31], Rouge [27]) can not measure how faithfully the generated responses reflect the world knowledge. Thus, we introduce an additional performance metric, referred to as Knowledge-verifying Question Answering (KQA), which evaluates whether generated responses contain the correct knowledge with an extractive question answering model. The experimental results show that SURGE generates responses that not only agree with the gold knowledge but are also consistent with the retrieved knowledge from the KG.

Our main contributions can be summarized as follows:

- We propose a context-relevant subgraph retrieval method for knowledge graph-augmented dialogue generation, to extract only the relevant piece of the knowledge for the given context from the entire knowledge graph, for generating more appropriate responses to the ongoing conversation.
- We propose an invariant yet efficient graph encoder and a multi-modal graph-text contrastive learning objective to ensure that the generated responses faithfully reflect the retrieved knowledge.
- We validate SURGE against relevant baselines, demonstrating its efficacy in generating responses that are more informative by retrieving and reflecting the relevant knowledge from the KG.

[1] In this work, we denote the knowledge as facts (i.e., a set of triplets) in the knowledge graph.

2 Related Work

Pre-trained Language Model Large Pre-trained Language Models (PLMs) [32, 23, 34] that use the encoder-decoder architecture based on Transformers [42] have achieved great successes on language generation tasks. As they can accurately contextualize the given context and then generate human-like sentences, they are often used as the base architecture for the neural dialogue systems [49, 15]. Moreover, when the PLMs become larger, dialogue generation models have shown to generate high-quality responses [1], suggesting that pre-trained parameters do contain certain knowledge. However, despite the fluency of such PLM-based dialogue agents, they often generate factually incorrect responses that are unfaithful to the dialogue context but look plausible – widely known as a hallucination problem [28]. Thus, generating responses requiring specific and valid factual knowledge is still challenging. To tackle this, recent works propose to retrieve knowledge from external sources, and then use it to augment the neural dialogue agents [37, 40], discussed below.

Knowledge-Grounded Dialogue The sources of external knowledge can be categorized into two types: documents from large unstructured corpora such as Wikipedia [7] or Web [30], and symbolic facts from Knowledge Graphs (KGs) [2, 43]. Firstly, Dinan et al. [7] propose a retrieval-based dialogue generation model, which links the pre-compiled documents retrieved from Wikipedia articles with the given dialogue context using the information retrieval [3]. Further, several works [19, 25, 40] propose to learn the document retrievers in an end-to-end fashion, to generate the knowledge-grounded responses for the given dialogue. However, KG-augmented dialogue generation models, which use structured KGs, are more efficient than the previous methods utilizing unstructured texts thanks to the efficacy of KG for encoding knowledge, and consequently thus more preferable when responsiveness is important [26]. Regarding the dialogue generation with the KG, Moon et al. [29] introduce a knowledge-grounded dialogue dataset where each dialogue comes with the large-scale KG. Before the era of pre-trained language models, several works [41, 45, 48, 5, 50] have suggested sequence-to-sequence models that generate dialogue by conditioning the output word distribution with the entities from the KG. Further, Galetzka et al. [9] propose an efficient way to encode all of the facts in the 1-hop neighbors of the entities that appear in the dialogue history in the given KG, in order to reduce the number of input tokens used in the pre-trained language model [32]. However, all of these methods simply match and retrieve all the facts for entities including irrelevant ones that appear in the dialogue history, which may mislead the agent into generating out-of-context responses. Our work differs from these existing works, since we aim at retrieving only the context-relevant subgraph among the 1-hop facts with a novel subgraph retriever, which is end-to-end trainable along with the dialogue generation model.

3 Method

We first discuss the basic ingredients: graph neural networks and transformers. We then formalize the dialogue generation problem and describe the key components for our **SU**bgraph **R**etrieval-augmented **GE**neration (**SURGE**) framework: context-relevant subgraph retrieval, invariant graph encoding, and graph-text contrastive learning. Figure 2 illustrates the overview of our framework.

3.1 Preliminaries

As we use two different modalities, namely text and graph, we first define them, and then describe the neural networks to encode them. In particular, a text is defined as a sequence of tokens $\boldsymbol{x} = [x_1, ..., x_N], \forall x_i \in \mathcal{V}$, where x_i is a token and $\mathcal{V}$ is a pre-defined vocabulary formed with specific tokenization algorithms [39]. On the other hand, a knowledge graph (KG) is a type of multi-relational graphs $\mathcal{G} = \{(\mathtt{e}_h, \mathtt{r}, \mathtt{e}_t)\} \in \mathcal{E} \times \mathcal{R} \times \mathcal{E}$, where $\mathtt{e}_h$ and $\mathtt{e}_t$ are head and tail entities along with their relation $\mathtt{r}$, and $\mathcal{E}$ and $\mathcal{R}$ are sets of entities and relations, respectively, i.e., $\mathtt{e}_h, \mathtt{e}_t \in \mathcal{E}$ and $\mathtt{r} \in \mathcal{R}$.

To easily access different modalities in the same framework, we define the mapping function that maps entities and relations in the KG to the tokens in the text as follows: $q_e : \mathcal{E} \rightarrow \mathcal{V}^l$ and $q_r : \mathcal{R} \rightarrow \mathcal{V}^l$. In other words, any entity $\mathtt{e} \in \mathcal{E}$ and relation $\mathtt{r} \in \mathcal{R}$ can be mapped to a sequence of l tokens $\boldsymbol{x} \in \mathcal{V}^l$: $q_e(\mathtt{e}) = \boldsymbol{x}_e$ and $q_r(\mathtt{r}) = \boldsymbol{x}_r$. Such functions enable us to associate the KG symbol with the text.

Transformer A Transformer [42] is a neural architecture that embeds a sequence of tokens while taking their relationships into account. It is a basic building block of recent PLMs [6, 32]. Formally, assume that we have a sequence of tokens $\boldsymbol{x} = [x_1, ..., x_N], \forall x_i \in \mathcal{V}$, then a goal of generative transformers is to generate a sequence of tokens $\boldsymbol{y}_{<t} = [y_1, ..., y_{t-1}], \forall y_i \in \mathcal{V}$, with encoder $\mathtt{Enc}$, decoder $\mathtt{Dec}$ and tokens' embedding function f. Thus, a hidden state at time t for generating

