A Review on Question Generation from Natural Language Text

RUQING ZHANG, JIAFENG GUO, LU CHEN, YIXING FAN, and XUEQI CHENG,
CAS Key Lab of Network Data Science and Technology, Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China; University of Chinese Academy of Sciences, Beijing, China

Question generation is an important yet challenging problem in Artificial Intelligence (AI), which aims to generate natural and relevant questions from various input formats, e.g., natural language text, structure database, knowledge base, and image. In this article, we focus on question generation from natural language text, which has received tremendous interest in recent years due to the widespread applications such as data augmentation for question answering systems. During the past decades, many different question generation models have been proposed, from traditional rule-based methods to advanced neural network-based methods. Since there have been a large variety of research works proposed, we believe it is the right time to summarize the current status, learn from existing methodologies, and gain some insights for future development. In contrast to existing reviews, in this survey, we try to provide a more comprehensive taxonomy of question generation tasks from three different perspectives, i.e., the types of the input context text, the target answer, and the generated question. We take a deep look into existing models from different dimensions to analyze their underlying ideas, major design principles, and training strategies We compare these models through benchmark tasks to obtain an empirical understanding of the existing techniques. Moreover, we discuss what is missing in the current literature and what are the promising and desired future directions.

CCS Concepts: • **Computing methodologies → Natural language generation;**

Additional Key Words and Phrases: Question generation, natural language generation, survey

ACM Reference format:
Ruqing Zhang, Jiafeng Guo, Lu Chen, Yixing Fan, and Xueqi Cheng. 2021. A Review on Question Generation from Natural Language Text. *ACM Trans. Inf. Syst.* 40, 1, Article 14 (August 2021), 43 pages.
https://doi.org/10.1145/3468889

14

This work was funded by the National Natural Science Foundation of China (NSFC) under Grant Nos. 62006218, 61902381, 61773362, and 61872338, Beijing Academy of Artificial Intelligence (BAAI) under Grant Nos. BAAI2019ZD0306, the Youth Innovation Promotion Association CAS under Grant Nos. 20144310, 2016102, and 2021100, the Lenovo-CAS Joint Lab Youth Scientist Project, the K.C. Wong Education Foundation, and the Foundation and Frontier Research Key Program of Chongqing Science and Technology Commission (No. cstc2017jcyjBX0059).
Authors' addresses: R. Zhang, J. Guo (corresponding author), L. Chen, Y. Fan, and X. Cheng, CAS Key Lab of Network Data Science and Technology, Institute of Computing Technology, Chinese Academy of Sciences, Beijing, China; University of Chinese Academy of Sciences, Beijing, China, NO. 6 Kexueyuan South Road, Haidian District, Beijing, China, 100190; emails: {zhangruqing, guojiafeng, chenlu19z, fanyixing, cxq}@ict.ac.cn.

1046-8188/2021/08-ART14 $15.00
https://doi.org/10.1145/3468889

1 INTRODUCTION

Question is an important research area in AI, which concerns the task of "automatically generating questions from some form of input varying from information in a database to a deep semantic representations to raw text" [185, 188, 221]. Recently, researchers have also widened the spectrum of sources to include knowledge bases [62, 96, 106, 196, 204] and images [65, 66, 128, 150, 256]. In this article, we focus on question generation from natural language text (QG for short), which aims to generate human-like questions from a source text and optionally a specific target answer. It is receiving increasing interest in recent years from both industrial and academic communities because of its huge potential benefits to various fields. For example, as a dual task of reading comprehension and question answering, QG can be leveraged as a data augmentation strategy to reduce the needed human labor for creating large-scale annotated question-answer pairs; Besides, QG is useful in conversational systems and chatbots, e.g., Siri, Cortana, Alexa, and Google Assistant, which can help them kick-start or continue a conversation with humans for better interactiveness and persistence. Furthermore, QG from educational content is often leveraged as an important component in intelligent tutor systems, which is crucial for evaluating students knowledge and stimulating self-learning.

Conventional methods for QG heavily rely on heuristic rules to transform a descriptive text to a related question. Existing rule-based methods can be broadly divided into three categories, i.e., template-based [151], syntax-based [110], and semantic-based [91] methods. Generally, these methods perform two steps, i.e., context selection and question construction, to consider the answer type and the question type, respectively. Given an input context text, content selection first finds a topic worthwhile to ask about via either a semantic or syntactic parsing method. Then, given the context with the selected topic, question construction converts the intermediate representations to a natural language question, taking either a transformation-based or template-based approach. However, such methods rely on effective handcrafted features, which is often time-consuming and requires domain-dependent expertise and experience. Moreover, they usually comprise the pipelines of several independent components, with low generalizability and scalability.

Recently, we have witnessed the bloom of deep neural models in the QG field. Neural QG models provide a fully data-driven and end-to-end trainable framework, in which content selection and question construction could be jointly optimized. Neural QG models have shown great advantages compared with previous rule-based methods both in terms of question fluency and diversity. Without loss of generality, most neural approaches formulate the QG task as a **Sequence-to-Sequence (Seq2Seq)** problem and design different types of encoders and decoders to improve the quality of generated questions. Perhaps the first neural QG model is introduced in 2017 by Reference [57], which achieves better performance than traditional rule-based approaches by employing a vanilla RNN-based Seq2Seq model with attention [14]. Later, many works are proposed to make RNN-based Seq2Seq framework more powerful by leveraging the question types [61, 145], answer position features [135, 264], answer separation [88, 203], and self-attention mechanism [25, 108]. In addition, some popular frameworks, such as pre-trained framework [53], variational autoencoder [228], graph-based framework [36], and adversarial network [16], have also gained much attention for question generation. Besides the widely used maximum likelihood estimation [94], some works employ multi-task learning [227], reinforcement learning [66], transfer learning [129], and other effective training strategies to optimize the neural QG models.

Up to now, we have seen exciting progress on QG models. Workshops and tutorials on QG have attracted extensive interest in the research community [19, 89, 187, 189]. Standard benchmark datasets [21, 172, 216], evaluation tasks [9, 193], and open-source toolkits [83, 245] have been created to facilitate research and rigorous comparison. Despite these exciting results, the QG area

lacks a comprehensive taxonomy for better understanding the existing QG tasks. Besides, there is little understanding and few guidelines on the design principles and learning strategies of different models. Therefore, it is the right moment to take a look back, summarize the current status, and gain some insights for future development.

There have been some related surveys on QG. On one hand, we find that the coverage is not enough to provide a hands-on guide for understanding the QG task and advancing the QG models. For example, Rakangor and Ghodasara [173] summarized the unsupervised QG methods before 2015, while Pan et al. [162] gave an introduction to the emerging research in neural QG until 2019. On the other hand, some researchers summarize the existing progress from specific aspects. For instance, Kurdi et al. [111] reviewed the automatic question generation for educational purposes, while Ch and Saha [24] performed a systematic review of automatic multiple choice question generation. Besides, Amidei et al. [9] analyzed the evaluation methodologies used in QG. In contrast to existing reviews, the advantages of our survey are as follows: First, we provide a more comprehensive taxonomy of QG tasks, in terms of the types of the input context text, the target answer, and the generated question. We then review existing QG models to analyze their underlying assumptions, major design principles, and learning strategies. We also compare these models through representative benchmark tasks to obtain an empirical understanding. We hope these discussions will help researchers in QG learn from previous successes and failures so they can develop better QG models in the future. In addition to the model discussion, we introduce some trending topics in QG, including diverse question generation, pre-training for question generation, question generation with higher cognitive levels, and question generation for information seeking. Some of these topics are important but have not been well addressed in this field, while others are very promising directions for future development.

In the following, we will first introduce some typical applications of QG in Section 2. We then provide a comprehensive task description and introduce the benchmark datasets in Section 3. From Section 4 to 6, we review the existing models with regard to different dimensions as well as making empirical comparisons between them. We discuss trending topics in Section 7 and conclude the article in Section 8.

2 MAJOR APPLICATIONS OF QUESTION GENERATION

In this section, we describe several major application scenarios where QG has been adopted and studied in the literature, including question answering, machine reading comprehension, automatic conversation, and intelligent tutor.

2.1 Question Answering

Question Answering (QA) aims to automatically return exact answers as either short facts or long passages to natural language questions issued by users. Sufficient labeled QA data is critical for modern QA methods based on deep learning models to achieve satisfactory performance. Although labeling efforts have been made, these datasets are still with limited sizes, as labeling is very expensive and time-consuming. Classical QG could be defined as the reverse task of QA, and thus has the potential to generate large-scale QA pairs to assist the QA systems. For example, Duan et al. [58] proposed to generate questions from given passages based on QA training pairs collected from Community-QA website. To ensure that QG is helpful for QA, the QA pair generation task is integrated into an end-to-end QA task. Fang et al. [67] leveraged QG for real-time QA, which first generates a large pool of QA pairs offline and then matches an input question with the candidate QA pool to predict the answer in real time.

2.2 Machine Reading Comprehension

Machine Reading Comprehension (MRC) aims to read and understand unstructured texts and then answer questions about it. Most state-of-the-art MRC models rely on large amount of human-annotated in-domain data to achieve the desired performance. Although there exists a number of large-scale MRC datasets, collecting such high-quality datasets is still expensive and time-consuming. Recently, QG is leveraged as a data augmentation strategy for MRC by generating questions to the point. For example, Du et al. [57] presented a fully data-driven neural networks approach to automatically generate questions for MRC. Gao et al. [70] presented a novel setting, i.e., difficulty controllable QG for RC, where the generated RC questions should satisfy the specified difficulty as much as possible. Wang et al. [225] first generated pseudo-questions given passages in a target domain and then used the generated data as augmentation to fine-tune a pre-trained MRC model from a source domain to a target domain.

2.3 Automatic Conversation

Automatic conversation (AC) aims to create an automatic human-computer dialogue process. However, manually building such conversational datasets is quite expensive. For example, CoQA [176] spent 3.6 USD per passage on crowdsourcing for conversation collection [161]. Also, the inability to ask questions based on previous turns may cause users' experience dissatisfaction [126]. Recently, to enhance the interactivity and persistence of dialogues, QG serves as an essential communication skill to help collect users' feedback and extend current conversational topics or start new ones. Specifically, **conversational question generation (CQG)** task is proposed to lead a QA-style conversation, which aims to generate a question given an input text and a conversation history. For example, Pan et al. [161] proposed an effective framework for CQG, which is equipped with a dynamic reasoning component to generate a conversational question and further fine-tuned via a reinforcement learning mechanism. Ling et al. [133] leveraged the potential topics from the conversational context to generate appropriate and informative questions for promoting interactivity and persistence of multi-turn dialogues. Wang et al. [230] considered asking questions in open-domain conversational systems with soft- and hard-typed decoders.

2.4 Intelligent Tutor

In traditional educational conditions (e.g., classrooms) or large-scale web-based conditions (e.g., online courses), people have found that providing students with quiz questions contributes to better learning outcomes than spending an equal amount of time studying educational content such as textbooks and lecture notes [44]. Although human-generated questions have been widely used, an unheard-of growth of educational content outpaces the manually written questions with respect to them. Moreover, manually writing a large set of high-quality questions is time-consuming due to the extensive efforts required of human domain experts. Therefore, there is a pressing need to find ways to automatically generate questions from educational content. Recently, many works are proposed to generate questions for developing intelligent tutor systems. For example, Wang et al. [233] leveraged several recent advances in summarization and QA and proposed an RNN-based QG model specifically designed for quiz question generation from educational content. Chen et al. [31] presented a novel educational QG dataset with over 230K document-question pairs, i.e., LearningQ, which covers a diverse set of educational topics. For more details of the automatic QG for educational applications, we refer readers to the related survey cited in Reference [112].

For the evaluation of education applications, most studies focused on the quality of the generated questions and the learning effectiveness contributed by the QG component. Since there are little metrics specifically designed to measure the quality of questions, people often adopt BLEU

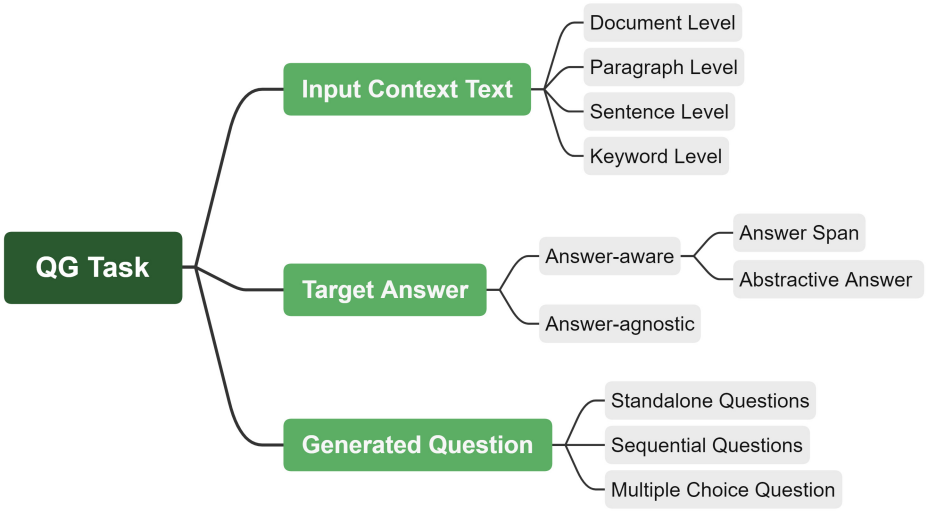

Fig. 1. Taxonomy of the QG tasks.

[166], METEOR [48], and ROUGE [131] as evaluation metrics. Besides, human evaluation is often conducted to judge the grammatical correctness, appropriateness, and helpfulness of the generated questions using different scores [134]. For the learning effectiveness, people often conduct the impact on students learning the educational system with the QG component or not [148].

3 PROBLEM STATEMENT

In this section, we introduce our proposed taxonomy of the QG tasks and describe the benchmark datasets for the evaluation.

3.1 Task Description

Given a context text and optionally a specific target answer, QG aims to generate natural questions relevant to the input. Previous works mainly divide the QG task into two streams, i.e., answer-aware and answer-agnostic. For the answer-aware QG task, the target answer is known and the generated questions ask towards the given answer based on the given context text.

The answer-agnostic QG task lifts the constraint of knowing the target answer before generating the question. In real applications such as intelligent tutor systems, people or machines are often required to create questions from natural language text without explicitly annotated answers.

In this article, as shown in Figure 1, we propose a more comprehensive taxonomy of the QG tasks, in terms of the types of the input context text, the target answer, and the generated question.

3.1.1 Input Context Text. Broadly speaking, the input context text for the QG task could be at four different levels of scope:

- **Document level** denotes that a single document [163, 241] or multiple documents [38, 39] are understood to obtain the relevant questions.
- **Paragraph level** denotes that the information of a single paragraph or multiple paragraphs is required to ask right questions [57, 259].
- **Sentence level** denotes that relevant questions are asked based on a single sentence or multiple sentences [4, 70].
- **Keyword level** denotes that the questions are generated from the given keywords [165, 177].

Table 1. An Example of QG Given a Single Paragraph and an Answer Span from the SQuAD Dataset [172]

Input Paragraph: Computational complexity theory is a branch of the theory of computation in theoretical computer science that focuses on classifying computational problems according to their **inherent difficulty**, and relating those classes to each other. A computational problem is understood to be a task that is in principle amenable to being solved by a computer, which is equivalent to stating that the problem may be solved by mechanical application of mathematical steps, such as an algorithm.
Answer Span: inherent difficulty
Question: By what main attribute are computational problems classified using computational complexity theory?

Bold text in the input paragraph indicates the answer span used to generate the question.

Table 2. An Example of QG Given Multiple Paragraphs and an Abstractive Answer Written by Human Editors from the MS MARCO Dataset [21]

Input Paragraph 1: Rachel Carson's essay on The Obligation to Endure, is a very convincing argument about the harmful uses of chemical, pesticides, herbicides and fertilizers on the environment.
Input Paragraph 2: Carson believes that as man tries to eliminate unwanted insects and weeds; however, he is actually causing more problems by polluting the environment with, for example, DDT and harming living things.
Input Paragraph 3: Carson subtly defers her writing in just the right writing for it to not be subject to an induction run rampant style that grabs the readers interest without biasing the whole article.
Abstractive Answer: Rachel Carson writes The Obligation to Endure because believes that as man tries to eliminate unwanted insects and weeds; however he is actually causing more problems by polluting the environment.
Question: Why did Rachel Carson write an obligation to endure?

3.1.2 Target Answer. Specifically, a widely used framework for categorizing the cognitive levels involved in question asking, i.e., multi-layered answer to a question, is Bloom's Revised Taxonomy [10]. It contains six learning objectives with cognitive skill levels from lower order to higher order, i.e., Reading, Understanding, Applying, Analyzing, Evaluating, and Creating.

As described above, most works are devoted to the answer-aware and answer-agnostic QG. For the answer-aware QG, the expected form of the answer typically has two settings, i.e., answer span and abstractive answer. Without loss of generality, answer span focuses on shallow levels of Bloom's taxonomy, while abstractive answer caters to higher levels of Bloom's taxonomy.

- **Answer span** denotes that a continuous subsequence of tokens is extracted from the input text as the target answer. It is usually the case based on shallow levels of Bloom's taxonomy, which enables *who*, *what*, *where*, *when*, *how much,* and other factoid questions. Table 1 shows a QG example that leverages an answer span to ask a question about.
- **Abstractive answer** is usually the case for non-factoid (deep) questions cater to higher levels of Bloom's taxonomy. The human-generated answers may not appear in the original context, which enables *why*, *how*, *what-if*, *what kind of,* and other non-factoid questions. Table 2 shows a QG example that leverages an abstractive answer to ask a question about.

Table 3. An Example of QG Only Given an Input Document from the LearningQ Dataset [31]

Input Document: ... if it was a perfect diode, made in some unknown technology, what would happen is in the reverse direction, if the voltage across the diode was negative, we 'll label the voltage this way, if the voltage across the diode was negative, that is, this terminal is at a higher voltage than this terminal, there would be zero current flowing. and then for any positive voltage, basically the diode would look like a wire. so i can call that, that's essentially model number zero of a diode ...
Question1: what would a diode look like on a circuit board?
Question2: what is the direction of current in a circuit (outside of a dc battery)?
Question3: electron flow vs proton flow?
Question4: does the current just go to infinity because there is no resistance or something?
Question5: why does a photodiode require biasing?

For the answer-agnostic QG, questions are generated without the supervision of the annotated answers. Table 3 shows a QG example without annotated answers. We can find that multiple diverse questions are related to the same document.

3.1.3 Generated Question. The type of the generated question is another piece of information used in guiding the generation, including,

- **Standalone Questions** denote a single question or a series of independent questions without interaction. The example questions in Tables 1, 2, and 3 are standalone questions.
- **Sequential Questions** are a series of interconnected questions with the availability of asking a sequence of answers. Intuitively, it is more natural for humans to ask questions based on previous discussions to get better user experiences. Table 4 shows an example of QG with sequential questions generated.
- **Multiple Choice Question** is a simple closed-ended question type with multiple options, one being the answer and the others being distractors. Under this setting, the question with respect to the given answer and its distractors should be jointly generated. Table 5 shows an example of QG with multiple choice questions generated.

3.2 Benchmark Datasets

In this section, we review the widely used benchmark datasets in the QG field. Researchers usually leverage large-scale QA and MRC datasets for the QG task, since the roles of questions and answers are switchable given the input texts. Building high-quality datasets specifically for QG will be an important future direction, which could largely boost the development of this field.

