ECAI 2024
U. Endriss et al. (Eds.)

doi:10.3233/FAIA240615

Bridging the Gap: Generating a Comprehensive Biomedical Knowledge Graph Question Answering Dataset

Xi Yan[a,*], Patrick Westphal[a], Jan Seliger[b] and Ricardo Usbeck[c]

[a]Universität Hamburg
[b]Universitätsklinikum Hamburg-Eppendorf
[c]Leuphana Universität Lüneburg
ORCID (Xi Yan): https://orcid.org/0000-0002-1829-6554, ORCID (Patrick Westphal): https://orcid.org/0000-0002-3855-4485, ORCID (Jan Seliger): https://orcid.org/0000-0003-1337-2850, ORCID (Ricardo Usbeck): https://orcid.org/0000-0002-0191-7211

Abstract. Despite the plethora of resources such as large-scale corpora and manually curated Knowledge Graphs (KGs), the ability to perform reasoning with natural language inputs over biomedical graphs remains challenging due to insufficient training data. We propose a novel method for automatically constructing a Biomedical Knowledge Graph Question Answering (BioKGQA) dataset sourced from PrimeKG, the largest precision medicine-oriented KG. In total, we create 85,368 question-answer pairs along with their respective SPARQL queries. Our approach generates a diverse array of contextually relevant questions covering a wide spectrum of biomedical concepts and levels of complexity. We evaluate our method based on automatic metrics alongside manual annotations. We establish novel standards tailored for KGQA systems to highlight the linguistic correctness and semantical faithfulness of the generated questions based on extracted KG facts. The compiled dataset – PrimeKGQA – serves as a valuable benchmarking resource for advancing knowledge-driven biomedical research and evaluating KGQA systems.

1 Introduction

Biomedical KGs offer a powerful framework for organizing and semantically linking heterogeneous biomedical data, enabling comprehensive exploration and analysis of the underlying biological phenomena. However, the effective use of these KGs for real-world applications, such as question answering (QA) systems [22], necessitates the availability of high-quality training datasets tailored to the intricacies of the biomedical domain.

There are three main challenges in developing large-scale bioKGQA datasets: (i) The costs of hiring annotators with professional backgrounds are usually high. (ii) There are numerous biomedical KGs not aligned with common (but evolving) ontologies. For instance, DisGeNET [28] is not fully aligned with common ontologies such as the Human Phenotype Ontology (HPO) [6] or the Disease Ontology (DO) [31]. (iii) Most existing automatic question-generation algorithms require (extensive) training data. As a result, there are only three bioKGQA datasets based on different KGs, with a total amount of 90 question-answer pairs.

* Corresponding Author. Email: xi.yan@uni-hamburg.de

Yet, recent advancements in pre-trained language models (PLMs) can tackle the above challenges by guiding the generation process with a small amount of annotated data without a costly training process, resolving the challenges (i) and (iii), by introducing prompt-based few-shot learning [38]. With a small number of samples ranging from one to twenty per class [7], PLMs generally demonstrate a strong capacity to generalize over unseen data.

Supporting the creation of a large-scale KG-based QA dataset in the biomedical domain, recently, a large database was built that integrates 20 high-quality and most cited databases in precision medicine,[1] PrimeKG [8]. It focuses on ten major biological scales, including disease-associated protein perturbations, biological processes and pathways, anatomical and phenotypic scales, and the entire range of approved drugs with their therapeutic action, considerably expanding previous efforts in disease-rooted databases. The underlying KGs of the existing bioKGQA datasets can be mapped to PrimeKG, providing a chance to integrate the other bioKGQA datasets, too.

Thus, we built a large-scale KGQA dataset on top of PrimeKG utilizing few-shot learning on PLMs. First, we transform the PrimeKG database into an RDF-based KG and set up a SPARQL endpoint. Next, we follow a general triple-to-question pipeline: (i) We sample subgraphs of specific structures from PrimeKG as reasoning paths of the question-answer pair. (ii) The answers are selected using a specific anchor selection strategy. (iii) The reasoning paths and the updated answers are linearized and sent to the PLM as parts of the input prompt to generate the underlying questions. (iv) We test several PLMs and validate them on our own as well as three well-known QA datasets. Hence, this dataset can serve as a vital resource for advancing biomedical research, enabling the development and evaluation of QA systems that efficiently retrieve relevant biomedical knowledge.

Therefore, the contribution of this work is three-fold: (i) We present the first large-scale biomedical KGQA dataset. PrimeKGQA is factor 1000 larger than the second-largest KGQA dataset. (ii) We develop a novel framework to generate questions based on the KG

[1] A data and KG-centered approach to disease diagnosis and treatment that accounts for the variability in genetics, environment, and lifestyle across individuals

Table 1. Statistics of the existing BioKGQA datasets

	QALD-4	Bgee-QA	OMA-QA	CORDIS-QA
# q-a pairs	50	20	10	30
Underlying KG	DrugBank	Bgee	OMA	CORDIS

triples. (iii) We initiate a novel anchored answer selection strategy. The developed model and dataset are publicly available on GitHub.[2]

2 Related work

In this section, we review the existing bioKGQA datasets, highlighting the need for new resources. We also discuss previous work on generating questions based on triples and the metrics used to evaluate the quality of these generated questions.