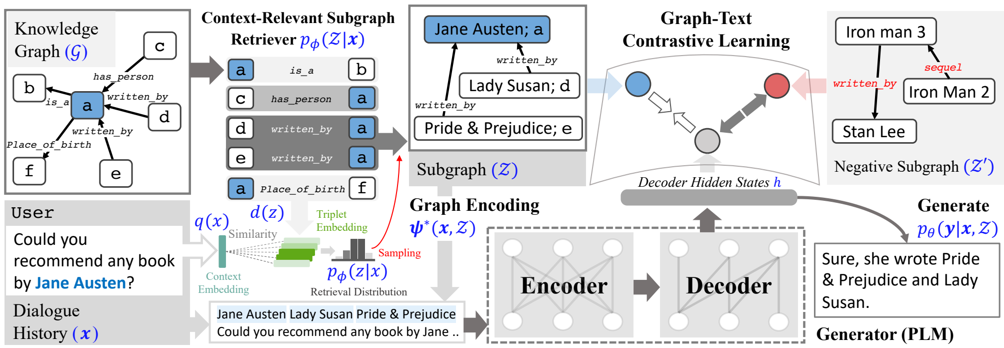

Figure 2: **Framework Overview.** Our framework, SURGE, consists of three parts. First, a context-relevant subgraph retriever $p_\phi(\mathcal{Z}|\boldsymbol{x})$ retrieves the subgraph $\mathcal{Z}$ relevant to the given dialogue history $\boldsymbol{x}$ from a knowledge graph $\mathcal{G}$ (e.g., 1-hop KG from entity *Jane Austen*; a). Specifically, we measure the similarity of a context and triplet embedding to compose the retrieval distribution $p_\phi(z|\boldsymbol{x})$ (§ 3.3). Then, we encode the retrieved subgraph $\mathcal{Z}$ into the input of the generator, using the graph encoding function $\boldsymbol{\psi}(\boldsymbol{x}, \mathcal{Z})$ (§ 3.4). Finally, we use a contrastive learning to enforce the model to generate a consistent response with the retrieved subgraph (§ 3.5).

y_t is $\boldsymbol{h}_t = \texttt{Dec}(\texttt{Enc}(\boldsymbol{X}), \boldsymbol{Y}_{<t})$, where $\boldsymbol{X} = f(\boldsymbol{x}) = [f(x_1), ..., f(x_N)]$ and $\boldsymbol{Y}_{<t} = f(\boldsymbol{y}_{<t}) = [f(y_1), ..., f(y_{t-1})]$. We note that both Enc and Dec functions are **permutation sensitive** with positional encodings as in generic Transformer architecture [42, 46].

Graph Neural Network A Graph Neural Network (GNN) represents a node with its neighboring nodes over the graph structure [12], which is formalized as follows:

$$\boldsymbol{e}_t^{(k+1)} = \texttt{GNN}^{(k)}(\boldsymbol{e}_t^{(k)}; \mathcal{G}) = \texttt{UPD}^{(k)}(\boldsymbol{e}_t^{(k)}, \texttt{AGG}^{(k)}(\{\boldsymbol{e}_h^{(k)} \mid \forall \mathtt{e}_h \in \mathcal{N}(\mathtt{e}_t; \mathcal{G})\})), \tag{1}$$

where $\boldsymbol{e}_t$ and $\boldsymbol{e}_h$ are embeddings of entities (nodes) $\mathtt{e}_t$ and $\mathtt{e}_h$, respectively, $\mathcal{N}(\mathtt{e}_t; \mathcal{G}) = \{\mathtt{e}_h \mid (\mathtt{e}_h, \mathtt{r}, \mathtt{e}_t) \in \mathcal{G}\}$ is a set of neighboring entities of $\mathtt{e}_t$, AGG is a function that aggregates embeddings of $\mathtt{e}_t$'s neighboring entities, and UPD is a function that updates a representation of $\boldsymbol{e}_t$ with the aggregated messages from AGG, at each iteration k.

3.2 Problem Statement

Here we formalize the problem of context-relevant subgraph retrieval for knowledge-grounded dialogue generation. Given a dialogue history $\boldsymbol{x} = [x_1, \ldots, x_N]$, a model with generative PLMs first encodes the input tokens, and then models a probabilistic distribution $p(\boldsymbol{y}|\boldsymbol{x})$ to generate an output response $\boldsymbol{y} = [y_1, \ldots, y_T]$. This problem requires a piece of specific knowledge for a conversation.

To that end, given a dialogue history $\boldsymbol{x}$, we aim at retrieving a subgraph $\mathcal{Z} \subseteq \mathcal{G}$ consisting of a set of triplets $z \in \mathcal{Z}$ where $z = (\mathtt{e}_h, \mathtt{r}, \mathtt{e}_t)$, which encodes relevant knowledge for ongoing conversation. Thus, the distribution of the context-relevant facts $\mathcal{Z}$ is $p(\mathcal{Z}|\boldsymbol{x})$, and our final likelihood of generating responses then becomes $p(\boldsymbol{y}|\boldsymbol{x}, \mathcal{Z})$. Then, to jointly optimize the objective of graph retrieval with response generation, we treat $\mathcal{Z}$ as a latent variable and then marginalize the likelihood of the generative model over all possible latent variables for retrieved subgraphs $\mathcal{Z}$, formalized as follows:

$$p(\boldsymbol{y}|\boldsymbol{x}) = \sum_{\mathcal{Z} \subseteq \mathcal{G}} p_\phi(\mathcal{Z}|\boldsymbol{x})\, p_\theta(\boldsymbol{y}|\boldsymbol{x}, \mathcal{Z}) = \sum_{\mathcal{Z} \subseteq \mathcal{G}} p_\phi(\mathcal{Z}|\boldsymbol{x}) \prod_t^T p_\theta(y_t|\boldsymbol{x}, \mathcal{Z}, \boldsymbol{y}_{1:t-1}), \tag{2}$$

where $p_\phi(\mathcal{Z}|\boldsymbol{x})$ is an output distribution of the context-relevant subgraph retriever, and $p_\theta(\boldsymbol{y}|\boldsymbol{x}, \mathcal{Z})$ is the target distribution of a knowledge-augmented generator, parameterized as ϕ and θ, respectively.

3.3 Context-Relevant Subgraph Retriever

We now provide a concrete description of our context-relevant subgraph retriever formalized in Eq. 2. We assume that a retrieval probability of each triplet in $\mathcal{Z} = \{z_1, \ldots, z_n\}$ is independent. Then, for simplicity, we decompose the probability of retrieving a set of triplets $p(\mathcal{Z}|\boldsymbol{x})$ into the product of individual triplet retrieval probabilities, as follows: $p(\mathcal{Z}|\boldsymbol{x}) = p(z_1|\boldsymbol{x})p(z_2|\boldsymbol{x}) \ldots p(z_n|\boldsymbol{x})$.

From the aforementioned Eq. about $p(\mathcal{Z}|\boldsymbol{x})$, we can now focus on retrieving the only one triplet. Therefore, we define the retrieval of one triplet with an inner product of dense vectors between the dialogue history $\boldsymbol{x}$ and the candidate triplet z, similarly to a dense retrieval model [11], as follows:

$$p_\phi(z|\boldsymbol{x}) \propto \exp(d(z)^\top q(\boldsymbol{x})), \tag{3}$$

where d is a triplet embedding function and q is a dialogue context embedding function. We can use a PLM for implementing q, but we need another effective method for d that can reflect the property of the graph. Therefore, we propose the GNN-based triplet embedding method for realizing d.