Here, we classify popular QG datasets based on the types of the target answer and the generated question in our proposed taxonomy.

3.2.1 Standalone Questions. Here, we introduce the representative standalone question generation datasets with different types of answers.

- **Answer Span.** The widely used datasets for generating standalone questions using answer spans include:
  - **SQuAD 1.1. S**tanford **Q**uestion **A**nswering **D**ataset (SQuAD)[1] is a large MRC dataset on Wikipedia articles with crowdsourced QA pairs. SQuAD 1.1 [172] contains 100K

[1] https://stanford-qa.com/.

Table 4. An Example of Sequential Question Generation Given a Document and a Series of Answers from the CoQA Dataset [176]

Input Document: The morning of that Wednesday of Corpus Christi, fateful to all concerned in this chronicle, dawned misty and grey, and the air was chilled by the wind that blew from the sea. The chapel bell tinkled out its summons, and the garrison trooped faithfully to Mass. Presently came Monna Valentina, followed by her ladies, her pages, and lastly, Peppe, wearing under his thin mask of piety an air of eager anxiety and unrest ...
Answer 1: the garrison first **Question 1:** Who arrived at the church?
Answer 2: Fra. Domenico **Question 2:** Who was followed by a clerk dressed in black?
Answer 3: Valentina **Question 3:** Who was crying?
Answer 4: her ladies **Question 4:** Who noticed it?
Answer 5: yes **Question 5:** Did any others arrive with her?

Table 5. An Example of Multiple Choice Question Generation from the RACE Dataset [118], Where the Target Questions and the Distractors Are Generated Simultaneously

Input Paragraph: In a small village in England about 150 years ago, a mail coach was standing on the street. It didn't come to that village often. People had to pay a lot to get a letter. The person who sent the letter didn't have to pay the postage, while the receiver had to. "Here's a letter for Miss Alice Brown," said the mailman ...
Answer: she had already known what was written in the letter
Question: The girl handed the letter back to the mailman because _____. **Distractors:** (1) she didn't know whose letter it was; (2) she had no money to pay the postage; 3) she received the letter but she didn't want to open it

crowd-sourced QA pairs on 536 Wikipedia articles. The questions are written by crowd-workers, and the answers are spans of tokens from the corresponding reading passage.

—**NewsQA.** NewsQA [216] is a large-scale MRC dataset including over 100K human-generated QA pairs. Crowd-workers post questions and answers based on a set of over 10K news articles from CNN, with answers consisting of spans of text from the corresponding articles. Also, each article-question pair is validated by an average of 2.48 crowd-workers.

—**SearchQA.** SearchQA [59] consists of more than 140K QA pairs with each pair having 49.6 snippets on average for MRC or QA. Different from SQuAD and NewsQA that start from an existing article and then generate a QA pair, SearchQA starts from an existing QA pair crawled from J! Archive and augments it with text snippets retrieved by Google.

—**HotpotQA.** HotpotQA [242] is a large-scale QA dataset with 113K Wikipedia-based QA pairs. Given multiple supporting context documents, crowd-workers are asked explicitly to come up with questions requiring reasoning about all of the documents and answer the question by extracting a span of text from the context.

—**NQ. N**atural **Q**uestions [114] is a QA dataset, with 307,373 training examples, 7,830 development examples, and 7,842 test examples. Each example consists of a Google query (i.e., question) and a related Wikipedia page. Specifically, each Wikipedia page has a long

answer (typically a paragraph or table in the page) that contains the information required to answer the question, and a short answer (a span or a set of spans) within the long answer that contains the actual answer.

—**TriviaQA.** TriviaQA [95] is an MRC dataset containing over 650K question-answer-evidence triples. The evidence documents are collected retrospectively from Wikipedia and the Web, and the questions are originated from trivia enthusiasts independent of the evidence documents.

- **Abstractive Answer.** The widely used datasets for generating standalone questions using abstractive answer include,
  —**MS MARCO. M**icro**S**oft **MA**chine **R**eading **CO**mprehension (MS MARCO) [21] is a large-scale real-world MRC dataset, which comprises of 1,010,916 anonymized questions from Bing's search query logs and 8,841,823 paragraphs extracted from 3,563,535 web documents retrieved by Bing. Each question is associated with a human-generated answer and 182,669 answers are completely human-rewritten-generated.
  —**NarrativeQA.** NarrativeQA [103] contains 46,765 human-written QA pairs based on 1,567 stories, evenly split between books and movie scripts. Compared with news and Wikipedia articles, the books and movie scripts are longer, requiring annotators to create questions and answers with more complicated understanding and reasoning abilities.
  —**DuReader.** DuReader [81] is a large-scale and open-domain Chinese MRC dataset built with real application data from Baidu search and Baidu Knows. DuReader contains 200K questions, 420K answers, and 1M documents, where answers are manually generated. Specifically, DuReader provides additional annotations for more question types, especially yes-no and opinion questions, that leaves more opportunity for the research community.
  —**SQuAD 2.0.** SQuAD 2.0 [171] combines the SQuAD 1.1 data with over 50K unanswerable questions written adversarially by crowd-workers to look similar to answerable ones.
- **Answer-agnostic.** The widely used answer-agnostic QG datasets include:
  —**LearningQ.** LearningQ [31] is a challenging educational QG dataset that consists of more than 230K document-question pairs collected from mainstream online learning platforms. In LearningQ, there are 7K instructor-designed questions assessing knowledge concepts being taught and 223K learner-generated questions seeking in-depth understanding of the taught concepts.

3.2.2 Sequential Questions. Here, we introduce the representative sequential question generation datasets with different types of answers.

- **Abstractive Answers.** The widely used datasets for generating sequential questions using abstractive answers include,
  —**CoQA. Co**nversational **Q**uestion **A**nswering (CoQA) [176] is built to enable machines to answer a series of questions that appear in a conversation. Based on 8K conversations about text passages from seven diverse domains, CoQA contains 127K conversational questions with answers that are free-form text with their corresponding evidence highlighted in the corresponding passage.
  —**QuAC. Qu**estion **A**nswering in **C**ontext (QuAC) [40] is a large-scale dataset of information-seeking dialogues over sections from Wikipedia articles, which contains 14K crowdsourced QA dialogues (100K total QA pairs). Unlike other QA datasets such as SQuAD and CoQA, crowd-workers do not know the answers to their questions prior to asking them in order to make QuAC similar to datasets that contain real user queries on search engines.

3.2.3 Multiple Choice Questions. Here, we introduce the representative multiple choice questions generation datasets with different types of answers.

- **Answer Span.** The widely used datasets for generating multiple choice questions using answer spans include,
  - —**CBT. C**hildren's **B**ook **T**est [85] is designed to measure directly how well language models can exploit wider linguistic context. It contains 687,343 questions from 108 books that are freely available in Project Gutenberg. Each question is provided with 10 candidate answers appearing in the context sentences and the query.
- **Abstractive Answer.** The widely used datasets for generating multiple choice questions using abstractive answers include,
  - —**RACE. R**e**A**ding **C**omprehension dataset from **E**xaminations (RACE) [118] is a multiple choice MRC dataset, designed to test Chinese students learning English as a foreign language. RACE contains 97,687 multiple choice questions and 27,933 passages generated by human experts. Each question is provided with four candidate answers and only one of them is correct.
  - —**ARC. A**I2 **R**easoning **C**hallenge (ARC) consists of 7,787 grade-school multiple choice science questions typically with four possible answers. Questions vary in their target student grade levels, ranging from third grade to ninth. ARC also provides a corpus of 14M science-related sentences with knowledge relevant to the challenge questions.
  - —**MCTest.** MCTest [181] is a multiple-choice MRC dataset, which contains 500 fictional stories, with four multiple choice questions per story. The stories and questions are carefully limited to those a young child would understand, reducing the world knowledge that is required for the benchmark task.

4 MODELS

In this section, we review the major QG models to better understand their basic assumptions and design principles. Existing methods on QG could be broadly categorized into two folds, i.e., rule-based methods and neural network-based methods.

4.1 Rule-based Methods

Conventional approaches usually rely on manually designed transformation rules to convert a piece of given text into corresponding questions. Existing rule-based works can be generally classified into template-based, syntax-based, and semantics-based approaches.

4.1.1 Template-based Approaches. Template-based approaches utilize the templates extracted from the training set to create questions for corresponding facts in the testing set, which are suitable for specific applications within a closed-domain. Wolfe [234] designed the earliest such experimental computer-based educational system to improve the independent study of any textual material. It uses a pattern matching method to compare a sentence against pre-defined patterns. Sammut and Banerji [191] introduced a program called Marvin, which is capable of automatically asking certain types of questions using pure logical rules to help investigate the learning of concepts. Mostow and Chen [151] and Chen and Aist [33] presented a template approach to generate self-questioning instructions from narrative and informational text, especially focusing on educational purpose. Ureel et al. [218] described the Ruminator, which takes summarized context as input and generates a large number of easy questions using simple rules and templates. Zheng et al. [260] proposed a template-based technique to construct questions from Chinese text, and the generated questions are ranked using a multi-feature neural ranking model. Recently, Liu et al. [139] combined template-based method and Seq2Seq learning to generate highly fluent and

diverse questions. Fabbri et al. [63] applied a template on a related retrieved sentence rather than the original context sentence.

4.1.2 Syntax-based Approaches. Syntax-based approaches first determine the syntactic structure of a given text and then apply syntactic transformation rules and question word placements to obtain the questions. Straach and Truemper [206] applied knowledge base and logic programming techniques to expert systems for learning to ask relevant questions. Mitkov and Ha [149] is one of the earliest attempts to automatically develop multiple choice tests, which involves a simple set of transformation rules and a shallow parser. Kunichika et al. [110] provided five fixed rules based on syntactic analyses for their intelligent English learning system. Harabagiu et al. [77] generated questions from answer passages using predicate-argument patterns. Ali et al. [4] first simplified the complex sentence and then generated questions based on predefined interaction rules. Varga [220] designed some syntactic rules applied for the similar question types. Ali et al. [5] generated the questions using syntactic parsing, Part Of Speech tagger, and Named Entity analyzer.

Moreover, some works explicitly perform transformation by searching tree patterns via Tregex, followed by their manipulation using Tsurgeon [122]. Specifically, Tregex is a utility for matching patterns in trees, based on tree relationships and regular expression matches on nodes. Gates [72] designed a QG system mainly using Stanford NLP's Tsurgeon for transforming declarative sentence trees into WH phrases and questions. Heilman and Smith [82, 84, 199] employed an overgenerate-and-rank strategy for QG. This strategy first generates multiple candidate questions via well-defined syntactic transformations mostly implemented using Tregex and Tsurgeon, and then ranks these questions via handcrafted features.

4.1.3 Semantic-based Approaches. Semantic-based approaches perform semantic analysis of texts to create questions. Yao and Zhang [246] proposed a novel QG approach based entirely on Minimal Recursion Semantics representation without templates or syntax information. Curto et al. [46] relied on patterns that convey lexical, syntactic, and semantic information, automatically learned from the Web. Agarwal et al. [2] generated questions using discourse connectives for different question types. Labutov et al. [116] developed an ontology-crowd-relevance workflow, consisting of representing text in the low-dimensional ontology and crowd-sourcing relevant question templates. Huang and He [91] introduced **Lexical Functional Grammar (LFG)** as the linguistic framework for QG, which enables systematic utilization of semantic and syntactic information. Following Mitkov and Ha [149], many works began to explore semantic-based approaches in the context of automatic generation of multiple choice questions [1, 3, 12, 130, 142]. Recently, Dhole and Manning [51] developed Syn-QG, a set of transparent syntactic rules leveraging universal dependencies, shallow semantic parsing, lexical resources, and custom rules.

Moreover, many works employed **semantic role labeling (SRL)** as an important analytic component for QG. Mannem et al. [143] proposed to exploit semantic roles along with predicate argument structures of sentences to generate questions. Lindberg et al. [132] incorporated SRL into a template-based system to generate questions for supporting online learning. As an extension to Lindberg's system Lindberg et al. [132], Odilinye et al. [159] aligned questions generated to the pedagogical goals and learner's behavior model. Unlike prior work primarily relying on one view of the sentence, Mazidi and Nielsen [144] created a tree structure using multiple views from different parsers to generate questions. This approach is built on a dependency parse, paired with information from semantic role labels and discourse cues. Later, Eldesoky [61] and Mazidi and Tarau [145] first classified sentences before generating questions to determine the question type, which significantly increases the percentage of acceptable questions. Rodrigues et al. [182] performed QG by using lexical, syntactic, and semantic information, where MatePlus SRL [184] is

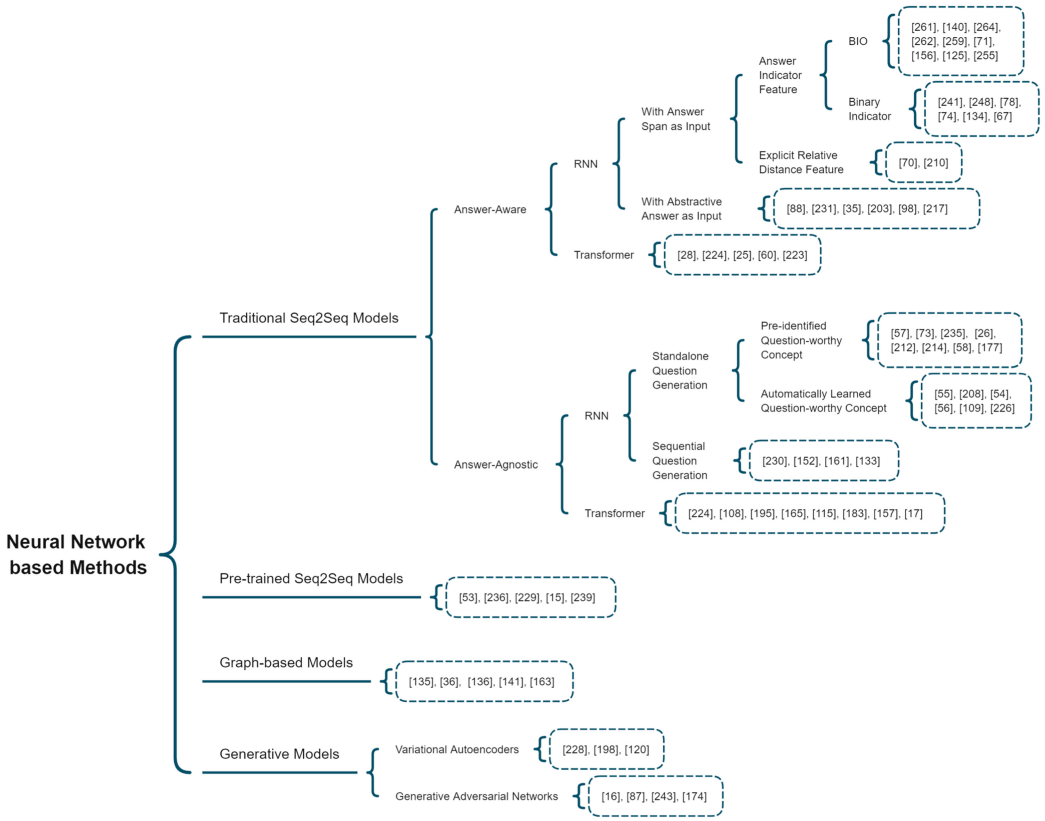

Fig. 2. Existing neural QG models with representative examples.

used to process the input text. Recently, some works [68, 69] leveraged SRL for text analysis for automatic generation of factual questions from sentences.

As described above, rule-based QG methods are easier to interpret and allow people greater control over model behaviors using little data. However, such methods strongly depend handcrafted transformation and generation rules, resulting in a significant lack of diversity in the generated questions and the limitation of flexibility in covering different domains.

4.2 Neural Network-based Methods

With the rise of data-driven learning approaches and the availability of large-scale datasets, neural network-based QG methods have gradually taken the mainstream. As shown in Figure 2, neural QG models could be mainly divided into Seq2Seq models, pre-trained models, variational autoencoder models, graph-based models, and adversarial network models.

4.2.1 Traditional Seq2Seq Models. The majority of neural QG models follow the Seq2Seq framework, which first converts an input context text and optionally a target answer into intermediate representations via an encoder, then use a decoder to generate questions from the intermediate representations. Here, we introduce the Seq2seq models using different types of the target answer, i.e., answer span, abstractive answer, and without answer information.

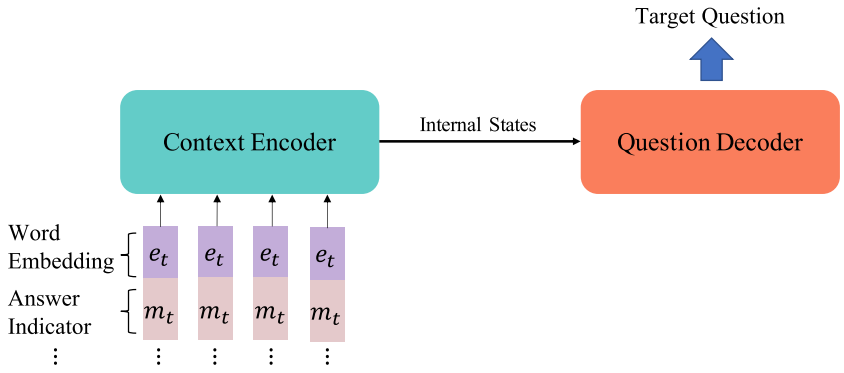

Fig. 3. Seq2Seq models with Answer Span as input.

- *Seq2Seq with Answer Span as Input.* To decide which information to focus on when generating questions, most Seq2Seq models leverage the answer position features to incorporate the answer spans. As shown in Figure 3, various works attempt to augment each word vector of the input context text with an extra answer indicator feature, indicating whether this word is within the answer span. Existing implementations of this feature can be generally categorized into BIO tagging scheme and binary indicator.

  In BIO tagging scheme, tag B denotes the start of an answer, tag I continues the answer, and tag O marks words that do not form part of an answer. Zhou et al. [261] encoded the BIO tags of answer position to real-valued vectors and then fed the answer position embeddings to the feature-rich encoder in RNN-based Seq2Seq model [14]. Later, many works employed this answer tagging that is similar to the techniques in Reference [261]. Ma et al. [140] leveraged the same enriched semantic and lexical features in Reference [261] and proposed sentence-level semantic matching and answer position inferring to improve question generation. Zhou et al. [264] proposed a feature-rich encoder with answer position embedding and predicted the question type and generated questions simultaneously. Following the features used in Reference [264], Zhou et al. [262] leveraged the pointer network [267] to explicitly select a sub-span from the source side to target side. Zhao et al. [259] proposed a gated self-attention encoder with answer tagging and a maxout pointer decoder, applicable to both sentence- and paragraph-level inputs. Gao et al. [71], Nema et al. [156] concatenated the word embedding, character-based embedding, and BIO features as described in Reference [259] for each word in a passage and performed a second pass to generate a revised question. Li et al. [125] proposed a method to jointly model the unstructured sentence and the structured answer-relevant relation for QG. To capture answer-relevant words in the sentence, they adopted a BIO tagging scheme to incorporate the answer position embedding in Seq2Seq learning. Following the model architecture in Reference [259], Zhang and Bansal [255] leveraged BIO tagging together with POS and NER linguistic features to enhance the word embedding from ELMo or BERT, targeted at paragraph-level inputs.

