Generating Relevant and Informative Questions for Open-Domain Conversations

2

YANXIANG LING, Science and Technology on Information Systems Engineering Laboratory, National University of Defense Technology, China
FEI CAI, Science and Technology on Information Systems Engineering Laboratory, National University of Defense Technology, China
JUN LIU, Department of Computer Science and Technology, Xi'an JiaoTong University, China
HONGHUI CHEN, Science and Technology on Information Systems Engineering Laboratory, National University of Defense Technology, China
MAARTEN DE RIJKE, University of Amsterdam, The Netherlands

Recent research has highlighted the importance of mixed-initiative interactions in conversational search. To enable mixed-initiative interactions, information retrieval systems should be able to ask diverse questions, such as information-seeking, clarification, and open-ended ones. **Question generation (QG)** of open-domain conversational systems aims at enhancing the interactiveness and persistence of human-machine interactions. The task is challenging because of the sparsity of QG-specific data in conversations. Current work is limited to single-turn interaction scenarios. We propose a ***context-enhanced neural question generation* (CNQG)** model that leverages the conversational context to predict question content and pattern, then perform question decoding. A hierarchical encoder framework is employed to obtain the discourse-level context representation. Based on this, we propose *Review* and *Transit* mechanisms to respectively select contextual

A preliminary version of this work appeared as a short paper in the Proceedings of TheWebConf 2020 [31]. In this extension, we (1) extend the context-enhanced neural question generation model by employing a hierarchical conversational context encoder, proposing a new question content prediction method, optimizing the question pattern prediction method, augmenting the question decoder with a joint attention, designing a multi-task learning based on self-supervised annotations to fully utilize the limited QG-specific training data, and exploring a decaying strategy to automatically set the weights of various loss functions; (2) conduct extensive experiments and provide more detailed discussions, including investigating model performance on question generation, question pattern prediction and question content prediction, adding human evaluation, providing detailed analysis of the multi-task learning based on self-supervised annotations, examining model performance under different context lengths, and providing case studies to illustrate both the positive and negative generated results; and (3) include more related work.
This research was supported by the Hybrid Intelligence Center, a 10-year program funded by the Dutch Ministry of Education, Culture and Science through the Netherlands Organisation for Scientific Research, https://hybrid-intelligence-centre.nl.
Authors' addresses: Y. Ling, Science and Technology on Information Systems Engineering Laboratory, National University of Defense Technology, No. 109, Deya Street, Changsha, Hunan, China; email: lingyanxiang@nudt.edu.cn; F. Cai (corresponding author) and H. Chen, Science and Technology on Information Systems Engineering Laboratory, National University of Defense Technology, No. 109, Deya Street, Changsha, Hunan, China; emails: caifei08@nudt.edu.cn, chenhonghui@nudt.edu.cn; J. Liu, Department of Computer Science and Technology, Xi'an JianTong University, No. 28, Xianning Street, Xi'an, Shaanxi, China; email: liukeen@xjtu.edu.cn; M. de Rijke, Information Institute, University of Amesterdam, Science Park 608B, 1098 XH Amsterdam, The Netherlands; email: m.derijke@uva.nl.

1046-8188/2023/01-ART2 $15.00
https://doi.org/10.1145/3510612

keywords and predict new topic words to further construct the question content. Conversational context and the predicted question content are used to produce the question pattern, which in turn guides the question decoding process implemented by a recurrent decoder with a joint attention mechanism. To fully utilize the limited QG-specific data to train our question generator, we perform multi-task learning with three auxiliary training objectives, i.e., question pattern prediction, *Review*, and *Transit* mechanisms. The required additional labeled data is obtained in a self-supervised way. We also design a weight decaying strategy to adjust the influences of various auxiliary learning tasks. To the best of our acknowledge, we are the first to extend the application of QG to the multi-turn open-domain conversational scenario. Extensive experimental results demonstrate the effectiveness of our proposal and its main components on generating relevant and informative questions, with robust performance for contexts with various lengths.

CCS Concepts: • **Information systems → Users and interactive retrieval;**

Additional Key Words and Phrases: Conversational search, neural question generation, open-domain conversations, context modeling

ACM Reference format:
Yanxiang Ling, Fei Cai, Jun Liu, Honghui Chen, and Maarten de Rijke. 2023. Generating Relevant and Informative Questions for Open-Domain Conversations. *ACM Trans. Inf. Syst.* 41, 1, Article 2 (January 2023), 30 pages.
https://doi.org/10.1145/3510612

1 INTRODUCTION

Open-domain conversational systems, also known as chit-chat dialogue systems, aim to converse with humans on various open-ended topics to maximize long-term user engagement [20]. They can be applied to provide natural human-machine interactions for conversational retrieval systems, and can also directly provide support for information-seeking activities [2, 11, 23, 45].

For an open-domain conversational system, asking questions is a necessary social skill as it can be used to provide suggestions, extend discussed topics, and solicit user feedback [61, 63], all of which serves to enhance dialogue engagement and achieve persistent multi-turn interactions. As a special kind of response, questions can be generated by ordinary ***response generation*** **(RG)** methods [48, 67, 71] or traditional ***question generation*** **(QG)** methods [14, 15, 18, 37–39, 52] for **machine reading comprehension (MRC)**. However, there is a clear demand to design QG models in the context of open-domain conversations due to two reasons:

— Given a conversational history, ordinary RG models are able to generate a question. However, this happens with random probabilities, meaning that we cannot control the form of responses to-be-generated. There is no large-scale $\langle context, question \rangle$ data in ordinary chat corpora [47], where RG methods may not be trained sufficiently to have ideal performance on generating questions.
— In MRC, QG aims to augment the training data for the reverse task, i.e., question answering, so the question is usually a factoid one and its answer is limited to a small scope of the given text. However, in open-domain conversations, the purpose of QG is to enhance dialogue engagement, so questions are required to be more diverse and flexible. This means that questions of open-domain conversations do not always have unique answers owing to rich language expressions, and fresh topics that have not appeared yet but are related to conversation are encouraged to come in. In addition, in question generation for MRC, the input text is usually informative and contains many entities, which is quite different from that in open-domain conversations. Chat texts are colloquial, sometimes inconsistent in word

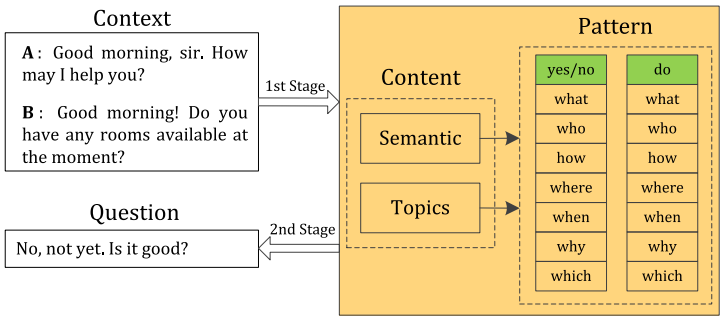

Fig. 1. Illustration of the two-stage process underlying CNQG.

expressions and ambiguous in semantics. This increases the difficulty for traditional QG to generate relevant and informative questions in open-domain conversations [27, 31, 61].

Despite a clear demand on pushing toward asking good questions in open-domain conversations, so far, the volume of work on the task appears to be limited. Several researchers have studied the problems of preference elicitation [e.g., 11, 45] and query clarification in the context of conversational search and recommendation [e.g., 2, 22]. Wang et al. [63] and Wang et al. [61] focus on QG in the setting of open-domain conversations but their approaches only consider a user's latest utterance to generate a question, ignoring the previous conversational history. In the multi-turn scenario, generating a random or free-style question without considering its context is not useful for enhancing conversational engagement [20, 59].

Our work focuses on leveraging conversational context to generate relevant and informative questions, which can extend the application of QG to the multi-turn open-domain conversational scenario. We introduce auxiliary learning tasks from conversational context modeling to make full use of the limited QG-specific data to train our proposed QG model in a multi-task learning framework. Our efforts lead to a ***context-enhanced neural question generation*** **(CNQG)** method that employs the idea of a *two-stage process* to first identify the question content and pattern, and then to generate a question, as shown in Figure 1. The CNQG model is implemented in an *end-to-end* manner; its input is the conversational context and output is the question.

To be specific, a hierarchical context encoder is adopted to obtain the semantic representation of conversational context. For question content, we design two parallel mechanisms, i.e., *Review* and *Transit*, to produce relevant topics. The *Review* mechanism focuses on selecting keywords that are worth being asked from the conversational context; this is expected to control the semantical consistence of the generated question. The *Transit* mechanism is designed to introduce new topics from a candidate corpus that has been defined using **point-wise mutual information (PMI)** by measuring coherence to the conversational context; this is expected to help extend the discussed topics and promote conversation. The context representation and the identified question content are jointly used for question pattern prediction. Driven by the predicted question pattern, the question is generated by a recurrent decoder augmented with a joint attention over the conversational context and the topic representations output from the *Review* and *Transit* mechanisms.

During training, the CNQG model is enhanced with a *multi-task learning* framework. We produce labeled data for the question pattern prediction, *Review*, and *Transit* mechanisms in a self-supervising way. Based on this, multi-task learning is used to improve the use of existing training data and boost the performance of the QG process. To improve the joint training process, we also design a decaying strategy to allow the loss weights of auxiliary learning tasks to adapt to changes in the corresponding losses.

To examine the effectiveness of the proposed CNQG model, we conduct extensive experiments on two human-to-human chat corpora, i.e., DailyDialog and PersonaChat. Experimental results demonstrate that CNQG outperforms various competitive baselines in terms of both automatic and human evaluations. It shows robust performance across conversational contexts of different lengths. We also perform detailed analyses on the predictions of the question pattern and content, and find that CNQG produces accurate patterns and highly relevant topic words. This helps to explain why CNQG performs well. An analysis of the multi-task learning behavior on self-supervised annotation highlights the distinct contributions of the auxiliary learning tasks and validates the effectiveness of the proposed loss weight decaying strategy.

The main contributions of our work can be summarized as follows:

— To the best of our acknowledge, we are the first to extend the application of QG to the multi-turn open-domain conversational scenario. We propose a context-enhanced neural question generation (CNQG) CNQG model that contains two crucial mechanisms, i.e., *Review* and *Transit*, to leverage conversational context for generating relevant and informative questions.
— To fully utilize the limited amount of QG-specific training data in the open-domain conversational corpora, we enhance CNQG with multi-task learning on self-supervised annotations and design a weight decaying strategy to adjust the influence of various auxiliary learning tasks.
— We conduct extensive experimental tests to examine the effectiveness of CNQG and its major components; we find that it outperforms the state-of-the-art models in both automatic and human evaluations while being robust to varying conversational context lengths.

2 RELATED WORK

We introduce related work along four dimensions: conversational systems in information retrieval, QG in open-domain conversations, conversational context modeling, and multi-task learning.

2.1 Conversational Systems in Information Retrieval

Human-machine conversation has attracted increasing attention due to its promising potential and societal impact [7]. The idea of viewing information retrieval systems as conversational systems has been around at least since the 1980s [4, 13]. After a long period of limited research activity, the topic of information-retrieval-as-conversation has seen considerable growth recently.

Some recent work focuses on the creation of theoretical frameworks that help to identify research directions as well as commonalities amongst algorithmic solutions [3, 41]. Other work examines the user experience and user expectations with conversational information retrieval systems [see, e.g., 56, 58]. To facilitate progress in the development of conversational search systems and to aid in the exploration of new conversational search scenarios, a growing number of datasets has been released [57], for conversations based on **search engine result pages** (**SERPs**) [44], conversational browsing [60], and spoken search interactions [55].

Of special interest to our work in this article is the algorithmic work that is aimed at making conversations more natural and engaging, through personalization, topic planning, knowledge grounding, or the addition of empathy. For instance, Zhang et al. [73] make conversations more engaging by conditioning them on pre-defined persona information. Wang et al. [62], Xing et al. [66], and Ling et al. [32] apply topic modeling and topic transitions to enhance the informativeness of a conversation. Vakulenko et al. [59] and Zhang et al. [72] analyze and organize conversations by reasoning over a commonsense graph. Rashkin et al. [43] incorporate emotions into open-domain conversations.

Our work focuses on generating relevant and informative questions to help enhance the engagement of conversational systems. It can be applied to support more natural human-machine interactions for conversational IR systems, and can also solicit user feedback or clarify user intent to enhance retrieval performance.

2.2 Question Generation for Conversations

Questions occur frequently in natural conversations, and asking questions is a necessary social skill for conversational systems. We classify questions in conversations into three major types, i.e., *information-seeking*, *clarification*, and *open-ended*, and review the related work for each type.

Information-seeking questions generally have specific goals and focus on soliciting feedback from users, which can be used to elicit users' preference for personalized search and recommendation. For instance, Zhang et al. [75] devise a "system ask-user respond" paradigm for conversational search, and design a memory network for product search and recommendation. Lei et al. [25] build a framework to achieve deep interactions between recommendation and conversation. Bi et al. [5] propose a conversational paradigm for product search, and an aspect-value likelihood model to incorporate feedback on non-relevant items. Besides IR systems, information-seeking questions can also be applied to the field of MRC. Previous research has studied the task of generating a series of interconnected questions to perform information seeking on a given document passage through a question-answering style conversation [18, 37–39].