Let consider a set of triplets associated to the entities that appear in the given dialogue context $\{(\mathtt{e}, \mathtt{r}, \mathtt{e}_t) \text{ or } (\mathtt{e}_h, \mathtt{r}, \mathtt{e}) \mid q_e(\mathtt{e}) \subseteq \boldsymbol{x}\}$, as the retrieval candidates. To effectively represent the triplets consisting of entities and their relations as items, we use GNNs described in Section 3.1 for the triplet embedding function d. In our triplet retrieval, representing both nodes and edges, which are equally essential components for the multi-relational graph, is worthwhile to represent an entire triplet. To do so, we adopt the existing edge message passing framework [17] that transforms edges of the original graph to nodes of the dual hypergraph [38] (i.e., transforming $\mathcal{G}$ to $\mathcal{G}^*$), which allows us to use existing node-level GNNs for representing edges of the original graph (See Section D.1 of the Supplementary File for more details). Formally, our triplet embedding function is denoted as follows:

$$d(z) = \mathtt{MLP}([\boldsymbol{e}_h \parallel \boldsymbol{r} \parallel \boldsymbol{e}_t]),\ \boldsymbol{e}_h = \mathtt{GNN}(\boldsymbol{e}_h; \mathcal{G}),\ \boldsymbol{r} = \mathtt{GNN}(\boldsymbol{r}; \mathcal{G}^*),\ \boldsymbol{e}_t = \mathtt{GNN}(\boldsymbol{e}_t; \mathcal{G}), \tag{4}$$

where $z = (e_h, r, e_t)$, and $\parallel$ is the concatenation operator.

3.4 Invariant Graph Encoding

In this subsection, we then now specify the remaining operations for $p_\theta(\boldsymbol{y}|\boldsymbol{x}, \mathcal{Z})$, which generates $\boldsymbol{y}$ conditioned on the two different modalities, namely text $\boldsymbol{x}$ and graph $\mathcal{Z}$. Before doing so, we first define the notion of graph encoding, whose goal is to leverage the retrieved subgraph information along with the dialogue history for response generation, which is formalized in Definition 3.1.

Definition 3.1. (Graph Encoding) *Let $\boldsymbol{\psi}(\boldsymbol{x}, \mathcal{Z})$ be a graph encoding function. Then, given a sequence of tokens $\boldsymbol{x} = [x_1, ..., x_N]$ and a subgraph $\mathcal{Z}$, it first yields a new sequence $\boldsymbol{x}' = [x'_1, ..., x'_m, x_1, ..., x_N]$ where $[x'_1, ..., x'_m]$ comes from $q_e(\mathtt{e}) = x'_e$ and $q_r(\boldsymbol{r}) = x'_r$ $\forall\ (\mathtt{e}, \boldsymbol{r}, *) \in \mathcal{Z}$. Then, it embeds a sequence $\boldsymbol{X}' = [f(x'_1), ..., f(x'_m), f(x_1), ..., f(x_N)] = f([x'_1, ..., x'_m, x_1, ..., x_N])$, where f is the token embedding function. Consequently, $\boldsymbol{X}' = \boldsymbol{\psi}(\boldsymbol{x}, \mathcal{Z})$.*

For instance, given a sequence $\boldsymbol{x} = [x_1, \ldots, x_N]$ and a subgraph $\mathcal{Z} = \{(\mathtt{a}, \mathtt{d}, \mathtt{b}), (\mathtt{b}, \mathtt{e}, \mathtt{a}), (\mathtt{a}, \mathtt{d}, \mathtt{c})\}$ from the retriever, $\boldsymbol{\psi}(\boldsymbol{x}, \mathcal{Z}) = f([a, d, b, b, e, a, a, d, c, x_1, ..., x_N])$ with $a = q_e(\mathtt{a})$, $b = q_e(\mathtt{b})$, $c = q_e(\mathtt{c})$, $d = q_r(\mathtt{d})$, $e = q_r(\mathtt{e})$, which we term as the naïve encoding. Due to its simplicity, it is widely used for a text-conditioned generation [24]. However, for graph encoding, it violates two important invariance properties: permutation invariance [47] and relation-inversion invariance, which are formalized in Definition 3.2, 3.3.

Definition 3.2. (Permutation Invariance) *For any set permutation π, $\boldsymbol{\psi}(\boldsymbol{x}, \mathcal{Z}) = \boldsymbol{\psi}(\boldsymbol{x}, \pi \cdot \mathcal{Z})$, i.e., an order of elements in a subgraph does not affect a representation.*

Definition 3.3. (Relation Inversion Invariance) *Let a relation $\neg\mathtt{d}$ be an inverse relation to $\mathtt{d}$, if $(\mathtt{a}, \mathtt{d}, \mathtt{b}) = (\mathtt{b}, \neg\mathtt{d}, \mathtt{a})\ \forall \mathtt{a}, \mathtt{b} \in \mathcal{E}$. Then, $\boldsymbol{\psi}(\boldsymbol{x}, \mathcal{Z} \cup \{(\mathtt{a}, \mathtt{d}, \mathtt{b})\}) = \boldsymbol{\psi}(\boldsymbol{x}, \mathcal{Z} \cup \{(\mathtt{b}, \neg\mathtt{d}, \mathtt{a})\})$ for any subgraph $\mathcal{Z}$.*

Invariant Graph Encoding To meet both properties, we consider two additional operations on a set of triplets up to the naïve encoding. We first define a SORT operator that returns the same output regardless of the order of input set elements, as follows:

$$\mathtt{SORT}(\pi \cdot \mathcal{Z}) = \mathtt{SORT}(\pi' \cdot \mathcal{Z}),\ \forall \pi, \pi' \in S_n, \tag{5}$$

where S_n is a set of all possible permutations for n elements. Moreover, we define a INV operator that adds the inverse triplet of each triplet in the subgraph $\mathcal{Z}$, as follows:

$$\mathtt{INV}(\mathcal{Z}) = \mathcal{Z} \cup \{(\mathtt{e}_t, \neg\mathtt{r}, \mathtt{e}_h) \mid (\mathtt{e}_h, \mathtt{r}, \mathtt{e}_t) \in \mathcal{Z}\}. \tag{6}$$

With above operations, we now define a more solid graph encoding function: $\boldsymbol{\psi}(\boldsymbol{x}, \mathtt{SORT}(\mathtt{INV}(\mathcal{Z})))$, which satisfies both permutation and relation inversion invariance.

Invariant and Efficient Graph Encoding However, above encoding is not efficient since it requires the $\mathcal{O}(n)$ space complexity for encoding a graph with n triplets. To be more efficient, we newly define $\tilde{\boldsymbol{\psi}}$ that only encodes the unique nodes (entities) along the sequence, formalized as follows:

$$\tilde{\boldsymbol{\psi}}(\boldsymbol{x}, \mathtt{SORT}(\mathtt{ENT}(\mathcal{Z}))) = f([a, b, c, x_1, \ldots, x_N]),$$

where $\mathtt{ENT}(\mathcal{Z})$ returns the set of unique nodes in $\mathcal{Z}$ and SORT is used to preserve the permutation invariance. This encoding is thus invariant but efficient since it only costs $\mathcal{O}(k)$, for a k-entity sequence where $k < n$. However, as it does not consider the relational information in $\mathcal{Z}$, we further perturb the entities' token embeddings with respect to their representations in $\mathcal{Z}$. Specifically, for each entity $\mathtt{a} \in \mathtt{ENT}(\mathcal{Z})$, we apply affine transformations from learnable Multi-Layer Perceptrons (MLP) on the token embedding of a as follows:

$$\boldsymbol{\beta}(f(a), \mathcal{Z}) = (1 + \boldsymbol{\gamma}) * f(a) + \boldsymbol{\delta}, \tag{7}$$
$$\boldsymbol{\gamma} = \mathtt{MLP}_1(\boldsymbol{\eta}), \quad \boldsymbol{\delta} = \mathtt{MLP}_2(\boldsymbol{\eta}), \quad \boldsymbol{\eta} = \mathtt{UPD}(f(a), \mathtt{AGGR}(\{f(b), \mathtt{r} \mid \forall \mathtt{b} \in \mathcal{N}(\mathtt{a}; \mathcal{Z})\})),$$