  For binary indicator, if a word token appears in the answer, then the answer position feature is set at one, otherwise zero. Yang et al. [241] and Yuan et al. [248] appended an additional binary feature to the word embeddings of the paragraph or document tokens, respectively. Following Reference [248], by combining the binary feature with additional linguistic features (e.g., named entity recognition, word case, and entity coreference resolution) and the question-specific sentence encoder, Harrison and Walker [78] achieved better results. Recently, Gupta et al. [74] leveraged the binary position information of the answer in

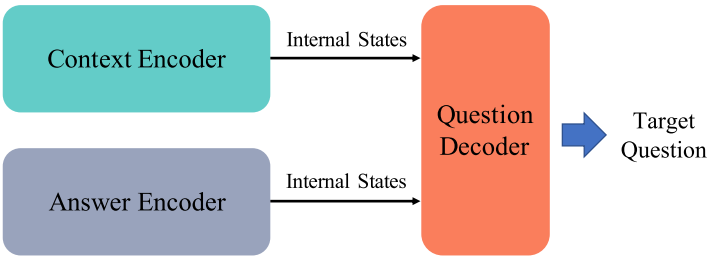

Fig. 4. Seq2Seq models with Abstractive Answer as input.

a list of documents and generated relevant questions based on supporting facts in the context. Liu et al. [134] proposed **Answer-Clue-Style-aware Question Generation (ACS-QG)** and utilized binary features to indicate the positions of answer and clue in input passage. Reference [67] calculated the ground-truth distribution of a position being the start and end of an answer span over all possible answers in each paragraph.

Different from the answer indicator feature, which is a weak representation of relative distance between answer and its context words, many works propose to explicitly model the relative distance between the context words and the answer. In this way, the model could put more emphasis on the answer-surrounded context words. For example, Gao et al. [70] incorporated the relative position embeddings capturing the proximity hints and the word embeddings of the input sentence as the input of encoder in the typical Seq2Seq framework. Sun et al. [210] proposed an answer-focused and position-aware neural QG model, where the relative distance is encoded as position embedding to help the model copy the context words that are relatively close and relevant to the answer.

Among these works, *CS2S-VR-ACS* [134] performs best in terms of BLEU 1-3 and ROUGE_L on SQuAD dataset following the data split in Reference [57]. For more details, see Table 7 in Section 6.1.1. *UniLM+DA-sim* [67] performs best in terms of BLEU4, METEOR, and ROUGE_L on HotpotQA dataset. For more details, see Table 10 in Section 6.1.3.

- *Seq2Seq with Abstractive Answer as Input.* Leveraging the answer position feature has a critical issue, i.e., a significant proportion of the generated questions include words in the target answer [98]. To leverage the abstractive answer information, as shown in Figure 4, many works focus on separately encoding the target answer and the context text. Note some researchers also encode the context text and the answer span separately. Here, we group all these models as Seq2Seq with Abstractive Answer.

  For example, Hu et al. [88] first identified the aspects shared in the given QA pairs, and then encoded the aspect and the answer information separately. Wang et al. [231] designed a weak supervision-based discriminator, which encodes the answer and the passage separately to capture the relations between them and focus on the answer-related parts of the passage. Chen et al. [35] first adopted a **Deep Alignment Network (DAN)** to explicitly model the global interactions between the passage and answer at multiple granularity levels and then generated a question using an RNN-based decoder. Song et al. [203] first encoded the context input and the answer via two separate LSTMs and then matched the answer with the passage before generating the question. Kim et al. [98] proposed an answer-separated Seq2Seq, which first replaces the target answer in the original passage with a special token (i.e., answer masking), and then encodes the masked passage and the answer separately for better utilization of the information from both sides. Different with Reference [98], which masked the target answer with a special token, Tuan et al. [217] directly masked the word

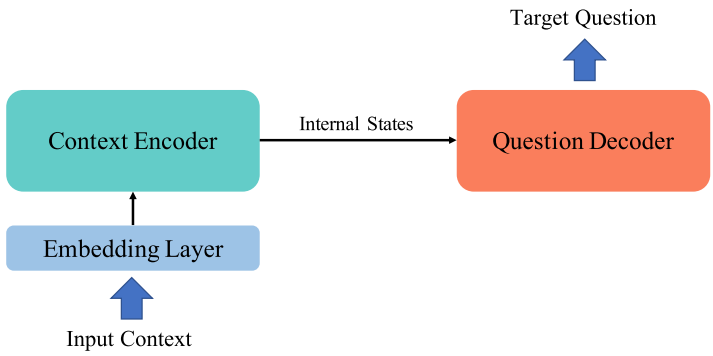

Fig. 5. Seq2Seq model without Answer Input, under the assumption that the input context contains a question-worthy concept.

representation at the position of the answer in the document representation with a special masking vector and denoted it as the final contextual attention representation of document for the input of the decoder. Note that researchers often leverage the above Seq2Seq models on the QG datasets with answer span by separately encoding the context texts and answer spans.

Actually, the above Seq2Seq models with answer span or abstractive answer rely on RNN units, e.g., LSTM [86] or GRU [37], augmented by the attention mechanism [14]. However, the inherent sequential nature of the RNN-based Seq2Seq models suffer from the consequent computational cost and the long-range dependency issues. Recently, the Transformer architecture [222] has been proved to be very effective on various NLP tasks, overcoming the issues caused by RNN. Transformer does not rely on any recurrent gate, which can be briefly described as a SeqSeq model with a symmetric encoder and decoder based on a self-attention mechanism. There exist some works to study the adaptation of Transformer architectures, i.e., Transformer-based Seq2Seq. For example, Chan and Fan [28] investigated the employment of the pre-trained BERT composed by transformer models for QG with the answer span information. Wang et al. [224] proposed to treat the answer spans as the hidden pivot for QG, and adopt the Transformer as the encoder and decoder module. Chai and Wan [25] proposed to generate questions using the answer span information in a semi-autoregressive way, where both encoders and decoder take the form of Transformer architecture. Some works [60, 223] fine-tuned a pre-trained BART language model [123], which combines Bidirectional and Auto-regressive Transformers, to generate questions. Specifically, Wang et al. [223] concatenated the answer to the source article with a special marker token in between, while Durmus et al. [60] masked the important text spans (i.e., the gold answer) in an input sentence. Note that in References [60, 223], the QG is used to evaluate the overall quality of abstractive summarization, which is a novel and interesting direction for researchers in the QG field.

- *Seq2Seq without Answer Input.* In real applications, people or machines are often required to produce questions by removing the dependency on the pre-defined answers. There have been many works to improve the quality of generated questions without the supervision of target answers.

  As shown in Figure 5, Du et al. [57] proposed the first neural QG model using an RNN-based Seq2Seq framework [37] for both sentence- and paragraph-level input text without leveraging the answer information. The attention mechanism [14] is applied to help decoder pay attention to the most relevant parts of the input text while generating a question. Later, many research works [73, 235] are studied to adopt the RNN-based

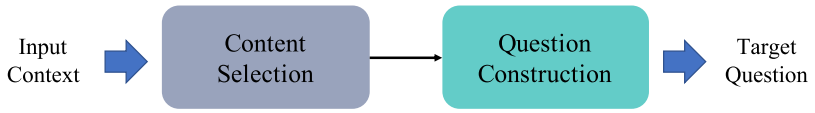

Fig. 6. Seq2Seq model without Answer Input, which automatically learns what is worthy to ask.

Seq2Seq framework for answer-agnostic QG with sentence-level input text. Guo et al. [73] proposed to generate a question given a sentence using an RNN-based Seq2Seq model similar to Reference [57]. Wu et al. [235] proposed a question-type driven framework for answer-agnostic QG, which first predicts the question types and then generates questions that follow the pattern of a specific question type. To prevent generated questions from repeating themselves, Chali and Baghaee [26] incorporated the coverage mechanisms into the RNN-based Seq2Seq framework, which takes sentences as the input text. To deal with the rare or unknown word problem, Tang et al. [212] and Tang et al. [214] incorporated the copy mechanism and post-processed replacing mechanism into Seq2Seq framework, respectively. Moreover, for the passage-level input text, Duan et al. [58] proposed to train two Seq2Seq models, where the former learns to generate most related question template from a passage, and the latter one learns to fill the blank of template with a topic phrase via the copy mechanism. For the keyword-level input text, Reddy et al. [177] leveraged the RNN-based Seq2Seq framework to generate questions from the given set of keywords.

The key assumption of the above models is that the input text contains a question-worthy concept that has been identified in advance. More practically, it is necessary to require the model to automatically learn what is worthy to ask. Recently, as shown in Figure 6, many works provide a two-stage framework in which content selection and question construction are jointly optimized in an end-to-end way [55, 56, 109, 152, 208, 226]. Du and Cardie [55] proposed a hierarchical neural sentence-level sequence tagging model to identify question-worthy sentence from given passages and then incorporated the sentence selection component into previous proposed neural QG system [57]. Similarly, Subramanian et al. [208] first identified key phrases within a passage or document that are likely to be picked by humans to ask questions about. Then, an RNN-based Seq2Seq model with a copy mechanism identical to Reference [248] is employed to construct questions about a given key phrases. Dong et al. [54] first employed the **Convolutional Neural Network (CNN)** to predict the question type and then incorporated the question type semantics into the QG process. Du and Cardie [56] decomposed the answer-agnostic QG process into two steps, i.e., candidate answer extraction and answer-specific QG, and proposed a neural Seq2Seq model for better encoding coreference knowledge. Kumar et al. [109] presented an end-to-end generator-evaluator framework for answer-agnostic QG. The generator first identifies the potentially best pivotal answer spans in the input sentence and then generates the question through an RNN-based Seq2Seq model. The evaluator directly optimizes for conformity to the structure of ground-truth questions. Wang, Wei, Fan, Liu, and Huang [226] proposed a multi-agent communication framework, which implements question-worthy phrase extraction and QG using pointer network and RNN-based Seq2Seq model, respectively, and learns these two tasks simultaneously via the message passing mechanism.

Furthermore, research on **sequential question generation (SQG)** that is unaware of the corresponding answers makes some achievements. Many works regarded SQG as a conversational QG task and recurrently produced each question taking a context text and the current conversation history as the input. Therefore, SOG is potentially more challenging, since it requires a deep understanding of what has been asked so far and what

information should be asked for the next round to make a coherent conversation. Wang et al. [230] presented soft- and hard-typed decoders to generate questions in open-domain conversational systems by capturing different roles of different word types. Nakanishi et al. [152] proposed to first locate the focus of a question in the text passage and then identify the question pattern that leads the sequential generation of the words in a question. Pan et al. [161] first selected a text span from the passage as the rationale at each conversation turn and then incorporated a dynamic reasoning procedure to the RNN-based Seq2Seq model. Besides, Ling et al. [133] leveraged the conversational context to generate appropriate and informative questions in the setting of multi-turn open-domain dialogue systems.

There are also some works leveraging the Transformer architecture to solve answer-agnostic QG. Wang et al. [224] leveraged the standard encoder-decoder architecture, with the multi-head attention as the building block. Kumar et al. [108] presented a cross-lingual model that effectively exploits resources in a secondary language to improve QG for a primary language, implemented by a Transformer-based encoder-decoder architecture. Scialom et al. [195] leveraged Transformers to complement the base architecture with a copying mechanism, placeholders, and contextual word embeddings for answer-agnostic QG. Pan et al. [165] built a Chinese diverse question dataset from Baidu Zhidao and incorporated context information and control signal to Transformer-based Seq2Seq model for generating diverse questions from keywords. Laban et al. [115] fine-tuned a GPT2 language model [168], which is a Transformer based architecture, on the QG task using the SQuAD 2.0 dataset. Roemmele et al. [183] leveraged Transformer-based Seq2Seq model with a copy mechanism and designed different means of augmenting the training data. To augment original passages in the MS MARCO dataset for better retrieval performance, Nogueira et al. [157] fed as input the passage and trained the Transformer-based Seq2Seq model T5 [170] to generate questions. Similarly, Bhambhoria et al. [17] leveraged T5 and rule-based method (i.e., syntactic parser) to generate QA pairs for COVID-19 literature.

Among these works, $GE_{\text{ROUGE+QSS+ANSS}}$ [109] performs best in terms of BLEU, METEOR, and ROUGE_L on SQuAD dataset following the data split in Reference [57]. For more details, see Table 7 in Section 6.1.1.

4.2.2 Pre-trained Seq2Seq Models. Large-scale pre-trained language models have substantially advanced the state-of-the-art across various NLP tasks, which can be fine-tuned to adapt to downstream tasks. Among the existing pre-training methods, BERT [50] is the most prominent one and some works have leveraged BERT to improve the quality of generated questions [28]. However, BERT is specially designed for language understanding tasks and thus directly applying a BERT on natural language generation tasks is not feasible. Recently, as shown in Figure 7, many works attempt to propose different Seq2Seq pre-training objectives for natural language generation, which are evaluated on various downstream text generation tasks including QG.

For example, Dong et al. [53] presented a new **UNIfied pre-trained Language Model (UNILM)**, which employs different masks for self-attention to control the access to the context of the word token to be predicted. Xiao et al. [236] proposed an enhanced multi-flow Seq2Seq pre-training and fine-tuning framework (ERNIE-GEN), which incorporates an infilling generation mechanism and a noise-aware generation method to alleviate the exposure bias. Following the experimental QG setting of Reference [53], ERNIE-GEN achieved better results than UNILM_{LARGE} for answer-aware QG. Wang et al. [229] proposed a simple and effective knowledge distillation model, named MiniLM, to compress large pre-trained Transformer-based language models. Bao et al. [15] proposed a pseudo-masked language model to jointly pre-train both autoencoding and partially autoregressive LM, and performed evaluations on answer-aware QG. Yan et al. [239]

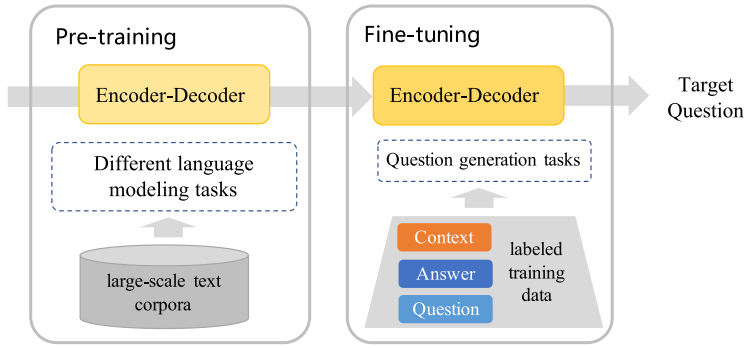

Fig. 7. Pre-trained Seq2Seq models.

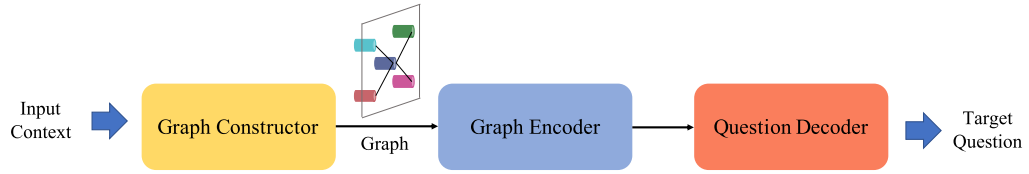

Fig. 8. Graph-based models.

proposed a pre-trained Seq2Seq model called ProphetNet, which learns to predict future n-gram based on previous context tokens at each timestep. They fine-tuned the ProphetNet on answer-aware QG task similar to References [53, 236].

Among these works, *ERNIE-GEN*$_{LARGE}$ [236] performs best in terms of BLEU4 and METEOR on SQuAD dataset following the data split in Reference [57]. Besides, *ERNIE-GEN*$_{LARGE}$ [236] performs best in terms of BLEU4 on SQuAD dataset following the data split in Zhou et al. [261]. *ProphetNet* [239] performs best in terms of METEOR and ROUGE$_L$ on SQuAD dataset following the data split in Reference [261]. For more details, see Tables 7 and 8 in Section 6.1.1.

4.2.3 Graph-based Models. Traditional Seq2Seq models only capture the surface linear structure of the context, which can not model the long-distance relationship between sentences. To address this issue, some recent studies began to focus on the graph-based neural models for QG, which are inspired by modeling highly structured objects (e.g., entity relationships and molecules) using graphs [100, 127]. These methods exploit the representational power of deep neural networks and the structural modeling ability of the relational sentence graphs, which can encode long-distance relationship between sentences. As shown in Figure 8, most of such methods first constructed a graph from the input context and then employed a graph-based model to effectively learn the graph embeddings from the constructed text graph.

For example, Liu et al. [135] designed a clue word predictor based on graph convolutional networks where the underlying intuition is that clue words will be more closely connected to the answer in dependency trees. Chen et al. [36] explored both syntax-based static and semantics-aware dynamic approaches to construct the directed passage graphs and then leveraged a novel bidirectional gated graph neural network to encode the passage graph. Liu et al. [136] used Graph Convolutional Networks onto a syntactic dependency tree representation of each passage to highlight the imperative aspects. Ma et al. [141] built an answer-centric entity graph from the document and leveraged relational graph convolutional network [194] to aggregate different granularity levels of semantic information. Pan et al. [163] proposed a novel framework that first constructs a

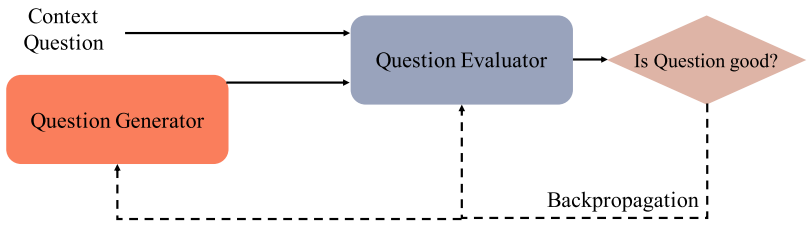

Fig. 9. Generative Adversarial Networks for QG.

semantic-level graph for the input document and then encodes the semantic graph by introducing an attention-based Gated Graph Neural Network.

Among these works, *multi-hop answer-focused reasoning model* [141] performs best in terms of BLEU 1-3 on HotpotQA dataset. For more details, see Table 10 in Section 6.2.

4.2.4 Generative Models. A generative model is a powerful way of learning the probability distribution that generated the data. Two of the widely adopted approaches in the QG tasks are are **Variational Autoencoder (VAE)** and **Generative Adversarial Networks (GAN)**. VAE aims at maximizing the lower bound of the data log-likelihood, while GAN aims at achieving an equilibrium between generator and discriminator. Here, we will introduce the working of VAE and GAN in the QG settings.

- **Variational Autoencoders.** The VAE [99, 180] is a popular latent variable generative model that simplifies learning by the introduction of a learned approximate posterior. To generate diverse questions, some recent works study to apply VAE and its variants [201, 238] to QG. In general, such works leveraged an encoder to map from an observed variable (i.e., question, and its corresponding answer and context) to a latent variable and then used a decoder to map from the latent variable to the observed variable. The model is usually trained by maximizing the variational lower bound and some auxiliary objectives.