2.1 Existing dataset

Two major problems of existing bioKGQA datasets are, that they are small in size and that they are built upon different KGs. Details are listed in Table 1. Four existing public datasets BgeeQA [34], OMA QA [34], CORDIS-QA [34] and Task 2 of QALD-4 [36] are dependent on KGs of various sub-domains of biomedicine. For instance, Bgee [3] contains information about genes and in which parts of the body (anatomical entity) a gene is expressed or absent, while DrugBank [18] is a pharmaceutical database. Theoretically, those KGs could be mapped and grouped into a bigger KG (by ontology or ID mapping) so that it serves as the underlying KG for all KGQA datasets, in order to fully utilize the training data and to enhance model generalizability. Yet, no work has been done in this direction.

2.2 BioKG

There is a growing focus on constructing large-scale biomedical KGs by integrating resources, like BioKG [43], Hetionet [15], OREGANO [5], and PrimeKG [8], among others. Of these, PrimeKG stands out as one of the largest open-source biomedical knowledge graphs, incorporating the most widely used datasets. Unlike BioKG, Hetionet, and OREGANO, PrimeKG includes more up-to-date resources such as Bgee[3], Drug Central[37], and Uberon,[3] making it the most diverse dataset available to date.

The original artifacts of the PrimeKG project [8] are publicly available.[4] They comprise the collection of build and preprocessing scripts to compile the main PrimeKG dataset from the respective sources, and the final data files for download. The collected and integrated information stems from a variety of prominent sources for biomedical data, namely Bgee,[5] Comparative Toxicogenomics Database(CTD),[6] DisGeNET,[7]DrugBank,[8] Drug Central,[9] Entrez Gene,[10] Gene Ontology (GO),[11] Human Phenotype Ontology (HPO),[12] Mondo Disease Ontology,[13] Reactome,[14] SIDER,[15] Uberon, and UMLS.[16]. The data is often in tabular form, except for the ontologies mentioned. The PrimeKG build scripts then generate an integrated view on the input sources with the core abstraction of having *nodes*, i.e. resources with certain properties and provenance information, and *edges*, which are typed relations between the nodes. This data constitutes the main PrimeKG, yet, it is provided in CSV format, not following the LinkedData principles.[17]

2.3 Triple-to-Question Generation

Most work in triple-to-question generation follows the supervised scheme of fitting and inferring, which means they fine-tune or pretrain a model based on a large-scale dataset and evaluate the trained model on the test set. GAIN [33] fine-tunes a T5 model [29], to convert two node triples from freebase to natural questions. Fock (2022) [12] fits triples and quadruples, extracted from a temporal KG, YAGO11k (a subset of YAGO3 [23]) into pre-defined question templates. JointGT [17] adds structural information of the input triple to the transformer layer and separates text generation into three sub-tasks to further pre-train the transformer models. Han and Gardent (2023) [13] designed a multitask model capable of generating questions from both textual and graphical inputs. They validated the generated questions by testing whether the corresponding answers from open-domain QA models based on the questions match the gold standard answer. Han, Ferreira, and Gardent (2022) [14] pretrained BART [20] for the triple-to-question task, incorporating two additional pieces of information: question type and property information from the underlying knowledge graph. Kumar et al. (2019) [19] feed the textualized graph to an encoder and decoder-based transformer with embedded answers and difficulty estimation to generate complex questions. Cheng et al. (2021) [10] fine-tune GPT-2 on a self-constructive dataset for guiding the model to rewrite the simple questions into difficult questions.

Note that Rangel et al. (2024) [30] utilize an automatic process to generate a large-scale biomedical KGQA dataset over Bgee. Yet, it is neither clear how the corresponding questions are generated, nor could we find the dataset under the given URL in their brief report.

All these supervised learning approaches use large training data and, thus, are not applicable in our case, where training data is hard to obtain due to high cost. Additionally, there are no existing triple-to-questions models in the biomedical domain and the available bioKGQA datasets are severely undersized for supervised training. Therefore, we decide to explore the power of few-shot learning using PLMs to synthesize a sufficiently large-scale dataset that is anchored in explicit domain facts.

2.4 Evaluation Metrics

There are two assessment strategies for examining quality and suitability: automatic evaluation and human evaluation [26]. Automatic evaluation metrics can be categorized into n-gram metrics, task-specific metrics, and information extraction metrics [26]. 15 metrics are adopted in triple-to-question research, with the top three being BLEU [27], METEOR [2] and ROUGE [21], which are established

[2] https://github.com/xixi019/primeKGQG
[3] https://github.com/obophenotype/uberon
[4] https://zitniklab.hms.harvard.edu/projects/PrimeKG/
[5] https://www.bgee.org/
[6] https://ctdbase.org
[7] https://www.disgenet.org
[8] https://www.drugbank.com/
[9] https://drugcentral.org/
[10] https://www.ncbi.nlm.nih.gov/gene
[11] https://geneontology.org/
[12] https://hpo.jax.org/app/
[13] https://mondo.monarchinitiative.org/
[14] https://reactome.org
[15] http://sideeffects.embl.de/
[16] https://www.nlm.nih.gov/research/umls/index.html
[17] https://www.w3.org/DesignIssues/LinkedData.html

n-gram-based metrics for evaluating text generation quality. Thus, we will use these three to evaluate our approach.