where $\boldsymbol{\beta} : \mathbb{R}^d \rightarrow \mathbb{R}^d$ perturbs the embedding according to $\mathcal{Z}$, AGGR is the relation-aware aggregation function for triplet $(\mathtt{b}, \mathtt{r}, \mathtt{a}) \in \mathcal{Z}$ with $a = q_e(\mathtt{a})$ and $b = q_e(\mathtt{b})$. In sum, we denote a relation-aware invariant and efficient encoder $\boldsymbol{\psi}^*$, formally represented as follows:

$$\boldsymbol{\psi}^*(\boldsymbol{x}, \mathcal{Z}) = \boldsymbol{\beta}(\tilde{\boldsymbol{\psi}}(\boldsymbol{x}, \mathtt{SORT}(\mathtt{ENT}(\mathcal{Z}))), \mathtt{INV}(\mathcal{Z})),$$

where $\boldsymbol{\beta}$ can be applied to the sequence of representations, $\boldsymbol{\beta} : \mathbb{R}^{n \times d} \rightarrow \mathbb{R}^{n \times d}$. We conclude that our graph encoding satisfies both properties. For proofs, please see Section C of the Supplementary File.

3.5 Consistent Generation with Graph-Text Contrastive Learning

Although the previous schemes allow retrieving and encoding subgraphs that are relevant to the input dialogue history, the consistent generation with the given subgraph is further required, when generating responses with the factual knowledge. In other words, the model should be able to generate different sequences given different graphs with the same dialogue history.

However, we only access the single ground-truth response regardless of the retrieved knowledge, while the generative model is trained with a teacher forcing. Thus, this setting can give rise to the problem of *exposure bias* [35]: the model is never exposed to other generated tokens during training. To overcome such limitations, we introduce a novel graph-text contrastive learning method motivated by multi-modal contrastive learning [33]. Formally, for a single pair of a graph and text, the contrastive learning objective is defined as follows:

$$\mathcal{L}_{cont} = \frac{1}{2} \log \frac{\exp(\mathtt{sim}(\zeta(\boldsymbol{z}), \xi(\boldsymbol{h}))/\tau)}{\sum_{\boldsymbol{h}'} \exp(\mathtt{sim}(\zeta(\boldsymbol{z}), \xi(\boldsymbol{h}'))/\tau)} + \frac{1}{2} \log \frac{\exp(\mathtt{sim}(\zeta(\boldsymbol{z}), \xi(\boldsymbol{h}))/\tau)}{\sum_{\boldsymbol{z}'} \exp(\mathtt{sim}(\zeta(\boldsymbol{z}'), \xi(\boldsymbol{h}))/\tau)}, \tag{8}$$

where $\boldsymbol{z} = \frac{1}{m} \sum_{i=1}^{m} \boldsymbol{z}'_i$ is the mean of graph representations from $[\boldsymbol{z}'_1, \ldots, \boldsymbol{z}'_m, \boldsymbol{z}_1, \ldots, \boldsymbol{z}_N] = \mathtt{Enc}(\boldsymbol{\psi}^*(\boldsymbol{x}, \mathcal{Z}))$, $\boldsymbol{h} = \frac{1}{T} \sum_{t=1}^{T} \boldsymbol{h}_t$ is the mean of decoder representations, sim is the cosine similarity, ζ and ξ are learnable linear projection layers, and τ is a learnable temperature parameter. Furthermore, $\sum_{\boldsymbol{h}'}$ and $\sum_{\boldsymbol{z}'}$ indicate the summation over negative samples, which are other texts or graphs within a same mini-batch as in previous contrastive literature [4, 10, 13, 20, 22, 33]. With Eq. 8, the model can embed the correlated pairs closer together in order to generate a consistent response to a given graph, i.e., given a different graph, the model would generate different tokens for the same text.

3.6 Training

We train the entire model by maximizing the log-likelihood $\log p(\boldsymbol{y}|\boldsymbol{x})$ defined in Eq. 2 with respect to parameters of the retriever ϕ and the generator θ. However, computing the marginal probability over all possible subgraphs $\sum_{\mathcal{Z} \subseteq \mathcal{G}} p_\phi(\mathcal{Z}|\boldsymbol{x}) p_\theta(\boldsymbol{y}|\boldsymbol{x}, \mathcal{Z})$ is impractical. Therefore, as in existing works [11, 24], we approximate this by instead summing over k sampled subgraphs. Moreover, for each subgraph, we samples n triplets without replacement from the categorical distribution, parameterized by ϕ: $z_1, \ldots, z_n \sim Cat(|\mathcal{G}|,\ p_\phi(z|\boldsymbol{x}))$, which results in $\mathcal{Z} = \{z_1, \ldots, z_n\}$.

Our whole end-to-end training objective for retrieval-augmented generation is then defined as follows:

$$\mathcal{L}_{ret} = \log \sum_{i=1}^{k} p_\phi(\mathcal{Z}_i|\boldsymbol{x}) p_\theta(\boldsymbol{y}|\boldsymbol{x}, \mathcal{Z}_i),\ \mathcal{Z}_i \sim p_\phi(\mathcal{Z}|\boldsymbol{x}), \tag{9}$$

where we simplify the sampling over n triplets as the sampling over the subgraph distribution $p_\phi(\mathcal{Z}|\boldsymbol{x})$. We assume that we can access the gold subgraph for some data in training. Thus, we further add the supervised retrieval loss to introduce a semi-supervised retriever learning as follows:

$$\mathcal{L}_{sup} = \log p_\phi(\mathcal{Z}^*|\boldsymbol{x}), \tag{10}$$

where $\mathcal{Z}^*$ is the available ground-truth subgraph. Combining all objectives in Eq. 8, 9, and 10, our final training objective is then defined as follows: $\mathcal{L} = \mathcal{L}_{ret} + \mathcal{L}_{sup} + \mathcal{L}_{cont}$.

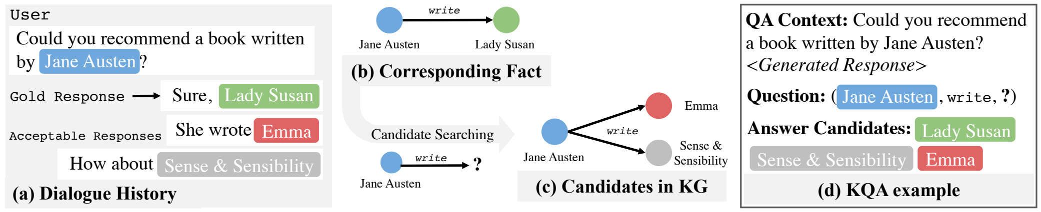

Figure 3: **KQA.** (Left) An example where multiple responses are acceptable but the gold response cannot reflect all of them. (Middle) We first find the fact from the KG that reflects the relation between entities within the user input and gold response (b), and then search candidate facts from the KG (c). (Right) Corresponding KQA example. If a generated response contains the one of answer candidates, the KQA can predict it (success).