  For example, Wang et al. [228] incorporated prior knowledge of question type into **conditional variational autoencoder (CVAE)** for controlling question generation with reasonable type. Specifically, the CVAE model incorporated with question type is trained by maximizing both the variational lower bound and question type prediction accuracy. Shinoda and Aizawa [198] proposed a variational QA pair generative model and introduced two independent latent random variables to model the two one-to-many problems in QG. Lee et al. [120] proposed a **hierarchical conditional variational autoencoder (HCVAE)** with two separate latent spaces for the question and the answer conditioned on the input text.
- **Generative Adversarial Networks.** GANs are composed of two components, a generator, which captures the real data distribution to generate fake samples and fool the discriminator, and a discriminator, which endeavors to distinguish the fake examples from real ones correctly. The training procedure of a GAN is a min-max game between the generator and the discriminator [119], which has been successfully applied in many tasks [41, 49, 79, 92, 138]. As shown in Figure 9, some recent works have also investigated adapting the generative adversarial networks directly to QG, where a generative model aims to generate questions and a discriminative model aims to evaluate the generated questions.

  For example, Bao et al. [16] designed a doubly adversarial net for QG, which involves two adversarial procedures between a question generator and two discriminators. One discriminator aims to help the generator learn domain-general representations of the input text, while another one provides more training data with estimated reward scores for generated

text-question pairs. Hosking and Riedel [87] adapted GAN to QG, where the discriminator aims to identify that a question was generated or came from the ground-truth. Similarly, Yao et al. [243] modified the discriminator in vanilla GAN, which not only evaluates the question authenticity, but distinguishes the types of questions, to generate diverse questions. Rao and Daumé [174] leveraged GAN for clarification question generation, where the generator is a Seq2Seq model, and the discriminator is a utility function to model the value of updating a context with the answer to a question.

5 TRAINING STRATEGIES OF NEURAL MODELS

Given the data available for training a neural QG model, an appropriate training strategy should be chosen. In this section, we briefly review a set of effective training strategies adopted by neural network-based QG models for comprehensive understanding, including maximum likelihood estimation, reinforcement learning, multi-task learning, and transfer learning.

5.1 Maximum Likelihood Estimation

The idea of **maximum likelihood estimation (MLE)** [94] is to find the model parameter values that make the observed data most likely. The negative of the log-likelihood function is often used, referred to generally as a **negative log-likelihood (NLL)** function [13, 57]. Specifically, given a training corpus $\mathcal{D}$ with a set of a context text T^i, optionally an answer A^i and a question Q^i, the MLE loss function is generally a probability over the training corpus with respect to all the parameters θ:

$$\mathcal{L}_{MLE} = -\sum_{i=1}^{|D|} \log p\left(Q^i \mid T^i, A^i; \theta\right) = -\sum_{i=1}^{|D|} \sum_{j=1}^{|Q^i|} \log p\left(Q^i_j \mid T^i, A^i, Q^i_{<j}; \theta\right). \tag{1}$$

MLE is the most commonly used learning objective in the above-introduced neural QG models due to: (1) The MLE loss function is computed based on each text-answer-question triple separately, which makes it simple and easy to scale; (2) the outputs of neural models learned with the MLE loss function often have real meanings and value in practice. However, most of standard MLE training would cause the test-time generation to be incrementally distorted because of the training-generation discrepancy [80]. Specifically, during training, the QG model is trained to predict the next word conditioned on prefix words sampled from the ground-truth data distribution. During test-time generation, the QG model generates words conditioned on prefix sequences sampled from the model itself. Due to the exposure to real data during training, the model could potentially be biased to only perform well with data prefixes, and the errors could accumulate along the generated sequence.

5.2 Reinforcement Learning

In the context of **reinforcement learning (RL)**, given the reward function R and the neural QG model G_θ, the objective is to maximize the expectation of reward $R(Q^i, Q^i_*)$, where Q^i_* is the ground-truth question and Q^i is the question generated by G_θ. The RL loss is usually designed as the negative expected reward for a generated question,

$$\mathcal{L}_{RL} = -\mathbb{E}_{\hat{Q}^i \sim G_\theta(\hat{Q}^i \mid T^i, A^i)}[R(\hat{Q}^i, Q^i)], \tag{2}$$

where the reward function $R(\hat{Q}^i, Q^i)$ is computed by comparing the generated question with corresponding ground-truth sequence based on some reward metrics.

RL-based QG models could be generally grouped into two categories: (1) first training with MLE and then fine-tuning the model using RL [109, 156, 161, 202, 248, 255] and (2) training with

the combination of RL and other objectives [36, 39, 74, 247]. Specifically, the major RL-based QG research focused on policy gradient RL algorithm.

The first category of research explored to pre-train the QG model with the MLE objective and then fine-tune with RL techniques to alleviate the exposure bias problem resulting from sequence learning with the MLE. Song et al. [202] first trained a QG model by optimizing the log-likelihood of the gold-standard output question and then fine-tuned the model with the policy gradient RL algorithm, where the reward is defined as the BLEU score [166]. Yuan et al. [248] fine-tuned the QG model pre-trained on MLE to directly optimize the combination of two individual rewards correlated to question quality, i.e., whether it can be answered correctly given the context document and whether it is expressed in suitable and grammatical English. Kumar et al. [109] proposed the task-specific rewards including three task-specific metrics and two new proposed reward functions designed to evaluate the conformity of the generated question and answer against the ground truth. To introduce human judgements on generated questions and provide feedback to QG, Pan et al. [161] used the quality of the answers predicted by the state-of-the-art RC model DrQA [29] as rewards and fine-tuned the model via RL. To solve the semantic drift problem in QG, Zhang and Bansal [255] proposed two semantics-enhanced rewards obtained from downstream question paraphrasing and QA tasks to regularize the QG model to generate semantically valid questions. Nema et al. [156] leveraged the fluency score and answerability score [154] as the reward to refine the initial question draft.

Beyond fine-tuning using RL, other works have explored how to train a QG model with the combination of RL and other learning objectives. Gupta et al. [74] focused on generating questions based on supporting facts in the context and combined the MLE and RL with a question-aware reward function to train the model. To consider the syntactic and semantic constraints, Chen et al. [36] used the combination of BLEU and word movers distance (WMD) [113] as the reward and fine-tuned the model by optimizing a mixed objective function combining both MLE loss and RL loss. Cho et al. [39] introduced the contrastive learning strategy for multi-document QG and defined the loss as the combination of the KL-divergence loss, MLE loss, and policy gradient loss with the reward designed to promote specificity of generated question.

A well-defined reward maximized via RL could provide the QG model with better information about how to distribute probability among sequences that do not appear in the training set [158, 248]. However, the widely used reward function (e.g., ROUGE [131] and BLEU [166]) are not quite suitable for evaluating the quality of the generated questions. Besides, RL practices in the QG field are likely to improve performance only when the pre-trained parameters are already close to yielding the correct generation. Therefore, better reward functions and policy gradient technologies should be explored in the future work.

5.3 Multi-task Learning

Multi-task learning (MTL) aims to optimize multiple related learning tasks at the same time. It has been successfully applied to many NLP tasks, including question answering [47], textual entailment [205], keyphrase generation [34], document summarization, [93] and machine translation [251]. Specifically, multi-task learning treats all the tasks equally, and the objective is to improve the generalization performance of all the tasks [258].

Recently, people have started to investigate how QG can contribute to other NLP tasks and vice versa. Existing works have employed multi-task learning for QG with an auxiliary task, such as language modeling [263], supporting fact prediction [75], summarization [73], cross-lingual QG [108], and question answering [190, 202, 213, 214, 227]. Zhou et al. [263] proposed to incorporate language modeling as an auxiliary task to help QG in a hierarchical multi-task learning structure, where language modeling and QG are treated as a low-level task and high-level task, respectively.

Gupta et al. [75] employed multi-task learning with the auxiliary task of answer-aware supporting fact prediction to guide the QG. Specifically, multiple relevant sentences from different paragraphs in a document that can answer a given question are regarded as the supporting facts. Guo et al. [73] proposed a soft, high-level (semantic) layer-specific multi-task learning framework of summarization, QG, and entailment generation task. Specifically, both the summarization and QG task require the ability to find the most salient information in the given passage and thus both the performance could be improved. Kumar et al. [108] proposed a cross-lingual QG model that exploits resources in a secondary language to improve QG performance for a primary language.

Furthermore, many works focus on jointly learning on QG and QA. Wang et al. [227] proposed to alternate the input data between QA and QG examples for the same RNN-based Seq2Seq model with attention. Tang et al. [213] leveraged the probabilistic correlation to guide the training process of both models, where the QA model judges whether the generated question of a QG model is relevant to the answer and the QG model provides the probability of generating a question given the answer. Song et al. [202] casted both the QG and QA tasks into one process by first matching the input passage against the query and then generating the output according to the matching results. Tang et al. [214] proposed to improve the QG model via incorporating additional QA-specific signal as the loss function and improve the QA model via adding artificially generated training instances from the QG model. Sachan and Xing [190] introduced self-training methods for jointly learning to answer and ask questions while leveraging unlabeled text along with labeled QA pairs for learning.

Multi-task learning has demonstrated the effectiveness in generating high-quality questions and efficiency in achieving different goals in less time. However, the evaluation scores are still lower than the state-of-the art results achieved by devoted QG models. The major reasons may be that the characteristic of the QG and jointly trained tasks are inconsistent. This may tend to bring noisy information when considering collaborative interaction among all the tasks. For example, QG aims to find the most question-worthy information in the context text, while summarization aims to search for the most salient information in the context. Thus, there is still room for future exploration of joint training between QG and other tasks.

5.4 Transfer Learning

Transfer learning (TL) techniques have already achieved significant success in many real-world applications [18, 52, 266], especially when we have sufficient data in one domain, but a similar domain of interest does not have enough data for learning purposes.

Rare labeled data of different domains may constrain the model training to produce more semantically similar questions in those area. Therefore, many works are proposed to leverage transfer learning techniques to adapt a well-performed QG model trained on one domain to another domain. As described in Section 4.2.2, fine-tuning the pre-trained Seq2Seq framework on the target QG dataset could be regarded as a transfer learning process. Such methods provide neural QG models with good initial weights learned from huge text corpus. Furthermore, some works [129, 265] proposed to train a neural model on source domain QG data and then utilized transfer learning techniques to further enhance the performance of the QG model on target domain QG data. Yang et al. [241] incorporated domain adaptation techniques in combination with generative models for semi-supervised learning. Yu et al. [247] proposed an iterative learning framework with adaptive instance transfer and augmentation to generate questions from reviews, where the major challenge is the lack of training instances.

Transfer learning is a key to ensure the breakthrough of neural models in different QG settings with small data. However, transfer learning only works if the initial and target problems are

Table 6. Overview of Traditional Seq2Seq Models for QG

Models	QG Task			Training Strategy
	Context	Answer	Question	
[70, 78, 125, 134, 140, 261, 262, 264]	Sentence	Answer Span	Standalone	MLE
[210, 259]	Paragraph	Answer Span	Standalone	MLE
[156, 255]	Paragraph	Answer Span	Standalone	RL
[71]	Paragraph	Answer Span	Sequential	MLE
[241]	Paragraph	Answer Span	Standalone	TL
[67]	Paragraph	Answer Span	Standalone	MTL
[74, 248]	Document	Answer Span	Standalone	RL
[98, 203, 231]	Paragraph	Abstractive Answer	Standalone	MLE
[35]	Paragraph	Abstractive Answer	Standalone	RL
[217]	Document	Abstractive Answer	Standalone	MLE
[26, 54, 57, 212, 226, 235]	Sentence	Without Answer	Standalone	MLE
[73, 214]	Sentence	Without Answer	Standalone	MTL
[55, 56, 58]	Paragraph	Without Answer	Standalone	MLE
[177]	Keyword	Without Answer	Standalone	MLE
[208]	Document	Without Answer	Standalone	MLE
[109]	Paragraph	Without Answer	Standalone	RL

similar enough. Therefore, it is important to exactly define which knowledge is appropriate to be transferred for QG.

Specifically, based on the above-mentioned training strategies, to better understand the task type and training strategy each individual paper uses, Table 6 summarizes some traditional Seq2Seq models (as described in Section 4.2.1), and the categories of different models refer to Figure 2.

6 MODEL COMPARISON

In this section, we mainly survey and analyze the empirical evaluation results of the previously introduced neural QG models for standalone question generation with answer span and abstractive answers and sequential question generation. Since there are few works on the public datasets with multiple choice questions and without annotated answers, we refer the readers to previous papers [30, 31, 147]. Note that sometimes it is difficult to compare published results across different papers, since small changes, e.g., different training/validation/testing split approaches and tokenization, can lead to significant differences. Therefore, we attempt to collect results from papers that contain comparisons across some models performed at a single site for fairness.

The evaluation methods for QG could be generally divided into two categories, i.e., human-centric evaluation metrics and automatic metrics (requiring no training). Here, we only show the automatic evaluation results for all the models. Specifically, the performance of QG is usually evaluated by the following three automatic metrics: (1) BLEU [166]: measures the average n-gram precision on a set of reference sentences with a penalty for overly short sentences. BLEU-n is the BLEU score that uses up to n-grams for counting co-occurrences. (2) Rouge$_L$ [131]: measures recall by how much the words in reference sentences appear in predictions using Longest Common Subsequence based statistics. (3) METEOR [48]: is a recall-oriented metric that calculates the similarity between generations and references by considering synonyms, stemming, and paraphrases. In general, automatic metrics can not correlate with human judgments of questions (e.g., appropriateness and coherence), since they only assess content selection. Hence, improved QG automatic metrics are required to be proposed in the future work.

6.1 Empirical Comparison on Standalone Question Generation with Answer Span

To better understand the performances of different neural QG models for generating standalone questions with answer spans, we show the published experimental results on three representative datasets, i.e., SQuAD, NewsQA, and HotpotQA.

6.1.1 Empirical Comparison on SQuAD. As shown in Tables 7 and 8, we give an overview of previous published results on the SQuAD dataset following the data split in Du et al. [57], which contains 70,484/10,570/11,877 (train/development/test) examples, and Zhou et al. [261], which contains 86,635/8965/8964 (train/development/test) examples, respectively. Existing models on the SQuAD dataset could be generally classified into two categories, i.e., without answer (ignoring the annotated answers) and answer span (using the annotated answers). We have included some well-known rule-based models as baselines. Based on the results, we have the following observations:

- The rule-based methods [51, 84], although simple, can already achieve reasonably good performance. The human-designed features/rules can be integrated into neural QG models to improve the generation performance.
- As described in Reference [57], nearly 30% of the questions in SQuAD rely on information beyond a single sentence. And, their experimental results showed that encoding the paragraph causes the performance to drop a little compared with only encoding the sentence. Later, many works explored how to consider paragraph-level context information (the subscript in Table 7 and 8 is P) or document-level context information (the subscript in Table 7 and 8 is D) to improve the performance of the QG system.
- The model proposed by Reference [134] achieves the best performance in terms of BLEU1, BLEU2, BLEU3, and ROUGE$_L$ on SQuAD following the data split in Reference [57]. The reason might be that it converts the one-to-many mapping problem into a one-to-one mapping problem, which makes the generation process more controllable, as well as improves the quality of generated questions.
- Recently proposed pre-trained Seq2Seq models [15, 53, 229, 236, 239] outperform the other models significantly in terms of BELU4, METEOR, and ROUGE$_L$. These results demonstrate that pre-trained models are beneficial for downstream QG task with the help of the representation extracted from the large unannotated corpora.
- Chan and Fan [28] achieves the best performance in terms of BLEU1, BLEU2, and BLEU3 on SQuAD following the data split in Reference [261]. Specifically, the authors introduced different neural architectures built on top of the pre-trained BERT language model for generating questions, which again demonstrates the effectiveness of the pre-trained models.

Table 7. Overview of Previously Published Results on SQuAD Dataset Following the Data Split in Du et al. [57]

		Models	BLEU1	BLEU2	BLEU3	BLEU4	METEOR	$ROUGE_L$
Without Answer	Seq2Seq	[57]$_S$ (2017)	43.09	25.96	17.50	12.28	16.62	39.75
		[73]$_S$ (2018)	-	-	-	13.80	17.50	40.10
		[195]$_S$ (2019)	43.33	26.27	18.32	13.23	-	40.22
		[109]$_S$ (2019d)	48.13	31.15	22.01	16.48	20.21	44.11
		[235]$_S$ (2020)	45.08	27.98	19.38	13.90	18.12	40.77
		[56]$_P$ (2018)	-	-	20.90	15.16	19.12	-
Answer Span	Seq2Seq	[259]$_S$ (2018)	43.47	28.23	20.40	15.32	19.29	43.91
		[78]$_S$ (2018)	-	-	-	19.98	22.26	48.23
		[125]$_S$ (2019)	45.66	30.21	21.82	16.27	20.36	44.35
		[28]$_S$ (2019)	48.29	33.12	24.78	19.14	22.89	47.07
		[202]$_P$ (2017)	-	-	-	13.98	18.77	42.72
		[259]$_P$ (2018)	45.07	29.58	21.60	16.38	20.25	44.48
		[231]$_P$ (2019)	32.65	22.14	15.86	12.03	20.25	32.36
		[98]$_P$ (2019)	-	-	-	16.20	19.92	43.96
		[255]$_P$ (2019)	-	-	-	18.37	22.65	46.68
		[28]$_P$ (2019)	49.73	34.60	26.13	20.33	23.88	48.23
		[67]$_P$ (2020)	-	-	-	23.72	26.07	52.23
		[134]$_P$ (2020)	**52.30**	**36.70**	**28.00**	22.05	25.11	**53.25**
		[217]$_D$ (2020)	45.13	30.44	23.40	17.09	21.25	45.81
	Pre-trained	[53]$_P$ (2019)	-	-	-	22.12	25.06	51.07
		[236]$_P$ (2020)	-	-	-	**25.40**	**26.92**	52.84
		[15]$_P$ (2020)	-	-	-	24.70	26.33	52.13
		[239]$_P$ (2020)	-	-	-	25.01	26.83	52.57
		[229]$_P$ (2020)	-	-	-	21.07	24.09	49.14
	VAE	[198]$_P$ (2020)	48.59	32.83	24.21	18.40	24.86	46.66

The results are cited from the original paper where the method is proposed. The subscripts denote the QG model takes the (D)ocument/(P)aragraph/(S)entence level input context text. The short dash symbols denote that there are no published results for the specific model on the specific dataset in the original paper.

6.1.2 Empirical Comparison on NewsQA. Table 9 shows an overview of previous published results on the NewsQA dataset. Specifically, the representative QG models on NewsQA are answer-aware. Based on the reported results, in general, we observe that the overall performance on NewsQA is worse than that on SQuAD. The major reason might be that the average answer length of NewsQA is larger than that of SQuAD, and long answers usually bring more key information needs and are more difficult to generate questions. Moreover, ground-truth questions in NewQA tend to have less strict grammars and more diverse phrasings [135].