As for task-specific metrics, embeddings or PLM-based metrics are used for enhancing the semantic alignment between text and the reference [25]. This is compliant with our use case since we want semantically aligned and linguistically varied question-answer pairs for the generalization capacity of the QA systems. Therefore, we also adopt BERTScore [42] and BLEURT [32] in the evaluation, which are the popular metrics under this category. Note that BLEURT is a learned metric that measures both fluency and the correspondence of the generated question to the reference in terms of the semantic meaning. It is a BERT model pre-trained on a synthetic sentence pair dataset and then fine-tuned based on public human ratings. The range of BLEURT is between -2 and around 1. The closer the score is to 1, the better the quality of the prediction is.

The information extraction metrics focus on content selection of the systems, often when multiple records are used as prediction sequences. Most of the question generation datasets do not contain multiple references. Therefore, this strategy is discarded.

Human evaluation is usually included since it is more precise in terms of semantic coherence, mismatch of the numerical values, and complexity. However, there are no established or unified standards for (costly) manual evaluation and different studies use various wording for a diverse range of aspects. According to a review paper in natural language generation [26], Fluency, Grammaticality, Correctness, Adequacy, Coherence, Faithfulness, Naturalness, Conciseness, and Similarity are the top 10 indexes used in NLG publications, with fluency being the most used standard. Based on those metrics, we conclude three with consideration to our dataset evaluation: Consistency, Grammaticality and Coverage. This is explained in Section 4.4.

3 Method

Based on PrimeKG, we aim to facilitate a generalizable approach for generating comprehensive KGQA datasets. Additionally, we aim to address energy efficiency concerns in the age of PLMs, which have significant requirements for training resources, leading to considerable carbon dioxide emissions. To achieve these goals, we propose a training-free and knowledge graph-independent method. On top, this method can be easily adapted to any knowledge graph of the user's choice, enhancing its flexibility and usability.

An illustration of our pipeline is shown in Figure 1. We first convert PrimeKG to an RDF KG which can be accessed via SPARQL. Then this SPARQL endpoint is used to extract the 2- to 4-node-subgraphs based on network motifs [24]. The subgraphs/triples are then linearized, i.e. transformed from formal KG triples into sentences, as part of the input for the PLM for generating the questions. On the other hand, based on the generated triples, we design SPARQL templates which take the entity and relations from each question to form a corresponding SPARQL query. And this query is then run against the endpoint to extract correct answers. For each subgraph, we collect the generated questions, SPARQL queries, and the answers as KGQA pairs.

3.1 Building an RDF KG for PrimeKG

The main motivation for developing an RDF-based KG is to build the downstream AI tasks (e.g., question answering, search engine, etc.) on established and standardized protocols and formats and to be able to ease further integration steps with other RDF-based data sources. To represent the original PrimeKG resources, as well as the relations between them in a unique way, a generic IRI scheme was applied, with the node IDs, resp. relation type strings, becoming the local parts of the IRIs. The original PrimeKG CSV files were then translated to RDF straightforwardly with relations between resources becoming object properties, any selected additional resource features becoming literals assigned to resources via datatype properties, and the resource types are assigned by means of `rdf:type` triples. In the current version, the relation types are not translated to RDF, as this would require RDF reifications which were considered too costly in terms of the storage overhead. We filtered out any MONDO group resources from the original PrimeKG dataset as they are essentially a *collection* of actual MONDO classes, which represent a new concept without an unique identifier. These resources were removed as they do not fit our downstream processing workflow. The generated RDF triples were then loaded into a triple store to make them publicly accessible via SPARQL.[18] The number of triples in the triple store amounts to 8,580,967.

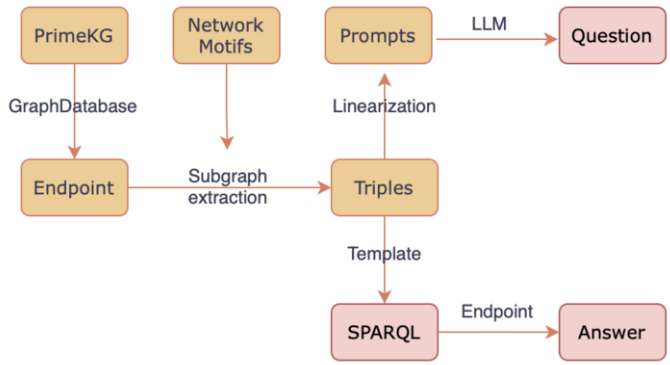

Figure 1. Our pipeline for automatic generation of PrimeKGQA. The pink blocks are the composing elements of the dataset, i.e., question in natural language, SPARQL query, and correct answer from the KG.

3.2 Subgraph Generation

To generate prompts which in turn should generate questions from triples, we need to extract subgraphs from PrimeKG. We sample triples with the numbers of nodes ranging from 2 to 4 using triple templates, namely, network motifs [24]. We focus on 2- to 4-node subgraphs for the following two reasons. (i) A comprehensive KGQA dataset typically consists of both simple and complex questions to enhance the model's ability to generalize across various complexities. The categorization of questions into *simple* and *complex* is based on the number of hops. Questions involving 1-hop patterns are considered simple, whereas those involving patterns with two or more hops are deemed complex, corresponding to 2-nodes and 3-nodes or more in the triples, respectively. (ii) We analyze a real-world biomedical QA dataset, BioASQ [35], which comprises questions with the most common number of entities ranging between two to four per question. Since the BioASQ dataset lacks named entity recognition (NER) annotations, and existing biomedical NER tools vary in granularity, we aim to compare the number of entities detected by tools designed for both fine and coarse granularities. Therefore, our approach begins by running two open-source biomedical named entity tagging tools with different granularities for the scientific and clin-

[18] http://sems-coypu-4.informatik.uni-hamburg.de:8890/sparql/

ical subdomain in biomedicine.[19,20] The results indicate that most manually curated questions contain around two to four entities. Consequently, we utilize subgraphs with two, three, and four nodes as the underlying triples.