4 A Novel Metric: Knowledge-verifying QA

Existing automatic evaluation metrics, namely BLEU and ROUGE [31, 27], are limited in that they only consider the lexical overlaps of words without measuring the factual correctness of the generated responses. As shown in Figure 3 (a), there could be multiple correct responses, but existing metrics score them lower due to the lexical mismatch. To solve this issue, we propose **K**nowledge-verifying **Q**uestion **A**nswering (**KQA**) which measures whether generated responses contain factually correct knowledge given the dialogue history. Compared to the existing metrics using question generation methods [14, 44], we automatically derive QA pairs for evaluation from the dialogue and the large-scale KG [2]. In particular, at first, the ground truth is an entity mentioned in the gold response, and a question is composed of an entity in a dialogue and the existing relation between two entities. Once the question is determined, more answer candidates can be found through the searching in KG, which allows more acceptable responses that contain the correct knowledge as in Figure 3 (d).

5 Experiment

5.1 Experimental Setup

We conduct experiments on the **OpendialKG** dataset [29], which is a dialogue corpus associated with a large-scale Knowledge Graph (KG), namely Freebase [2] with 100k entities and 1M facts. This dataset has 15K dialogues with 91K utterances. We note that 49% of the responses come with the gold knowledge, whereas others are not. Since the dataset does not provide a predefined data split, we randomly split it into train (70%), validation (15%), and test sets (15%). We use **T5-small** [34] for all experiments. We also conduct experiments on another dataset and language model and present the results in Section E of the Supplementary File. For details, see Section D of the Supplementary File.

5.2 Baselines and Our Models

We compare different variants of our SURGE framework against various KG-augmented dialogue generation models. **No Knowledge.** This model is only provided with the dialog history, thus no external knowledge is used. **All Knowledge.** This model is provided with entire facts within a 1-hop subgraph of entities associated with the dialog history. **Gold Knowledge.** This model is provided with the exact gold knowledge, even in the test time if the gold knowledge exists. **Space Efficient Encoding.** This model takes all facts from the 1-hop subgraph of the entities as input. We use two different encoding methods introduced in [9], namely Space Efficient (series) and Space Efficient (parallel). **EARL.** This is an RNN-based model, where the entities are conditioned in response generation [50]. **Random/Sparse Retrieval.** These models are provided with selected facts from a 1-hop subgraph, via the random sampling or the sparse retrieval – BM25 [36]. **Text-based Retrieval.** This model is a variant of our framework where T5 encoder [34] is used for d in Eq. 4 instead of GNNs similar to [16]. **SURGE (unsupervised).** Ours with retrieved context-relevant facts from 1-hop subgraph, where the retrieval is trained without any supervision. **SURGE (semi-supervised).** Ours but the retriever is trained with supervision if the data has a gold fact. **SURGE (contrastive).** Our full model jointly trains the retriever in a semi-supervised manner with the contrastive learning term. By default, all our models are trained with an invariant and efficient graph encoding.

5.3 Evaluation Metrics

We evaluate the generated responses using BLEU [31], ROUGE [27] and unigram overlap (F1) with the gold response. Along with these conventional text evaluation metrics, we also evaluate the results

Table 1: Experimental results on OpendialKG dataset. † indicates the model under the oracle setting using the gold facts even in the test time.

		KQA		BLEU				ROUGE			Unigram
	Method	EM	F1	B-1	B-2	B-3	B-4	R-1	R-2	R-L	F1
Baselines	**No Knowledge**	7.62	13.2	15.79	9.19	5.61	3.43	19.67	7.13	19.02	22.21
	All Knowledge	30.06	34.95	15.95	9.98	6.72	4.65	20.96	8.50	20.21	24.34
	Space Efficient *(series)*	26.88	31.15	16.15	10.03	6.66	4.50	21.15	8.56	20.44	24.55
	Space Efficient *(parallel)*	28.90	33.19	16.33	10.22	6.81	4.64	21.42	8.85	20.68	24.87
	EARL	24.52	27.09	11.49	6.34	4.06	2.75	15.36	4.37	14.61	16.88
Retrieval variants	**Random Retrieval**	21.05	26.09	15.70	9.52	6.12	3.99	20.21	7.88	19.55	23.28
	Sparse Retrieval (BM25)	19.32	24.55	15.63	9.44	6.05	3.96	20.05	7.67	19.37	23.10
	Text-based Retrieval	31.00	35.95	16.87	10.64	7.23	5.07	20.63	8.53	19.89	24.16
Ours	**SURGE** *(unsupervised)*	37.35	42.24	18.10	11.65	7.99	5.59	22.14	9.50	21.23	25.91
	SURGE *(semi-supervised)*	**39.57**	**44.13**	**18.21**	**11.74**	**8.08**	**5.68**	22.11	9.41	21.22	25.91
	SURGE *(contrastive)*	39.52	43.96	17.72	11.53	7.96	5.61	**22.19**	**9.77**	**21.34**	**25.94**
Oracle	**Gold Knowledge**†	49.76	53.41	18.47	12.79	9.32	6.92	24.93	11.97	24.03	28.82
	Gold Response	83.88	86.22	100.0	100.0	100.0	100.0	100.0	100.0	100.0	100.0

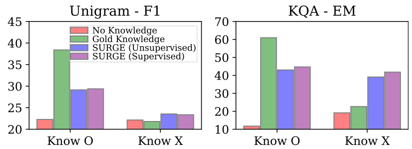

Figure 4: Results of whether gold knowledge exists (Know O) or not (Know X) for the dialogue history. We note that T5 + Gold Knowledge exactly uses the gold knowledge for generating responses.

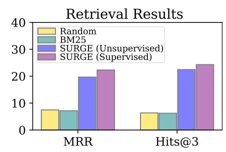

Figure 5: Knowledge retrieval results on the OpendialKG dataset, with MRR and Hits@3.

Table 2: Results on knowledge-consistent response generation, where we compare three variants of our SURGE – unsupervised, semi-supervised and contrastive, on F1 and KF1 as metrics.

Method	F1	KF1
SURGE (unsupervised)	27.78	24.09
SURGE (semi-supervised)	**28.30**	26.38
SURGE (contrastive)	28.17	**27.58**

using our new metric, KQA (Section 4), which measures whether the generated responses contain proper knowledge. Lastly, we compute the Knowledge F1 (KF1) similarly as in Shuster et al. [40], which is a measure of unigram overlap between the retrieved knowledge and generated response.

5.4 Experimental Results and Analysis

In Table 1, we report the knowledge-grounded response generation performances of baselines and our SURGE. As shown in Table 1, our models significantly outperform all the baseline models, excluding the upper bound measure, *Gold Knowledge*, in all evaluation metrics. The high BLEU, ROUGE, and F1 refer that ours sufficiently learns the syntactic and semantic structure of the responses. Our models also achieve high F1 and EM scores in KQA. The high KQA scores indicate that the generated responses are formed with the correct facts, which are relevant to the dialog context. Even the baseline models such as *All Knowledge*, *Space Efficient Encoding* [9], and *EARL* [50], which are provided with all of 1-hop facts, underperform than ours. The result demonstrates that selecting relevant knowledge is critical in knowledge-augmented response generation.

Figure 4 examines the generation performance further by categorizing the dialogues into two groups: those with gold knowledge and those without. When there is no gold knowledge, the model using *Gold Knowledge* suffers a significant drop in all metrics and performs similarly to No Knowledge. On the contrary, ours significantly improves the unigram F1 and KQA scores even with the retrieved knowledge without using the exact gold knowledge.