Clarification questions are aimed at asking about missing information in the context. They have broad applications in practice, for example guiding users to complete a query in search engine, mining the intent of interlocutors in a conversation, and so on. Aliannejadi et al. [2] and Zamani et al. [70] formulate the task of asking clarifying questions in conversational search. Aliannejadi et al. [1] propose generating clarifying questions for open-domain dialogue systems. Xu et al. [68] study asking clarification questions in knowledge-based question answering.

As part of a conversation, open-ended questions have no strict restrictions on their answers; they are used to enhance dialogue engagement and trigger more interactions. Wang et al. [63] generate questions to keep open-domain conversations interactive and persistent; they design a typed decoder to first predict word type and then conduct generation. Wang et al. [61] leverage the semantic coherence between question and answer to enhance QG; they use a coherence score as a reward function, and incorporate reinforcement learning and generative adversarial networks into a conditional variational auto-encoder.

All three kinds of question occur naturally in the context of a conversation. They provide different ways for machines to interact with humans. This article focuses on generating relevant and informative questions for open-domain conversations. Our prior work [31] introduces the conversational context to produce questions for open-domain conversations; the PMI-based question content prediction that we have previously proposed is susceptible to semantic noise. Thus, in this article, we optimize our modeling of the conversational context by employing a hierarchical framework and designing two different mechanisms, i.e., *Review* and *Transit*, that can produce relevant and informative topic words as question content. Moreover, we leverage multi-task learning to alleviate the sparsity of QG-specific data in chat corpora.

2.3 Conversational Context Modeling

In conversational systems, the notion of context is multi-dimensional, as it may concern persona, emotion, physical, or linguistic environments. In our work, we particularly focus on the linguistic context, i.e., the conversational history. Current approaches to conversational context modeling can be grouped into two types, non-hierarchical and hierarchical.

Non-hierarchical approaches to conversational context modeling process the conversational context as a whole, concentrating on word-level semantic and sequential relations. Early work directly concatenates historical contextual utterances into a sentence and adopts **recurrent neural Networks (RNNs)** to obtain a context representation [50, 51]. These approaches may be challenged by the long-term dependency problem that RNNs face. Thus, some researches use transformer-based architectures as they better display and are better at representing long sentences than RNNs for context modeling. For instance, DialoGPT [76] and T5 [42] both adopt the transformer as a basic encoding unit and have achieved impressive progress on context modeling. An essential problem of non-hierarchical context modeling is that it ignores the semantic relations within contextual utterances, which can actually reflect dynamic topic flow across a conversation.

Hierarchical approaches to conversational context modeling represent the conversational context at both the utterance and discourse level. Serban et al. [48] presents the first hierarchical approach, HRED, that first uses an RNN to get the embedding of each utterance and then employs another RNN to integrate the utterance embeddings into a context representation. Based on this, HRED has been combined with memory networks [8], latent variable models [49], and conditional auto-encoders [77]. To capture the distinct influence of each utterance in a context, some models [such as 67, 71] utilize attention mechanisms to enhance the hierarchical context modeling. Compared to non-hierarchical approaches, the advantage of hierarchical approaches is that they consider the discourse semantics contained in the context, which can allow one to explore conversational dynamics such as topic transitions across multi-turn interactions. In our work, we adopt a hierarchical framework to produce a context representation, which is then used to conduct the QG.

2.4 Multi-Task Learning

Multi-task learning (MTL) is meant to help a model generalize better on a given task by sharing representations and jointly training with related tasks [6, 34]. MTL is an implicit data augmentation method. It can leverage supervised labels from auxiliary tasks, which can provide a deeper analysis of existing data, especially in the setting of limited labeled data. In QG, Zhou et al. [79] propose to employ language modeling to enhance QG by MTL. Duan et al. [16] and Tang et al. [54] leverage the intrinsic connections between **question answering (QA)** and QG, and propose to improve QA with QG. In our work, we employ self-supervised annotations on the existing training data to obtain additional labeled data, and then introduce related learning tasks to enhance the final QG. Moreover, we design a loss weight decaying strategy to balance the influence of various training objectives.

On top of the related work discussed above, we add the following: (1) we initially apply QG to obtain engaging conversations in the setting of multi-turn scenario by proposing two efficient mechanisms, i.e., *Review* and *Transit*, for context utilization. (2) We alleviate the data sparsity issue existing in QG of open-domain conversations through multi-task learning on self-supervised annotations.

3 APPROACH

The task of QG in multi-turn open-domain conversations can be defined as follows: given conversational context $\mathcal{X} = \{U_1, \ldots, U_{|\mathcal{X}|}\}$ consisting of $|\mathcal{X}|$-length utterances, the model should generate a relevant and informative question Q by computing the conditional probability $P(Q \mid \mathcal{X})$. Following previous work [16, 19, 39, 80], the essence of question Q includes two parts, i.e., *question content* Q_c and *question pattern* Q_p. Thus, $P(Q \mid \mathcal{X})$ can be approximated through a two-stage process: (1) identifying Q_c and Q_p based on $\mathcal{X}$; and (2) decoding Q word-by-word based on Q_c, Q_p, and $\mathcal{X}$.

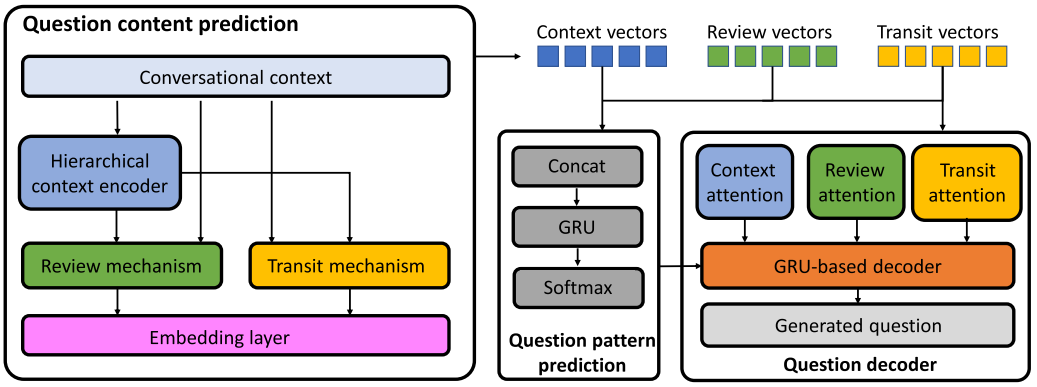

Fig. 2. Overview of the context-enhanced neural question generation (CNQG) framework.

Figure 2 provides a high-level overview of the proposed *context-enhanced neural question generation* (CNQG) CNQG model, which consists of three major components, i.e., (1) a *question content prediction* module (see Section 3.1) to generate topic words that are worth being asked, i.e., Q_c, by *Review* and *Transit* mechanisms; (2) a *question pattern prediction* module (see Section 3.2) to classify the question to-be-generated into a certain pattern Q_p conditioned on the topic words Q_c as well as the conversational context $\mathcal{X}$; and (3) a *question decoder* module (see Section 3.3) to implement the surface realization of the question through a recurrent decoder and a joint attention mechanism over $\mathcal{X}$, Q_c, and Q_p. CNQG implements the above modules in an end-to-end framework; it attempts to minimize the amount of manual work required while providing reasonable model interpretability through a two-stage generation process.

3.1 Question Content Prediction

The question content prediction module of the CNQG framework (the leftmost block in Figure 2) aims to predict what topics will be discussed in the question to-be-generated; this is the most important step in question generation. The set of topics discussed during a natural, multi-turn conversation tends to grow [20, 62]. Hence, the question content predictor should achieve an effective balance between selecting existing topics so as to maintain coherence and introducing fresh topics to further the conversation. The question content Q_c in CNQG has two parts: *review topics* and *transit topics*. We leverage the conversational context to produce candidate existing topics and transit topics, and further utilize a semantic representation of the conversational context to select the question content based on these candidates.

3.1.1 Hierarchical Context Encoder. Given a conversational context $\mathcal{X} = \{U_1, \ldots, U_{|\mathcal{X}|}\}$, the hierarchical context encoder [48] first employs an *utterance encoder* to obtain a vector representation of each utterance in $\mathcal{X}$, and then integrates these utterance representations through a *context encoder*.

Specifically, given $U_i = \{w_{1,i}, \ldots, w_{N_i,i}\}$ ($U_i \in \mathcal{X}$), the utterance encoder employs a **bidirectional gated recurrent unit** (**BiGRU**) [10] to convert each word $w_{n,i}$ ($n \in [1, N_i]$) into a hidden vector $\mathbf{h}^{utt}_{n,i}$ as follows:

$$\begin{cases} \overrightarrow{\mathbf{h}^{utt}_{n,i}} = \overrightarrow{\text{BiGRU}}(\overrightarrow{\mathbf{h}^{utt}_{n-1,i}}, \mathbf{e}_{w_{n,i}}), \\ \overleftarrow{\mathbf{h}^{utt}_{n,i}} = \overleftarrow{\text{BiGRU}}(\overleftarrow{\mathbf{h}^{utt}_{n-1,i}}, \mathbf{e}_{w_{n,i}}), \\ \mathbf{h}^{utt}_{n,i} = \overrightarrow{\mathbf{h}^{utt}_{n,i}} + \overleftarrow{\mathbf{h}^{utt}_{n,i}}, \end{cases} \tag{1}$$

where $\mathbf{e}_{w_{n,i}}$ is the initialized word embedding of $w_{n,i}$; and $\overrightarrow{\mathbf{h}_{n,i}^{utt}}$ and $\overleftarrow{\mathbf{h}_{n,i}^{utt}}$ are the respective hidden vectors of $w_{n,i}$ for the forward and backward passes. Then, the context encoder uses a unidirectional **gated recurrent unit** (**GRU**) [9] to obtain the context representation as follows:

$$\mathbf{h}_i^{con} = \text{GRU}(\mathbf{h}_{i-1}^{con}, \mathbf{h}_i^{utt}), \tag{2}$$

where $\mathbf{h}_i^{utt}$ is the last hidden vector $\mathbf{h}_{N_i,i}^{utt}$ of U_i, and $\mathbf{h}_i^{con}$ is the discourse-level representation of U_i. We write $\{\mathbf{h}_1^{con}, \ldots, \mathbf{h}_{|\mathcal{X}|}^{con}\}$ for the semantic representation of conversational context, which not only carries the semantics of various utterances but also captures the sequential relations within context.

3.1.2 Review Mechanism. A natural conversation is a coherent process, where utterances may have diverse local focuses while the global theme remains consistent. Asking a non-relevant question without contextual coherence may lead to an unnatural user experience and lead to a breakdown of the conversation. Inspired by the question content selection of QG in MRC [46, 78], the intuition behind the *Review* mechanism is to select question content from the conversational context, which can help the to-be-generated question maintain contextual coherence.

Specifically, for the words in the conversational context, after removing stop words, we view a conversational session as a document and a word as a term in that document so that we can compute the TF-IDF [65] score of each word. Then we choose at most $|\mathcal{K}|$ words with the highest TF-IDF scores as the *context keywords* for $\mathcal{X}$, denoted as $\mathcal{K}$. Based on $\mathcal{K}$, the *Review* mechanism selects several words from $\mathcal{K}$ as the question content. Figure 3 (left) illustrates the *Review* mechanism.

Given $\mathcal{K} = \{k_1, \ldots, k_{|\mathcal{K}|}\}$, where k_j ($j \in [1, |\mathcal{K}|]$) denotes a context keyword, a deep model consisting of a stack of H **multi-layer perceptrons** (**MLPs**) is designed to predict the *review scores*. The computation proceeds as follows:

$$\begin{cases} \mathbf{o}_0 = & \mathbf{h}_{|\mathcal{X}|}^{con}, \\ \mathbf{o}_1 = & \text{MLP}_1^{relu}(\mathbf{o}_0), \\ \vdots & \vdots \\ \mathbf{o}_H = & \text{MLP}_H^{relu}(\mathbf{o}_{H-1}), \end{cases} \tag{3}$$

$$\mu = \text{sigmoid}(\mathbf{o}_H), \tag{4}$$

where MLP_h^{relu} ($h \in [1, H]$) is a one-layer MLP with a ReLU as the activating function; $\mu = \{\mu_1, \ldots, \mu_{|\mathcal{K}|}\}$, where μ_j denotes the review score of k_j indicating the probability that k_j will be included in Q_c.

We sort $\{\mu_1, \ldots, \mu_{|\mathcal{K}|}\}$ by value and select the top-L context keywords with the highest review scores to be the *review topics* $\mathcal{K}_R$, which will become the first part of Q_c. As $\mathcal{K}_R$ has appeared in the given conversational context, its relevance to the conversation theme can be guaranteed, thus we expect it to control the contextual coherence of Q_c. We further input $\mathcal{K}_R$ into an embedding layer to obtain the *review vectors*, i.e., $\{\mathbf{e}_{k_1}, \ldots, \mathbf{e}_{k_L}\}$.