6.1.3 Empirical Comparison on HotpotQA. Table 10 shows an overview of previous published results on the HotpotQA dataset. Different from the SQuAD and NewsQA dataset with simple questions involving single-hop relations, the HotpotQA dataset contains complex and semantically relevant questions from multiple documents through multi-hop reasoning. Therefore, it is more challenging than the existing single-hop QG task. Recently, multi-hop QG has received increasing attention, since it has broad real-world applications in future intelligent systems. For example, in educational system, these questions require higher-order cognitive-skills that are crucial for evaluating a student's knowledge and stimulating self-learning. The published experimental results

Table 8. Overview of Previously Published Results on SQuAD Dataset Following the Data Split in Zhou et al. [261]

		Models	BLEU1	BLEU2	BLEU3	BLEU4	METEOR	ROUGE$_L$
Rule-based	Syntax	[84]$_S$ (2010b)	28.77	17.81	12.64	9.47	-	-
	Semantic	[51]$_S$ (2020)	45.55	30.24	23.84	18.72	-	-
Answer Span	Seq2Seq	[259]$_S$ (2018)	44.51	29.07	21.06	15.82	19.67	44.24
		[264]$_S$ (2019b)	43.11	29.13	21.39	16.31	-	-
		[125]$_S$ (2019)	44.40	29.48	21.54	16.37	20.68	44.73
		[156]$_S$ (2019)	47.27	31.88	23.65	18.16	23.40	47.14
		[28]$_S$ (2019)	50.71	35.44	26.95	21.20	24.02	48.68
		[140]$_S$ (2020a)	44.71	29.89	21.77	16.32	20.84	44.79
		[202]$_P$ (2017)	-	-	-	13.91	-	-
		[210]$_P$ (2018)	43.02	28.14	20.51	15.64	-	-
		[259]$_P$ (2018)	45.69	30.25	22.16	16.85	20.62	44.99
		[261]$_P$ (2018b)	-	-	-	13.29	-	-
		[203]$_P$ (2018)	-	-	-	13.91	-	-
		[98]$_P$ (2019)	-	-	-	16.17	-	-
		[156]$_P$ (2019)	46.41	30.66	22.42	16.99	21.10	45.03
		[28]$_P$ (2019)	**51.54**	**36.45**	**27.96**	22.17	24.80	49.68
		[224]$_P$ (2020b)	48.26	29.23	22.37	16.42	18.95	43.07
		[217]$_D$ (2020)	46.60	31.94	23.44	17.76	21.56	45.89
	Pre-trained	[53]$_P$ (2019)	-	-	-	23.75	25.61	52.04
		[236]$_P$ (2020)	-	-	-	**26.95**	27.57	53.77
		[15]$_P$ (2020)	-	-	-	26.30	27.09	53.19
		[239]$_P$ (2020)	-	-	-	26.72	**27.64**	**53.79**
		[229]$_P$ (2020)	-	-	-	23.27	25.15	50.60
	Graph	[36]$_P$ (2020)	-	-	-	18.30	21.70	45.98
		[135]$_P$ (2019a)	46.58	30.90	22.82	17.55	21.24	44.53
	Adversarial	[16]$_S$ (2018)	33.14	16.66	9.52	5.58	-	-

The results are cited from the original paper where the method is proposed. The subscripts denote the QG model takes the (D)ocument/(P)aragraph/(S)entence level input context text. The short dash symbols denote that there are no published results for the specific model on the specific dataset in the original paper.

Table 9. Overview of Previously Published Results on NewsQA Dataset

		Models	BLEU1	BLEU2	BLEU3	BLEU4	METEOR	ROUGE$_L$
Rule-based	Syntax	[199] (2011)*	16.90	7.94	4.72	3.08	23.78	3.74
Answer Span	Seq2seq	[261] (2018b)*	40.33	22.47	14.83	9.94	**42.25**	16.72
		[203] (2018)*	35.70	17.16	9.64	5.65	39.85	14.13
		[217] (2020)	**42.54**	**26.14**	**17.30**	**12.36**	19.04	**44.05**
	Graph	[135] (2019a)	40.45	23.52	15.68	11.06	17.43	43.16

The superscript * denotes that the results are cited from Reference [135], and others are cited from the original paper where the method is proposed.

have shown successful results for multi-hop QG, and it is necessary to advance the generation of such deep questions for considering how human intelligence embodies the skills of curiosity and integration. Specifically, we find that the graph-based method [141] performs better than Seq2Seq models in terms of BLEU 1–3. The reason might be that it leverages different granularity levels

Table 10. Overview of Previously Published Results on HotpotQA Dataset

		Models	BLEU1	BLEU2	BLEU3	BLEU4	METEOR	ROUGE$_L$
Answer Span	Seq2Seq	[261] (2018b)[1]	44.55	33.18	26.57	21.99	24.35	41.08
		[259] (2018)[2]	42.58	30.91	24.61	20.39	20.36	35.31
		[98] (2019)[2]	39.08	29.06	23.45	19.66	22.84	36.98
		[156] (2019)	45.45	33.13	26.05	21.17	25.81	43.12
		[74] (2020a)	46.80	34.94	28.21	23.57	22.88	39.68
		[140] (2020a)[1]	46.95	35.76	29.02	24.34	24.30	42.32
		[67] (2020)	-	-	-	**28.53**	**28.36**	**48.78**
	Graph	[141] (2020b)	**50.93**	**38.93**	**31.78**	26.70	25.40	43.88
		[135] (2019a)[3]	31.18	22.55	17.69	14.36	25.20	40.94
		[163] (2020a)	40.55	27.21	20.13	15.53	20.15	36.94

The superscript 1–3 denotes that the results are cited from References [141], [74], and [163], respectively, and others are cited from the original paper where the method is proposed.

Table 11. Overview of Previously Published Results on MS MARCO Dataset

		Model	BLEU1	BLEU2	BLEU3	BLEU4	METEOR	ROUGE$_L$
Answer Span	Seq2Seq	[261] (2018b)[2]	46.62	32.67	22.98	16.13	20.22	46.35
		[210] (2018)	48.24	35.95	25.79	19.45	-	-
		[259]$_S$ (2018)	-	-	-	16.02	-	-
		[259]$_P$ (2018)	-	-	-	17.24	-	-
		[214] (2018)	-	-	-	9.89	-	-
		[264] (2019b)	55.67	38.16	28.12	21.59	-	-
		[217] (2020)	41.43	29.97	23.01	18.25	42.77	19.43
		[140] (2020a)	50.33	37.10	27.23	20.46	24.69	49.89
Without Answer	Seq2Seq	[57] (2017)[1]	-	-	-	10.46	-	-
		[58] (2017)	-	-	-	11.46	-	-
		[212] (2017a)	-	-	-	9.31	-	-

The subscripts denote the QG task takes the (P)aragraph/(S)entence level context text. The superscript 1–2 denotes that the results are cited from References [58] and [140], respectively, and others are cited from the original paper where the method is proposed.

in the grounded answer-centric entity graph, producing precise and enriched semantics for the decoder.

6.2 Empirical Comparison on Standalone Question Generation with Abstractive Answer

To better understand the performance of different neural QG models, as shown in Table 11, we survey the previously published results on the MS MARCO dataset for generating standalone questions with abstractive answers. However, most works [140, 210, 214, 217, 259, 261, 264] extracted a subset of MS MARCO where the answers are sub-spans within the passages and leveraged MS MARCO to learn a QG model for generating factoid questions instead of non-factoid questions. Some works [57, 58, 212] directly generated questions from the input sentences based on the assumption that the sentences contain the correct answer span. Therefore, it is unreasonable to compare published results across these papers, which could only provide a rough comprehension of existing works. In the future work, more efforts should be put in the generation of questions that are associated with abstractive answers summarizing the said information in the passages.

Table 12. Overview of Previously Published Results on CoQA Dataset

		Model	BLEU1	BLEU2	BLEU3	BLEU4	METEOR	ROUGE$_L$
Answer Span	Seq2Seq	[161] (2019b)	-	-	-	**19.69**	-	34.05
		[71] (2019b)	**37.38**	**22.81**	**16.25**	-	-	**46.90**
		[25] (2020)	35.70	19.64	12.06	-	**17.26**	38.15
Without Answer	Seq2Seq	[57] (2017)[1]	-	-	-	13.97	-	31.75
		[56] (2018)[2]	35.56	21.14	14.84	-	-	45.58
		[152] (2019)	27.62	13.67	0.13	0.04	-	-

The superscript 1–2 denotes that the results are cited from References [161] and [71], and others are cited from the original paper where the method is proposed.

Enabling a QG system with the ability to ask questions towards abstractive answers will help us build curious machines that can interact with humans in a better manner. We will discuss such ability in the Trending Topics Section in detail.

6.3 Empirical Comparison on Sequential Question Generation with Abstractive Answer

Here, we show the published experimental results for sequential question generation on the representative CoQA dataset. Due to the frequently occurred information omission and coreference between questions, sequential question generation is usually more challenging than standalone question generation. Existing sequential question generation models mainly focused on modeling complex context dependencies and frequently occurred coreference between questions. Specifically, abstractive answers in the CoQA are mainly small modifications to spans occurring in the context. Therefore, most works leveraged the region in the passage that overlaps with the rationales for actual answers or ignored the answer information. As shown in Table 12, Gao et al. [71] achieved the best performance that generates sequential questions via coreference alignment and conversation flow modeling. To further push forward this research, it would be necessary to (1) Devise better evaluation metrics to properly reflect the conversational natures; and (2) generate different questions according to the responder's answers instead of the pre-defined answers for real applications.

6.4 Human-centric Evaluations

Human evaluations are typically viewed as the most important form of evaluation [23]. On one hand, the ultimate goal of QG systems is to generate questions that are valuable to people. On the other hand, automatic metrics still fall short of replicating human decisions. Therefore, many QG works include some form of human evaluation on the generated questions.

Among the various human evaluation methodologies, eliciting quality judgments is most common. Human evaluators are asked to assess the quality of a question based on criteria such as question's grammaticality/fluency and coherence/consistency. For example, Du et al. [57] considered two modalities: naturalness, which indicates the grammaticality and fluency, and difficulty, which measures the sentence-question syntactic divergence and the reasoning needed to answer the question. Hosking and Riedel [87] rated the questions from 1 (worst) to 5 (best) on two separate criteria, i.e., the fluency of the language used and the relevance of the question to the context document and answer. Further, Chen et al. [36] used three categories to rate the questions, i.e., whether the questions are syntactically correct, semantically correct, and relevant to the passage.

While human evaluations give the best insight into how well a QG model performs, it still poses several challenges. First, human evaluations are usually expensive and time-consuming, especially

for questions with high-cognitive level questions that require extensive domain expertise. Then, there is also a lack of consistency among different labelers, resulting in the unreliability of evaluation results. For more details of the evaluation methodologies in question generation, we refer readers to the related survey [9]. Therefore, more research efforts should focus on the automatic and human evaluation metrics that can better measure the model quality from different perspectives.

7 TRENDING TOPICS

In this section, we discuss several trending topics related to QG. Some of these topics are important but have not been well addressed in this field, while some are very promising directions for future research.

7.1 Diverse Question Generation

The ability of exhibiting certain personality with diverse traits is essential for practical QG systems to interact well with users in a more natural and coherent way. Although there have been some attempts on the diversity problem in many text generation tasks, e.g., conversation [124, 197, 237, 253, 254] and summarization [45, 64, 76, 121, 137, 155], few works consider these special characteristics for QG.

Diverse QG could be formulated as a 1-to-n mapping problem, which aims to produce diverse questions with different focuses for the same context input. The major challenge in diverse QG is to identify different question-worthy context words dynamically and actively control the question generation with respect to the topic or other factors. Traditional QG methods mainly investigate how to generate one question based on a given context input, i.e., 1-to-1 mapping problem. As a result, they fail to generate multiple diverse questions or control the generation of the questions, which can not well meet users' needs and improve users' satisfaction.

Despite still having a long way to go, some researchers have explored how to generate diverse questions. Based on the standard RNN-based Seq2Seq framework, some works enable to control the generation of numerous diverse questions from a given context text via the consideration of the question type [235], specificity [22], relevant topic [27, 90], clue [134], difficulty [107], and answer relevant relations [78, 125]. Recently, Krishna and Iyyer [105] proposed to use QG as a pedagogic way of representing documents, which aims to control the specificity of generated hierarchies of QA pairs. Pan et al. [165] incorporated the context information and control signal into the Transformer-based Seq2Seq model to produce diverse questions from a given fixed set of keywords. As described in Section 4.2.4, there have been some works employing the VAE framework to produce more diverse, yet still authentic-looking, questions [120, 198, 228]. For the evaluation of diversity in QG, Schlichtkrull and Cheng [193] extended existing commonly used evaluation metrics into F1-like scoring functions. Furthermore, to evaluate the importance of diversity in QG for QA, Sultan et al. [209] fine-tuned a pre-trained transformer-based masked language model for QG. The results showed that producing diverse yet accurate questions indeed yields better QA results than previous approaches of generating only one question.

In summary, with the emergence of interactive search system and educational system, diverse question generation would be an indispensable technology in these scenarios. In some sense, diverse question generation simulates human behavior, as people often ask various questions depending on their own purpose (which might be affected by a variety of underlying factors such as their current mood, knowledge state, and so on). However, existing works mainly considered the effect of the input text and the target answer and ignored the subjective factor of human, e.g., his/her emotion or knowledge state, which is a major factor that affects the QG in practice. More research work is expected in this direction in the short future.

7.2 Pre-training Tailored for Question Generation

Recent advances have shown that language representation models pre-trained on large-scale corpora, such as OpenAI GPT [167], BERT [50], and XLNET [240], can well capture rich semantic information and be fine-tuned to achieve state-of-the-art performance in many language-understanding tasks, e.g., sentiment classification [200], question answering [207], and named entity recognition [192]. Recently, some researchers have attempted to fine-tune the pre-trained models, e.g., BERT [28, 36, 219] and GPT [39, 101, 102], for improving the QG performance. However, directly applying such pre-trained language models (e.g., BERT and GPT) on the QG task is not appropriate, since they are generally designed for language understanding instead of language generation. Therefore, it is necessary and important to design pre-training objectives tailored for QG.

As described in Section 4.2.2, some pre-training objectives for the language generation tasks have been proposed, which are usually based on the encoder-decoder learning framework. Recently, Narayan et al. [153] have collected large-scale English QA pairs from community QA resources to pre-train a Transformer-based Seq2Seq framework for text generation. Despite the exciting performance of pre-training objectives for language generation, however, pre-training objectives tailored for QG have not been well explored. QG task typically handles context texts with different lengths, answers with different granularities, and questions with different types. It requires not only understanding the diverse content of a context, an answer and a question, but also modeling the relevance relationship between the three.

When we look at those existing Seq2Seq-based pre-training objectives from the QG perspective, we may find that: (1) Seq2Seq-based pre-training tasks could in general contribute to build good contextual representations for the text, the answer, and the question; (2) most Seq2Seq-based pre-training objectives aim to reconstruct a sentence fragment given the remaining part of the sentence, which are quite diverged from the QG requirement not just due to the input difference (sentence-pair vs. context-answer-question triples) but also the relation type (coherence between two sentences vs. relevance among contexts, answers, and questions). It is generally hypothesized that using a pre-training objective that more closely resembles the downstream task leads to better fine-tuning performance [252]. In this sense, we argue that the power of pre-training has not been fully exploited for QG. Pre-training objectives tailored for QG is a very challenging and promising direction for researchers.

7.3 Question Generation with Higher Cognitive Levels

Asking questions is a cognitively demanding process, where questions are expected to require different levels of cognitive efforts to answer. As described in Section 3.1.2, a typical framework that attempts to categorize the cognitive levels involved in question asking comes from Bloom's taxonomy.

Overall, there are two main cognitive levels of questions, i.e., high-cognitive and low-cognitive level questions. High-cognitive level questions are those requiring complex applying, analyzing, evaluating, or creating, while low-cognitive level questions are those at reading, understanding, and lower level applying. In many applications, especially for educational purposes, it is necessary to generate questions from low-level recalling factual details to high-level re-organizing a coherent new whole. Existing QG research has typically focused on generating low-cognitive level questions relevant to one fact obtainable from a text. Unsurprising, the higher-cognitive questions have the greatest educational value [11, 146], which require deep thinking and reasoning over multiple pieces of information of the context text.

High-cognitive level questions are harder to ask than factoid-oriented questions [186]. However, little effort has been made to comprehend aspects of questioning, resulting in questions that are

shallow and can not well reflect the mechanism behind human question asking. Early works have attempted to generate higher cognitive questions by building a semantic representation of the arbitrary text [160, 244] to reason over concepts in the text. Labutov et al. [117] first represented the original text in a low-dimensional ontology, then solicited high-level candidate question templates through crowdsourcing, and finally retrieved potentially relevant templates for a novel region of text. Recently, Pan et al. [164] incorporated semantic graphs to enhance the input document representations and performed jointly training of content selection and question generation.

In summary, asking higher cognitive level questions is of great value concerning how intelligence reflects the depth and breadth of human knowledge and will have broad applications in future intelligent systems. More research efforts are needed to select relevant information and model the reasoning chains in the question generation with higher cognitive level.

7.4 Question Generation for Information Seeking

Information seeking systems such as conversational search and recommendation have grown in popularity in recent years, which aim to satisfy users' complex information needs through multi-turn interactions. However, it is often the case that users fail to express their information need adequately in a single query. Consequently, they may have to scan multiple result pages or reformulate their queries, which requires much effort and is often unsuccessful. Beyond the "user asks, system responds" paradigm, an alternative solution is that systems can clarify the user information intents by actively asking a question. This could assist users to refine their information need and then increase the chance of retrieving a satisfactory result.

Up to now, there has been much effort to automatically generate good clarifying questions for conversational search. First, analyzing user interaction with clarifying questions could help researchers better understand the search clarification. To achieve this, some works analyze the effect of clarifying questions from different perspectives. For example, Radlinski and Craswell [169] raised the importance of asking for clarification in conversational search. Braslavski et al. [20] studied the clarification questions asked by users and analyzed their behavior and the types of clarification questions asked. The user study done by Reference [97] showed that there is no penalty in user satisfaction when the system asks for clarification. Reference [104] analyzed the effect of clarifying questions on the quality of ranking in conversational search and highlighted the importance of effectively understanding and incorporating explicit conversational feedback. Zamani et al. [250] conducted a comprehensive study about user engagements with clarifying questions and presentation bias in user interactions with the clarification panes. Tavakoli [215] defined a novel taxonomy to investigate clarifying questions and further discovered the patterns and types of such clarifying questions.

Besides, many researchers proposed different ways to ask clarifying questions for conversational search. Earlier in the TREC HARD Track [8], participants could ask clarifying questions through clarification forms. Later, Coden et al. [43] studied clarifying questions for entity disambiguation. However, the format was mostly restricted to a "did you mean A or B?," which makes it non-practical for real-world scenarios. Recently, Aliannejadi et al. [7] proposed an offline evaluation methodology and collected a dataset through crowdsourcing to study the task of selecting and asking clarifying questions for open-domain information seeking. Rao and Daumé [174] developed an adversarial training approach for generating clarification questions, while Rao and Daumé III [175] proposed a clarification question selection model based on the expected value of perfect information after obtaining the answer to the clarification question. Aliannejadi et al. [6] presented a shared task where the challenge includes when to ask clarifying questions in multi-turn interactions and how to generate them. Zamani et al. [249] proposed clarifying question generation models trained via weak supervision for open-domain search queries. Most recently, to balance

the risk of answering user's query and asking clarifying questions, Wang and Ai [232] proposed a risk-aware conversational search agent model trained using reinforcement learning approach.

Furthermore, much work has been done on interacting with users for recommendation by asking a clarifying question. Christakopoulou et al. [42] developed a conversational restaurant recommender that interacts with users to collect more detailed users' preferences by asking questions. Sun and Zhang [211] utilized a semi-structured user query with facet-value pairs to represent a conversation history and built a deep RL-based conversational recommender system. Similarly, some works [32, 179] also trained a policy agent for conversational recommendation. Zhang et al. [257] proposed a multi-memory network to ask aspect-based questions in the right order for better performance of e-commerce recommendation. Recently, Ren et al. [178] proposed a knowledge-based question generation method to generate personalized clarifying questions for conversational recommendation.