Sampling based on network motifs allows us to include diverse and complicated reasoning paths. Network motifs are well-defined network structures used across many fields of science, such as the World Wide Web, networks from biochemistry, neurobiology, ecology, and engineering [24]. Motifs are patterns of interconnections occurring in complex networks at numbers that are significantly higher than those in randomized networks.

As for 2-node-subgraphs, there is only one pattern, as can be seen in Figure 2, since we do not consider cyclic graphs. In terms of 3-node-subgraphs, according to Milo et al. (2022) [24], there are 13 types, as shown in Figure 2. Certain types of them contain fully connected graphs. The extracted triples in this structure are, despite being meaningful subgraphs, hard to convert into a valuable question in the later step. For instance, type 5 (N3_5 in Figure 2) is a graph G, formally written as $G = \langle V, E \rangle$, with V denoting a set of nodes x_1, x_2, x_3, E denoting a set of edges $\{\langle x_1, x_2 \rangle, \langle x_3, x_2 \rangle, \langle x_3, x_1 \rangle \mid x_1 \neq x_2, x_1 \neq x_3, x_2 \neq x_3\}$. Under such a triangular structure, we observe that inevitably one edge would not be connected directly to a certain node in the graph. For instance, $\langle x_3, x_1 \rangle$ is not related to node x_1. In this case, the edge is not needed to generate a reasoning path to the question node, i.e., the information needed to generate a path is the same as contained in the fourth structure of 3-node-subgraph (denoted as N3_4 in Figure 2). And this shall hold for all the triangular-shaped subgraph patterns. Therefore, we decide to discard seven motifs, leaving only six as plausible motifs, due to the occurrence of the pattern. Also, we remove the subgraphs with duplicate edges.

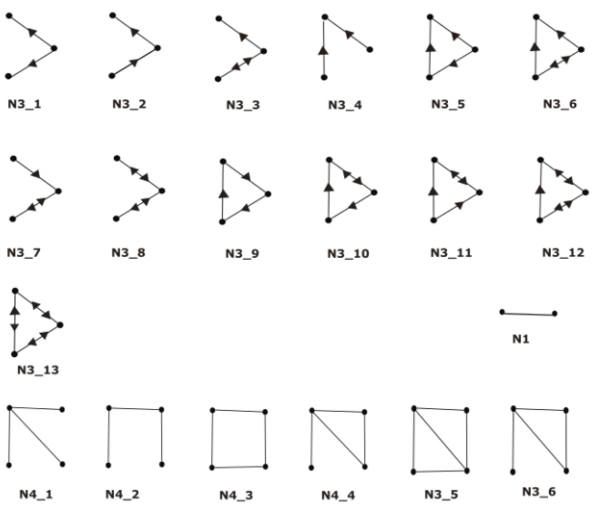

Figure 2. All types of network motifs for graphs with node numbers from two to four. N3_1 stands for "node number 3 subgraph type 1". Note that for 3-node-subgraphs, we discard N3_5, N3_6, N3_9, N3_10, N3_11, N3_12 and N3_13.

Regarding 4-node motifs, the number of possible motifs explodes, factoring the count of the possible 3-node-subgraph motifs. The basic structures of the motifs are listed in Figure 2, according to [1]. However, according to our statistics real-world QA datasets, 4-node questions are not the majority of the whole corpus, taking up 0.04% and 0.19% according to different NER tagging tools. Consequently, we only sample motifs 1 (N4_1) and 2 (N4_2) amongst the set of 4-node motifs listed in Figure 2 and abandon other types for the same reason as explained for 3-node-motifs.

Based on those motifs, we generate SPARQL queries to extract subgraphs from the PrimeKG. An example SPARQL query based on the type 1 motif for 3-node-subgraph is illustrated in Figure 3:

```
SELECT
DISTINCT ?subj ?prop1 ?obj1 ?prop2 ?obj2
WHERE {{
        ?subj ?prop1 ?obj1.
        ?subj ?prop2 ?obj2.
        }}
```

Figure 3. An example SPARQL query.

Note, we also make use of the `BIND()`, `RANDOM()` and `FILTER()` functions to extract more diverse subgraphs and exclude terminological information. An example SPARQL query is contained in our project repository on GitHub.[21] In total, we have nine motifs from which we extract 90,000 subgraphs.

3.2.1 *Answer selection*

Most of the work on triple-to-question generation or automatic KGQA construction chooses the tail entity as the answer. In a KG, a triple is formed by a relation r connecting two entities h and t. It is noted as $\langle h, r, t \rangle$, where h is the *head* entity, and t is the *tail* entity with $h, t \in V$ and $r \in E$. Hence, the generated dataset is usually homogeneous and the reasoning path is easier to learn for the model. Besides, most of the answers are only nodes, sofar the edges are ignored in the subgraphs, which are also important in composing the subgraphs and reasoning path. Therefore, we decide on an anchor answer selection strategy. First, we include both edges and nodes. Second, we choose the edges or nodes with the highest connectivity in the graph. Specifically, we locate the node with at least two outgoing/incoming edges. Next, the chosen nodes and the edges are combined into a candidate answer list. We randomly sample one entry from the candidates list to be the designated answer. Based on the subgraph and the answer, a SPARQL query is formatted with the answer masked. This SPARQL query is used for SPARQL validation.