3.1.3 Transit Mechanism. In open-domain conversations, a question can not only be used to deepen or clarify existing topics, but also enables the introducing of new topics that have not appeared yet but that are related to conversational context. Choosing such transiting topics from an open domain is a challenging task, as there may be numerous candidates making it highly susceptible to noise, which eventually hurt the consistency of the generated question. The *Transit* mechanism leverages the conversational context to first produce coarse *transit candidates* and then

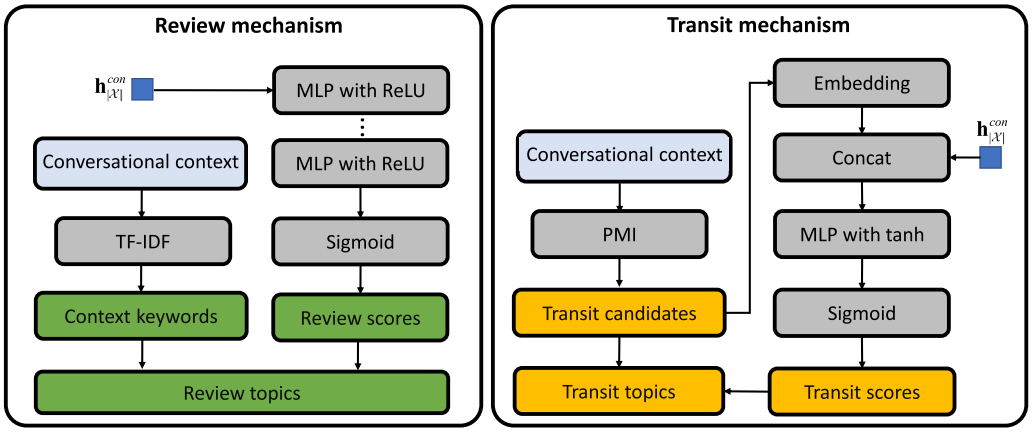

Fig. 3. Illustrations of the *Review* and *Transit* mechanisms.

filter out non-relevant ones resulting in more accurate *transit topics*. Figure 3 (right) illustrates the *Transit* mechanism.

First, we construct a PMI point-wise mutual information (PMI) matrix [12] based on the ⟨*context, question*⟩ pairs in training corpus, and use it to identify transit candidates from the entire vocabulary. Specifically, nouns, verbs, and adjectives from the conversational context $\mathcal{X}$ are referred to as *triggers*, and those in the ground-truth question Q as *targets*. The PMI score of word w_1 to word w_2 ($w_1, w_2 \in V$, where V is the pre-defined vocabulary) is calculated as

$$PMI(w_1, w_2) = \log \frac{p_{\langle trigger, target \rangle}(w_1, w_2)}{p_{trigger}(w_1) \cdot p_{target}(w_2)}, \tag{5}$$

where $p_{\langle trigger, target \rangle}(w_1, w_2)$ is the co-occurrence probability of w_1 occurring in triggers and w_2 occurring in targets, simultaneously; $p_{trigger}(w_1)$ and $p_{target}(w_2)$ denote the independent probabilities of w_1 occurring as a trigger and w_2 as a target, respectively. The PMI matrix is asymmetric. Based on the PMI matrix, given a word $w \in V$, its relevance to $\mathcal{X}$ is calculated as

$$relevance(w, \mathcal{X}) = \sum_{j=1}^{|\mathcal{K}|} PMI(k_j, w). \tag{6}$$

We select at most 50 words with the highest relevance scores as the transit candidates, denoted as $\mathcal{T}$.

Although $\mathcal{T}$ provides a limited and focused range of topics, it may still contain some non-relevant words. To increase the accuracy of the transit topics, we measure the *coherence* between each transit candidate and the conversational context. Here, coherence is a discourse-level feature concerned with the logical and semantical organization of a text sequence. It has been widely used in discourse analysis [36] to determine whether two sentences are semantically coherent. A transit candidate will be assigned a higher probability occurring in the question content, if it obtains a higher coherence to the given conversational context. Inspired by the Neural Coherence Model proposed by Xu et al. [69], we employ a neural framework to compute the coherence.

For each transit candidate $t_m \in \mathcal{T}$ ($m \in [1, |\mathcal{T}|]$), we first concatenate its word embedding $\mathbf{e}_{t_m}$ with the last state of the context representation, i.e., $\mathbf{h}^{con}_{|\mathcal{X}|}$ as Concat$[\mathbf{e}_{t_m}; \mathbf{h}^{con}_{|\mathcal{X}|}]$. Then, we input the concatenation into a one-layer MLP with tanh as activating function and use a linear projection matrix $\mathbf{W}$ followed by a sigmoid layer. Finally, t_m will receive a predicted score δ_m, i.e., its *transit*

score, which indicates its coherence to $\mathcal{X}$. Formally,

$$\delta_m = \text{sigmoid}\left(\mathbf{W} * \text{MLP}^{tanh}\left(\text{Concat}\left[\mathbf{e}_{t_m}; \mathbf{h}^{con}_{|\mathcal{X}|}\right]\right)\right). \tag{7}$$

We sort the transit candidates in $\mathcal{T}$ by their transit scores and select the top-L highest scoring candidates as the *transit topics* $\mathcal{T}_T$; these topics serve as the second part of Q_c. Compared to $\mathcal{T}$, $\mathcal{T}_T$ not only further narrows down the scope of the topics, but also filters out words that are less relevant to $\mathcal{X}$. In $\mathcal{T}_T$, some words may have already appeared in $\mathcal{X}$ while some are new. Finally, we embed $\mathcal{T}_T$ into a vector space and obtain the *transit vectors*, i.e., $\{\mathbf{e}_{t_1}, \ldots, \mathbf{e}_{t_L}\}$.

3.2 Question Pattern Prediction

The question pattern denotes the question type, which plays an important role in guiding the question generation process. An accurate question pattern will help to generate a relevant and informative question. Following [19, 80], most question patterns are divided into eight types: *yes/no*, *what*, *who*, *how*, *where*, *when*, *why*, and *which*. Moreover, each pattern can be identified by one or several representative question words or phrases, e.g., the pattern *when* corresponds to "when," "what time." We view question pattern prediction as a classification task and assume that the question pattern Q_p is jointly determined by the conversational context $\mathcal{X}$ and the predicted question content Q_c.

Formally, we first concatenate the context vectors, the review vectors, and the transit vectors into a sequence, and then input this $(|\mathcal{X}|+|\mathcal{K}_R|+|\mathcal{T}_T|)$-length sequence of vectors into a GRU-based recurrent network as follows:

$$\mathbf{h}^{pat}_l = \text{GRU}\left(\mathbf{h}^{pat}_{l-1}, \mathbf{s}_l\right), \tag{8}$$

$$\mathbf{s}_0 = \mathbf{h}^{con}_{|\mathcal{X}|},\ \mathbf{s}_l \in \left\{\mathbf{h}^{con}_1, \ldots, \mathbf{h}^{con}_{|\mathcal{X}|}, \mathbf{e}_{k_1}, \ldots, \mathbf{e}_{k_{|\mathcal{K}_R|}}, \mathbf{e}_{t_1}, \ldots, \mathbf{e}_{t_{|\mathcal{T}_T|}}\right\}, \tag{9}$$

where $\mathbf{e}_k$ and $\mathbf{e}_t$ are the word embeddings of a review topic and a transit topic, respectively; $\mathbf{h}^{pat}_l$ is the l-th hidden state, and $l \in [1, |\mathcal{X}| + |\mathcal{K}_R| + |\mathcal{T}_T|]$. Such GRU-based encoding [19, 80] can put the representation of topic words in the conversational context, which further makes them adapt to specific linguistic environment, not just limit to word-level meanings.

After that, we feed the last state $\mathbf{h}^{pat}_{|\mathcal{X}|+|\mathcal{K}_R|+|\mathcal{T}_T|}$ into a linear projection layer followed by a softmax operation as

$$P(Q_p) = \text{softmax}\left(\mathbf{W}_Q * \mathbf{h}^{pat}_{|\mathcal{X}|+|\mathcal{K}_R|+|\mathcal{T}_T|}\right), \tag{10}$$

where $\mathbf{W}_Q$ is the linear projection matrix and $P(Q_p)$ is the probability distribution over the predefined eight question types.

3.3 Question Decoder

The question decoder generates a question Q based on the conversational context $\mathcal{X}$, the predicted question content Q_c, and the classified question pattern Q_p. We employ a GRU-based recurrent decoder jointly augmented by three types of attention to implement the generation process.

Specifically, the n-th hidden state of question decoder, i.e., $\mathbf{h}^{dec}_n$, is obtained as

$$\mathbf{h}^{dec}_n = \text{GRU}\left(\mathbf{h}^{dec}_{n-1}, \mathbf{e}_{\hat{w}_{n-1}}, \mathbf{c}^{\mathcal{X}}_n, \mathbf{c}^{\mathcal{K}_R}_n, \mathbf{c}^{\mathcal{T}_T}_n\right), \tag{11}$$

$$\mathbf{h}^{dec}_0 = \mathbf{h}^{con}_{|\mathcal{X}|}, \tag{12}$$

where $\mathbf{e}_{\hat{w}_{n-1}}$ is the embedding of the generated word $\hat{w}_{n-1}$ at the $(n-1)$-th step. $\mathbf{c}_n^{\mathcal{X}}$, $\mathbf{c}_n^{\mathcal{K}_R}$, and $\mathbf{c}_n^{\mathcal{T}_T}$ are obtained as follows:

$$\begin{cases} \mathbf{c}_n^{\mathcal{X}} = \sum_{i=1}^{|\mathcal{X}|} \alpha_{i,n} \mathbf{h}_i^{con}, \text{ where } \alpha_{i,n} = \text{softmax}(\text{MLP}^{tanh}(\mathbf{h}_{n-1}^{dec}, \mathbf{h}_i^{con})), \\ \mathbf{c}_n^{\mathcal{K}_R} = \sum_{j=1}^{|\mathcal{K}_R|} \rho_{j,n} \mathbf{e}_{k_j}, \text{ where } \rho_{j,n} = \text{softmax}(\text{MLP}^{tanh}(\mathbf{h}_{n-1}^{dec}, \mathbf{e}_{k_j})), \text{ and} \\ \mathbf{c}_n^{\mathcal{T}_T} = \sum_{m=1}^{|\mathcal{T}_T|} \beta_{m,n} \mathbf{e}_{t_m}, \text{ where } \beta_{m,n} = \text{softmax}(\text{MLP}^{tanh}(\mathbf{h}_{n-1}^{dec}, \mathbf{e}_{t_m})). \end{cases} \tag{13}$$

Here, $\alpha_{i,n}$, $\rho_{j,n}$, and $\beta_{m,n}$ are weights produced by *context attention*, *review attention*, and *transit attention*, respectively; $\mathbf{h}_i^{con}$ is the i-th context vector; $\mathbf{e}_{k_j}$ and $\mathbf{e}_{t_m}$ are the word embeddings of review topic k_j and transit topic t_m, respectively.

Based on the hidden state $\mathbf{h}_n^{dec}$, the word probability distribution at the n-th step is obtained by

$$P(\hat{w}_n) = \text{softmax}\left(\mathbf{W}_{\mathbf{dec}} * \mathbf{h}_n^{dec}\right), \tag{14}$$

where $\mathbf{W}_{dec}$ is a matrix to project the dimension of $\mathbf{h}_n^{dec}$ to the vocabulary size $|V|$. Through the above process, we can generate the target question Q by an auto-regressive manner, i.e., $Q = \{\hat{w}_1, \ldots, \hat{w}_{|Q|}\}$.

Different from previous generation-based work [15, 48, 53, 63, 67] that inputs a special ⟨GO⟩ token into the first step of the recurrent decoder, we follow Zhou et al. [80] and use the question word w_{Q_p} corresponding to the classified pattern Q_p as the first input token of the question decoder. Specifically, we directly use words "what, who, how, where, when, why, which" for their corresponding question patterns, i.e., *what, who, how, where, when, why, which*. As the *yes/no* question pattern often features diverse interrogatives, like "do, is, may, can,...", we choose the representative "do" as its universal question word. Our purpose is to enhance the contextual consistence of question through making the decoding process guided by a specific pattern, rather than to achieve optimal pattern prediction accuracy. During training, we employ a "teach-forcing" mode to input the ground-truth question word to the first step of decoder, which can prompt model convergence.

3.4 Multi-Task Learning on Self-Supervised Annotations

To fully utilize the limited QG-specific training data to train our question generator, we propose to first obtain additional labeled data through a self-supervised annotation process, and then perform multi-task learning on the parallel annotations so as to enhance the question generation process. In the remainder of this section, we will first introduce the self-supervised annotations and the multi-task learning process, and then present a loss weight decaying strategy that is designed to balance the influence of various auxiliary tasks.

3.4.1 Self-Supervised Annotations. Despite the fact that CNQG is an end-to-end framework, the predicted question pattern, the selected review topic, and the predicted transit topics are intermediate outputs of the question generation. We generate training material for the question pattern prediction module, the *Review*, and the *Transit* mechanisms.

Specifically, based on the original paired $\langle context, question \rangle$ data, we first employ rules from [19, 80] to identify the question pattern of each question in the training data, which is then used as a ground-truth label for the question pattern prediction task. As for the *Review* mechanism, we label each context keyword in $\mathcal{K}$ that simultaneously appears in the corresponding question as 1, otherwise 0. The labeled data is regarded as the ground truth for the review scores $\{\mu_1, \ldots, \mu_{|\mathcal{K}|}\}$. Likewise, for the *Transit* mechanism, transit candidates in $\mathcal{T}$ that simultaneously occur in the corresponding question will get the label 1, which is then used as a supervised signal for the transit scores $\{\delta_1, \ldots, \delta_{|\mathcal{T}|}\}$. This annotation process is fully automatic and follows a self-supervising paradigm that needs no additional labeled data except the raw training data.

