SI630 Final Project: Answer-aware Question Generation

Era Parihar and **Tzu-Jo Hsu**

1 Introduction and motivation

Formulating/Asking the right questions is essential for learning and comprehension, and it is important for both human and machine intelligence. The task of Question Generation (QG) involves automatically generating questions from a given paragraph of text. Such systems can be applied in a wide range of domains, such as improving question-answer systems and conversational systems/chatbots; they have also been considerably experimented with within the educational sector. However, most existing QG systems focus on short factoid answers [10], while in real-world scenarios, we encounter a diverse set of answer types. This project explores how expanding/diversifying the types of questions for finetuning affects the performance of QG, and explores better ways to evaluate the QG result. Our project also differs in terms of integrating datasets from three different sources. By fine-tuning t5-large and BART-large on enlarged datasets enriched with paraphrased content, we have demonstrated that our approach significantly improves performance across all metrics. We also provide insights on evaluating various large language models. Solving the problem of automated question generation is crucial as it has significant implications in educational technology, enabling personalized learning and automated content creation, which can make educational resources more interactive and accessible.

2 NLP Task Definition

Question Generation can be formulated as finding a function:

$$f(C, A) = Q' \approx Q$$

,where C is the context document, A is the corresponding answer to the question Q. Our goal is to generate Q' that is semantically equivalent to Q. How similar is the approximation will be evaluated through some evaluation metrics, which will be covered in the section `Evaluation`. We focus on the standalone question generation setting, where questions are generated independently of each other. This is different from a multi-round conversational/sequential question generation setting. The questions in QG can be classified as answer-aware and answer-agnostic. This project focuses on the answer-aware setting where the generated questions ask towards the given context from the answer.

3 Data

The training or fine-tuning for Question Generation tasks can directly apply Question Answering datasets, but many datasets' answer contain only 2-4 tokens. Instead of applying just one dataset, we gathered these data from three different sources, ensuring our model can deal with different type of context and questions. These datasets include SQuAD 2.0 [12], AdversarialQA [3] and MS Marco [2]. In this way, the model could adapt to more diverse scenarios. We try to avoid the case where the questions and answers are simply factoid check.

We looked into how much data is required for fine-tuning a task. It has been widely suggested that at least 1000 data entries for one single task; also if the data quality is high, we can apply less data. Combining the fact that we only have limited computational resources, we've decided to construct a dataset with 5000 entries for training set and 1500 entries for testing set. Here are some statistics of our dataset. The average tokens length (split by white space) for context is 95.17; 7.258 for answer and 8.951 for questions.

The Stanford Question Answering Dataset (SQuAD) stands out as one of the most commonly utilized. SQuAD [12] is a reading comprehension dataset that includes questions formulated by crowd workers based on a collection of Wikipedia articles. Each question's answer is a text segment or span from the relevant reading passage, although some questions may be deemed unanswerable.

The next one is AdversarialQA [3]. It features three datasets created via an adversarial approach, challenging models to answer difficult questions using SQuADv1.1 passages. It leverages models BiDAF, BERT-Large, and RoBERTa-Large to produce datasets D(BiDAF), D(BERT), and D(RoBERTa), each with 10,000 training, 1,000 validation, and 1,000 test examples, focusing on questions that defy state-of-the-art model capabilities. We ensured the passages we sampled have no overlap with the SQuAD.

The MS MARCO (Microsoft MAchine Reading Comprehension) collection comprises various datasets designed to advance deep learning in search applications. Initially, it featured a question-answering dataset containing 100,000 real Bing questions accompanied by human-generated answers. Subsequently, the collection expanded to include additional datasets: one with 1,000,000 questions, a natural language generation dataset, a passage ranking dataset, a keyphrase extraction dataset, a crawling dataset, and one focused on conversational search..