3.2.2 *SPARQL validation*

We utilize the SPARQL queries generated in the last step for extracting the answer from the PrimeKG. When the answer from the endpoint differs from the one extracted from the subgraph, we update the answer by the answers extracted from the SPARQL endpoint.

3.3 *Question Generation*

The input of this triple-to-question task is a subgraph $G_i = \langle V_i, E_i \rangle$ and an answer $a_i \in G_i$. We evaluate various prompts and incrementally add settings such as one-shot, few-shot, and chain-of-thought (CoT) [39]. We collect the output and evaluate a few samples manually to choose the best settings. Upon initial exploration and experimentation, we narrow down the set of experiments. Our experiments

[19] https://huggingface.co/d4data/biomedical-ner-all
[20] https://huggingface.co/Clinical-AI-Apollo/Medical-NER
[21] https://github.com/xixi019/primeKGQG/blob/main/primekg/appendices.pdf

show that enclosing edges and nodes in brackets (e.g., [nasal cavity epithelium] [presents the expression] [Anterior segment of eye aplasia]) in CoT prompts yields the most faithful and grammatically fluent results. An example of our final prompt setting can be found in the project repository on GitHub. We set up three metrics to check the question quality: *Grammaticality*, *Coverage*, and *Consistency*. They are explained in Section 4.4. We test ChatGPT,[22] Mistral,[23] [16] and a LLaMA-based medicine-PLM (med-PLM) [9].[24] We keep the initial prompt as the baseline and the best group of prompt settings for generating the actual dataset.

4 Evaluation

We evaluate our model and the generated dataset using both automatic and manual metrics. We use Mistral [16] without few-shot samples, bracket, and CoT in the prompts as the baseline. Similar to the past work on question generation, and due to a lack of testing bioKGQA datasets, we examine the model on KGQA datasets in the encyclopedic domain. For manual evaluation, we sample from the generated dataset and ask three experts (two biologists and one IT expert) to annotate the samples. The sample size is 90 (10 samples from each type of network motif) due to the high annotation costs.

4.1 Dataset Description

We use SQB [40], LC-QuAD [11], and WebquestionSP (WebQSP) [41] as the testing dataset since they cover simple and complex triples. Each dataset includes a natural language question, an answer, and a query based on Freebase or DBpedia. The questions from SQB and LC-QuAD are created by filling the entity and relations in question templates, while WebQSP is generated by annotating the natural language questions against a KG.

To obtain the input subgraph, we first process the SPARQL queries or the inference path (i.e., reasoning path) in the dataset, including the removal of namespace prefixes, converting IRIs to corresponding entity names, linearizing the relation representation, etc. Detailed statistics of the datasets can be accessed in Table 2.

Table 2. Statistics of the evaluation. *Simple* and *Complex* stand for simple and complex questions in the dataset. *Paraphrase* indicates whether the edges and nodes in the triple are replaced by the synonyms in the generated question, which makes it harder for the model to generate a similar question based on n-gram metrics. We use the test/validation sets for evaluation.

	Simple	Complex	Eval_size	Rephrase	Template
SQB	✓	✗	21,483	✓	✓
LC-QuAD	✓	✓	2,000	✓	✓
WebQSP	✓	✓	1,639	✓	✗

4.2 Automatic Evaluation

Our methods are compared to GAIN [33], which is tested on the same datasets (SQB, LC-QuAD, and WebQSP). We intended to compare our approach to DiffQG [4], however, we were unable to obtain the necessary resources and models from the authors. The evaluation results show that our method significantly improves upon the baseline across all three datasets. The settings used in our best-performing model outperform GAIN by a large margin on most evaluation metrics across almost all datasets, except for SQB. This difference can primarily be attributed to SQB being a dataset comprised solely of 2-node triple-to-question pairs, with shorter question text lengths. In contrast, our models tend to utilize all information from the triple and generate longer questions that are semantically more faithful to the original questions. For datasets with more complex questions (LC-QuAD and WebQuestionsSP), our models demonstrate better performance. This would be further explained in Section 4.5.

[22] https://platform.openai.com/docs/models/gpt-3-5-turbo
[23] https://huggingface.co/mistralai/Mixtral-8x7B-Instruct-v0.1
[24] https://huggingface.co/AdaptLLM/medicine-LLM

Table 3. Evaluation result of different methods on SQB

Result	BLEU	ROUGE	METEOR	BS	BLEURT
GAIN	**0.3060**	**0.5927**	**0.5361**	**0.8709**	**-0.2934**
med-PLM.	0.0750	0.4507	0.4744	0.8208	-0.4904
Mistral	0.0520	0.4020	0.4502	0.8140	-0.4423
baseline	0.0198	0.2501	0.3340	0.7674	-0.9215

Table 4. Evaluation result of different methods on WebquestionSP

Result	BLEU	ROUGE	METEOR	BS	BLEURT
GAIN	**0.0585**	**0.4790**	0.4110	0.8320	-0.3213
med-PLM	0.0487	0.4471	**0.4929**	**0.8785**	-0.1675
Mistral	0.0151	0.3710	0.4528	0.8460	**1.0370**
baseline	0.0108	0.3046	0.4201	0.8075	-0.5606

Automatic Metrics As mentioned in Section 2.4 We utilize both n-gram and PLM-based metrics. n-gram based metrics include BLEU, METEOR, and ROUGE. As for PLM-based metrics, we utilize BERTSCORE(BS) and BLEURT.