3.4.2 Multi-Task Learning. Based on the self-supervised annotations, we introduce auxiliary learning tasks to enhance the training of question generation. Specifically, the loss function $\mathcal{L}_\Theta$ of our model is as follows:

$$\mathcal{L}_\Theta = \mathcal{L}_{dec} + \lambda_{Q_p}\mathcal{L}_{Q_p} + \lambda_R\mathcal{L}_R + \lambda_T\mathcal{L}_T, \tag{15}$$

where Θ denotes the trainable parameter set of our model. $\mathcal{L}_{dec}$ is the question decoding loss that plays a predominant role in the training process. $\mathcal{L}_{Q_p}$, $\mathcal{L}_R$, and $\mathcal{L}_T$ are auxiliary losses from the question pattern prediction, *Review* mechanism, and *Transit* mechanism, respectively. The weights λ_{Q_p}, λ_R, and λ_T are weights to balance the influence of various auxiliary losses, whose values range from 0 to 1.

$\mathcal{L}_{dec}$ is computed as the cross-entropy function based on the negative log-likelihood of $P(Q)$, and Q denotes the generated question $\{\hat{w}_1, \ldots, \hat{w}_{|Q|}\}$,

$$\mathcal{L}_{dec} = -\frac{1}{|Q|}\sum_{n=1}^{|Q|}\log P(\hat{w}_n = w_n), \tag{16}$$

where $\hat{w}_n$ and w_n are the generated word and the ground-truth word at the n-th step, respectively.

Question pattern prediction, the *Review* mechanism, and the *Transit* are essentially classification tasks. Hence, a natural way to compute $\mathcal{L}_{Q_p}$, $\mathcal{L}_R$, and $\mathcal{L}_T$ is to use cross-entropy. However, there are *class imbalance* and *hard sample mining* issues with these learning tasks. First, the distribution of different question patterns is quite imbalanced. For example, in the DailyDialog dataset, more than 50% of the training samples feature the *yes/no* pattern, while the remaining mass is distributed across the other seven patterns. Such extreme class imbalance makes it difficult to produce accurate predictions on question pattern classification, especially for relatively infrequent patterns. For the *Review* mechanism, the positive samples that have been assigned the label 1, i.e., context keywords that also occur in the corresponding question, are more meaningful than negative ones with a label of 0, while they only occupy a relatively small fraction in the set $\mathcal{K}$. A similar comment can be made about the *Transit* mechanism. During the learning process, the easy negative samples may overwhelm training and lead to poor performance on question content prediction.

To deal with the above issues, we borrow the *Focal Loss* [30] idea from the field of computer vision. In the focal loss function, loss is computed as follows:

$$\mathrm{FL}(\hat{y}) = \begin{cases} -\psi(1-\hat{y})^\gamma \log \hat{y}, & y = 1; \\ -(1-\psi)\hat{y}^\gamma \log(1-\hat{y}), & y = 0, \end{cases} \tag{17}$$

where $\hat{y}$ is the predicted probability, y is the ground-truth label, ψ is used to address the class imbalance issue, and γ can adjust the weight of hard samples. Thus, in our model, $\mathcal{L}_{Q_p}$, $\mathcal{L}_R$, and $\mathcal{L}_T$ are computed as

$$\mathcal{L}_{Q_p} = \frac{1}{8}\sum_{q=1}^{8}\mathrm{FL}\left(P\left(\hat{Q}_p = Q_p^q\right)\right), \tag{18}$$

$$\mathcal{L}_R = \frac{1}{|\mathcal{K}|}\sum_{j=1}^{|\mathcal{K}|}\mathrm{FL}(\mu_j), \tag{19}$$

$$\mathcal{L}_T = \frac{1}{|\mathcal{T}|}\sum_{m=1}^{|\mathcal{T}|}\mathrm{FL}(\delta_m), \tag{20}$$

where $P(\hat{Q}_p = Q_p^q)$ denotes the probability that the predicted question pattern $\hat{Q}_p$ is the q-th pre-defined pattern Q_p^q. Here, $q = 1, \ldots, 8$ denotes the pre-defined question patterns, i.e., *yes/no*, *what*, *who*, *how*, *where*, *when*, *why*, *which*.

3.4.3 Loss Weight Decaying. In Equation (15), λ_{Q_p}, λ_R, and λ_T determine the respective importance of the auxiliary learning tasks in the joint training process, which has been proven to have a positive impact on the model performance [21]. Previous work [62, 63, 80] usually uses a naive sum or manually tuned parameters to set these loss weights. Unfortunately, it is expensive to search for an optimal choice. In our work, we relate loss weight to the convergence extent of the corresponding learning task, and propose a loss weight decaying strategy.

Generally in a multi-task learning process, different tasks may have different convergence speeds due to the training data and the feature to be learned itself. In our model, the aim of multi-task learning is to train the question generator in an optimal way, other than to keep every auxiliary loss to a minimum. Thus, we assume that when an auxiliary loss displays a rising that means the corresponding learning task is close to convergence, its contribution to the dominant question generation task tends to decline, thus its weight in the total loss $\mathcal{L}_\Theta$ should be reduced. In practice, we are inspired by the widely used exponential learning rate decaying to implement the loss weight decay strategy, where λ (a universal notation for λ_{Q_p}, λ_R, and λ_T) is decayed as

$$\lambda^* = \lambda \epsilon^\sigma, \tag{21}$$

where ϵ is the decay rate and σ denotes how often λ has decayed in training. In our experiments, the loss weight will decay once its corresponding loss of the current epoch is higher than that of the last epoch.

4 EXPERIMENTAL SETUP

The following research questions guide our experiments:

- (RQ1) How does CNQG perform in terms of question relevance and informativeness? Does it outperform the state-of-the-art question generation models?
- (RQ2) Can CNQG predict accurate question patterns in a question?
- (RQ3) How does CNQG perform on question content prediction? Can it generate accurate topic words in the question?
- (RQ4) What is CNQG's performance in terms of human evaluation?
- (RQ5) Does multi-task learning on self-supervised annotations help to improve the performance of CNQG? How does each auxiliary learning task affect the model performance? Can the proposed loss weight decaying strategy boost the joint training process?
- (RQ6) How does CNQG perform under different context lengths?

4.1 Datasets and Pre-Processing

We conduct experiments on two benchmark datasets, i.e., DailyDialog [28] and PersonaChat [73]. We choose these two datasets because they are both chat corpora capturing real-time interactions between users; they share much similarity with spoken dialogues between humans [57]. They contain multi-turn interactions about various open-domain topics. Utterances contained in the two datasets are natural and colloquial, allowing us to provide a realistic experimental environment for our experiments.

Table 1. Major Statistics of the Datasets Used for Evaluation

Variable	DailyDialog		PersonaChat	
	Training	Testing	Training	Testing
Number of samples	25,939	2,883	39,195	4,356
Avg. context length	4.60	4.53	6.99	7.07
Avg. utterance length	13.23	13.12	11.80	11.79

"Context length" denotes the number of utterances contained in a context.
"Utterance length" denotes the number of words contained in an utterance.

Table 2. Fractions of Question Patterns in the Used Datasets

	who	which	when	where	why	how	what	yes/no	others
DailyDialog dataset									
Training Set	1.22%	1.38%	3.25%	2.83%	4.74%	12.83%	21.26%	48.39%	4.10%
Testing Set	1.28%	1.27%	3.16%	2.57%	4.86%	13.49%	21.89%	47.73%	3.75%
PersonaChat dataset									
Training Set	0.96%	0.38%	0.60%	3.97%	1.78%	11.76%	27.18%	48.14%	5.23%
Testing Set	0.83%	0.53%	0.37%	4.02%	2.09%	12.08%	27.73%	47.02%	5.33%

"Others" denotes questions that cannot be classified by rules [19, 80].

DailyDialog is collected from human-to-human talks in daily life. It contains 11,318 human-written dialogue sessions and covers diverse topics such as culture, education, tourism, health, and so on.[1]

PersonaChat contains 12,949 dialogue sessions, where two interlocutors are assigned with provided personas and chat naturally to get to know each other.[2]

Compared to PersonaChat, topics and language expressions of DailyDialog are more diverse and the interactions it contains are closer to real life.

To train the models that we consider, we perform several pre-processing steps on the raw texts. First, for both datasets, we employ the official version of training/test splits. Secondly, for the training and test sets, given a conversation session $\{U_1, \ldots, U_M\}$ $(M \geq 2)$, we construct $\langle context, response \rangle$ pairs, where *context* is $\{U_1, \ldots, U_{m-1}\}$ and *response* is U_m $(m \in (1, M])$. Then we select samples whose *response* contains the question mark "?" and obtain $\langle context, question \rangle$ paired data. Contexts longer than 15 turns and the utterances longer than 50 words are truncated. Each *question* is assigned to a certain question type by rules designed in [19, 80]. Table 1 presents the major statistics of the pre-processed datasets. Table 2 shows the fractions of the different question patterns in the used datasets. We also provide conversation examples and their corresponding question patterns in Table 3.

4.2 Baselines

To conduct performance comparisons, we employ three kinds of baselines. Table 4 presents detailed descriptions of the key features of the baselines and the CNQG model.

— *Traditional QG.* Methods that are designed to produce training data for machine comprehension, including

[1]The dataset can be downloaded at http://yanran.li/dailydialog.html.
[2]The dataset can be downloaded at https://github.com/facebookresearch/ParlAI.

Table 3. Conversation Examples from the used Datasets

Context: Good afternoon. Can I help you?→I need some remedies for an upset stomach. **Question:** Are you also suffering from pain and fever? (**Pattern:** yes/no)
Context: Good evening! How are you?→I'm great, just had transitional surgery.→Nice! How are you recovering?→So far so good. What are you up to?→I was just playing Nintendo. And my dog is here with me. **Question:** Cool. What kind of do you have? (**Pattern:** what)

"→" separates utterances in different turns in the context. The first example comes from DailyDialog dataset and the second from the PersonaChat dataset.

NQG a Seq2Seq-based question generation model augmented with attention mechanism and feature-enriched encoder [15];

QType a model that fuses a question pattern prediction module and a feature-enriched encoder into the Seq2Seq framework to guide the question generation [80];

T5-QG a model that uses transformer-based pre-trained model, i.e., T5 [42], to fine-tune on the question generation task [35].

— *Response Generation.* Methods that generate context-aware responses for open-domain conversations, including

HRED a multi-turn response generation model using a hierarchical context encoder [48];

HRAN a hierarchical attention framework modeling the importance of conversational context at both the word and utterance level for multi-turn response generation [67];

ReCoSa a state-of-the-art model for multi-turn response generation, which leverages self-attention to detect relevant conversational context [71].

— *QG for Open-Domain Conversations.* This group includes four approaches:

STD and HTD two typed decoders proposed to generate questions only based on the last utterance for open-domain conversations, which first estimate a type distribution over word types and then use the type distribution to modulate the final word generation; STD models the word type in an implicit manner, while HTD explicitly classifies words from the vocabulary into three types, namely, interrogative, topic word, and ordinary word [63];

STD+ and HTD+ two variants of STD and HTD, respectively, which take the concatenation of the conversation history and the last utterance as input.

4.3 Evaluation Methods

4.3.1 Automatic Evaluation. We follow previous work [15, 27, 31, 63, 80] to evaluate the generated questions on two aspects: *relevance*, aimed at determing how relevant a question is to its given conversational context, and *informativeness*, aimed at assessing how informative a question is in terms of sentence semantics, which should contrast with generic and dull questions.

For *relevance*, we adopt three metrics:

BLEU is a frequently used metric in QG, which measures the N-gram overlap between the generated question and the ground-truth one. N-gram refers to the number of consecutive words, where N is usually set as 1, 2, 3, and 4. As BLEU-4 considers the influence of all N-gram, we choose it as our representative of BLEU metrics [40].

ROUGE-L is similar to BLEU. Given a generated question and the ground-truth one, it measures their longest matching sequence of words using **LCS (longest common subsequence)** [29].

BERTScore leverages the pre-trained contextual embeddings from BERT and matches words in generated and ground-truth questions by cosine similarity. Unlike the BLEU and ROUGE metrics, which focus on word overlap, the BERTScore can measure semantic relevance [74].

Table 4. Descriptions of Key Features of the Baselines and the Proposed CNQG Model

Type	Model	Feature				
		Pattern	Content	Context	Feature-enriched encoder	MTL
Traditional QG	NQG			✓	✓	
	QType	✓		✓	✓	✓
	T5-QG			✓		
Response generation	HRED			✓		
	HRAN			✓		
	ReCoSa			✓		
QG for open-domain conversations	STD	✓	✓			✓
	HTD	✓	✓			✓
	STD+	✓	✓	✓		✓
	HTD+	✓	✓	✓		✓
	CNQG	✓	✓	✓		✓

"Pattern" and "Content" denote that the question pattern and question content are pre-predicted in a model. "Context" means that the conversational context is used in a model. "Feature-enriched encoder" indicates that the word embedding in the sentence encoder of a model is enriched by lexical features and answer position. "MTL" indicates that model is augmented with multi-task learning.

For *informativeness*, we use the following metrics:

Word entropy measures how non-generic the generated question is. Here we employ the bi-gram version of averaged word entropy denoted as $H(w)$. Higher $H(w)$ values indicate that the generated question is more informative [49].

Distinct is used to evaluate the sentence diversity and a higher value denotes the generated question is more diverse. We adopt *Distinct-1* and *Distinct-2* to respectively measure the number of distinct uni-grams and bi-grams in the generated questions [26].

Moreover, to obtain further insights into the performance on the question pattern and content prediction tasks, we also evaluate all models with pattern-related and content-related metrics:

Accuracy and F1 are commonly used metrics for classification tasks. Accuracy measures the overall performance on pattern prediction, which denotes the percentage of generated questions that are featured with ground-truth patterns in test sets. We employ the F1 score to investigate the predictive accuracy of a model on each specific pattern.