Knowledge Retrieval Figure 5 shows the performances of knowledge retrieval methods, where we only measure the retrieval performance on dialogues that contain the gold knowledge. We use Mean Reciprocal Rank (MRR) and Hits@k as metrics. Note that our SURGE is a differentiable retriever, which jointly learns to retrieve the context-relevant knowledge and then generate the corresponding responses, whereas *Random* and *BM25* [36] retrieve the knowledge without learning. Therefore, our models outperform other retrieval approaches by a large margin (See Section G with Figure 4 of the Supplementary File for examples of baselines and ours). In addition, when our retriever is trained in a semi-supervised manner, we observe the substantial performance gains from unsupervised learning, as the model can learn to retrieve the ground truth knowledge during training.

Knowledge-Consistent Generation We conduct an ablation study on our models to validate the knowledge consistency performance of the response generation by computing the Knowledge F1 (KF1) score [40]. To focus solely on the case where a given knowledge is consistently reflected in the generated responses, we use the gold knowledge rather than the retrieved one. We randomly perturb

	Context	Gold response	Baseline response	SURGE response
(a)	I loved Moby Dick. Can you recommend something similar?	It was written by Herman Melville in 1851. It's sometimes called The Whale.	Moby Dick is a sailor. Do you like her work?	Moby Dick was written by Herman Melville. He also wrote The Whale.
(b)	Do you know anything the actor Adam Brown?	Yes, he was in the movie The Hobbit: An Unexpected Journey.	Adam Brown starred in King Kong. Have you seen it?	Adam Brown starred in The Hobbit: The Desolation of Smaug and The Hobbit: The Battle of the Five Armies.

(a) Retrieved Subgraph from SURGE	(b) Retrieved Subgraph from SURGE
(Moby Dick; or, The Whale, written_by, Herman Melville)	(The Hobbit: The Battle of the Five Armies, starred_actors, Adam Brown)
(Moby Dick, written_by, Norman Corwin)	(The Hobbit: An Unexpected Journey, starred_actors, Adam Brown)
(Moby Dick, written_by, Ray Bradbury)	(The Hobbit: The Desolation of Smaug, starred_actors, Adam Brown)

Figure 6: Examples of the baseline (Space Efficient, parallel) responses and SURGE responses. On both examples, the baseline generates statements which are factually wrong while SURGE successfully retrieves appropriate facts and generate the good response.

Table 3: Performance comparisons of variants of graph encodings, described in Section 3.4.

Method	KQA EM	KQA F1	Knowledge Length
Naïve	38.18	42.18	62
Invariant	39.54	43.28	117
Efficient (entity only)	38.80	43.06	39
Invariant & Efficient	**39.57**	**44.13**	**39**

Table 4: Human evaluation on **Consis**tency, **Info**rmativeness, and **Fluency**.

Method	Consis.	Info.	Fluency
All Knowledge	**2.52**	1.99	2.62
Space Efficient	2.47	1.75	2.46
SURGE *(ours)*	**2.71**	**2.39**	**2.92**

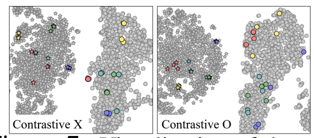

Figure 7: Visualization of the embedding space from our graph(star)-text(circle) contrastive learning.

the gold knowledge to ensure that responses are generated from the given knowledge rather than the trained knowledge. Table 2 shows that our model with a contrastive learning term outperforms all others in the KF1, implying that the generated responses accurately reflect the encoded knowledge.

Sensitive Analysis on Graph Encoding We further conduct an analysis on graph encoding variants introduced in Section 3.4. The knowledge length in Table 3 indicates the average token length used for graph encoding. Our *Invariant & Efficient* ψ^* performs the best against other variants, while using the lesser space at the graph encoding phase. Notably, simple *Invariant* achieves a comparable performance against *Invariant & Efficient*, but yields a longer sequence.

Retrieval and Generation Examples Figure 6 shows the examples of generated responses along with the retrieved knowledge. In particular, we compare our SURGE against *Space Efficient (parallel)* baseline. In example (a), the baseline response contains an incorrect fact distracted by the contextually irrelevant entity 'sailor'. Contrarily, SURGE successfully retrieves relevant facts from the KG then generates the factually correct response. This tendency is similar in example (b), where the baseline incorrectly generates the response with a wrong fact containing 'King Kong', meanwhile our SURGE retrieves context-relevant facts and generates a informative response.

Human Evaluation We sample 30 responses of SURGE, *All Knowledge*, and *Space Efficient* on the OpendialKG test dataset [29], then conduct a human study of them. We recruit 46 annotators, and ask them to evaluate the quality of the generated responses by the 3 models given in a random order, with 3 criteria – consistency, informativeness, and fluency – using a 3 point Likert-like scale. As shown in Table 4, ours obtains significantly (p-value < 0.05) higher scores than others in all criteria, which is another evidence that our framework generates consistent, informative, and fluent responses.

Embedding Space Visualization We further visualize the multi-modal graph-text latent space in Figure 7. The visualization shows that, for the same dialogue with different subgraphs, our SURGE with graph-text contrastive learning (right) generates distinct response embeddings pertraining to different subgraphs, unlike the one without graph-text contrastive learning which shows less variety over responses for the same dialogue (left). We include zoomed Figure 7 in the **Supplementary File**.

6 Conclusion

We proposed a novel end-to-end framework for knowledge graph-augmented dialogue generation which retrieves context-relevant subgraph, encodes a subgraph with the text, and generates knowledge-consistent responses, called as **SU**bgraph **R**etrieval-augmented **GE**neration (**SURGE**). Our results demonstrate the effectiveness of our framework in both quantitative and qualitative experiments in knowledge retrieval and response generation tasks. The analysis shows the contribution of each proposed component: retrieval, encoding, and graph-text representation learning. Our work suggests a new direction to generate informative responses for knowledge graph-based dialogue task by empirically showing the importance of retrieving the more relevant subgraph knowledge rather than using all the relevant knowledge graphs when generating knowledge-grounded responses.

References

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

[2] Kurt D. Bollacker, Colin Evans, Praveen Paritosh, Tim Sturge, and Jamie Taylor. Freebase: a collaboratively created graph database for structuring human knowledge. In *Proceedings of the ACM SIGMOD International Conference on Management of Data, SIGMOD 2008, Vancouver, BC, Canada, June 10-12, 2008*, pages 1247–1250, 2008.

[3] Danqi Chen, Adam Fisch, Jason Weston, and Antoine Bordes. Reading wikipedia to answer open-domain questions. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics, ACL 2017, Vancouver, Canada, July 30 - August 4, Volume 1: Long Papers*, pages 1870–1879. Association for Computational Linguistics, 2017.

[4] Ting Chen, Simon Kornblith, Mohammad Norouzi, and Geoffrey Hinton. A simple framework for contrastive learning of visual representations. In *International conference on machine learning*, pages 1597–1607. PMLR, 2020.