4.3 Automatic Evaluation Result and Analysis

The performance of GAIN and the performance of our methods over three different PLMs are listed in Table 3, Table 4 and Table 5. The best performances are marked in bold. We also explain why on SQB our methods are worse than GAIN.

Note that for the dataset SQB, we turn the relational facts from Freebase into a triple format to include structural and domain knowledge, which shows improvement in several metrics than non-relational included counterparts.

Overall, med-PLM has the best performance overall metrics across different datasets. On SQB, med-PLM has a similar score compared to the best GAIN model. In terms of WebquestionSP, med-PLM has the highest score on METEOR and BS, with a small difference (less than 0.15) on BLEU, ROUGE, and BLEURT. As for LC-QuAD, med-PLM performs the best across all metrics with a big margin.

Upon n-gram-based metrics, for all the evaluated models there is still room for improvement. This might be due to the paraphrasing applied in the original dataset generation process. For instance, in SQB, the question *"What is a hong kong netflix film?"* is related to triple: *"hong kong"* (head entity), *"media_common.netflix_genre.titles"* (relation), *"Saviour of the Soul"* (tail entity and answer). Note that this pair is problematic since the tail entity is not the only node corresponding to the reasoning path.[25] To predict the mention of *"film"* from the relation "media_common.netflix_genre.titles" seems to be a daunting task for PLMs in general, since different subchunks of the relation can all be suitable for generating such a question. The predictions of the PLM for this triple vary from:

[25] There are multiple entities connected to the entity *"Hong Kong"* by the relation *"media_common.netflix_genre.titles"* in Freebase. The answer should be validated by running SPARQL queries which correspond to the reasoning path and extract the connected tail entities. This problem is fixed in our dataset creation process by the SPARQL validation step.

Table 5. Evaluation result of different methods on LC-QuAD

Result	BLEU	ROUGE	METEOR	BS	BLEURT
GAIN	0.0692	0.3575	0.2396	0.8008	-0.7671
med-PLM	**0.1649**	**0.4151**	**0.4314**	**0.8399**	**-0.4550**
Mistral	0.0822	0.3440	0.3651	0.8249	-0.6351
baseline	0.0590	0.2756	0.3309	0.7601	-0.8353

- *"what is the title of hong kong"* (GAIN)
- *"Which Hong Kong action film from the 1990s was particularly popular and went by the name Saviour of the Soul?"* (baseline)
- *"What is an example of a popular Hong Kong TVB drama series from the 1990s?"* (Mistral)
- *"What is the title of the movie that is available on Netflix and is set in Hong Kong?"* (med-PLM)
- *"what is a hong kong netflix film?"* (reference)

As humans, we can discern that med-PLM preserves the closest semantic meaning to the original sentence. However, because n-gram-based metrics assess similarity on the string-based gram level and disregard contextual meaning, the scores are relatively low.

This example also explains why our models are worse than GAIN on SQB. From our observation of the generated questions, the PLMs we use tend to produce lengthy questions, which can contribute to low scores on BLEU, METEOR, and ROUGE, since on SQB the question spans are relatively short. Meanwhile, GAIN is optimized for SQB, resulting in shorter generated text. Besides, GAIN effectively learns the template after the fine-tuning, resulting in better performance on SQB. Mistral and med-PLM, on the other hand, have not been specifically optimized for simple or complex questions and strive to retain all the information provided to preserve semantics. Consequently, they achieve relatively low scores across different metrics for SQB, as the reference questions in SQB have shorter spans. However, for LC-QuAD and WebQuestionsSP, which contain both simple and complex triples and reference questions with longer text spans, our models have better scores.

4.4 Manual Metrics

As discussed in the related work, we have scrutinized the existing manual evaluation metrics and identified key indices that are pertinent to the demands of a comprehensive KGQA dataset. Specifically, we find that Consistency, Grammaticality, and Coverage are essential for nurturing high-performing KGQA systems.

Consistency pertains to the fidelity of the generated question with respect to the provided answer. Annotators are tasked with assessing the alignment between questions and the designated answer node/edge extracted from the SPARQL endpoint. This evaluation criterion is pivotal for determining whether the generated question can be effectively answered and accurately reflects real entities or relationships within the original subgraph. For example, given the subgraph with the head entity *"muscle organ"*, relation *"is associated with"*, and tail entity *"esophagus carcinoma in situ"* (the answer), a question such as *"What are the possible associations of muscle organ with esophagus carcinoma in situ?"* would be deemed incorrect, as it focuses on the relation rather than the tail entity. A preferred formulation would be *"What are the possible associations of the muscle organ?"* which directly targets the tail entity.

Grammaticality assesses whether the question adheres to linguistic correctness in terms of vocabulary, grammar, and structure. This criterion is crucial for ensuring that questions are understandable and interpretable by domain experts.

Table 6. Aggregated annotation result on the sample question pair.