Passage	**Question**	**Answer**
Martin Luther married Katharina von Bora, one of 12 nuns he had helped escape from the Nimbschen Cistercian convent in April 1523, when he arranged for them to be smuggled out in herring barrels. "Suddenly, and while I was occupied with far different thoughts," he wrote to Wenceslaus Link, "the Lord has plunged me into marriage." At the time of their marriage, Katharina was 26 years old and Luther was 41 years old.	In a letter who did Luther credit for his union with Katharina?	the Lord

Table 1: Example of a QA Table

	Adv. QA	**SQuAD**	**MS MARCO**
Training set	1,667	1,666	1,666
Enlarged Train	10624		
Testing set	500	500	500

Table 2: Number of instances for our dataset

After we had the dataset ready, we decided to enlarge the dataset by paraphrasing the ground truth questions. In other words, during the fine-tuning process, the model was presented with various rephrased and reworked versions of the same question, allowing it to learn from multiple acceptable formulations. The goal is to let the model be more expressive when generating the questions since there is no single correct way to ask a question. We will discuss the detailed approach in the methodology section. After the enlargement, we have around 10,000 instances for our training data.

4 Related Work

The availability of extensive datasets, alongside advances in data-driven learning techniques, has catalyzed the prominence of neural network-driven approaches in question generation (QG) methodologies. This domain has experienced substantial progression over the years with the integration of cutting-edge deep learning frameworks.

Traditional methods for question generation tasks are mostly rule-based, which involves using heuristic rules to transform a descriptive text into a related question. As the development of neural networks went on, neural models were widely applied in the QG field as they provide an end-to-end trainable framework and enable joint optimization for content selection and question construction [13]. RNN seq2seq framework with attention has been a popular direction. In more recent developments, the use of transformers-based models and Generative AI techniques has become prevalent, which builds upon a pre-trained large language model by finetuning the QG task in a supervised way. Some research has utilized knowledge graphs [6]and retrieval-based augmentation techniques[5] to further improve the performance of QG tasks. [10] addressed the diversity of question generation, they have utilized pre-trained models T5 and BART on different datasets to allow for a generation of questions that leads to different answer types, such as yes/no, extractive and abstractive answers etc. This modern advancement has resulted in significant improvements across various NLP tasks, including Question Answering [11, 8], Text Summarization [7, 1], and numerous classification challenges [4].

Most research uses Exact Match, F1-score, BLEU, ROUGE and METEOR etc. as their metrics to evaluate the performance of generation, which will be explained in detail in the next section. In Data, we have mentioned datasets widely used for QG tasks. In this project, we plan to improve the quality of question generation in the sense of diversity and robustness, by designing and implementing novel methodologies; more specifically, it allows the generation of more sophisticated question forms, not just asking for a short answer. As the project progresses, we will continue to refine and establish the scope of our research objectives.

5 Methodology

As we looked into the topic of question generation, we identified one important bottleneck: the evaluation of question generation systems. In an answer-aware task, the model-generated question is compared against a golden standard question in the dataset in most research studies. However, there are various ways to form the same question that is grammatically and semantically correct, yet common NLP metrics (including exact match, f1-score, BLEU, ROUGE, etc.) focus on the n-gram similarities of the sentence. This fails to consider other questions that serve the equivalent purpose. Seeing these limitations, some researchers have proposed to fine-tune a pre-trained model to classify whether the generated question is answerable. Drawing inspiration from recent advances, we have come up with the following pipeline:

To diversify the questions in the dataset, we employed fine-tuned paraphraser models, including Vamsi/T5_Paraphrase_Paws and eugenesiow/bart-paraphrase. Namely, for each context, question, and answer data instance c_i, q_i, a_i, we obtain $f(q_i) = \{q_{i1}, q_{i2}, ..., q_{ik}\}$, where f is some paraphraser. Here the k is a hyper-parameter, we set it to 6. To ensure the quality of the paraphrased questions, we utilized a BERT-based sentence embedding similarity score (all-MiniLM-L6-v2) to filter out those deviating significantly from the original question or being identical. Specifically, we retained paraphrased questions with similarity scores between 0.96 and 0.99, a range determined based on our experience. Paraphrased questions with scores lower than 0.96 were discarded to avoid deviating excessively from the original meaning. This pipeline is visualized in Figure 1.