Overall, existing works on clarifying question generation have shown significant promise in information seeking. However, beyond these promising results, there is still a long way to go for clarifying question in information seeking. First, the clarifying question generation data for open-domain search queries is typically limited. More research efforts are needed to construct standard benchmark datasets and evaluation tasks to facilitate research and rigorous comparison. Then, how to leverage clarifying question to resolve a user's information need more quickly and efficiently has been relatively unstudied and evaluated. Besides, most works generated questions based on the current and previous states of a conversation, which ignores the future possible states, i.e., the long-term utility. Finally, how to cooperatively learn the result retrieval model with clarifying questions and the question generation model with user responses is a promising direction in information-seeking research.

8 CONCLUSION

Asking questions about a text plays a vital role for both the growth of human beings and the improvement of AI systems. In this survey, we summarized the current research status on question generation from natural language text and gain some insights for future development. We introduced a more comprehensive taxonomy of the QG tasks, in terms of the types of the input context text, the input answer, and the target question. We reviewed existing models from different dimensions under model architecture and model learning. For model architecture analysis, we reviewed existing models to understand their underlying assumptions and major design principles, including how to treat the input context texts, how to consider the answer features, and how to model the relevance relations. We find that the pre-trained language generation models are beneficial for QG tasks, and it is necessary to further explore the pre-training objectives tailored for QG. Besides, more efforts should be put in generating higher-cognitive questions using the abstractive answers as the input or without the answers as the input. For model learning analysis, we reviewed the major training strategies adopted for neural QG methods. To better understand the current status of QG models on major application, we surveyed published empirical results on the representative benchmark tasks to conduct a comprehensive comparison. In addition, we discussed several trending topics that are important or might be promising in the future. We hope this survey can help researchers who are interested in this direction and will motivate new ideas by looking at past attempts and achieve significant breakthroughs in this domain in the near future.

REFERENCES

[1] Naveed Afzal and Ruslan Mitkov. 2014. Automatic generation of multiple choice questions using dependency-based semantic relations. *Soft Comput.* 18 (07 2014), 1269–1281.

[2] Manish Agarwal, Rakshit Shah, and Prashanth Mannem. 2011. Automatic question generation using discourse cues. In *Proceedings of the 6th Workshop on Innovative Use of NLP for Building Educational Applications*. Association for Computational Linguistics, 1–9.
[3] Renlong Ai, Sebastian Krause, Walter Kasper, Feiyu Xu, and Hans Uszkoreit. 2015. Semi-automatic generation of multiple-choice tests from mentions of semantic relations. In *Proceedings of the 2nd Workshop on Natural Language Processing Techniques for Educational Applications*. Association for Computational Linguistics, 26–33.
[4] Husam Ali, Yllias Chali, and Sadid A. Hasan. 2010. Automation of question generation from sentences. In *Proceedings of the Third Workshop on Question Generation*. 58–67.
[5] Husam Ali, Yllias Chali, and Sadid A. Hasan. 2011. Automation of question generation from sentences. In *Proceedings of the Actes de la 17e Conference sur le Traitement Automatique des Langues Naturelles Articles courts*. 213–218.
[6] Mohammad Aliannejadi, Julia Kiseleva, Aleksandr Chuklin, Jeff Dalton, and Mikhail Burtsev. 2020. ConvAI3: Generating clarifying questions for open-domain dialogue systems (ClariQ). *arXiv preprint arXiv:2009.11352* (2020).
[7] 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*. 475–484.
[8] James Allan. 2005. *HARD Track Overview in TREC 2003 High Accuracy Retrieval from Documents*. Technical Report. Massachusetts University Amherst Center For Intelligent Information Retrieval.
[9] Jacopo Amidei, P. Piwek, and A. Willis. 2018. Evaluation methodologies in automatic question generation 2013–2018. In *INLG*.
[10] L. Anderson, D. Krathwohl, and B. Bloom. 2000. A taxonomy for learning, teaching, and assessing: A revision of bloom's taxonomy of educational objectives Longman.
[11] R. Anderson and W. B. Biddle. 1975. On asking people questions about what they are reading. *Psychol. Learn. Motiv.* 9 (1975), 89–132.
[12] J. Araki, Dheeraj Rajagopal, S. Sankaranarayanan, Susan Holm, Yukari Yamakawa, and T. Mitamura. 2016. Generating questions and multiple-choice answers using semantic analysis of texts. In *COLING*.
[13] Tina Baghaee. 2017. *Automatic Neural Question Generation using Community-based Question Answering Systems*. Ph.D. Dissertation. University of Lethbridge, Deptartment of Mathematics, Lethbridge, Alberta, Canada.
[14] Dzmitry Bahdanau, Kyunghyun Cho, and Yoshua Bengio. 2015. Neural machine translation by jointly learning to align and translate. In *ICLR*.
[15] Hangbo Bao, Li Dong, Furu Wei, Wenhui Wang, Nan Yang, X. Liu, Yu Wang, Songhao Piao, Jianfeng Gao, M. Zhou, and H. Hon. 2020. UniLMv2: Pseudo-masked language models for unified language model pre-training. *ArXiv* abs/2002.12804 (2020).
[16] Junwei Bao, Yeyun Gong, N. Duan, M. Zhou, and T. Zhao. 2018. Question generation with doubly adversarial nets. *IEEE/ACM Trans. Audio, Speech Lang. Process.* 26 (2018), 2230–2239.
[17] Rohan Bhambhoria, Luna Feng, Dawn Sepehr, John Chen, Conner Cowling, Sedef Kocak, and Elham Dolatabadi. 2020. A smart system to generate and validate question answer pairs for COVID-19 literature. In *Proceedings of the 1st Workshop on Scholarly Document Processing*. 20–30.
[18] Parminder Bhatia, Kristjan Arumae, and B. Celikkaya. 2020. Dynamic transfer learning for named entity recognition. In *Proceedings of the International Workshop on Health Intelligence*. Springer, 69–81.
[19] K. Boyer and P. Piwek. 2010. In *Proceedings of the 3rd Workshop on Question Generation*. questiongeneration.org.
[20] Pavel Braslavski, Denis Savenkov, Eugene Agichtein, and Alina Dubatovka. 2017. What do you mean exactly? Analyzing clarification questions in CQA. In *Proceedings of the Conference on Human Information Interaction and Retrieval*. 345–348.
[21] Daniel Fernando Campos, T. Nguyen, M. Rosenberg, Xia Song, Jianfeng Gao, Saurabh Tiwary, Rangan Majumder, L. Deng, and Bhaskar Mitra. 2016. MS MARCO: A human-generated machine reading comprehension dataset. *ArXiv* abs/1611.09268 (2016).
[22] Yang Trista Cao, Sudha Rao, and Hal Daumé III. 2019. Controlling the specificity of clarification question generation. In *Proceedings of the Workshop on Widening NLP*. Association for Computational Linguistics, 53–56.
[23] A. Celikyilmaz, E. Clark, and J. Gao. 2020. Evaluation of text generation: A survey. *arXiv preprint arXiv:2006.14799*.
[24] D. Ch and S. Saha. 2020. Automatic multiple choice question generation from text: A survey. *IEEE Trans. Learn. Technol.* 13 (2020), 14–25.
[25] Zi Chai and Xiaojun Wan. 2020. Learning to ask more: Semi-autoregressive sequential question generation under dual-graph interaction. In *Proceedings of the 58th Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 225–237. DOI: https://doi.org/10.18653/v1/2020.acl-main.21
[26] Yllias Chali and Tina Baghaee. 2018. Automatic opinion question generation. In *Proceedings of the 11th International Conference on Natural Language Generation*. Association for Computational Linguistics, 152–158. DOI: https://doi.org/10.18653/v1/W18-6518

[27] Yllias Chali and Sadid A. Hasan. 2015. Towards topic-to-question generation. *Comput. Ling.* 41 (2015), 1–20.
[28] Ying-Hong Chan and Yao-Chung Fan. 2019. A recurrent BERT-based model for question generation. In *MRQA@EMNLP*.
[29] Danqi Chen, A. Fisch, J. Weston, and Antoine Bordes. 2017. Reading Wikipedia to answer open-domain questions. *ArXiv* abs/1704.00051 (2017).
[30] Guanliang Chen, Jie Yang, and Dragan Gasevic. 2019. A comparative study on question-worthy sentence selection strategies for educational question generation. In *Artificial Intelligence in Education*, Seiji Isotani, Eva Millán, Amy Ogan, Peter Hastings, Bruce McLaren, and Rose Luckin (Eds.). Springer International Publishing, Cham, 59–70.
[31] Guanliang Chen, Jie Yang, C. Hauff, and G. Houben. 2018. LearningQ: A large-scale dataset for educational question generation. In *ICWSM*.
[32] Tong Chen, Hongzhi Yin, Guanhua Ye, Zi Huang, Yang Wang, and Meng Wang. 2020. Try this instead: Personalized and interpretable substitute recommendation. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 891–900.
[33] Wei Chen and Gregory Aist. 2009. Generating questions automatically from informational text. In *Proceedings of AIED Workshop on Question Generation*. 17–24.
[34] Wang Chen, Hou Pong Chan, Piji Li, Lidong Bing, and Irwin King. 2019. An integrated approach for keyphrase generation via exploring the power of retrieval and extraction. In *NAACL-HLT*.
[35] Yu Chen, Lingfei Wu, and Mohammed J. Zaki. 2019. Natural question generation with reinforcement learning based graph-to-sequence model. *CoRR* abs/1910.08832 (2019).
[36] Yu Chen, Lingfei Wu, and Mohammed J. Zaki. 2020. Reinforcement learning based graph-to-sequence model for natural question generation. *ArXiv* abs/1908.04942 (2020).
[37] 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 Conference on Empirical Methods in Natural Language Processing*.
[38] W. S. Cho, Y. Zhang, Sudha Rao, Chris Brockett, and S. Lee. 2019. Generating a common question from multiple documents using multi-source encoder-decoder models. *ArXiv* abs/1910.11483 (2019).
[39] W. S. Cho, Y. Zhang, Sudha Rao, A. Çelikyilmaz, Chenyan Xiong, Jianfeng Gao, M. Wang, and B. Dolan. 2019. Contrastive multi-document question generation. *arXiv preprint arXiv:1911.03047*.
[40] Eunsol Choi, H. He, Mohit Iyyer, Mark Yatskar, W. Yih, Yejin Choi, Percy Liang, and L. Zettlemoyer. 2018. QuAC : Question answering in context. *ArXiv* abs/1808.07036 (2018).
[41] Yunjey Choi, Min-Je Choi, Munyoung Kim, Jung-Woo Ha, S. Kim, and J. Choo. 2018. StarGAN: Unified generative adversarial networks for multi-domain image-to-image translation. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 8789–8797.
[42] Konstantina Christakopoulou, Filip Radlinski, and Katja Hofmann. 2016. Towards conversational recommender systems. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*. 815–824.
[43] Anni Coden, Daniel Gruhl, Neal Lewis, and Pablo N. Mendes. 2015. Did you mean A or B? Supporting clarification dialog for entity disambiguation. In *SumPre-HSWI@ ESWC*.
[44] P. Connor-Greene. 2000. Assessing and promoting student learning: Blurring the line between teaching and testing. *Teach. Psychol.* 27 (2000), 84–88.
[45] P. Costa and Ivandré Paraboni. 2019. Personality-dependent neural text summarization. In *RANLP*.
[46] Sérgio Curto, Ana Cristina Mendes, and Luísa Coheur. 2011. Exploring linguistically-rich patterns for question generation. In *Proceedings of the UCNLG+Eval: Language Generation and Evaluation Workshop (UCNLG+EVAL'11)*. Association for Computational Linguistics, 33–38.
[47] Yang Deng, Yuexiang Xie, Y. Li, Min Yang, Nan Du, W. Fan, Kai Lei, and Ying Shen. 2019. Multi-task learning with multi-view attention for answer selection and knowledge base question answering. *ArXiv* abs/1812.02354 (2019).
[48] Michael J. Denkowski and A. Lavie. 2014. Meteor universal: Language specific translation evaluation for any target language. In *WMT@ACL*.
[49] Emily L. Denton, Soumith Chintala, Arthur Szlam, and R. Fergus. 2015. Deep generative image models using a laplacian pyramid of adversarial networks. In *NIPS*.
[50] Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2018. BERT: Pre-training of deep bidirectional transformers for language understanding. *arXiv preprint arXiv:1810.04805* (2018).
[51] Kaustubh Dhole and Christopher D. Manning. 2020. Syn-QG: Syntactic and shallow semantic rules for question generation. In *Proceedings of the 58th Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 752–765.
[52] Chuong B. Do and A. Ng. 2005. Transfer learning for text classification. In *NIPS*.

[53] Li Dong, Nan Yang, Wenhui Wang, Furu Wei, Xiaodong Liu, Yu Wang, Jianfeng Gao, Ming Zhou, and Hsiao-Wuen Hon. 2019. Unified language model pre-training for natural language understanding and generation. In *Advances in Neural Information Processing Systems 32*, H. Wallach, H. Larochelle, A. Beygelzimer, F. d'Alché-Buc, E. Fox, and R. Garnett (Eds.). Curran Associates, Inc., 13063–13075. Retrieved from http://papers.nips.cc/paper/9464-unified-language-model-pre-training-for-natural-language-understanding-and-generation.pdf.

[54] Xiaozheng Dong, Yu Hong, Xin Chen, Weikang Li, Min Zhang, and Qiaoming Zhu. 2018. Neural question generation with semantics of question type. In *Natural Language Processing and Chinese Computing*, Min Zhang, Vincent Ng, Dongyan Zhao, Sujian Li, and Hongying Zan (Eds.). Springer International Publishing, Cham, 213–223.

[55] Xinya Du and Claire Cardie. 2017. Identifying where to focus in reading comprehension for neural question generation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 2067–2073. DOI: https://doi.org/10.18653/v1/D17-1219

[56] X. Du and Claire Cardie. 2018. Harvesting paragraph-level question-answer pairs from Wikipedia. In *ACL*.

[57] Xinya Du, Junru Shao, and Claire Cardie. 2017. Learning to ask: Neural question generation for reading comprehension. In *Proceedings of the 55th Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, 1342–1352.

[58] Nan Duan, Duyu Tang, Peng Chen, and Ming Zhou. 2017. Question generation for question answering. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 866–874. DOI: https://doi.org/10.18653/v1/D17-1090

[59] M. Dunn, Levent Sagun, M. Higgins, V. U. Güney, Volkan Cirik, and K. Cho. 2017. SearchQA: A new Q&A dataset augmented with context from a search engine. *ArXiv* abs/1704.05179 (2017).

[60] Esin Durmus, He He, and Mona Diab. 2020. FEQA: A question answering evaluation framework for faithfulness assessment in abstractive summarization. *arXiv preprint arXiv:2005.03754* (2020).

[61] Ibrahim Eldesoky. 2015. Semantic question generation using artificial immunity. *I.J. Mod. Educ. Comput. Sci.* 7 (01 2015), 1–8.

[62] Hady ElSahar, C. Gravier, and F. Laforest. 2018. Zero-shot question generation from knowledge graphs for unseen predicates and entity types. *ArXiv* abs/1802.06842 (2018).

[63] A. R. Fabbri, Patrick Ng, Zhiguo Wang, Ramesh Nallapati, and B. Xiang. 2020. Template-based question generation from retrieved sentences for improved unsupervised question answering. In *ACL*.

[64] Angela Fan, David Grangier, and M. Auli. 2018. Controllable abstractive summarization. In *NMT@ACL*.

[65] Zhihao Fan, Zhongyu Wei, Piji Li, Yanyan Lan, and X. Huang. 2018. A question type driven framework to diversify visual question generation. In *IJCAI*.

[66] Zhihao Fan, Zhongyu Wei, S. Wang, Y. Liu, and X. Huang. 2018. A reinforcement learning framework for natural question generation using bi-discriminators. In *COLING*.

[67] Yuwei Fang, Shuohang Wang, Zhe Gan, S. Sun, and Jing jing Liu. 2020. Accelerating real-time question answering via question generation. *ArXiv* abs/2009.05167 (2020).

[68] M. Flor and Brian Riordan. 2018. A semantic role-based approach to open-domain automatic question generation. In *BEA@NAACL-HLT*.

[69] Michael Flor and Brian Riordan. 2018. A semantic role-based approach to open-domain automatic question generation. In *Proceedings of the 13th Workshop on Innovative Use of NLP for Building Educational Applications*. Association for Computational Linguistics, 254–263.

[70] Yifan Gao, Lidong Bing, Wang Chen, Michael Lyu, and Irwin King. 2019. Difficulty controllable generation of reading comprehension questions. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence*. International Joint Conferences on Artificial Intelligence Organization, 4968–4974. DOI: https://doi.org/10.24963/ijcai.2019/690

[71] 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 Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 4853–4862. DOI: https://doi.org/10.18653/v1/P19-1480

[72] Donna Gates. 2008. *Automatically Generating Reading Comprehension Look-back Strategy: Questions from Expository Texts*. Technical Report. Carnegie-Mellon Univ Pittsburgh Pa School of Computer Science.

[73] Han Guo, Ramakanth Pasunuru, and Mohit Bansal. 2018. Soft layer-specific multi-task summarization with entailment and question generation. In *ACL*.

[74] Deepak Gupta, Hardik Chauhan, Asif Ekbal, and Pushpak Bhattacharyya. 2020. Reinforced multi-task approach for multi-hop question generation. *CoRR* abs/2004.02143 (2020).

[75] D. Gupta, H. Chauhan, Asif Ekbal, and P. Bhattacharyya. 2020. Reinforced multi-task approach for multi-hop question generation. *ArXiv* abs/2004.02143 (2020).

[76] Xu-Wang Han, Hai-Tao Zheng, J. Chen, and Cong-Zhi Zhao. 2019. Diverse decoding for abstractive document summarization. *Appl. Sci.* 9 (2019), 386.