	Grammar	Coverage	Consistency
Evaluation scores	0.6111	0.7555	0.4555

Table 7. The detailed scores from different annotators

	Grammaticality	Coverage	Consistency
Biologist-1	55	8	24
Biologist-2	22	33	41
IT Expert	55	61	16

Coverage evaluates the fidelity of the generated question to the underlying subgraph or reasoning path. This aspect is vital for our KGQA dataset, as many KGQA systems, whether based on information retrieval or semantic parsing, heavily rely on the alignment between natural language questions and reasoning paths. For instance, a 2-hop subgraph should not be associated with a 1-hop question.

These standards are also highlighted as the weak points of current PLMs in the sense that the hallucinated generation of the model would be detected as false. **Model hallucination** is a focus of evaluation on the generated text by big models nowadays. In the context of triple-to-question, hallucination refers to the generation of content that lacks fidelity or is not supported by the source data provided. In this work, hallucination can be seen as a divergence of the generated questions to the input: i.e., the corresponding subgraph and the answer, which are the consistency and coverage in our manual metrics.

4.5 Manual Evaluation Result and Analysis

We ask three English-proficient annotators to independently provide feedback on each generated question based on the following three criteria, including two biomedical experts and one IT expert. The unannotated sample can be found in our project repository on GitHub. We aggregate the result and list it in Table 6. A detailed score from the annotators is also illustrated in Table 7.

As can be seen in Table 6, we have relatively high scores for Grammar and Coverage, while Consistency appears to be lower compared to the other two metrics. This can be due to the complexity of reasoning: Questions that follow a reasoning path often require understanding complex relationships, concepts, and logical structures within the graph. In the context of KGQA, the reasoning path refers to the multiple-fact triples in the KG corresponding to capturing this complexity. Replicating it in a question generation system can be difficult even for the PLM, especially for open-ended or higher-order thinking questions. Reasoning has always been proven challenging for PLMs and they're not optimized for this special task.

While Grammaticality and Coverage seem relatively satisfactory, efforts could be directed towards enhancing Consistency, possibly through refining the generation process or providing more context for the generated content or some post-editing/filtering techniques.

4.6 Inter-Annotator Agreement

We utilize Fleiss' Kappa as the metric for checking the reliability, i.e., coefficients of agreement among our annotators. The κ score for each metric is listed in Table 8.

The measured agreement is rather low, showing a disparity between annotators. Biologist-1 and the IT Expert have given higher scores for Grammaticality compared to Biologist-2. There's some

Table 8. κ scores for the Grammaticality, Coverage and Consistency metrics

	Grammaticality	Coverage	Consistency
κ	0.12	0.17	0.15

Table 9. The distribution of questions based on the number of nodes in their corresponding subgraphs. Also, the total number of relations (# rel.) and entities (# ent.) are listed.

	2-node	3-node	4-node	# q-a pairs	# rel.	# ent.
Train	5,769	34,118	11,333	51,220	131,775	263,792
Test	1,955	11,272	3,847	17,074	44,035	87,786
Val.	2,008	11,276	3,790	17,074	43,932	87,840

disparity between annotators, especially evident in the Grammaticality and Consistency scores.

The disparity can be attributed to external reasons. This can be due to the inexperience of the biologist experts hired who have limited knowledge in annotating an NLP dataset and with KGs. On the other hand, the IT Expert gives a Grammaticality score similar to Biologist-1 and a different score in Consistency. This is mostly due to the lack of domain knowledge in the biomedicine domain to match the concept in the question and answer. Nonetheless, we decided to base our quality analyses on the majority vote. On this basis, we did not deem it necessary to remove any examples for a lack of quality.

5 Generated Dataset

In total we have 85,368 question-answer pairs, since we filter out MONDO group resources from PrimeKG, as explained in Section 3.1. The generated dataset is partitioned into *train*, *test*, and *validation* set with a ratio of 6:2:2.

5.1 Statistics

The numbers of 2-node, 3-node, and 4-node based questions, question-answer pairs, relations, and entities in each separation are exhibited in Table 9.

The majority of questions in each subset has 3 nodes, followed by 2-node and 4-node questions. As the number of nodes in the subgraph increases, the number of questions decreases, which is expected as subgraphs with more nodes are likely to be less common or more complex. This also aligns with the analysis from the existing biomedical QA datasets. The distribution of questions across different node counts is consistent across subsets, indicating that the dataset is well-balanced in terms of subgraph complexity across training, testing, and validation sets. This will ensure representative sampling during different stages of developing a model, such as training and testing.

6 Ethical Statement and Acknowledgement

All subjects gave their informed consent for inclusion before they participated in the study. This project was supported by the Ministry of Research and Education within the SifoLIFE project RESCUE-MATE (project number 13N16836), and by the Federal Ministry for Economic Affairs and Climate Action of Germany in the project CoyPu (project number 01MK21007G). We utilized two NVIDIA RTX A5000 graphics cards with 24GB of RAM, kindly provided by the NVIDIA Academic Hardware Grant Program.