[5] Fuwei Cui, Hui Di, Hongjie Ren, Kazushige Ouchi, Ze Liu, and Jinan Xu. Syntactically diverse adversarial network for knowledge-grounded conversation generation. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih, editors, *Findings of the Association for Computational Linguistics: EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 16-20 November, 2021*, pages 4620–4630. Association for Computational Linguistics, 2021. URL https://doi.org/10.18653/v1/2021.findings-emnlp.394.

[6] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. BERT: pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL-HLT 2019, Minneapolis, MN, USA, June 2-7, 2019, Volume 1 (Long and Short Papers)*, pages 4171–4186, 2019. URL https://doi.org/10.18653/v1/n19-1423.

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

[8] Nouha Dziri, Andrea Madotto, Osmar Zaïane, and Avishek Joey Bose. Neural path hunter: Reducing hallucination in dialogue systems via path grounding. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih, editors, *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 2197–2214. Association for Computational Linguistics, 2021.

[9] Fabian Galetzka, Jewgeni Rose, David Schlangen, and Jens Lehmann. Space efficient context encoding for non-task-oriented dialogue generation with graph attention transformer. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing, ACL/IJCNLP 2021, (Volume 1: Long Papers), Virtual Event, August 1-6, 2021*, pages 7028–7041. Association for Computational Linguistics, 2021.

[10] Jean-Bastien Grill, Florian Strub, Florent Altché, Corentin Tallec, Pierre Richemond, Elena Buchatskaya, Carl Doersch, Bernardo Avila Pires, Zhaohan Guo, Mohammad Gheshlaghi Azar, et al. Bootstrap your own latent-a new approach to self-supervised learning. *Advances in Neural Information Processing Systems*, 33:21271–21284, 2020.

[11] Kelvin Guu, Kenton Lee, Zora Tung, Panupong Pasupat, and Ming-Wei Chang. Retrieval augmented language model pre-training. In *Proceedings of the 37th International Conference on Machine Learning, ICML 2020, 13-18 July 2020, Virtual Event*, volume 119 of *Proceedings of Machine Learning Research*, pages 3929–3938. PMLR, 2020.

[12] William L. Hamilton. Graph representation learning. *Synthesis Lectures on Artificial Intelligence and Machine Learning*, 14(3):1–159, 2020.

[13] Kaiming He, Haoqi Fan, Yuxin Wu, Saining Xie, and Ross Girshick. Momentum contrast for unsupervised visual representation learning. In *Proceedings of the IEEE/CVF conference on computer vision and pattern recognition*, pages 9729–9738, 2020.

[14] Or Honovich, Leshem Choshen, Roee Aharoni, Ella Neeman, Idan Szpektor, and Omri Abend. q^2: Evaluating factual consistency in knowledge-grounded dialogues via question generation and question answering. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih, editors, *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 7856–7870. Association for Computational Linguistics, 2021. URL https://doi.org/10.18653/v1/2021.emnlp-main.619.

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

[16] Samuel Humeau, Kurt Shuster, Marie-Anne Lachaux, and Jason Weston. Poly-encoders: Architectures and pre-training strategies for fast and accurate multi-sentence scoring. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020. URL https://openreview.net/forum?id=SkxgnnNFvH.

[17] Jaehyeong Jo, Jinheon Baek, Seul Lee, Dongki Kim, Minki Kang, and Sung Ju Hwang. Edge representation learning with hypergraphs. *CoRR*, abs/2106.15845, 2021.

[18] Vladimir Karpukhin, Barlas Oguz, Sewon Min, Patrick S. H. Lewis, Ledell Wu, Sergey Edunov, Danqi Chen, and Wen-tau Yih. Dense passage retrieval for open-domain question answering. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing, EMNLP 2020, Online, November 16-20, 2020*, pages 6769–6781. Association for Computational Linguistics, 2020.

[19] Byeongchang Kim, Jaewoo Ahn, and Gunhee Kim. Sequential latent knowledge selection for knowledge-grounded dialogue. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020.

[20] Minseon Kim, Jihoon Tack, and Sung Ju Hwang. Adversarial self-supervised contrastive learning. *Advances in Neural Information Processing Systems*, 33:2983–2994, 2020.

[21] Mojtaba Komeili, Kurt Shuster, and Jason Weston. Internet-augmented dialogue generation. *CoRR*, abs/2107.07566, 2021.

[22] Seanie Lee, Dong Bok Lee, and Sung Ju Hwang. Contrastive learning with adversarial perturbations for conditional text generation. In *International Conference on Learning Representations*, 2021. URL https://openreview.net/forum?id=Wga_hrCa3P3.

[23] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. BART: denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel R. Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 7871–7880. Association for Computational Linguistics, 2020.

[24] Patrick S. H. Lewis, Ethan Perez, Aleksandra Piktus, Fabio Petroni, Vladimir Karpukhin, Naman Goyal, Heinrich Küttler, Mike Lewis, Wen-tau Yih, Tim Rocktäschel, Sebastian Riedel, and Douwe Kiela. Retrieval-augmented generation for knowledge-intensive NLP tasks. In Hugo Larochelle, Marc'Aurelio Ranzato, Raia Hadsell, Maria-Florina Balcan, and Hsuan-Tien Lin, editors, *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.

[25] Linxiao Li, Can Xu, Wei Wu, Yufan Zhao, Xueliang Zhao, and Chongyang Tao. Zero-resource knowledge-grounded dialogue generation. In *Advances in Neural Information Processing Systems 33: Annual Conference on Neural Information Processing Systems 2020, NeurIPS 2020, December 6-12, 2020, virtual*, 2020.

[26] Yu Li, Baolin Peng, Yelong Shen, Yi Mao, Lars Liden, Zhou Yu, and Jianfeng Gao. Knowledge-grounded dialogue generation with a unified knowledge representation. In Marine Carpuat, Marie-Catherine de Marneffe, and Iván Vladimir Meza Ruíz, editors, *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, NAACL 2022, Seattle, WA, United States, July 10-15, 2022*, pages 206–218. Association for Computational Linguistics, 2022. URL https://aclanthology.org/2022.naacl-main.15.

[27] Chin-Yew Lin. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain, July 2004. Association for Computational Linguistics. URL https://aclanthology.org/W04-1013.

[28] Joshua Maynez, Shashi Narayan, Bernd Bohnet, and Ryan T. McDonald. On faithfulness and factuality in abstractive summarization. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel R. Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 1906–1919. Association for Computational Linguistics, 2020.

[29] Seungwhan Moon, Pararth Shah, Anuj Kumar, and Rajen Subba. Opendialkg: Explainable conversational reasoning with attention-based walks over knowledge graphs. In Anna Korhonen, David R. Traum, and Lluís Màrquez, editors, *Proceedings of the 57th Conference of the Association for Computational Linguistics, ACL 2019, Florence, Italy, July 28- August 2, 2019, Volume 1: Long Papers*, pages 845–854. Association for Computational Linguistics, 2019.

[30] Reiichiro Nakano, Jacob Hilton, Suchir Balaji, Jeff Wu, Long Ouyang, Christina Kim, Christopher Hesse, Shantanu Jain, Vineet Kosaraju, William Saunders, Xu Jiang, Karl Cobbe, Tyna Eloundou, Gretchen Krueger, Kevin Button, Matthew Knight, Benjamin Chess, and John Schulman. Webgpt: Browser-assisted question-answering with human feedback. *arXiv preprint*, arXiv:2112.09332, 2021.