Next, the dataset was prepared in a certain format and fine-tune it using a pre-trained language model. We applied the following formats for finetuning:

$$< answer > a_i < context > c_i$$

and

$$answer : a_i, context : c_i$$

But after some experiments, the second one was chosen since their performance made no difference. We tried out a number of models for fine-tuning. However, most models are too big to load into GPU memory, raising CUDA out of memory errors. According to HuggingFace Model Memory calculator, training t5-large requires at least 10.99 GB for vRAM using ADAM as optimizer using a batch size of 1. Moreover, we only got 2 hours max usage for 32GB memory on Great Lakes, we eventually narrowed down to T5-large (770m) and BART-large (406m).

T5, or Text-to-Text Transfer Transformer, is a Transformer-based model that adopts a text-to-text framework for processing various text-based tasks. This model treats every NLP challenge—whether it be translation, question answering, or classification—as a problem of converting input text into target text. This universal approach allows the same model architecture, loss function, and hyperparameters to be applied consistently across a wide range of tasks.

BART is a denoising autoencoder that pretrains sequence-to-sequence models through a two-step process: corrupting text with noising techniques, then training a model to reconstruct the original text. It uses a standard Transformer-based architecture seen in neural machine translation (NMT), featuring a bidirectional encoder like BERT for complete input visibility and a left-to-right decoder like GPT with a causal attention mask, making it effective for various sequence-to-sequence tasks.

We trained t5-large models with a batch size of 4, learning rate 1e-4 and 5 epochs (as the loss was converging), with 0.2 of the training set assigned to be the validation set.

As for Bart-large models, we used a batch size of 2 and 3 epochs (saw a steady decline) and used a learning late of 5e-5, with 0.15 of the training set assigned to be the validation set.

For evaluation, we applied common metrics, including BLEU, ROUGE-L, METEOR to indi-

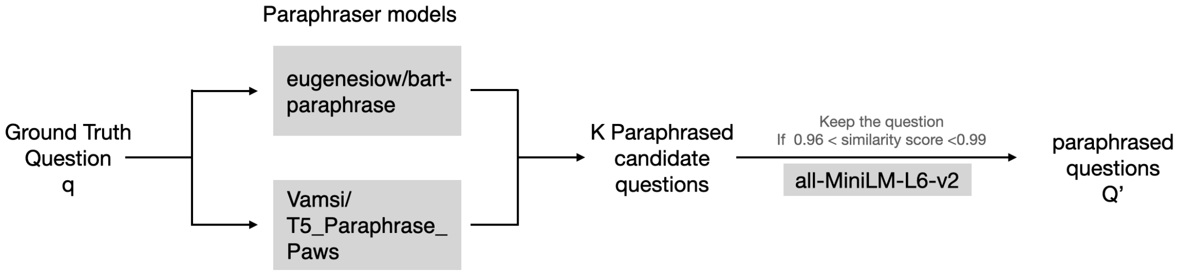

Figure 1: Illustration of the data preparation pipeline

cate the performance in terms of n-gram overlap with the ground truth question. We also included BERTscore and RQUGE for evaluating generated questions. Furthermore, we also assessed with ChatGPT - 3.5, we will discuss them in detail in the following section.

6 Evaluation and Results

6.1 Metrics

The commonly used evaluation metrics used in QG research are:

- BLEU(Bilingual Evaluation Understudy): it is a precision-based metric that compares the lexical relationship between the generated answer and the ground truth answer by computing the n-gram overlap between them [Banerjee, Meteor]. On a higher level, we multiply the brevity penalty by geometric average precision scores, where the brevity penalty penalizes sentences that are too short. $BLEU(N) = BrevityPenalty * GeometricAveragePrecisionScores(N)$

- ROUGE(Recall-Oriented Understudy for Gisting Evaluation): it is a set of metrics often used to evaluate NLP tasks such as summarization tasks and machine translations. ROUGE-N refers to the direct n-gram overlaps between the prediction and the ground truth answer. For ROUGE-L, it considers the longest common subsequence (LCS), that is, the overlap of n-consecutive words, between the prediction and the ground truth. Note that it is different from the f1-score as the f1-score measures the overlapping of the words themselves, instead of calculating n-gram, LCS etc.

- METEOR (Metric for Evaluation of Translation with Explicit Ordering): it is based on a weighted mix of unigram precision and recall; it also extends to include more advanced matching (such as stemming and synonymy matching), along with the standard exact word matching.