Average, Greedy, and Extrema are embedding-based topic similarity metrics proposed in [33]. They can measure how semantically relevant the content of a generated question is to that of the ground truth.

4.3.2 Human Evaluation. We first randomly select 300 samples from the DailyDialog dataset and conduct predictions using various models. We choose the DailyDialog dataset for subjective evaluation, as it does not require annotators to adapt to various personas and as its conversations concern daily-life topics, which makes it easy for human annotators to understand and provide judgments. We invite three students who have passed the **CET-4** (**College English Test Band 4** in China) and are not involved in our work to be the annotators. Then, each question was evaluated by all three annotators in terms of the following four metrics.

Grammaticality. Is the generated question grammatically correct?

Relevance. Is the generated question semantically relevant to the given conversational context?

Informativeness. Is the generated question a meaningful and informative response, which is distinct from generic and dull ones like "what's the matter," "do you have any questions?"

Interactiveness. Is the generated question engaging for users? Can it trigger more interactions? Does it contribute to dialogue persistence?

All four metrics are assessed using a five-point scale (1, 2, 3, 4, 5), where higher values denote better performance. For each model, we average the ratings of three annotators as the final evaluated result. We also employ the Kappa score [64] for each model on each metric to indicate how the three annotators agree in their judgments. The Kappa score usually ranges from 0 to 1 and higher values denote better consistency among annotators. Generally, a Kappa score in $[0.0, 0.20]$ indicates slight consistency, $[0.21, 0.40]$ fair consistency, $[0.41, 0.60]$ moderate consistency, $[0.61, 0.80]$ substantial consistency, and $[0.81, 1.0]$ almost perfect consistency.

The human evaluation is conducted in a reference-free manner, which means that the annotators do not access to the ground-truth questions. On the one hand, this will drive human annotators to provide evaluations based on their own comprehensive understanding of the conversational context. On the other hand, this respects the fact that there may be no standard responses in open-domain conversations, which can provide a different view of validation to investigate model performance.

4.4 Implementation Details

In our implementation of the CNQG model, we employ `NLTK`[3] for pos-tagging and the `scikit_learn`[4] package to conduct the TF-IDF-based context keywords extraction. The PMI matrix is calculated on the training corpora. The maximum number of context keywords and transit candidates are set as 50 and 20, respectively, since too few may fail to cover words related to current conversation while too many will introduce common words affecting informativeness. The number of review topics or transit topics, i.e., L, is set to 5 according to a tuning process. All recurrent units, like GRU and BiGRU, have a one-layer structure with 512 hidden cells. The word embeddings are randomly initialized and trainable with dimension of 512. The deep model in the *Review* mechanism consists of four-layer MLP structures, whose hidden sizes are, respectively, 1,024, 512, 128, and 5. For the joint training process, the loss weights λ_{Q_p}, λ_R, and λ_T are all initialized to 1, and are decayed with 0.5 when their corresponding losses in the current epoch are higher than that in the last epoch. We keep the origin parameter settings of the focal loss [30], where ψ is 0.25 and γ is 2.

For the NQG, T5-QG, STD, and HTD baselines, we adopt implementations that have been open-sourced by the corresponding authors.[5] Implementations of HRED, HRAN, and ReCoSa are released by [24]. We re-produce QType and implement the CNQG model within the Tensorflow framework.[6] The shared parameters between baselines and CNQG are set to the same values, and the remainder is fine-tuned to ensure the best performance. All models are trained for at most 20 epochs and are optimized with the Adam Optimizer with a learning rate of 0.001. The mini-batch size is 64.

5 RESULTS AND DISCUSSION

5.1 Performance on Relevance and Informativeness

To answer (RQ1), we compare the CNQG model to the baselines listed in Section 4.2; we evaluate their performance on the DailyDialog and PersonaChat datasets introduced in Section 4.1, in terms

[3] https://www.nltk.org/.
[4] https://scikit-learn.org/stable/install.html.
[5] NQG: https://github.com/yanghoonkim/neural_question_generation. T5-QG: https://github.com/ThilinaRajapakse/simpletransformers. STD and HTD: https://github.com/victorywys/Learning2Ask_TypedDecoder.
[6] The code of our own implementations will be open-sourced at https://github.com/katherinelyx/CNQG.

Table 5. Performance Comparison on Question Relevance and Informativeness

Type	Model	BLEU-4	ROUGE-L	BERTScore	$H(w)$	Dis-1	Dis-2
DailyDialog dataset							
Traditional QG	NQG	0.1904	0.2502	0.1685	12.3102	0.0453	0.1901
	QType	0.1971	0.3499	0.2572	12.6762	0.0681	0.2889
	T5-QG	0.1269	0.1684	0.1503	12.3782	0.0568	0.2282
Response generation	HRED	0.1492	0.2365	0.1191	11.4568	0.0170	0.0571
	HRAN	0.1953	0.1926	0.0663	11.5793	0.0219	0.1084
	ReCoSa	0.1625	0.2237	0.1310	12.2164	0.0198	0.0609
QG for open-domain conversations	STD	0.1632	0.2424	0.1566	11.6802	0.0162	0.0751
	HTD	0.1317	0.2250	0.1156	12.3736	0.0369	0.2332
	STD+	0.1380	0.2394	0.1562	11.5999	0.0199	0.0903
	HTD+	0.1295	0.2340	0.1230	12.3181	0.0423	0.2284
	CNQG	**0.2258**▲	**0.3606**▲	**0.2735**▲	**12.8659**▲	**0.0793**▲	**0.3504**▲
PersonaChat dataset							
Traditional QG	NQG	0.1786	0.2681	0.1608	10.4772	0.0247	0.0807
	QType	0.1434	0.2598	0.1421	11.1053	0.0354	0.1434
	T5-QG	0.1253	0.1674	0.0906	9.9738	0.0290	0.0945
Response generation	HRED	0.1687	0.2740	0.1554	10.0925	0.0130	0.0400
	HRAN	0.1899	0.2660	0.1570	10.7962	0.0104	0.0314
	ReCoSa	0.1725	0.2669	0.1605	10.5046	0.0189	0.0585
QG for open-domain conversations	STD	0.1783	**0.3001**	0.1717	9.3490	0.0036	0.0118
	HTD	0.1865	0.2673	0.1405	10.4253	0.0181	0.1012
	STD+	0.1851	0.2924	**0.1836**	9.6749	0.0033	0.0118
	HTD+	0.1824	0.2694	0.1345	10.3991	0.0190	0.1121
	CNQG	**0.1920**	0.2780	0.1729	**11.8370**▲	**0.0411**	**0.2428**

"Dis-1" and "Dis-2" are short for Distinct-1 and Distinct-2, respectively. The results of the best performer and the best baseline are set in boldface and underlined, respectively. ▲ denotes significantly better than the best baseline in a paired t-test with $\alpha = 0.05$.

of the metrics described in Section 4.3. We also conduct significance tests (t-test with $\alpha = 0.05$). The results are presented in Table 5.

Let us first discuss the baselines. In terms of the question relevance metrics, we see that QType achieves the best performance on the DailyDialog dataset; STD and its variant STD+ perform better in terms of ROUGE-L and BERTScore, respectively, on the PersonaChat dataset. It may be attributed to the fact that these methods produce specific predictions about pattern and content to guide the question decoding process, which enhances the contextual relevance of the generated questions. Compared to the response generation methods, QG models specializing on open-domain conversations generally obtain better performance on question relevance, especially in terms of BERTScore. This may be attributed to the fact that asking a relevant question not only concerns providing a response with a special form, but also requires identifying what is worth being asked in the preceding conversation. This further indicates the need to design a special QG model for open-domain conversations.

As to the baseline performance in terms of the question informativeness metrics, i.e., $H(w)$, Distinct-1, and Distinct-2, we see that traditional QG methods generally outperform response generation methods. However, the open-domain conversational QG methods, particularly STD and

STD+, do not seem to have a stable advantage. For instance, STD obtains the lowest value on $H(w)$ and STD+ performs worst on the Distinct metrics, even augmented with conversational context. This may be attributed to the fact that these methods do not properly utilize conversational context. Context provides background information for the QG process; it can help to characterize the semantics of a conversation, leading to the generation of informative and diverse questions. Inappropriate use of context may negatively impact question quality. STD misses the information contained in the context, while in STD+ the context introduces considerable noise in the topic words without effective filtering.

Next, let us focus on the CNQG model. CNQG significantly outperforms all baselines in terms of all metrics on the DailyDialog dataset. On the PersonaChat dataset, CNQG outperforms the baselines on almost all metrics, except for ROUGE-L and BERTScore, where STD and STD+ perform best. Specifically, compared to the traditional QG method, CNQG performs consistently better on all metrics. We attribute the strong performance of CNQG to the *Review* and *Transit* mechanisms, which can not only select consistent topics from the conversational context but also introduce relevant and new topics. Furthermore, compared to STD, HTD, and their variants, CNQG wins on almost all metrics except for ROUGE-L and BERTScore. CNQG achieves particularly large improvements in terms of the Distinct metrics. This confirms the importance of properly utilizing conversational context in QG for open-domain conversations. CNQG's multi-task learning based on self-supervised annotations essentially mines the conversational context, which also contributes to its solid performance.

5.2 Question Pattern Prediction

To answer (RQ2), given a question generated by one of the models, we first identify its corresponding question pattern by the same rules that are employed in [19, 80], and then calculate the Accuracy to evaluate how well the pattern matches the ground-truth pattern. We also use the F1 score to assess the model performance on each pattern prediction. We only present the results on the DailyDialog dataset; the results on the PersonaChat dataset are qualitatively similar. Table 2 presents the relative fractions of the question patterns in the DailyDialog dataset. The pattern distribution is imbalanced, i.e., the majority of questions feature *what* or *yes/no* patterns. Many questions in natural conversations may not have typical or formal question words, as people tend to use colloquial expressions in daily communication. As a consequence, *what* and *yes/no* become the generic question patterns in open-domain conversations. It also suggests that it is a difficult task to accurately predict a question pattern especially for infrequent patterns, such as *who*, *which*, *when*. The performance on question pattern prediction is presented in Table 6.

Let us focus on the baselines first. As shown in Table 6, T5-QG and QType achieve higher values in terms of Accuracy compared to other baselines, indicating they can well approximate to the ground-truth pattern distribution on the whole. The response generation methods (HRED, HRAN, ReCoSa) fail to generate some relatively infrequent patterns such as *who* and *which*. NQG, HTD, and HTD+ are able to predict most patterns except *which*, *where*, and *who*. Surprisingly, STD achieves poor performance on pattern diversity, only providing generic patterns in the generated questions. The implicit word type prediction of STD cannot effectively learn features from minority question patterns. QType presents a diverse coverage of question patterns, demonstrating the effectiveness of additional question pattern prediction. T5-QG also covers all types of question patterns.

By zooming in on the F1 score of each pattern prediction, we observe that T5-QG obtains a worse performance than QType on most patterns except *yes/no*. As the *yes/no* pattern covers almost half of the samples in the DailyDialog dataset, it is clear why T5-QG achieves the highest Accuracy score amongst the baselines. The prior knowledge learned from pre-training helps T5-QG obtain

Table 6. Performance Comparison on Question Pattern Prediction in the DailyDialog Dataset

Model	Accuracy	F1							
		who	which	when	where	why	how	what	yes/no
NQG	0.4083	0.1143	–	0.1165	–	0.1926	0.1751	0.3464	0.5465
QType	0.5104	0.3461	**0.1905**	0.1987	0.3066	0.3282	0.3094	0.3757	0.6557
T5-QG	0.5231	0.1212	0.0606	0.0769	0.0964	0.1789	0.1636	0.1998	0.6906
HRED	0.4162	–	–	0.0301	–	–	0.0452	0.2782	0.5811
HRAN	0.4259	–	–	–	–	0.0199	0.1283	0.2119	0.6105
ReCoSa	0.3909	–	–	–	0.0294	0.1959	0.1601	0.2957	0.5379
STD	0.3802	–	–	–	–	–	0.0442	0.3152	0.5200
HTD	0.2688	–	–	0.0343	0.0421	0.1067	0.1811	0.2598	0.3736
STD+	0.2404	–	–	–	–	0.0744	0.2228	0.2553	0.2922
HTD+	0.2196	–	–	0.0427	–	0.0897	0.1785	0.3101	0.2023
CNQG	**0.5646**	**0.3902**	0.1569	**0.2533**	**0.3200**	**0.3759**	**0.3471**	**0.4452**	**0.6954**

The results of the best performer and the best baseline are set in boldface and underlined, respectively. "–" denotes that none of the generated questions feature the corresponding pattern.

diverse patterns in questions it generates, but it fails to predict accurate question patterns for certain conversations due to the lack of context understanding and specialized pattern prediction.

Compared to the baselines, the CNQG model achieves the highest Accuracy scores on the question pattern prediction task. CNQG not only covers all types of ground-truth patterns, but also obtains the highest F1 scores on almost all patterns except *which*. The question pattern prediction module and the multi-task learning based on self-supervised annotations help CNQG learn how to ask relevant questions from a limited amount of data.

5.3 Question Content Prediction

To answer (RQ3), given the ground-truth and the generated questions, we use NLTK to identify the nouns, verbs, and adjectives from each question and filter out stop words. The remaining words are regarded as question content. Then, we calculate the embedding-based topic similarity metrics, i.e., Average, Greedy, and Extrema, to measure how semantically relevant the content of the generated question is to that of the ground truth. We only present the results on the DailyDialog dataset, since qualitatively similar findings are obtained for the PersonaChat dataset. The results are shown in Table 7.