[31] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. Bleu: a method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics, July 6-12, 2002, Philadelphia, PA, USA*, pages 311–318. ACL, 2002.

[32] Alec Radford, Jeff Wu, Rewon Child, David Luan, Dario Amodei, and Ilya Sutskever. Language models are unsupervised multitask learners. *OpenAI blog*, 1(8):9, 2019.

[33] Alec Radford, Jong Wook Kim, Chris Hallacy, Aditya Ramesh, Gabriel Goh, Sandhini Agarwal, Girish Sastry, Amanda Askell, Pamela Mishkin, Jack Clark, Gretchen Krueger, and Ilya Sutskever. Learning transferable visual models from natural language supervision. In Marina Meila and Tong Zhang, editors, *Proceedings of the 38th International Conference on Machine Learning, ICML 2021, 18-24 July 2021, Virtual Event*, volume 139 of *Proceedings of Machine Learning Research*, pages 8748–8763. PMLR, 2021.

[34] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *J. Mach. Learn. Res.*, 21:140:1–140:67, 2020.

[35] Marc'Aurelio Ranzato, Sumit Chopra, Michael Auli, and Wojciech Zaremba. Sequence level training with recurrent neural networks. In Yoshua Bengio and Yann LeCun, editors, *4th International Conference on Learning Representations, ICLR 2016, San Juan, Puerto Rico, May 2-4, 2016, Conference Track Proceedings*, 2016.

[36] Stephen Robertson and Hugo Zaragoza. The probabilistic relevance framework: BM25 and beyond. *Found. Trends Inf. Retr.*, 3(4):333–389, 2009.

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

[38] Edward Scheinerman and Daniel Ullman. *Fractional graph theory: a rational approach to the theory of graphs*. Courier Coporation, 2011.

[39] Rico Sennrich, Barry Haddow, and Alexandra Birch. Neural machine translation of rare words with subword units. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 1: Long Papers*. The Association for Computer Linguistics, 2016.

[40] Kurt Shuster, Spencer Poff, Moya Chen, Douwe Kiela, and Jason Weston. Retrieval augmentation reduces hallucination in conversation. In Marie-Francine Moens, Xuanjing Huang, Lucia Specia, and Scott Wen-tau Yih, editors, *Findings of the Association for Computational Linguistics: EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 16-20 November, 2021*, pages 3784–3803. Association for Computational Linguistics, 2021.

[41] Yi-Lin Tuan, Yun-Nung Chen, and Hung-yi Lee. Dykgchat: Benchmarking dialogue generation grounding on dynamic knowledge graphs. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing, EMNLP-IJCNLP 2019, Hong Kong, China, November 3-7, 2019*, pages 1855–1865. Association for Computational Linguistics, 2019.

[42] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, Lukasz Kaiser, and Illia Polosukhin. Attention is all you need. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 5998–6008, 2017.

[43] Denny Vrandecic and Markus Krötzsch. Wikidata: a free collaborative knowledgebase. *Commun. ACM*, 57(10):78–85, 2014.

[44] Alex Wang, Kyunghyun Cho, and Mike Lewis. Asking and answering questions to evaluate the factual consistency of summaries. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel R. Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 5008–5020. Association for Computational Linguistics, 2020. URL https://doi.org/10.18653/v1/2020.acl-main.450.

[45] Sixing Wu, Ying Li, Dawei Zhang, Yang Zhou, and Zhonghai Wu. Diverse and informative dialogue generation with context-specific commonsense knowledge awareness. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel R. Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 5811–5820. Association for Computational Linguistics, 2020. URL https://doi.org/10.18653/v1/2020.acl-main.515.

[46] Chulhee Yun, Srinadh Bhojanapalli, Ankit Singh Rawat, Sashank J. Reddi, and Sanjiv Kumar. Are transformers universal approximators of sequence-to-sequence functions? In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020.

[47] Manzil Zaheer, Satwik Kottur, Siamak Ravanbakhsh, Barnabás Póczos, Ruslan Salakhutdinov, and Alexander J. Smola. Deep sets. In Isabelle Guyon, Ulrike von Luxburg, Samy Bengio, Hanna M. Wallach, Rob Fergus, S. V. N. Vishwanathan, and Roman Garnett, editors, *Advances in Neural Information Processing Systems 30: Annual Conference on Neural Information Processing Systems 2017, December 4-9, 2017, Long Beach, CA, USA*, pages 3391–3401, 2017.

[48] Houyu Zhang, Zhenghao Liu, Chenyan Xiong, and Zhiyuan Liu. Grounded conversation generation as guided traverses in commonsense knowledge graphs. In Dan Jurafsky, Joyce Chai, Natalie Schluter, and Joel R. Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics, ACL 2020, Online, July 5-10, 2020*, pages 2031–2043. Association for Computational Linguistics, 2020. URL `https://doi.org/10.18653/v1/2020.acl-main.184`.

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

[50] Hao Zhou, Minlie Huang, Yong Liu, Wei Chen, and Xiaoyan Zhu. EARL: informative knowledge-grounded conversation generation with entity-agnostic representation learning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing, EMNLP 2021, Virtual Event / Punta Cana, Dominican Republic, 7-11 November, 2021*, pages 2383–2395. Association for Computational Linguistics, 2021.

Checklist

1. For all authors...
   (a) Do the main claims made in the abstract and introduction accurately reflect the paper's contributions and scope? [Yes]
   (b) Did you describe the limitations of your work? [Yes] See Section A of the Supplementary File
   (c) Did you discuss any potential negative societal impacts of your work? [Yes] See Section A of the Supplementary File
   (d) Have you read the ethics review guidelines and ensured that your paper conforms to them? [Yes]
2. If you are including theoretical results...
   (a) Did you state the full set of assumptions of all theoretical results? [N/A]
   (b) Did you include complete proofs of all theoretical results? [Yes] See Section C of the Supplementary File
3. If you ran experiments...
   (a) Did you include the code, data, and instructions needed to reproduce the main experimental results (either in the supplemental material or as a URL)? [Yes] We provide it as the supplemental material.
   (b) Did you specify all the training details (e.g., data splits, hyperparameters, how they were chosen)? [Yes] See Section D of the Supplementary File
   (c) Did you report error bars (e.g., with respect to the random seed after running experiments multiple times)? [N/A]
   (d) Did you include the total amount of compute and the type of resources used (e.g., type of GPUs, internal cluster, or cloud provider)? [Yes] See Section D.3 of the Supplementary File
4. If you are using existing assets (e.g., code, data, models) or curating/releasing new assets...
   (a) If your work uses existing assets, did you cite the creators? [Yes]
   (b) Did you mention the license of the assets? [N/A]
   (c) Did you include any new assets either in the supplemental material or as a URL? [N/A]
   (d) Did you discuss whether and how consent was obtained from people whose data you're using/curating? [N/A]
   (e) Did you discuss whether the data you are using/curating contains personally identifiable information or offensive content? [N/A]
5. If you used crowdsourcing or conducted research with human subjects...

(a) Did you include the full text of instructions given to participants and screenshots, if applicable? [Yes] See Section F of the Supplementary File

(b) Did you describe any potential participant risks, with links to Institutional Review Board (IRB) approvals, if applicable? [N/A]

(c) Did you include the estimated hourly wage paid to participants and the total amount spent on participant compensation? [N/A]