The determination of the most suitable metrics will be made as we delve deeper into additional research, enabling us to conduct a more thorough assessment.

RQUGE (Reference-free QUestion Generation Evaluation): RQUGE (Reference-free QUestion Generation Evaluation) is a novel metric introduced in 2022 by Alireza Mohammadshahi et al. [9]. It is designed to evaluate the quality of generated questions in the context of natural language processing (NLP) without requiring a reference question for comparison. Unlike ROUGE and other metrics, RQUGE assesses the quality of a candidate question based solely on its context and answer span. It employs a general question-answering module to determine how well the question captures the essence of the answer given the context. This is followed by a span scorer that evaluates how accurately the question leads to the correct answer span within the provided context. The score ranges from 1 to 5, where a higher score indicates a better quality question in terms of relevance and coherence with respect to the provided context and answer. Thus, RQUGE provides a unique approach to question generation evaluation by focusing on the functional and contextual alignment of the question-answer pair rather than direct textual overlap. Here is a sample for reference.

context = "Manuel has created RuPERTa-base with the support of HF-Transformers and Google" answer = "Manuel" question: Who created the RuPERTa-base?
Mean RQUGE Score: 4.958300240039826

Additionally, we have incorporated BERTScore, an automatic evaluation metric for text generation. Similar to conventional metrics, BERTScore calculates a similarity score for each token in the candidate sentence relative to each token in the reference sentence. Unlike methods that rely on exact matches, our approach involves computing token similarity based on contextual embeddings.

Furthermore, to refine our methodology, we employ prompt engineering with GPT-3.5. We present the model with a specific prompt and subsequently request it to evaluate the output from the Question Generation (QG) model in terms of grammatical correctness and semantic accuracy. This technique enables us to effectively measure the quality of the questions generated.For the system role section, "You are a professional sentence evaluator. " is provided.

The prompts we are using are the following:

- **On a scale of 1 to 5, where 1 is completely incorrect and 5 is perfectly correct, rate the grammatical correctness of the generated question. Output the score only.**
  Generated question: Q
  Ground truth question: GT

- **On a scale of 1 to 5, where 1 is not at all relevant and 5 is highly relevant, rate how semantically aligned the generated question is with the ground truth question. Output the score only.**
  Generated question: Q
  Ground truth question: GT

6.2 Result Analysis

For our baselines, we selected three models: a question generation (QG) specific fine-tuned t5-base model from Hugging Face, the original t5-large, and the original BART-large. As shown in Table 1, the fine-tuned t5-base outperforms the raw t5-large on almost all metrics, validating our motivation for fine-tuning. Our model, trained on an enlarged dataset through paraphrasing, achieved the highest scores for BLEU (0.099), RougeL (0.336), and Meteor (0.335). These results indicates that providing variants of question form can bring improvements in terms of n-gram overlap and alignment with reference texts. Furthermore, our model's RQuGE score of 2.810 is marginally higher than the fine-tuned t5-large, indicating improved question-answer relevance.

Models	BLEU	RougeL	Meteor
iarfmoose/t5-base-question-generator	0.017	0.205	0.228
google-t5/t5-large	0.009	0.109	0.148
Finetuned t5-large	0.057	0.234	0.242
Finetuned t5-large (enlarged)	**0.099**	**0.336**	**0.335**
facebook/bart-large	0.014	0.139	0.206
Finetuned bart-large	0.016	0.096	0.227
Finetuned bart-large (enlarged)	0.063	0.263	0.246

Table 3: **Performance across all models: traditional metrics**

Models	RQuGE	BERTscore
iarfmoose/t5-base-question-generator	1.906	0.838
google-t5/t5-large	2.51	0.821
Finetuned t5-large	2.63	0.876
Finetuned t5-large (enlarged)	**2.810**	0.857
facebook/bart-large	2.946	0.828
Finetuned bart-large		0.803
Finetuned bart-large (enlarged)	2.689	**0.889**

Table 4: **Performance across all models: RQuGE and BERTscore**

In the case of the original Bart-large and the fine-tuned Bart-large models, the latter shows improved performance. Our model, trained on an expanded dataset enhanced with paraphrased content, achieved the highest scores in BLEU (0.063), RougeL (0.263), and Meteor (0.246). These metrics suggest incorporating varied question formulations enhances n-gram overlap and alignment with reference texts. However, the RQuGE score of our model stands at 2.689, lower than that of the original, indicating a shift in the question formulation structure due to the enlarged dataset. Additionally, there is a notable increase in the BERTScore, indicating enhanced text generation quality.