T5-QG is the best performing baseline in terms of Average and QType is best performing in terms of Extrema and Greedy. QType employs a feature-enriched encoder to represent the conversational context, highlighting the semantically relevant words according to lexical feature and answer position. This helps QType to select accurate question content. T5-QG contains prior knowledge learned from pre-training corpora, which makes it generate more informative words in questions and further achieve good performance on question content prediction. Compared to STD and HTD, their variants STD+ and HTD+ perform better on most metrics, indicating the importance of conversational context to question content prediction.

CNQG significantly outperforms all baselines on the question content prediction task. The strong performance of CNQG can be attributed to (1) the *Review* and *Transit* mechanisms in CNQG, which provide two sources for question content—one for emphasizing topics in the conversational context, the other to help transit the conversational focus to new but relevant topics; this combination allows CNQG to adapt to open-domain conversations even if they have different conversational aims; and (2) multi-task learning on self-supervised annotations fully leverages the topic

Table 7. Performance Comparison on Question Content Prediction in the DailyDialog Dataset

Model	Average	Extrema	Greedy
NQG	0.5531	0.3505	0.4491
QType	0.5913	0.4577	0.5329
T5-QG	0.6122	0.4219	0.5081
HRED	0.4829	0.3147	0.3957
HRAN	0.5729	0.3310	0.4399
ReCoSa	0.5282	0.3242	0.4212
STD	0.5135	0.3351	0.4242
HTD	0.4943	0.3182	0.4073
STD+	0.4921	0.3477	0.4270
HTD+	0.5007	0.3331	0.4200
CNQG	**0.6240**▲	**0.4963**▲	**0.5654**▲

The results of the best performer and the best baseline are set in boldface and underlined, respectively.
▲ denotes significantly better than the best baseline in a paired t-test with $\alpha = 0.05$.

Table 8. Performance Comparison on Human Evaluation on the DailyDialog Dataset

Model	Grammaticality	Relevance	Informativeness	Interactiveness
NQG	3.8122 (0.2465)	2.3322 (0.2543)	3.0656 (0.3119)	3.6433 (0.4835)
QType	3.9000 (0.1720)	2.4622 (0.2063)	2.2533 (0.4324)	3.1722 (0.5878)
T5-QG	3.9500 (0.1893)	2.2944 (0.2621)	2.8200 (0.4709)	3.7302 (0.8180)
HRED	3.9767 (0.1535)	2.7944 (0.3364)	2.1911 (0.4204)	2.8667 (0.6486)
HRAN	3.9244 (0.2637)	2.2822 (0.2852)	2.1578 (0.3675)	2.4400 (0.7919)
ReCoSa	**4.1056** (0.1172)	2.4211 (0.2393)	2.4011 (0.3213)	2.8478 (0.2567)
STD	3.9456 (0.1158)	2.9389 (0.3612)	2.5022 (0.2782)	2.9033 (0.1617)
HTD	3.8667 (0.0723)	2.8878 (0.3331)	2.3578 (0.4827)	2.9100 (0.2200)
STD+	3.9256 (0.1608)	2.9622 (0.3363)	2.0778 (0.3455)	2.2911 (0.1114)
HTD+	3.9244 (0.1410)	2.9422 (0.3551)	2.1467 (0.3745)	2.6533 (0.2012)
CNQG	4.0256 (0.2635)	**3.1084**▲ (0.3663)	**3.2744**▲ (0.5121)	**3.7600** (0.1444)

The results of the best performer and the best baseline are set in boldface and underlined, respectively.
▲ denotes significantly better than the best baseline in a paired t-test with $\alpha = 0.05$. The values in brackets denote the Kappa scores.

relevance and the topic transitions reflected in the training data, which further boosts the effect of the *Review* and *Transit* mechanisms.

5.4 Human Evaluation

To answer (RQ4), we conduct human evaluation based on randomly selected samples from the DailyDialog dataset in terms of grammaticality, relevance, informativeness, and interactiveness. We also present the Kappa score for each model on each metric to show how the three annotators agree in their judgments. The results are shown in Table 8.

All models perform well on grammaticality, indicating they all can provide grammatically correct questions. Surprisingly, the Kappa scores on grammaticality are relatively low. It may be attributed to the fact that the annotators have different tolerance to colloquial expressions. In terms of

Table 9. Performance Comparison with Various Settings for λ_R, λ_T, and λ_{Q_p}, whose Values are Set as 0 when their Corresponding Loss Functions are Removed from Training

$(\lambda_R, \lambda_T, \lambda_{Q_p})$	**BLEU-4**	**ROUGE-L**	**BERTScore**	$H(w)$	**Dis-1**	**Dis-2**
(0, 0, 0)	0.2369	0.3799	0.2865	12.7351	0.0716	0.3325
(1, 0, 0)	**0.2568**	**0.3824**	**0.2966**	12.7987	0.0761	0.3468
(0, 1, 0)	0.1383	0.2149	0.1195	12.7574	0.0712	0.3320
(0, 0, 1)	0.1487	0.2642	0.1674	12.8448	0.0754	0.3449
(1, 0, 1)	0.2471	0.3801	0.2908	12.5811	0.0697	0.3224
(1, 1, 1)	0.1562	0.2691	0.1704	12.7121	0.0747	0.3321
LWD	0.2258	0.3606	0.2735	**12.8659**	**0.0793**	**0.3504**

LWD denotes the Loss Weight Decaying strategy. The result of the best performer is set in boldface.

relevance and informativeness, our CNQG model achieves the best performance with the highest Kappa scores. These results are consistent with the results of automatic evaluation (Table 5), validating the effectiveness of CNQG on QG for open-domain conversations from the perspective of real human-machine interaction. In addition, we find that QG models specializing on open-domain conversations, i.e., STD, HTD, STD+, HTD+, and our CNQG, significantly outperform other models in terms of relevance. This further validates the uniqueness of this domain. We cannot directly apply current response generation models or traditional QG models to generating relevant questions for open-domain conversations and achieve state-of-the-art performance.

As to interactiveness, although our CNQG model performs better than the best baseline T5-QG, its corresponding Kappa score is much lower than T5-QG. On the one hand, annotators may have diverse understandings about what "interactiveness" is due to different background knowledge; on the other hand, T5-QG contains prior knowledge refined from pre-training corpora, which makes it powerful on text generation. This suggests a promising way to further enhance the CNQG model: first pre-train CNQG on large-scale conversational datasets where questions naturally occur, and then fine-tune it on target chat corpora. It resembles the process of human learning; first obtaining the ability of asking and then learning to ask questions for open-domain conversations.

As an aside, disagreement between automatic and human evaluation reflects a typical feature of open-domain conversations. It is *one-to-many*, i.e., the same context may have diverse responses displaying different information, emotions, or attitudes; and the same information may be expressed using different realizations. The creativity of open-domain conversations makes it hard to have a standard response, as a result of which relative performance may differ between automatic and human evaluations. Despite this observation, CNQG outperforms all baselines on both automatic and human evaluations, underlining its advantage for generating questions for open-domain conversations.

5.5 Analysis of Multi-Task Learning Based on Self-supervised Annotations

To answer (RQ5), we conduct a detailed empirical test of the multi-task learning based on self-supervised annotations of CNQG. We train CNQG with various settings of the λ_R, λ_T, and λ_{Q_p} parameters, and then compare the resulting performance in terms of automatic metrics. Specifically, when $\lambda_R = 0$, the loss function of the *Review* mechanism, i.e., $\mathcal{L}_R$, does not take part in the multi-task learning (see Equation (15)), while the *Review* mechanism is still kept in CNQG, and similarly for λ_T and λ_{Q_p}. The results are presented in Table 9.

First, we discuss the impact of each auxiliary learning task. The setting $(0, 0, 0)$ denotes that no auxiliary learning is employed. Comparing the performance of $(1, 0, 0)$ to that of $(0, 0, 0)$, we can see improvements in terms of most metrics, which validates that introducing the loss $\mathcal{L}_R$ in the

Review mechanism enhances both the relevance and the informativeness of generated questions. In contrast, the setting where $(\lambda_R, \lambda_T, \lambda_{Q_p})$ is set to $(0, 1, 0)$ suffers from performance drops on most metrics. The self-supervised annotations on the *Transit* mechanism are affected by the class imbalance (i.e., most transit candidates get label 0), as the PMI-based transit candidates may contain considerable noise. When jointly trained with only auxiliary learning on the *Transit* mechanism, question generation is misled by non-relevant transit topics and eventually gets a poor performance on question relevance and informativeness. As to $(0, 0, 1)$, we see that model performance improves in terms of informativeness metrics, i.e., $H(w)$ and Distinct, while it drops in terms of relevance metrics, i.e., BLEU, D^u_{kl}, and D^b_{kl}.

Based on these observations, we conclude the following: (1) $\mathcal{L}_R$, $\mathcal{L}_T$, and $\mathcal{L}_{Q_p}$ have a different impact on question generation; and (2) $\mathcal{L}_R$ can boost the quality of the generated questions, $\mathcal{L}_{Q_p}$ works on question informativeness, while $\mathcal{L}_T$ has a negative impact on QG. However, does $\mathcal{L}_T$ really contribute nothing to the performance of CNQG? To answer this question, we pay particular attention to the performance of $(1, 0, 1)$ and see that the informativeness metrics all decline compared to $(0, 0, 0)$. This shows that $\mathcal{L}_T$ can actually benefit the QG if we apply it properly.

A natural way for comprehensive performance improvement is to apply a combination of the three loss functions. Following [80], we have assigned λ_R, λ_T, and λ_{Q_p} with equal weights, i.e., $(1, 1, 1)$, but we have found that it performs worse than $(0, 0, 0)$. Another widely used way is to search for a relatively optimal choice through training with different hand-crafted settings [62, 63]. However, this is prohibitively time-consuming due to the size of the search space. The proposed **loss weight decaying** (**LWD**) strategy provides a reasonable solution. As shown in Table 9, the LWD strategy achieves the best performance in terms of question informativeness. In particular, compared to the $(1, 1, 1)$ setting, LWD presents obvious performance gains. The key point is that LWD is fully automatic and enables us to achieve good performance without manual work involved in setting loss weights.

5.6 Impact of Context Length

To answer (RQ6), we examine the performance of various models under different context lengths. Figure 4 shows the distribution of the number of samples under various context lengths in the DailyDialog dataset. We can see that the context length of most conversations is less than 6: people tend to have relatively short conversations in daily life. Thus, we first divide the test samples of the DailyDialog dataset into three groups according to their context length, i.e., $[1, 5]$, $(5, 10]$, and >10. Then, for simplicity, we calculate the BLEU-4 and $H(w)$ scores of each group for baselines and the CNQG model. We choose the above two metrics as they are representatives for relevance and informativeness, respectively. The results are plotted in Figure 5.

Based on Table 5, we concluded that QType is the best baseline in terms of all metrics on the DailyDialog dataset. Nevertheless, in Figure 5 we see that the performance of QType in terms of BLEU-4 declines as the context length increases, and the drop is much more obvious when the context length is larger than 10. In terms of $H(w)$, we can also see fluctuating performance of QType with variations of the context length. A similar phenomenon can be observed for other baselines. Based on these observations, we conclude that it is non-trivial for a model to maintain stable performance under different context lengths, since long and short contexts offer different challenges. Long contexts usually contain more words or sentences, which bring well-known long-term dependency and memory decay issues in context modeling [8], making it increasingly difficult to capture the focus of a conversation. Short contexts may carry too little information to be able to characterize its focus.

Interestingly, despite the variation in context length, compared to the baselines, the CNQG model consistently maintains a strong performance in terms of all presented metrics under

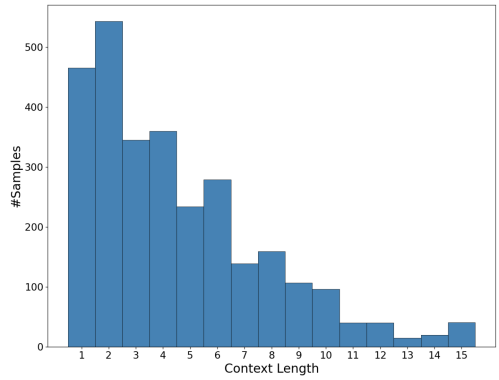

Fig. 4. Distribution of the number of samples of varying context lengths in the DailyDialog test set.

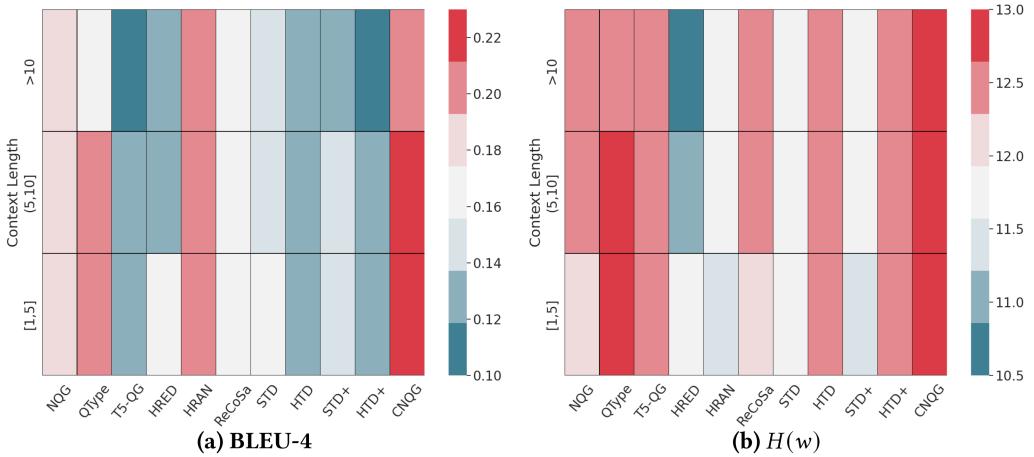

Fig. 5. Comparison of model performance under different context lengths on the DailyDialog test set.

every context length. We attribute this robustness to the question content prediction module of CNQG. For long contexts, the hierarchical encoder can produce a discourse-level context representation that will capture high-level semantics contained in the conversational context and alleviate the long-term dependency issue. Meanwhile, the *Review* and *Transit* mechanisms filter out non-essential information from the context, which helps to capture the conversational focus. As to short contexts, the *Transit* mechanism introduces relevant topic words based on PMI that will augment the limited semantic features contained in the original conversational context.