[77] Sanda Harabagiu, Andrew Hickl, John Lehmann, and Dan Moldovan. 2005. Experiments with interactive question-answering. In *Proceedings of the 43rd Meeting of the Association for Computational Linguistics (ACL'05)*. 205–214.
[78] V. Harrison and M. Walker. 2018. Neural generation of diverse questions using answer focus, contextual and linguistic features. In *INLG*.
[79] Amartya Hatua, T. T. Nguyen, and A. Sung. 2019. Dialogue generation using self-attention generative adversarial network. In *Proceedings of the IEEE International Conference on Conversational Data & Knowledge Engineering (CDKE'19)*. 33–38.
[80] T. He, J. Zhang, Z. Zhou, and J. Glass. 2021. Quantifying exposure bias for neural language generation. *arXiv preprint arXiv:1905.10617* (2021).
[81] Wei He, Kai Liu, Jing Liu, Yajuan Lyu, Shiqi Zhao, X. Xiao, Yulong Liu, Yizhong Wang, H. Wu, Qiaoqiao She, Xuan Liu, T. Wu, and H. Wang. 2018. DuReader: A Chinese machine reading comprehension dataset from real-world applications. *ArXiv* abs/1711.05073 (2018).
[82] Michael Heilman and Noah A. Smith. 2009. *Question Generation via Overgenerating Transformations and Ranking*. Technical Report. Carnegie-Mellon Univ Pittsburgh pa language technologies insT.
[83] Michael Heilman and Noah A. Smith. 2010. Extracting simplified statements for factual question generation.
[84] Michael Heilman and Noah A. Smith. 2010. Good question! Statistical ranking for question generation. In *HLT-NAACL*.
[85] Felix Hill, Antoine Bordes, S. Chopra, and J. Weston. 2016. The goldilocks principle: Reading children's books with explicit memory representations. *CoRR* abs/1511.02301 (2016).
[86] Sepp Hochreiter and Jürgen Schmidhuber. 1997. Long short-term memory. *Neur. Comput.* 9, 8 (1997), 1735–1780.
[87] T. Hosking and S. Riedel. 2019. Evaluating rewards for question generation models. In *NAACL-HLT*.
[88] Wenpeng Hu, Bing Liu, Jinwen Ma, Dongyan Zhao, and Rui Yan. 2018. Aspect-based question generation. In *Proceedings of the 6th International Conference on Learning Representations (ICLR'18)*. OpenReview.net. Retrieved from https://openreview.net/forum?id=rkRR1ynIf.
[89] Wenpeng Hu, B. Liu, Jinwen Ma, Dongyan Zhao, and R. Yan. 2018. Workshop track-ICLR 2018 Aspect-based question generation.
[90] Wenpeng Hu, B. Liu, R. Yan, Dongyan Zhao, and Jinwen Ma. 2018. Topic-based question generation. In *ICLR*.
[91] Yan Huang and L. He. 2016. Automatic generation of short answer questions for reading comprehension assessment. *Nat. Lang. Eng.* 22 (2016), 457–489.
[92] Phillip Isola, Jun-Yan Zhu, Tinghui Zhou, and A. Efros. 2017. Image-to-image translation with conditional adversarial networks. *2017 IEEE Conference on Computer Vision and Pattern Recognition (CVPR)* (2017), 5967–5976.
[93] Masaru Isonuma, Toru Fujino, Junichiro Mori, Y. Matsuo, and I. Sakata. 2017. Extractive summarization using multi-task learning with document classification. In *EMNLP*.
[94] S. Johansen and K. Juselius. 2005. Maximum likelihood estimation and inference on cointegration—with applications to the demand for money. *Oxford Bull. Econ. Statist.* 52 (2005), 169–210.
[95] Mandar Joshi, Eunsol Choi, Daniel S. Weld, and Luke Zettlemoyer. 2017. TriviaQA: A large scale distantly supervised challenge dataset for reading comprehension. In *ACL*.
[96] Mitesh M. Khapra, Dinesh Raghu, S. Joshi, and Sathish Reddy. 2017. Generating natural language question-answer pairs from a knowledge graph using an RNN-based question generation model. In *EACL*.
[97] Johannes Kiesel, Arefeh Bahrami, Benno Stein, Avishek Anand, and Matthias Hagen. 2018. Toward voice query clarification. In *Proceedings of the International ACM SIGIR Conference on Research & Development in Information Retrieval*. 1257–1260.
[98] Yanghoon Kim, Hwanhee Lee, Joongbo Shin, and K. Jung. 2019. Improving neural question generation using answer separation. *ArXiv* abs/1809.02393 (2019).
[99] Diederik P. Kingma and M. Welling. 2014. Auto-encoding variational Bayes. *CoRR* abs/1312.6114 (2014).
[100] Thomas Kipf and M. Welling. 2017. Semi-supervised classification with graph convolutional networks. *ArXiv* abs/1609.02907 (2017).
[101] T. Klein and M. Nabi. 2019. Learning to answer by learning to ask: Getting the best of GPT-2 and BERT worlds. *ArXiv* abs/1911.02365 (2019).
[102] Wei-Jen Ko, Te-Yuan Chen, Yiyan Huang, Greg Durrett, and Junyi Jessy Li. 2020. Inquisitive question generation for high level text comprehension. *ArXiv* abs/2010.01657 (2020).
[103] Tomás Kociský, Jonathan Schwarz, P. Blunsom, Chris Dyer, K. Hermann, Gábor Melis, and E. Grefenstette. 2018. The narrative QA reading comprehension challenge. *Trans. Assoc. Comput. Ling.* 6 (2018), 317–328.
[104] 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 ACM SIGIR on International Conference on Theory of Information Retrieval*. 129–132.

[105] Kalpesh Krishna and Mohit Iyyer. 2019. Generating question-answer hierarchies. *arXiv preprint arXiv:1906.02622* (2019).
[106] V. Kumar, Yuncheng Hua, G. Ramakrishnan, G. Qi, L. Gao, and Yuan-Fang Li. 2019. Difficulty-controllable multi-hop question generation from knowledge graphs. In *SEMWEB*.
[107] Vishwajeet Kumar, Yuncheng Hua, Ganesh Ramakrishnan, Guilin Qi, Lianli Gao, and Yuan-Fang Li. 2019. Difficulty-controllable multi-hop question generation from knowledge graphs. In *International Semantic Web Conference*. Springer, 382–398.
[108] Vishwajeet Kumar, Nitish Joshi, Arijit Mukherjee, Ganesh Ramakrishnan, and Preethi Jyothi. 2019. Cross-lingual training for automatic question generation. In *Proceedings of the 57th Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 4863–4872. DOI: https://doi.org/10.18653/v1/P19-1481
[109] V. Kumar, G. Ramakrishnan, and Yuan-Fang Li. 2019. Putting the horse before the cart: A generator-evaluator framework for question generation from text. In *CoNLL*.
[110] Hidenobu Kunichika, Tomoki Katayama, Tsukasa Hirashima, and Akira Takeuchi. 2004. Automated question generation methods for intelligent English learning systems and its evaluation. In *Proceedings of the ICCE*.
[111] Ghader Kurdi, Jared Leo, Bijan Parsia, Uli Sattler, and Salam Al-Emari. 2019. A systematic review of automatic question generation for educational purposes. *Int. J. Artif. Intell. Educ.* 30 (2019), 121–204.
[112] Ghader Kurdi, Jared Leo, Bijan Parsia, Uli Sattler, and Salam Al-Emari. 2020. A systematic review of automatic question generation for educational purposes. *Int. J. Artif. Intell. Educ.* 30, 1 (2020), 121–204.
[113] Matt J. Kusner, Yu Sun, Nicholas I. Kolkin, and Kilian Q. Weinberger. 2015. From word embeddings to document distances. In *ICML*.
[114] T. Kwiatkowski, J. Palomaki, Olivia Redfield, Michael Collins, Ankur P. Parikh, C. Alberti, D. Epstein, Illia Polosukhin, J. Devlin, Kenton Lee, Kristina Toutanova, Llion Jones, Matthew Kelcey, Ming-Wei Chang, Andrew M. Dai, Jakob Uszkoreit, Q. Le, and Slav Petrov. 2019. Natural questions: A benchmark for question answering research. *Trans. Assoc. Comput. Ling.* 7 (2019), 453–466.
[115] Philippe Laban, John Canny, and Marti A. Hearst. 2020. What's the latest? A question-driven news chatbot. In *Proceedings of the 58th Meeting of the Association for Computational Linguistics: System Demonstrations*. 380–387.
[116] Igor Labutov, Sumit Basu, and Lucy Vanderwende. 2015. Deep questions without deep understanding. In *Proceedings of the 53rd Meeting of the Association for Computational Linguistics and the 7th International Joint Conference on Natural Language Processing (Volume 1: Long Papers)*. Association for Computational Linguistics, 889–898.
[117] I. Labutov, Sumit Basu, and Lucy Vanderwende. 2015. Deep questions without deep understanding. In *ACL*.
[118] Guokun Lai, Qizhe Xie, Hanxiao Liu, Y. Yang, and E. Hovy. 2017. RACE: Large-scale reading comprehension dataset from examinations. In *EMNLP*.
[119] C. Ledig, L. Theis, Ferenc Huszár, J. Caballero, Andrew Aitken, Alykhan Tejani, J. Totz, Zehan Wang, and W. Shi. 2017. Photo-realistic single image super-resolution using a generative adversarial network. In *Proceedings of the IEEE Conference on Computer Vision and Pattern Recognition (CVPR'*17). 105–114.
[120] D. Lee, Seanie Lee, Woo Tae Jeong, Donghwan Kim, and Sung Ju Hwang. 2020. Generating diverse and consistent QA pairs from contexts with information-maximizing hierarchical conditional VAEs. In *ACL*.
[121] Wen-Yu Lee, Y. Kuo, Peng-Ju Hsieh, Wen-Feng Cheng, Ting-Hsuan Chao, Hui-Lan Hsieh, Chieh-En Tsai, Hsiao-Ching Chang, Jia-Shin Lan, and W. Hsu. 2015. Unsupervised latent aspect discovery for diverse event summarization. In *MM'15*.
[122] Roger Levy and Galen Andrew. 2006. Tregex and Tsurgeon: Tools for querying and manipulating tree data structures. In *Proceedings of the 5th International Conference on Language Resources and Evaluation (LREC'06)*. European Language Resources Association (ELRA).
[123] Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Ves Stoyanov, and Luke Zettlemoyer. 2019. Bart: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461* (2019).
[124] J. Li, M. Galley, Chris Brockett, Jianfeng Gao, and W. Dolan. 2016. A diversity-promoting objective function for neural conversation models. In *HLT-NAACL*.
[125] Jingjing Li, Yifan Gao, Lidong Bing, Irwin King, and Michael R. Lyu. 2019. Improving question generation with to the point context. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*. Association for Computational Linguistics, 3216–3226. DOI: https://doi.org/10.18653/v1/D19-1317
[126] J. Li, Alexander H. Miller, S. Chopra, Marc'Aurelio Ranzato, and J. Weston. 2017. Learning through dialogue interactions by asking questions. In *ICLR*.
[127] Qimai Li, Zhichao Han, and Xiao-Ming Wu. 2018. Deeper insights into graph convolutional networks for semi-supervised learning. *ArXiv* abs/1801.07606 (2018).

[128] Yikang Li, N. Duan, B. Zhou, X. Chu, Wanli Ouyang, and X. Wang. 2018. Visual question generation as dual task of visual question answering. In *Proceedings of the IEEE/CVF Conference on Computer Vision and Pattern Recognition*. 6116–6124.

[129] Yin-Hsiang Liao and Jia-Ling Koh. 2020. Question generation through transfer learning. In *IEA/AIE*.

[130] Chenghua Lin, Dong Liu, Wei Pang, and Edward Apeh. 2015. Automatically predicting quiz difficulty level using similarity measures. In *Proceedings of the 8th International Conference on Knowledge Capture (K-CAP'15)*. Association for Computing Machinery.

[131] Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *ACL 2004*.

[132] David Lindberg, Fred Popowich, John Nesbit, and Phil Winne. 2013. Generating natural language questions to support learning on-line. In *Proceedings of the 14th European Workshop on Natural Language Generation*. Association for Computational Linguistics, 105–114.

[133] 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, Taipei, Taiwan, April 20–24, 2020*, Yennun Huang, Irwin King, Tie-Yan Liu, and Maarten van Steen (Eds.). ACM / IW3C2, 2486–2492. DOI: https://doi.org/10.1145/3366423.3379996

[134] B. Liu, Haojie Wei, Di Niu, Haolan Chen, and Yancheng He. 2020. Asking questions the human way: Scalable question-answer generation from text corpus. In *Proceedings of the Web Conference*.

[135] Bang Liu, Mingjun Zhao, Di Niu, Kunfeng Lai, Yancheng He, Haojie Wei, and Yu Xu. 2019. Learning to generate questions by learning what not to generate. *CoRR* abs/1902.10418 (2019).

[136] B. Liu, Mingjun Zhao, Di Niu, K. Lai, Yancheng He, Haojie Wei, and Y. Xu 2019. Learning to generate questions by learning what not to generate. In *Proceedings of the World Wide Web Conference*. 1106–1118.

[137] Dayiheng Liu, Yeyun Gong, Jie Fu, Wei Liu, Yu Yan, B. Shao, Daxin Jiang, J. Lv, and N. Duan. 2020. Diverse, controllable, and keyphrase-aware: A corpus and method for news multi-headline generation. *ArXiv* abs/2004.03875 (2020).

[138] L. Liu, Yao Lu, Min Yang, Q. Qu, Jia Zhu, and H. Li. 2018. Generative adversarial network for abstractive text summarization. *ArXiv* abs/1711.09357 (2018).

[139] T. Liu, Bingzhen Wei, B. Chang, and Z. Sui. 2017. Large-scale simple question generation by template-based seq2seq learning. In *NLPCC*.

[140] Xiyao Ma, Qile Zhu, Yanlin Zhou, and Xiaolin Li. 2020. Improving question generation with sentence-level semantic matching and answer position inferring. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence, the 32nd Innovative Applications of Artificial Intelligence Conference, the 10th AAAI Symposium on Educational Advances in Artificial Intelligence*. AAAI Press, 8464–8471. Retrieved from https://aaai.org/ojs/index.php/AAAI/article/view/6366.

[141] Xiyao Ma, Qile Zhu, Y. Zhou, Xiaolin Li, and Dapeng Wu. 2020. Asking complex questions with multi-hop answer-focused reasoning. *ArXiv* abs/2009.07402 (2020).

[142] Mukta Majumder and Sujan Kumar Saha. 2015. A system for generating multiple choice questions: With a novel approach for sentence selection. In *Proceedings of the 2nd Workshop on Natural Language Processing Techniques for Educational Applications*. Association for Computational Linguistics, 64–72.

[143] Prashanth Mannem, Rashmi Prasad, and Aravind Joshi. 2010. Question generation from paragraphs at U Penn: QGSTEC system description. In *Proceedings of the QG2010: The Third Workshop on Question Generation*. 84–91.

[144] Karen Mazidi and Rodney D. Nielsen. 2015. Leveraging multiple views of text for automatic question generation. In *Artificial Intelligence in Education*, Cristina Conati, Neil Heffernan, Antonija Mitrovic, and M. Felisa Verdejo (Eds.). Springer International Publishing, Cham, 257–266.

[145] Karen Mazidi and Paul Tarau. 2016. Infusing NLU into automatic question generation. In *INLG 2016 - Proceedings of the Ninth International Natural Language Generation Conference, September 5-8, 2016, Edinburgh, UK*, Amy Isard, Verena Rieser, and Dimitra Gkatzia (Eds.). The Association for Computer Linguistics, 51–60.

[146] J. Mcmillan. 2005. Secondary teachers' classroom assessment and grading practices. *Educ. Measur.: Iss. Pract.* 20 (2005), 20–32.

[147] R. Mitkov. 2003. Computer-aided generation of multiple-choice tests. In *Proceedings of the International Conference on Natural Language Processing and Knowledge Engineering*.

[148] R. Mitkov, L. Ha, and N. Karamanis. 2006. A computer-aided environment for generating multiple-choice test items. *Nat. Lang. Eng.* 12 (2006), 177–194.

[149] Ruslan Mitkov and Le An Ha. 2003. Computer-aided generation of multiple-choice tests. In *Proceedings of the HLT-NAACL 03 Workshop on Building Educational Applications Using Natural Language Processing*. 17–22.

[150] N. Mostafazadeh, I. Misra, J. Devlin, M. Mitchell, X. He, and L. Vanderwende. 2016. Generating natural questions about an image. *ArXiv* abs/1603.06059 (2016).

[151] Jack Mostow and Wei Chen. 2009. *Generating Instruction Automatically for the Reading Strategy of Self-Questioning*. IOS Press, NLD, 465–472.

[152] 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*. Association for Computational Linguistics, 63–71. DOI: https://doi.org/10.18653/v1/D19-5809

[153] S. Narayan, Gonçalo Simães, Ji Ma, Hannah Craighead, and R. McDonald. 2020. QURIOUS: Question generation pretraining for text generation. *ArXiv* abs/2004.11026 (2020).

[154] Preksha Nema and Mitesh M. Khapra. 2018. Towards a better metric for evaluating question generation systems. *ArXiv* abs/1808.10192 (2018).

[155] Preksha Nema, Mitesh M. Khapra, Anirban Laha, and B. Ravindran. 2017. Diversity driven attention model for query-based abstractive summarization. In *ACL*.

[156] Preksha Nema, Akash Kumar Mohankumar, Mitesh M. Khapra, Balaji Vasan Srinivasan, and Balaraman Ravindran. 2019. Let's ask again: Refine network for automatic question generation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP'19)*. Association for Computational Linguistics, 3314–3323. DOI: https://doi.org/10.18653/v1/D19-1326

[157] Rodrigo Nogueira, Jimmy Lin, and AI Epistemic. 2019. From doc2query to docTTTTTquery. *Online Preprint* (2019).

[158] Mohammad Norouzi, S. Bengio, Z. Chen, Navdeep Jaitly, Mike Schuster, Y. Wu, and Dale Schuurmans. 2016. Reward augmented maximum likelihood for neural structured prediction. In *NIPS*.

[159] L. Odilinye, F. Popowich, E. Zhang, J. Nesbit, and P. H. Winne. 2015. Aligning automatically generated questions to instructor goals and learner behaviour. In *Proceedings of the IEEE 9th International Conference on Semantic Computing (IEEE ICSC'15)*. 216–223.

[160] A. Olney, A. Graesser, and N. Person. 2012. Question generation from concept maps. *Dialog. Disc.* 3 (2012), 75–99.

[161] Boyuan Pan, Hao Li, Ziyu Yao, Deng Cai, and Huan Sun. 2019. Reinforced dynamic reasoning for conversational question generation. In *Proceedings of the 57th Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 2114–2124. DOI: https://doi.org/10.18653/v1/P19-1203

[162] Liangming Pan, Wenqiang Lei, Tat-Seng Chua, and Min-Yen Kan. 2019. Recent advances in neural question generation. *ArXiv* abs/1905.08949 (2019).

[163] 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 Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 1463–1475. DOI: https://doi.org/10.18653/v1/2020.acl-main.135

[164] Liangming Pan, Yuxi Xie, Yansong Feng, Tat-Seng Chua, and Min-Yen Kan. 2020. Semantic graphs for generating deep questions. In *ACL*.

[165] Youcheng Pan, Baotian Hu, Qingcai Chen, Yang Xiang, and X. Wang. 2020. Learning to generate diverse questions from keywords. In *Proceedings of the IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP'20)*. 8224–8228.

[166] Kishore Papineni, S. Roukos, T. Ward, and Wei-Jing Zhu. 2002. Bleu: A method for automatic evaluation of machine translation. In *ACL*.

[167] Alec Radford, Karthik Narasimhan, Tim Salimans, and Ilya Sutskever. 2018. Improving language understanding by generative pre-training. (2018).

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

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

[170] Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J. Liu. 2019. Exploring the limits of transfer learning with a unified text-to-text transformer. *arXiv preprint arXiv:1910.10683* (2019).

[171] Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for SQuAD. *ArXiv* abs/1806.03822 (2018).

[172] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100, 000+ questions for machine comprehension of text. *ArXiv* abs/1606.05250 (2016).

[173] Sheetal Rakangor and Dr. Y. R. Ghodasara. 2015. Literature review of automatic question generation systems. *International Journal of Scientific and Research Publications* 5, 1 (2015), 1–5.

[174] Sudha Rao and Hal Daumé. 2019. Answer-based adversarial training for generating clarification questions. In *NAACL-HLT*.

[175] Sudha Rao and Hal Daumé III. 2018. Learning to ask good questions: Ranking clarification questions using neural expected value of perfect information. *arXiv preprint arXiv:1805.04655* (2018).

[176] Siva Reddy, Danqi Chen, and Christopher D. Manning. 2019. CoQA: A conversational question answering challenge. *Trans. Assoc. Comput. Ling.* 7 (2019), 249–266.