Models	GPT3.5-grammar	GPT3.5-semantic
Finetuned t5-large	2.064	2.290
Finetuned t5-large (enlarged)	**3.422**	**2.918**
Finetuned bart-large	1.258	1.024
Finetuned bart-large (enlarged)	3.161	2.265

Table 5: Comparing the how well the models perform on grammar and semantics using GPT3.5

We further utilized GPT-3.5 to assess the grammatical correctness and semantic relevance of the generated questions. It is important to note that we conducted multiple evaluations to ensure the stability of ChatGPT's outputs. Given that we specified "output the score only" in the prompt, 95% of the responses comprised solely numerical scores, without additional explanations. As depicted in Table 2, models trained on expanded question sets demonstrably outperformed in both assessed dimensions, with a particularly notable improvement in grammatical correctness. By exposing the model to multiple accepted forms or paraphrases of the same question during the fine-tuning process, it likely acquired the ability to produce more grammatically accurate and well-structured questions. The paraphrased questions presumably encompassed a broad spectrum of grammatically correct methods to convey the same semantic intent. This varied exposure facilitated the model's learning of proper grammar and phrasing nuances. Furthermore, without such paraphrasing, the model might have been predisposed to generating questions biased towards the specific phrasing found in the original dataset. The introduction of paraphrased variations enabled the model to generalize better and create grammatically correct questions that extend beyond the initial phrasing.

Table 6 shows the comparison of generated questions and standard questions. We can see that the pre-trained t5-large was unable to understand the instructions, outputting repeating and many random tokens and unable to form a complete sentence. After the fine-tuning, our model is able to capture the format and output a complete question. Our models can perform well on short and factual-check questions as in example 2. It can also correctly identify the subjects and entities in the given context; for example, "NYC City Council", "Jacksonville" were mentioned. However, some questions still takes the form of a statement rather than a direct question; like in example 1, the ground truth question is more specific and directly asks about the role of solar energy in limiting climate change. In example 4, the generated question and the ground truth question are related to the topic of freezing food, but they have different perspectives. Overall, the examples demonstrate that the generated questions can sometimes capture the overall topic or intent of the ground truth question, but there can be variations in phrasing, specificity, and perspective.

7 Discussion

Our experiments have confirmed several ideas and shed light on future directions for this topic. First of all, the data quality for fine-tuning matters. By providing various forms of questioning, the performance of our models improved across all metrics. This could potentially be a pre-processing technique when we are dealing with other open-ended generation tasks. These enlarged questions could be further tested with different prompt design skills. Secondly, by fine-tuning for specific tasks, a smaller model may perform as well as larger models, which gives us a lower inference cost and higher speed. In our case, raw t5-large or BART-large will output irrelevant or random tokens (such as punctuation) given the same prompt, while after the tuning, the output has none of these noises. This repeating behaviour of raw t5-large's output could be because during the next-token prediction, the built-in attention-mechanism is reinforcing certain tokens. This is alleviated after fine-tuning.

Additionally, during the fine-tuning of BART with both the original and enlarged datasets, we

	Models	Generated Question	Ground Truth
1	google-t5/ t5-large	/answer answer 51 query The City Council is theelected body of the City. answer What is the City Council? context anchor	How many members are on the NYC city council?
2		question humid subtropical climate context mild question mild mild mild context mild subtropical context mild mi ldquestions mild	What is winter like in Jacksonville?
1	Finetuned t5-large (enlarged)	solar energy plays a key role in decarbonizing the global economy?	What could the sun's energy do to help limit climate change?
2		what is pyrroloquinoline quinine?	what is pyrroloquinoline quinine
3		what is the weather in winters in Jacksonville?	What is winter like in Jacksonville?
4		why do some foods freeze well?	How does freezing affect the nutrients in food?
5		how many council members are there on the NYC City Council?	How many members are on the NYC city council?