5.7 Case Studies

To gain further insights into how the proposed CNQG model works and understand its strengths and weaknesses, we present a positive and a negative example of questions generated by CNQG and several baselines in Table 10. We also provide the intermediate review topics and transit topics produced by CNQG. Furthermore, to provide an intuitive explanation of the automatic and human evaluations, we report the specific scores that each model achieves on various metrics.

In Example 1 in Table 10, the context is relatively long and the interlocutors are "discussing their food." Given the long context, the *Review* mechanism selects five topic words to filter out trivial

Table 10. Case Study

1	**Context:** Hey, George. How is your chicken? → My chicken tastes all right, but it is pretty dry. How is your fish? → My fish is pretty dry too. → It is almost as if this good has been sitting a little too long. It does not seem fresh. → Yes, it seems that way to me also. **Reference:** How are your vegetables?	**CNQG**	Are your vegetables OK?			
			Review topics: sitting, fish, long, chicken, tastes *Transit topics*: raving, vegetables, fish, prediction, colder			
			BLEU-4: 0.1257	ROUGE-L: 0.5999	BERTScore: 0.5209	$H(w)$: 15.0480
			Gra.:5	Rel.:5	Info.:4	Inte.:4
		QType	Do you really think so?			
			BLEU-4: 0.0408	ROUGE-L: 0.1818	BERTScore: 0.0929	$H(w)$: 12.4243
			Gra.:5	Rel.:4	Info.:2	Inte.:3
		ReCoSa	How about mah jong?			
			BLEU-4: 0.0639	ROUGE-L: 0.3999	BERTScore: 0.1709	$H(w)$: 10.4887
			Gra.:5	Rel.:1	Info.:3	Inte.:3
		STD	What do you mean?			
			BLEU-4: 0.0537	ROUGE-L: 0.1999	BERTScore: 0.2577	$H(w)$: 10.5747
			Gra.:5	Rel.:3	Info.:1	Inte.:2
		HTD	How much is it dating for a pushcart?			
			BLEU-4: 0.0285	ROUGE-L: 0.2857	BERTScore: 0.2293	$H(w)$: 12.0592
			Gra.:2	Rel.:1	Info.:4	Inte.:2
2	**Context:** Are you being served, madam? → No. Um, do you carry any padded coat? → Yes, what size is in your mind? → Small size. → **Reference:** Any particular brand?					
	CNQG: And what is your charge today? *Review topics*: served, mind, small, size, padded *Transit topics*: 60, charge, nice, frisk, flavors					

Gra., *Rel.*, *Info.*, and *Inte.* are short for *Grammaticality*, *Relevance*, *Informativeness*, and *Interactiveness*, respectively.

information and characterize what is worth being asked out. Despite the fact that the generated transit topics contain noise, e.g., "raving" and "prediction," they introduce a relevant and fresh topic word, i.e., "vegetables." This makes CNQG accurately capture a suitable topic transition and produce a relevant question. Compared to the CNQG model, QType and STD provide proper but meaningless questions, which can serve as responses to their corresponding context but contribute little to dialogue persistence. Questions produced by ReCoSa and HTD are totally irrelevant.

Let us now focus on the model performance in terms of various metrics. CNQG performs best on both automatic and human evaluation metrics. However, there are also conflicts among these evaluation metrics. For instance, the question generated by HTD is irrelevant from the subjective perspective and also obtains a low BLEU-4 score, but its performance in terms of BERTScore is relatively high; ReCoSa obtains a relatively high ROUGE-L score indicating good performance on question relevance, which conflicts with its performance on the relevance of human evaluation. This shows that evaluation for open-domain conversation is a non-trivial problem, due to its open-ended goal and the richness of natural language. In this article, we employ various metrics to provide a comprehensive evaluation as insightful as possible.

Example 2 in Table 10 is a negative example. Given the context, the *Review* mechanism of CNQG produces five topic words that can indicate the gist of the given conversation, i.e., "buying a padded coat." Nonetheless, the topic words predicted by the *Transit* mechanism digress from the main gist. The semantics of the final generated question is dominated by the transit topic words, which are

inappropriate. This reflects a potential disadvantage of the CNQG model. Topic words from the *Review* and *Transit* mechanisms have equal influence on the question decoding process, which may lead to non-ideal questions, especially when the *Transit* mechanism produces noisy words.

6 CONCLUSIONS

In this article, we have proposed a CNQG context-enhanced neural question generation (CNQG) model for open-domain conversations. CNQG uses *Review* and *Transit* mechanisms to identify what is worth being asked in the question; the first mechanism is used to emphasize topics in the conversational context in order to achieve coherence; the second mechanism is used to introduce new but relevant topics so as to promote multi-turn interactions. To fully utilize the limited question generation (QG) QG-specific training data available in chat corpora, CNQG performs multi-task learning on self-supervised annotations that are obtained from the question pattern prediction, *Review* mechanism, and *Transit* mechanism. A decaying strategy is proposed to automatically adjust the influence of multiple training objectives.

Extensive experimental results on two open-domain conversational datasets demonstrate the strong performance of the proposed CNQG model compared to competitive state-of-the-art baselines on generating relevant and informative questions. By performing detailed assessments of the predictive performance for the question pattern and content tasks, we find that CNQG enables us to produce accurate patterns and semantically relevant topics, which provides an explanation for its strong performance. We have also found that CNQG is robust to variations in context length, making it a suitable choice for diverse conversational scenarios.

As to broader implications of our work, the proposed CNQG model can be applied to generate relevant and informative questions for conversational search and recommendation systems [17]. It will help to seek information from users in terms of questions, products, or features; clarify users' intent, and provide a humanized interacting service.

Naturally, CNQG leaves room for improvement. In particular, when handling generic conversational contexts, the *Review* mechanism cannot capture recognizable context keywords, while the *Transit* mechanism may mostly introduce non-relevant topics under such conditions. In addition, the question decoder assigns equal influences for the *Review* and *Transit* mechanisms by default, which may make those noisy words produced by the *Transit* mechanism overwhelm relevant ones from the *Review* mechanism.

A potential direction for future work is to enhance CNQG by dynamically assigning different degrees of influence of the *Review* and *Transit* mechanisms on the question decoder. This can help the model adapt to diverse conversational scenarios, e.g., chatting around fixed topics, or dynamically transiting to different topics. Another interesting direction for future work is to study the "when to ask" problem, since identifying a proper time to increase engagingness and ask questions, is important in multi-turn conversations.

ACKNOWLEDGMENTS

We are grateful to our associate editor and reviewers for providing extensive feedback on earlier versions of this article.

All content represents the opinion of the authors, which is not necessarily shared or endorsed by their respective employers and/or sponsors.

REFERENCES

[1] Mohammad Aliannejadi, Julia Kiseleva, Aleksandr Chuklin, Jeff Dalton, and Mikhail S. Burtsev. 2020. ConvAI3: Generating clarifying questions for open-domain dialogue systems (ClariQ). *CoRR* abs/2009.11352 (2020).

[2] Mohammad Aliannejadi, Hamed Zamani, Fabio Crestani, and W. Bruce Croft. 2019. Asking clarifying questions in open-domain information-seeking conversations. In *Proceedings of the 42nd International ACM SIGIR Conference on Research and Development in Information Retrieval*. Association for Computing Machinery, 475–484.

[3] Leif Azzopardi, Mateusz Dubiel, Martin Halvey, and Jeffery Dalton. 2018. Conceptualizing agent-human interactions during the conversational search process. In *The 2nd Workshop on Conversational Approaches to Information Retrieval*.

[4] Nicholas J. Belkin. 1980. Anomalous states of knowledge as a basis for information retrieval. *Can. J. Inf. Sci.* 5, 1 (1980), 133–143.

[5] Keping Bi, Qingyao Ai, Yongfeng Zhang, and W. Bruce Croft. 2019. Conversational product search based on negative feedback. In *Proceedings of the 28th ACM International Conference on Information & Knowledge Management (CIKM'19)*. 359–368.

[6] Rich Caruana. 1997. Multitask learning. *Mach. Learn.* 28, 1 (1997), 41–75. https://doi.org/10.1023/A:1007379606734

[7] Hongshen Chen, Xiaorui Liu, Dawei Yin, and Jiliang Tang. 2017. A survey on dialogue systems: Recent advances and new frontiers. *SIGKDD Explor.* 19, 2 (2017), 25–35.

[8] Hongshen Chen, Zhaochun Ren, Jiliang Tang, Yihong Eric Zhao, and Dawei Yin. 2018. Hierarchical variational memory network for dialogue generation. In *Proceedings of the 2018 World Wide Web Conference*. 1653–1662.

[9] Kyunghyun Cho, Bart van Merriënboer, Dzmitry Bahdanau, and Yoshua Bengio. 2014. On the properties of neural machine translation: Encoder–decoder approaches. In *Proceedings of the 8th Workshop on Syntax, Semantics and Structure in Statistical Translation*. 103–111.

[10] Kyunghyun Cho, Bart van Merriënboer, Caglar Gulcehre, Dzmitry Bahdanau, Fethi Bougares, Holger Schwenk, and Yoshua Bengio. 2014. Learning phrase representations using RNN encoder–decoder for statistical machine translation. In *Proceedings of the 2014 Conference on Empirical Methods in Natural Language Processing*. 1724–1734.

[11] Konstantina Christakopoulou, Filip Radlinski, and Katja Hofmann. 2016. Towards conversational recommender systems. In *KDD 2016: 22nd ACM SIGKDD Conference on Knowledge Discovery and Data Mining*. ACM, 815–824.

[12] Kenneth Ward Church and Patrick Hanks. 1989. Word association norms, mutual information and lexicography. In *27th Annual Meeting of the Association for Computational Linguistic*. 76–83.

[13] W. Bruce Croft and R. H. Thompson. 1987. I3R: A new approach to the design of document retrieval systems. *J. Assoc. Inf. Sci. Technol.* 38, 6 (1987), 389–404.

[14] Kaustubh D. Dhole and Christopher D. Manning. 2020. Syn-QG: Syntactic and shallow semantic rules for question generation. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. 752–765.

[15] Xinya Du, Junru Shao, and Claire Cardie. 2017. Learning to ask: Neural question generation for reading comprehension. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*. 1342–1352.

[16] Nan Duan, Duyu Tang, Peng Chen, and Ming Zhou. 2017. Question generation for question answering. In *Proceedings of the 2017 Conference on Empirical Methods in Natural Language Processing*. 866–874.

[17] Chongming Gao, Wenqiang Lei, Xiangnan He, Maarten de Rijke, and Tat-Seng Chua. 2021. Advances and challenges in conversational recommender systems: A survey. *AI Open* 2 (July 2021), 100–126.

[18] Yifan Gao, Piji Li, Irwin King, and Michael R. Lyu. 2019. Interconnected question generation with coreference alignment and conversation flow modeling. In *Proceedings of the 57th Conference of the Association for Computational Linguistics*. 4853–4862.

[19] Wenpeng Hu, Bing Liu, Jinwen Ma, Dongyan Zhao, and Rui Yan. 2018. Aspect-based question generation. In *6th International Conference on Learning Representations*.

[20] Minlie Huang, Xiaoyan Zhu, and Jianfeng Gao. 2020. Challenges in building intelligent open-domain dialog systems. *ACM Trans. Inf. Syst. (TOIS)* 38, 3 (2020), 1–32.

[21] Alex Kendall, Yarin Gal, and Roberto Cipolla. 2018. Multi-task learning using uncertainty to weigh losses for scene geometry and semantics. In *2018 IEEE Conference on Computer Vision and Pattern Recognition*. 7482–7491.

[22] Johannes Kiesel, Arefeh Bahrami, Benno Stein, Avishek Anand, and Matthias Hagen. 2018. Toward voice query clarification. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*. Association for Computing Machinery, 1257–1260.

[23] Antonios Minas Krasakis, Mohammad Aliannejadi, Nikos Voskarides, and Evangelos Kanoulas. 2020. Analysing the effect of clarifying questions on document ranking in conversational search. In *Proceedings of the 2020 ACM SIGIR on International Conference on Theory of Information Retrieval*. 129–132.

[24] Tian Lan, Xianling Mao, Wei Wei, Xiaoyan Gao, and Heyan Huang. 2020. PONE: A novel automatic evaluation metric for open-domain generative dialogue systems. *CoRR* abs/2004.02399 (2020).