[177] Sathish Reddy, Dinesh Raghu, Mitesh M. Khapra, and Sachindra Joshi. 2017. Generating natural language question-answer pairs from a knowledge graph using an RNN-based question generation model. In *Proceedings of the 15th Conference of the European Chapter of the Association for Computational Linguistics: Volume 1, Long Papers*. Association for Computational Linguistics. 376–385. Retrieved from https://www.aclweb.org/anthology/E17-1036.

[178] Xuhui Ren, Hongzhi Yin, Tong Chen, Hao Wang, Zi Huang, and Kai Zheng. 2021. Learning to ask appropriate questions in conversational recommendation. *arXiv preprint arXiv:2105.04774* (2021).

[179] Xuhui Ren, Hongzhi Yin, Tong Chen, Hao Wang, Nguyen Quoc Viet Hung, Zi Huang, and Xiangliang Zhang. 2020. CRSAL: Conversational recommender systems with adversarial learning. *ACM Trans. Inf. Syst.* 38, 4 (2020), 1–40.

[180] D. Rezende, S. Mohamed, and Daan Wierstra. 2014. Stochastic backpropagation and approximate inference in deep generative models. In *ICML*.

[181] M. Richardson, C. Burges, and Erin Renshaw. 2013. MCTest: A challenge dataset for the open-domain machine comprehension of text. In *EMNLP*.

[182] H. Rodrigues, Luísa Coheur, and Eric Nyberg. 2016. QGASP: A framework for question generation based on different levels of linguistic information. In *INLG*.

[183] Melissa Roemmele, Deep Sidhpura, Steve DeNeefe, and Ling Tsou. 2021. AnswerQuest: A system for generating question-answer items from multi-paragraph documents. *arXiv preprint arXiv:2103.03820* (2021).

[184] Michael Roth and K. Woodsend. 2014. Composition of word representations improves semantic role labelling. In *EMNLP*.

[185] Vasile Rus and Graesser Art. 2009. The question generation task and evaluation challenge. In *Proceedings of the Workshop Report*.

[186] V. Rus, Z. Cai, and A. Graesser. 2007. Experiments on generating questions about facts. In *CICLing*.

[187] V. Rus and J. Lester. 2009. The 2nd workshop on question generation. In *AIED*.

[188] Vasile Rus, Brendan Wyse, Paul Piwek, Mihai Lintean, Svetlana Stoyanchev, and Cristian Moldovan. 2010. The first question generation shared task evaluation challenge. In *Proceedings of the 6th International Natural Language Generation Conference*. 251–257.

[189] V. Rus, B. Wyse, P. Piwek, Mihai C. Lintean, Svetlana Stoyanchev, and C. Moldovan. 2012. A detailed account of the first question generation shared task evaluation challenge. *Dialog. Disc.* 3 (2012), 177–204.

[190] Mrinmaya Sachan and E. Xing. 2018. Self-training for jointly learning to ask and answer questions. In *NAACL-HLT*.

[191] Claude Sammut and Ranan B. Banerji. 1986. Learning concepts by asking questions. *Mach. Learn.: Artif. Intell. Appr.* 2 (1986), 167–192.

[192] Erik F. Sang and Fien De Meulder. 2003. Introduction to the CoNLL-2003 shared task: Language-independent named entity recognition. *arXiv preprint cs/0306050* (2003).

[193] M. Schlichtkrull and Weiwei Cheng. 2020. Evaluating for diversity in question generation over text. *ArXiv* abs/2008.07291 (2020).

[194] M. Schlichtkrull, Thomas Kipf, P. Bloem, R. V. Berg, Ivan Titov, and M. Welling. 2018. Modeling relational data with graph convolutional networks. In *ESWC*.

[195] Thomas Scialom, Benjamin Piwowarski, and Jacopo Staiano. 2019. Self-attention architectures for answer-agnostic neural question generation. In *Proceedings of the 57th Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 6027–6032. DOI: https://doi.org/10.18653/v1/P19-1604

[196] I. Serban, Alberto García-Durán, Çaglar Gülçehre, Sungjin Ahn, A. Chandar, A. Courville, and Y. Bengio. 2016. Generating factoid questions with recurrent neural networks: The 30M factoid question-answer corpus. *ArXiv* abs/1603.06807 (2016).

[197] 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*.

[198] Kazutoshi Shinoda and Akiko Aizawa. 2020. Variational question-answer pair generation for machine reading comprehension. *ArXiv* abs/2004.03238 (2020).

[199] Michael Heilman. 2011. *Automatic Factual Question Generation from Text*. Carnegie Mellon University.

[200] Richard Socher, Alex Perelygin, Jean Wu, Jason Chuang, Christopher D. Manning, Andrew Y. Ng, and Christopher Potts. 2013. Recursive deep models for semantic compositionality over a sentiment treebank. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. 1631–1642.

[201] Kihyuk Sohn, H. Lee, and Xinchen Yan. 2015. Learning structured output representation using deep conditional generative models. In *NIPS*.

[202] Linfeng Song, Zhiguo Wang, and Wael Hamza. 2017. A unified query-based generative model for question generation and question answering. *CoRR* abs/1709.01058 (2017).

[203] Linfeng Song, Z. Wang, W. Hamza, Yue Zhang, and Daniel Gildea. 2018. Leveraging context information for natural question generation. In *NAACL-HLT*.

[204] Linfeng Song and Lin Zhao. 2016. Domain-specific question generation from a knowledge base. *ArXiv* abs/1610.03807 (2016).
[205] Asa Cooper Stickland and I. Murray. 2019. BERT and PALs: Projected attention layers for efficient adaptation in multi-task learning. In *ICML*.
[206] Janell Straach and Klaus Truemper. 1999. Learning to ask relevant questions. *Artif. Intell.* 111, 1 (1999), 301–327.
[207] Dan Su, Yan Xu, Genta Indra Winata, Peng Xu, H. Kim, Zihan Liu, and Pascale Fung. 2019. Generalizing question answering system with pre-trained language model fine-tuning. In *MRQA@EMNLP*.
[208] Sandeep Subramanian, Tong Wang, Xingdi Yuan, and Adam Trischler. 2017. Neural models for key phrase detection and question generation. *CoRR* abs/1706.04560 (2017).
[209] Md Arafat Sultan, Shubham Chandel, Ramón Fernández Astudillo, and V. Castelli. 2020. On the importance of diversity in question generation for QA. In *ACL*.
[210] 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 Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 3930–3939. DOI: https://doi.org/10.18653/v1/D18-1427
[211] Yueming Sun and Yi Zhang. 2018. Conversational recommender system. In *Proceedings of the 41st International Acm Sigir Conference on Research & Development in Information Retrieval*. 235–244.
[212] Duyu Tang, Nan Duan, Tao Qin, and Ming Zhou. 2017. Question answering and question generation as dual tasks. *CoRR* abs/1706.02027 (2017).
[213] Duyu Tang, Nan Duan, T. Qin, and M. Zhou. 2017. Question answering and question generation as dual tasks. *ArXiv* abs/1706.02027 (2017).
[214] 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 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long Papers)*. Association for Computational Linguistics, 1564–1574. DOI: https://doi.org/10.18653/v1/N18-1141
[215] Leila Tavakoli. 2020. Generating clarifying questions in conversational search systems. In *Proceedings of the 29th ACM International Conference on Information & Knowledge Management*. 3253–3256.
[216] Adam Trischler, T. Wang, Xingdi Yuan, J. Harris, Alessandro Sordoni, Philip Bachman, and Kaheer Suleman. 2017. NewsQA: A machine comprehension dataset. In *Rep4NLP@ACL*.
[217] Luu Anh Tuan, Darsh J. Shah, and Regina Barzilay. 2020. Capturing greater context for question generation. In *Proceedings of the 34th AAAI Conference on Artificial Intelligence, the 32nd Innovative Applications of Artificial Intelligence Conference, the 10th AAAI Symposium on Educational Advances in Artificial Intelligence*. AAAI Press, 9065–9072. Retrieved from https://aaai.org/ojs/index.php/AAAI/article/view/6440.
[218] Leo C. Ureel, Kenneth D. Forbus, Christopher K. Riesbeck, and Lawrence A. Birnbaum. 2005. Question generation for learning by reading. In *AAAI Workshop*. 22–26.
[219] Stalin Varanasi, Saadullah Amin, and G. Neumann. 2020. CopyBERT: A unified approach to question generation with self-attention. In *NLP4CONVAI*.
[220] A. Varga. 2010. WLV: A question generation system for the QGSTEC 2010 task B.
[221] Cai Zhiqiang Vasile Rus and Graesser Art. 2008. Question generation: Example of a multi-year evaluation campaign. In *WS on the QGSTEC*.
[222] Ashish Vaswani, Noam Shazeer, Niki Parmar, Jakob Uszkoreit, Llion Jones, Aidan N. Gomez, L. Kaiser, and Illia Polosukhin. 2017. Attention is all you need. *ArXiv* abs/1706.03762 (2017).
[223] Alex Wang, Kyunghyun Cho, and Mike Lewis. 2020. Asking and answering questions to evaluate the factual consistency of summaries. *arXiv preprint arXiv:2004.04228* (2020).
[224] Bingning Wang, Xiao chuan Wang, Ting Tao, Qi Zhang, and Jingfang Xu. 2020. Neural question generation with answer pivot. In *AAAI*.
[225] Huazheng Wang, Zhe Gan, Xiaodong Liu, Jingjing Liu, Jianfeng Gao, and Hongning Wang. 2019. Adversarial domain adaptation for machine reading comprehension. *ArXiv* abs/1908.09209 (2019).
[226] Siyuan Wang, Zhongyu Wei, Zhihao Fan, Yang Liu, and Xuanjing Huang. A multi-agent communication framework for question-worthy phrase extraction and question generation. In *Proceedings of the 33rd AAAI Conference on Artificial Intelligence, the 31st Innovative Applications of Artificial Intelligence Conference, the 9th AAAI Symposium on Educational Advances in Artificial Intelligence*. 7168–7175.
[227] T. Wang, Xingdi Yuan, and Adam Trischler. 2017. A joint model for question answering and question generation. *ArXiv* abs/1706.01450 (2017).
[228] W. Wang, Shi Feng, D. Wang, and Y. Zhang. 2019. Answer-guided and semantic coherent question generation in open-domain conversation. In *EMNLP/IJCNLP*.
[229] Wenhui Wang, Furu Wei, Li Dong, Hangbo Bao, Nan Yang, and M. Zhou. 2020. MiniLM: Deep self-attention distillation for task-agnostic compression of pre-trained transformers. *ArXiv* abs/2002.10957 (2020).

[230] 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 Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, 2193–2203. DOI: https://doi.org/10.18653/v1/P18-1204

[231] Yutong Wang, Jiyuan Zheng, Qijiong Liu, Zhou Zhao, Jun Xiao, and Yueting Zhuang. 2019. Weak supervision enhanced generative network for question generation. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI'19)*. International Joint Conferences on Artificial Intelligence Organization, 3806–3812.

[232] Zhenduo Wang and Qingyao Ai. 2021. Controlling the risk of conversational search via reinforcement learning. *arXiv preprint arXiv:2101.06327* (2021).

[233] Z. Wang, Andrew S. Lan, Weili Nie, A. Waters, Phillip J. Grimaldi, and Richard Baraniuk. 2018. QG-net: A data-driven question generation model for educational content. In *Proceedings of the 5th ACM Conference on Learning at Scale*.

[234] John H. Wolfe. 1976. Automatic question generation from text—An aid to independent study. In *SIGCSE*. Association for Computing Machinery, 104–112.

[235] Xiuyu Wu, Nan Jiang, and Yunfang Wu. 2020. A question type driven and copy loss enhanced framework for answer-agnostic neural question generation. *ArXiv* abs/2005.11665 (2020).

[236] Dongling Xiao, Han Zhang, Yukun Li, Yu Sun, Hao Tian, Hua Wu, and Haifeng Wang. 2020. ERNIE-GEN: An enhanced multi-flow pre-training and fine-tuning framework for natural language generation. In *Proceedings of the 29th International Joint Conference on Artificial Intelligence (IJCAI'20)*. International Joint Conferences on Artificial Intelligence Organization, 3997–4003.

[237] Chen Xing, W. Wu, Yu Wu, J. Liu, Yalou Huang, M. Zhou, and W. Ma. 2017. Topic aware neural response generation. In *AAAI*.

[238] Xinchen Yan, Jimei Yang, Kihyuk Sohn, and H. Lee. 2016. Attribute2Image: Conditional image generation from visual attributes. *ArXiv* abs/1512.00570 (2016).

[239] Yu Yan, Weizhen Qi, Yeyun Gong, Dayiheng Liu, Nan Duan, Jiusheng Chen, Ruofei Zhang, and Ming Zhou. 2020. ProphetNet: Predicting future n-gram for sequence-to-sequence pre-training. *CoRR* abs/2001.04063 (2020).

[240] Zhilin Yang, Zihang Dai, Yiming Yang, Jaime Carbonell, Russ R. Salakhutdinov, and Quoc V. Le. 2019. Xlnet: Generalized autoregressive pretraining for language understanding. In *Proceedings of the International Conference on Advances in Neural Information Processing Systems*. 5753–5763.

[241] Zhilin Yang, Junjie Hu, Ruslan Salakhutdinov, and William Cohen. 2017. Semi-supervised QA with generative domain-adaptive nets. In *Proceedings of the 55th Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*. Association for Computational Linguistics, 1040–1050. DOI: https://doi.org/10.18653/v1/P17-1096

[242] Z. Yang, Peng Qi, Saizheng Zhang, Y. Bengio, W. Cohen, R. Salakhutdinov, and Christopher D. Manning. 2018. HotpotQA: A dataset for diverse, explainable multi-hop question answering. *ArXiv* abs/1809.09600 (2018).

[243] Kaichun Yao, L. Zhang, Tiejian Luo, Lili Tao, and Y. Wu. 2018. Teaching machines to ask questions. In *IJCAI*.

[244] Xuchen Yao. 2010. Question generation with minimal recursion semantics. In *Proceedings of QG2010: The Third Workshop on Question Generation*. 68–75.

[245] Xuchen Yao, Emma Tosch, G. Chen, E. Nouri, Ron Artstein, A. Leuski, Kenji Sagae, and D. Traum. 2012. Creating conversational characters using question generation tools. *Dialog. Disc.* 3 (2012), 125–146.

[246] Xuchen Yao and Yi Zhang. 2010. Question generation with minimal recursion semantics. In *Proceedings of the 3rd Workshop on Question Generation*. Citeseer, 68–75.

[247] Qian Yu, Lidong Bing, Qiong Zhang, Wai Lam, and Luo Si. 2020. Review-based question generation with adaptive instance transfer and augmentation. In *Proceedings of the 58th Meeting of the Association for Computational Linguistics*. Association for Computational Linguistics, 280–290. DOI: https://doi.org/10.18653/v1/2020.acl-main.26

[248] Xingdi Yuan, Tong Wang, Caglar Gulcehre, Alessandro Sordoni, Philip Bachman, Saizheng Zhang, Sandeep Subramanian, and Adam Trischler. 2017. Machine comprehension by text-to-text neural question generation. In *Proceedings of the 2nd Workshop on Representation Learning for NLP*. Association for Computational Linguistics, 15–25. DOI: https://doi.org/10.18653/v1/W17-2603

[249] Hamed Zamani, Susan Dumais, Nick Craswell, Paul Bennett, and Gord Lueck. 2020. Generating clarifying questions for information retrieval. In *Proceedings of the Web Conference*. 418–428.

[250] Hamed Zamani, Bhaskar Mitra, Everest Chen, Gord Lueck, Fernando Diaz, Paul N. Bennett, Nick Craswell, and Susan T. Dumais. 2020. Analyzing and learning from user interactions for search clarification. In *Proceedings of the 43rd International ACM SIGIR Conference on Research and Development in Information Retrieval*. 1181–1190.

[251] Poorya Zaremoodi, Wray L. Buntine, and Gholamreza Haffari. 2018. Adaptive knowledge sharing in multi-task learning: Improving low-resource neural machine translation. In *ACL*.

[252] Jingqing Zhang, Y. Zhao, Mohammad Saleh, and Peter J. Liu. 2019. PEGASUS: Pre-training with extracted gap-sentences for abstractive summarization. *ArXiv* abs/1912.08777 (2019).

[253] R. Zhang, J. Guo, Yixing Fan, Yanyan Lan, and Xueqi Cheng. 2020. Dual-factor generation model for conversation. *ACM Trans. Inf. Syst.* 38 (2020), 1–31.
[254] R. Zhang, J. Guo, Y. Fan, Yanyan Lan, J. Xu, and X. Cheng. 2018. Learning to control the specificity in neural response generation. In *ACL*.
[255] Shiyue Zhang and Mohit Bansal. 2019. Addressing semantic drift in question generation for semi-supervised question answering. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP'19)*. Association for Computational Linguistics, 2495–2509. DOI: https://doi.org/10.18653/v1/D19-1253
[256] S. Zhang, Lizhen Qu, Shaodi You, Zhenglu Yang, and Jiawan Zhang. 2017. Automatic generation of grounded visual questions. *ArXiv* abs/1612.06530 (2017).
[257] 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 and Knowledge Management*. 177–186.
[258] Y. Zhang and Qiang Yang. 2017. A survey on multi-task learning. *ArXiv* abs/1707.08114 (2017).
[259] Yao Zhao, Xiaochuan Ni, Yuanyuan Ding, and Qifa Ke. 2018. Paragraph-level neural question generation with maxout pointer and gated self-attention networks. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing*. Association for Computational Linguistics, 3901–3910. DOI: https://doi.org/10.18653/v1/D18-1424
[260] Hai-Tao Zheng, Jinxin Han, J. Chen, and A. K. Sangaiah. 2018. A novel framework for automatic Chinese question generation based on multi-feature neural network model. *Comput. Sci. Inf. Syst.* 15 (2018), 487–499.
[261] Qingyu Zhou, Nan Yang, Furu Wei, Chuanqi Tan, Hangbo Bao, and Ming Zhou. 2018. Neural question generation from text: A preliminary study. In *Natural Language Processing and Chinese Computing*, Xuanjing Huang, Jing Jiang, Dongyan Zhao, Yansong Feng, and Yu Hong (Eds.). Springer International Publishing, Cham, 662–671.
[262] Qingyu Zhou, Nan Yang, Furu Wei, and M. Zhou. 2018. Sequential copying networks. In *AAAI*.
[263] Wenjie Zhou, Minghua Zhang, and Yunfang Wu. 2019. Multi-task learning with language modeling for question generation. *ArXiv* abs/1908.11813 (2019).
[264] Wenjie Zhou, Minghua Zhang, and Yunfang Wu. 2019. Question-type driven question generation. In *Proceedings of the Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP'19)*. Association for Computational Linguistics, 6032–6037. DOI: https://doi.org/10.18653/v1/D19-1622
[265] Kangli Zi, Xingwu Sun, Yanan Cao, Shi Wang, X. Feng, Zhaobo Ma, and C. Cao. 2019. Answer-focused and position-aware neural network for transfer learning in question generation. In *KSEM*.
[266] Barret Zoph, Deniz Yuret, Jonathan May, and K. Knight. 2016. Transfer learning for low-resource neural machine translation. *ArXiv* abs/1604.02201 (2016).
[267] Çaglar Gülçehre, Sungjin Ahn, Ramesh Nallapati, Bowen Zhou, and Yoshua Bengio. 2016. Pointing the unknown words. *ArXiv* abs/1603.08148 (2016).

Received March 2021; revised May 2021; accepted May 2021