Table 6: Generated questions vs. Ground Truth Questions

observed that the model performed better when trained on the enlarged dataset, which contained a greater variety of question iterations. In contrast, the BART model trained with the original dataset predominantly produced repetitive outputs, thereby impeding its ability to learn the underlying logic of question formation.

Moreover, the evaluation of question generation remains a challenge. Unlike Question-Answering tasks, where the model locates and extracts the answers from the given passage (for factoid questions), question generation is more open-ended. In Tables 6 and 7, numerous examples show that while the meaning remains consistent, the phrasing differs. Relying solely on overlapping-based metrics in such cases does not accurately reflect performance, as these scores may not capture the nuanced variations in wording. In response to this issue, numerous researchers have begun utilizing larger language models for evaluation purposes. We adopted this approach by prompting GPT-3.5 (ChatGPT) to assess the generated questions for grammatical correctness and semantic relevance. Our results indicated a significant improvement in grammar by incorporating a diverse range of question forms. Although our final results still have a lot of room for improvement (0.336 for RougeL and 0.335 for Meteor, are not the best), we believe that our approach, performed on a larger-size model (such as 7b, 13b), could yield promising results. The current version of model is not serviceable to end-users. In real-world application, we think that the task question generation is valuable in assisting learning (AI for education). It could be part of an integrated application, along with summarization, translation, other abilities.

Finally, we learned that computation resources are extremely important, especially when dealing with millions and billions of parameter sizes. Being able to estimate the time, speed, and memory required for each experiment is a valuable skill. The future direction of this line would be to try out efficient fine-tuning skills, such as freezing some parameters and doing some approximation (such as LoRA.). Another line would be knowledge distillation, where we train a student model to align with the output of a teacher model. This could potentially allow us to boost the performance of the smaller model.

8 Conclusion

In conclusion, our project aimed at improving answer-aware question generation through the finetuning of original t5-large and BART-large models on enlarged datasets enriched with paraphrased content. Our evaluations demonstrate significant performance enhancements across several metrics such as BLEU, RougeL, Meteor, RQuGE, and

	Models	Generated Question	Ground Truth
1	facebook/ bart-large	WhatWhowhathowWhichInHowThewhere WherewhenWhenWhy iswhyA wasForOn whichIfwhoOfDuringToAccordingAboutBey Are did are does long	The methods kennel clubs used to classify dogs is what?’
1	Finetuned bart-large (enlarged)	what is kennel clubs	The methods kennel clubs used to classify dogs is what?
2		What is another term for Avestan?	What is the contemporary name of the religion Avesta was part of?
3		What was given to the fourth Dalai Lama in 1616?	Shared by what title was granted to the fourth Dalai Lama?

Table 7: Generated questions vs. Ground Truth Questions

BERTScore, highlighting the efficacy of incorporating diverse question formulations to optimize text alignment and n-gram overlap. Despite the slightly lower RQuGE score from the fine-tuned BART-large model, the findings suggest positive shifts in question formulation attributable to the broader dataset. Utilizing GPT-3.5 for rigorous assessments of grammatical correctness and semantic relevance, we observed that models trained on these comprehensive datasets consistently outperformed their counterparts. These results affirm the critical importance of data quality in fine-tuning and propose that tailored smaller models can deliver significant efficiencies in inference costs and speed. Our study not only deepens the understanding of question generation dynamics but also establishes a foundation for future advancement in related topics.

9 Other Things We Tried

We investigated five to six distinct datasets and tested numerous models for paraphrasing and question generation tasks, including facebook/opt and llama-2. Additionally, we developed a knowledge distillation pipeline for T5 models, although the outcomes were less than promising. Our dataset comprised 8,000 entries for training and 1,000 entries for the testing set. After observing no further improvements in metrics and loss after a few epochs, we decided to cease these efforts and redirected our focus back to the original fine-tuning methodology. We also experimented with various prompts to determine the most effective ones for evaluation and fine-tuning.