[25] Wenqiang Lei, Xiangnan He, Yisong Miao, Qingyun Wu, Richang Hong, Min-Yen Kan, and Tat-Seng Chua. 2020. Estimation-action-reflection: Towards deep interaction between conversational and recommender systems. In *Proceedings of the 13th International Conference on Web Search and Data Mining (WSDM'20)*. 304–312.

[26] Jiwei Li, Michel Galley, Chris Brockett, Jianfeng Gao, and Bill Dolan. 2016. A diversity-promoting objective function for neural conversation models. In *The 2016 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 110–119.

[27] Jiwei Li, Alexander H. Miller, Sumit Chopra, Marc'Aurelio Ranzato, and Jason Weston. 2017. Learning through dialogue interactions by asking questions. In *5th International Conference on Learning Representations*.

[28] Yanran Li, Hui Su, Xiaoyu Shen, Wenjie Li, Ziqiang Cao, and Shuzi Niu. 2017. DailyDialog: A manually labelled multi-turn dialogue dataset. In *Proceedings of the E8th International Joint Conference on Natural Language Processing*. 986–995.

[29] Chin-Yew Lin and Eduard H. Hovy. 2003. Automatic evaluation of summaries using n-gram co-occurrence statistics. In *Human Language Technology Conference of the North American Chapter of the Association for Computational Linguistics (HLT-NAACL'03)*.

[30] Tsung-Yi Lin, Priya Goyal, Ross Girshick, Kaiming He, and Piotr Dollár. 2017. Focal loss for dense object detection. In *Proceedings of the IEEE International Conference on Computer Vision*. 2980–2988.

[31] Yanxiang Ling, Fei Cai, Honghui Chen, and Maarten de Rijke. 2020. Leveraging context for neural question generation in open-domain dialogue systems. In *WWW'20: The Web Conference 2020*. 2486–2492.

[32] Yanxiang Ling, Fei Cai, Xuejun Hu, Jun Liu, Wanyu Chen, and Honghui Chen. 2021. Context-controlled topic-aware neural response generation for open-domain dialog systems. *Inf. Process. & Manage.* 58, 1 (2021), 102392.

[33] Chia-Wei Liu, Ryan Lowe, Iulian Vlad Serban, Mike Noseworthy, Laurent Charlin, and Joelle Pineau. 2016. How NOT to evaluate your dialogue system: An empirical study of unsupervised evaluation metrics for dialogue response generation. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*. 2122–2132.

[34] Xiaodong Liu, Pengcheng He, Weizhu Chen, and Jianfeng Gao. 2019. Multi-task deep neural networks for natural language understanding. In *Proceedings of the 57th Conference of the Association for Computational Linguistics*, Anna Korhonen, David R. Traum, and Lluís Màrquez (Eds.). 4487–4496.

[35] Luis Enrico Lopez, Diane Kathryn Cruz, Jan Christian Blaise Cruz, and Charibeth Cheng. 2020. Transformer-based end-to-end question generation. *CoRR* abs/2005.01107 (2020).

[36] Mohsen Mesgar and Michael Strube. 2018. A neural local coherence model for text quality assessment. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. 4328–4339.

[37] Mao Nakanishi, Tetsunori Kobayashi, and Yoshihiko Hayashi. 2019. Towards answer-unaware conversational question generation. In *Proceedings of the 2nd Workshop on Machine Reading for Question Answering (MRQA@EMNLP'19)*. 63–71.

[38] Boyuan Pan, Hao Li, Ziyu Yao, Deng Cai, and Huan Sun. 2019. Reinforced dynamic reasoning for conversational question generation. In *Proceedings of the 57th Conference of the Association for Computational Linguistics*. 2114–2124.

[39] Liangming Pan, Yuxi Xie, Yansong Feng, Tat-Seng Chua, and Min-Yen Kan. 2020. Semantic graphs for generating deep questions. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. 1463–1475.

[40] Kishore Papineni, Salim Roukos, Todd Ward, and Wei-Jing Zhu. 2002. BLEU: A method for automatic evaluation of machine translation. In *Proceedings of the 40th Annual Meeting of the Association for Computational Linguistics*. 311–318.

[41] Filip Radlinski and Nick Craswell. 2017. A theoretical framework for conversational search. In *Proceedings of the 2017 Conference on Conference Human Information Interaction and Retrieval (CHIIR'17)*. 117–126.

[42] 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. *CoRR* abs/1910.10683 (2019).

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

[44] Pengjie Ren, Zhumin Chen, Zhaochun Ren, Evangelos Kanoulas, Christof Monz, and Maarten de Rijke. 2021. Conversations with search engines. *ACM Trans. Inf. Syst.* 30, 2 (2021).

[45] Anna Sepliarskaia, Julia Kiseleva, Filip Radlinski, and Maarten de Rijke. 2018. Preference elicitation as an optimization problem. In *RecSys 2018: The ACM Conference on Recommender Systems*. ACM, 172–180.

[46] Iulian Vlad Serban, Alberto García-Durán, Çaglar Gülçehre, Sungjin Ahn, Sarath Chandar, Aaron C. Courville, and Yoshua Bengio. 2016. Generating factoid questions with recurrent neural networks: The 30M factoid question-answer corpus. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics*.

[47] Iulian Vlad Serban, Ryan Lowe, Peter Henderson, Laurent Charlin, and Joelle Pineau. 2018. A survey of available corpora for building data-driven dialogue systems: The journal version. *Dialogue Discourse* 9, 1 (2018), 1–49.

[48] Iulian V Serban, Alessandro Sordoni, Yoshua Bengio, Aaron Courville, and Joelle Pineau. 2016. Building end-to-end dialogue systems using generative hierarchical neural network models. In *Proceedings of the 30th AAAI Conference on Artificial Intelligence*. 3776–3783.

[49] Iulian Vlad Serban, Alessandro Sordoni, Ryan Lowe, Laurent Charlin, Joelle Pineau, Aaron Courville, and Yoshua Bengio. 2017. A hierarchical latent variable encoder-decoder model for generating dialogues. In *Proceedings of the 31st AAAI Conference on Artificial Intelligence*. 3295–3301.

[50] Lifeng Shang, Zhengdong Lu, and Hang Li. 2015. Neural responding machine for short-text conversation. In *Proceedings of the 53rd Annual Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing of the Asian Federation of Natural Language Processing*. 1577–1586.

[51] Alessandro Sordoni, Michel Galley, Michael Auli, Chris Brockett, Yangfeng Ji, Margaret Mitchell, Jian-Yun Nie, Jianfeng Gao, and Bill Dolan. 2015. A neural network approach to context-sensitive generation of conversational responses. In *Proceedings of the 2015 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*. 196–205.

[52] Xingwu Sun, Jing Liu, Yajuan Lyu, Wei He, Yanjun Ma, and Shi Wang. 2018. Answer-focused and position-aware neural question generation. In *Proceedings of the 2018 Conference on Empirical Methods in Natural Language Processing*. 3930–3939.

[53] Ilya Sutskever, Oriol Vinyals, and Quoc V. Le. 2014. Sequence to sequence learning with neural networks. In *Proceedings of the 27th International Conference on Neural Information Processing Systems*. 3104–3112.

[54] Duyu Tang, Nan Duan, Zhao Yan, Zhirui Zhang, Yibo Sun, Shujie Liu, Yuanhua Lv, and Ming Zhou. 2018. Learning to collaborate for question answering and asking. In *Proceedings of the 2018 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, Marilyn A. Walker, Heng Ji, and Amanda Stent (Eds.). 1564–1574.

[55] Paul Thomas, Mary Czerwinski, Daniel McDuff, Nick Craswell, and Gloria Mark. 2018. Style and alignment in information-seeking conversation. In *Proceedings of the 2018 Conference on Human Information Interaction and Retrieval (CHIIR'18)*. 42—51.

[56] Johanne R. Trippas, Damiano Spina, Lawrence Cavedon, Hideo Joho, and Mark Sanderson. 2018. Informing the design of spoken conversational search: Perspective paper. In *Proceedings of the 2018 Conference on Human Information Interaction & Retrieval (CHIIR'18)*. 32–41.

[57] Svitlana Vakulenko, Evangelos Kanoulas, and Maarten de Rijke. 2021. A large-scale analysis of mixed initiative in information-seeking dialogues for conversational search. *ACM Trans. Inf. Syst.* 39, 4 (August 2021), Article 49.

[58] Svitlana Vakulenko, Ilya Markov, and Maarten de Rijke. 2017. Conversational exploratory search via interactive storytelling. In *1st International Workshop on Search-Oriented Conversational AI*.

[59] Svitlana Vakulenko, Kate Revoredo, Claudio Di Ciccio, and Maarten de Rijke. 2019. QRFA: A data-driven model of information-seeking dialogues. In *ECIR 2019: 41st European Conference on Information Retrieval*. Springer, 541–557.

[60] Svitlana Vakulenko, Vadim Savenkov, and Maarten de Rijke. 2020. Conversational browsing. *arXiv:2012.03704*. https://arxiv.org/abs/2012.03704.

[61] Weichao Wang, Shi Feng, Daling Wang, and Yifei Zhang. 2019. Answer-guided and semantic coherent question generation in open-domain conversation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*. 5065–5075.

[62] Wenjie Wang, Minlie Huang, Xin-Shun Xu, Fumin Shen, and Liqiang Nie. 2018. Chat more: Deepening and widening the chatting topic via a deep model. In *The 41st International ACM SIGIR Conference on Research & Development in Information Retrieval*. 255–264.

[63] Yansen Wang, Chenyi Liu, Minlie Huang, and Liqiang Nie. 2018. Learning to ask questions in open-domain conversational systems with typed decoders. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*. 2193–2203.

[64] Matthijs J. Warrens. 2021. Kappa coefficients for dichotomous-nominal classifications. *Adv. Data Anal. Classif.* 15, 1 (2021), 193–208.

[65] Ho Chung Wu, Robert Wing Pong Luk, Kam Fai Wong, and Kui Lam Kwok. 2008. Interpreting tf-idf term weights as making relevance decisions. *ACM Trans. Inf. Syst. (TOIS)* 26, 3 (2008), 1–37.

[66] Chen Xing, Wei Wu, Yu Wu, Jie Liu, Yalou Huang, Ming Zhou, and Wei-Ying Ma. 2017. Topic aware neural response generation. In *Proceedings of the 31st AAAI Conference on Artificial Intelligence*. 3351–3357.

[67] Chen Xing, Yu Wu, Wei Wu, Yalou Huang, and Ming Zhou. 2018. Hierarchical recurrent attention network for response generation. In *Proceedings of the 32nd AAAI Conference on Artificial Intelligence*. 5610–5617.

[68] Jingjing Xu, Yuechen Wang, Duyu Tang, Nan Duan, Pengcheng Yang, Qi Zeng, Ming Zhou, and Xu Sun. 2019. Asking clarification questions in knowledge-based question answering. 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'19)*. 1618–1629.

[69] Peng Xu, Hamidreza Saghir, Jin Sung Kang, Teng Long, Avishek Joey Bose, Yanshuai Cao, and Jackie Chi Kit Cheung. 2019. A cross-domain transferable neural coherence model. In *Proceedings of the 57th Conference of the Association for Computational Linguistics*. 678–687.

[70] Hamed Zamani, Susan Dumais, Nick Craswell, Paul Bennett, and Gord Lueck. 2020. Generating clarifying questions for information retrieval. In *Proceedings of the 29th International Conference on World Wide Web (WWW'20)*.

[71] Hainan Zhang, Yanyan Lan, Liang Pang, Jiafeng Guo, and Xueqi Cheng. 2019. ReCoSa: Detecting the relevant contexts with self-attention for multi-turn dialogue generation. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*. 3721–3730.

[72] Houyu Zhang, Zhenghao Liu, Chenyan Xiong, and Zhiyuan Liu. 2020. Grounded conversation generation as guided traverses in commonsense knowledge graphs. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*. 2031–2043.

[73] Saizheng Zhang, Emily Dinan, Jack Urbanek, Arthur Szlam, Douwe Kiela, and Jason Weston. 2018. Personalizing dialogue agents: I have a dog, do you have pets too?. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics*. 2204–2213.

[74] Tianyi Zhang, Varsha Kishore, Felix Wu, Kilian Q. Weinberger, and Yoav Artzi. 2020. BERTScore: Evaluating text generation with BERT. In *8th International Conference on Learning Representations (ICLR'20)*.

[75] Yongfeng Zhang, Xu Chen, Qingyao Ai, Liu Yang, and W. Bruce Croft. 2018. Towards conversational search and recommendation: System ask, user respond. In *Proceedings of the 27th ACM International Conference on Information & Knowledge Management (CIKM'18)*. 177–186.

[76] 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 *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*. 270–278.

[77] Tiancheng Zhao, Ran Zhao, and Maxine Eskenazi. 2017. Learning discourse-level diversity for neural dialog models using conditional variational autoencoders. In *Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics*. 654–664.

[78] Qingyu Zhou, Nan Yang, Furu Wei, Chuanqi Tan, Hangbo Bao, and Ming Zhou. 2017. Neural question generation from text: A preliminary study. In *Natural Language Processing and Chinese Computing—6th CCF International Conference*. 662–671.

[79] Wenjie Zhou, Minghua Zhang, and Yunfang Wu. 2019. Multi-task learning with language modeling for question generation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*, Kentaro Inui, Jing Jiang, Vincent Ng, and Xiaojun Wan (Eds.). 3392–3397.

[80] Wenjie Zhou, Minghua Zhang, and Yunfang Wu. 2019. Question-type driven question generation. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing*. 6031–6036.

Received 16 January 2021; revised 2 October 2021; accepted 6 January 2022