10 What You Would Have Done Differently or Next

Reflecting on our project on question generation, we identified the evaluation of question generation systems as a significant bottleneck. While our use of fine-tuned paraphraser models and diverse datasets improved n-gram overlap and question-answer relevance, the existing metrics still fall short in fully capturing the nuances of question validity and diversity. If given the chance to revisit certain aspects of the project, we would explore more advanced paraphrasing techniques and expand the datasets further to include more domain-specific content. This would potentially address the observed discrepancies in question formulation and enhance the model’s ability to generate contextually rich and varied questions. Moreover, if we apply parameter-efficient finetuning techniques, we could have worked on more advanced or larger models. Figuring out a way to systematically evaluate the computation resources could be very useful to avoid many trial-and-error iterations. Additionally, integrating domain-specific datasets could pave the way for more tailored and precise question generation, which is particularly valuable in specialized fields such as medical or legal questioning. This approach not only promises to refine the outputs but also opens up new avenues for future research in applying question generation technology more effectively across different domains.

References

[1] Siwar Abbes, Sarra Ben Abbès, Rim Hantach, and Philippe Calvez. 2021. Automatic text summarization using transformers. In *Knowledge Graphs and*

Semantic Web: Third Iberoamerican Conference and Second Indo-American Conference, KGSWC 2021, Kingsville, Texas, USA, November 22–24, 2021, Proceedings 3, pages 308–320. Springer.

[2] Payal Bajaj, Daniel Campos, Nick Craswell, Li Deng, Jianfeng Gao, Xiaodong Liu, Rangan Majumder, Andrew McNamara, Bhaskar Mitra, Tri Nguyen, et al. 2016. Ms marco: A human generated machine reading comprehension dataset. *arXiv preprint arXiv:1611.09268*.

[3] Max Bartolo, Alastair Roberts, Johannes Welbl, Sebastian Riedel, and Pontus Stenetorp. 2020. Beat the ai: Investigating adversarial human annotation for reading comprehension. *Transactions of the Association for Computational Linguistics*, 8:662–678.

[4] Wei-Cheng Chang, Hsiang-Fu Yu, Kai Zhong, Yiming Yang, and Inderjit S Dhillon. 2020. Taming pretrained transformers for extreme multi-label text classification. In *Proceedings of the 26th ACM SIGKDD international conference on knowledge discovery & data mining*, pages 3163–3171.

[5] Chao-Chun Hsu, Eric Lind, Luca Soldaini, and Alessandro Moschitti. 2021. Answer generation for retrieval-based question answering systems. *arXiv preprint arXiv:2106.00955*.

[6] Sathish Reddy Indurthi, Dinesh Raghu, Mitesh M Khapra, and Sachindra Joshi. 2017. Generating natural language question-answer pairs from a knowledge graph using a 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*, pages 376–385.

[7] Urvashi Khandelwal, Kevin Clark, Dan Jurafsky, and Lukasz Kaiser. 2019. Sample efficient text summarization using a single pre-trained transformer. *arXiv preprint arXiv:1905.08836*.

[8] Alok Kumar, Aditi Kharadi, Deepika Singh, and Mala Kumari. 2021. Automatic question-answer pair generation using deep learning. In *2021 Third International Conference on Inventive Research in Computing Applications (ICIRCA)*, pages 794–799. IEEE.

[9] Alireza Mohammadshahi, Thomas Scialom, Majid Yazdani, Pouya Yanki, Angela Fan, James Henderson, and Marzieh Saeidi. 2022. Rquge: Reference-free metric for evaluating question generation by answering the question. *arXiv preprint arXiv:2211.01482*.

[10] Lidiya Murakhovs' ka, Chien-Sheng Wu, Philippe Laban, Tong Niu, Wenhao Liu, and Caiming Xiong. 2021. Mixqg: Neural question generation with mixed answer types. *arXiv preprint arXiv:2110.08175*.

[11] Khalid Nassiri and Moulay Akhloufi. 2023. Transformer models used for text-based question answering systems. *Applied Intelligence*, 53(9):10602–10635.

[12] Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. Squad: 100,000+ questions for machine comprehension of text. *arXiv preprint arXiv:1606.05250*.

[13] Ruqing Zhang, Jiafeng Guo, Lu Chen, Yixing Fan, and Xueqi Cheng. 2021. A review on question generation from natural language text. *ACM Transactions on Information Systems (TOIS)*, 40(1):1–43.