7 Conclusion and Future Work

In this paper, we introduced a novel approach for addressing the challenge of generating high-quality question-answer pairs for BioKGQA systems. Leveraging PLMs and the PrimeKG, we devised a methodology to automatically construct a large-scale BioKGQA dataset. Our approach resulted in the creation of PrimeKGQA, a benchmarking resource comprising 83999 question-answer pairs alongside their corresponding SPARQL queries. This is so far the largest dataset in BioKGQA and is 1000 factors more than the second biggest dataset in this domain. Through a rigorous evaluation process involving both automatic metrics and manual annotations by domain experts, we established novel standards tailored specifically for assessing the linguistic correctness and semantic faithfulness of the generated questions. This ensures that PrimeKGQA serves as a reliable benchmark for evaluating the performance of KGQA systems in the biomedical domain. On top, the dataset generation framework is training-free, adaptable to other domains, and supports evolving KGs, making it suitable for automatic dataset generation across various fields for automatic dataset generation.

While our work represents a significant step forward in addressing the dearth of large-scale BioKGQA datasets, several avenues for future research and improvement remain: Refined Question Generation, i.e., investigating methodologies to refine the generated questions, enabling examination of the output in desired dimensions and facilitating post-editing strategies for error correction. Application-oriented Evaluation, i.e., conducting current KGQA systems using PrimeKGQA to assess their effectiveness in supporting real-world biomedical tasks, such as clinical decision support and drug discovery.

References

[1] A. Al-Thaedan and M. Carvalho. Online estimation of motif distribution in dynamic networks. In *2019 IEEE 9th Annual Computing and Communication Workshop and Conference (CCWC)*, pages 0758–0764. IEEE, 2019.

[2] S. Banerjee and A. Lavie. Meteor: An automatic metric for mt evaluation with improved correlation with human judgments. In *Proceedings of the acl workshop on intrinsic and extrinsic evaluation measures for machine translation and/or summarization*, pages 65–72, 2005.

[3] F. B. Bastian, J. Roux, A. Niknejad, A. Comte, S. Fonseca Costa, T. M. de Farias, S. Moretti, G. Parmentier, V. R. de Laval, M. Rosikiewicz, J. Wollbrett, A. Echchiki, A. Escoriza, W. H. Gharib, M. Gonzales-Porta, Y. Jarosz, B. Laurenczy, P. Moret, E. Person, P. Roelli, K. Sanjeev, M. Seppey, and M. Robinson-Rechavi. The Bgee suite: integrated curated expression atlas and comparative transcriptomics in animals. *Nucleic Acids Research*, 49(D1):D831–D847, 10 2020. ISSN 0305-1048. doi: 10.1093/nar/gkaa793.

[4] S. Bi, J. Liu, Z. Miao, and Q. Min. Difficulty-controllable question generation over knowledge graphs: A counterfactual reasoning approach. *Information Processing & Management*, 61(4):103721, 2024. ISSN 0306-4573. doi: https://doi.org/10.1016/j.ipm.2024.103721.

[5] M. Boudin, G. Diallo, M. Drancé, and F. Mougin. The OREGANO knowledge graph for computational drug repurposing. *Scientific data*, 10(1):871, 2023.

[6] A. J. Brookes and P. N. Robinson. Human genotype–phenotype databases: aims, challenges and opportunities. *Nature Reviews Genetics*, 16(12):702–715, 2015.

[7] T. Cao, M. T. Law, and S. Fidler. A theoretical analysis of the number of shots in few-shot learning. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*. OpenReview.net, 2020.

[8] P. Chandak, K. Huang, and M. Zitnik. Building a knowledge graph to enable precision medicine. *Scientific Data*, 10(1):67, 2023.

[9] D. Cheng, S. Huang, and F. Wei. Adapting large language models via reading comprehension. *arXiv preprint arXiv:2309.09530*, 2023.

[10] Y. Cheng, S. Li, B. Liu, R. Zhao, S. Li, C. Lin, and Y. Zheng. Guiding the growth: Difficulty-controllable question generation through step-by-step rewriting. In C. Zong, F. Xia, W. Li, and R. Navigli, editors,

Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 1: Long Papers), pages 5968–5978, Online, Aug. 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.acl-long.465.

[11] M. Dubey, D. Banerjee, A. Abdelkawi, and J. Lehmann. LC-QuAD 2.0: A large dataset for complex question answering over Wikidata and DBpedia. In C. Ghidini, O. Hartig, M. Maleshkova, V. Svátek, I. F. Cruz, A. Hogan, J. Song, M. Lefrançois, and F. Gandon, editors, *The Semantic Web - ISWC 2019 - 18th International Semantic Web Conference, Auckland, New Zealand, October 26-30, 2019, Proceedings, Part II*, volume 11779 of *Lecture Notes in Computer Science*, pages 69–78. Springer, 2019. doi: 10.1007/978-3-030-30796-7_5.

[12] H. A. Fock. Knowledge graph expansion using Question Answering by leveraging pre-trained language models. Master's thesis, Utrecht University, 2022.

[13] K. Han and C. Gardent. Generating and answering simple and complex questions from text and from knowledge graphs. In *Proceedings of the 13th International Joint Conference on Natural Language Processing and the 3rd Conference of the Asia-Pacific Chapter of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 285–304, 2023.

[14] K. Han, T. C. Ferreira, and C. Gardent. Generating questions from Wikidata triples. In *13th Edition of its Language Resources and Evaluation Conference*, 2022.

[15] D. S. Himmelstein, A. Lizee, C. Hessler, L. Brueggeman, S. L. Chen, D. Hadley, A. Green, P. Khankhanian, and S. E. Baranzini. Systematic integration of biomedical knowledge prioritizes drugs for repurposing. *Elife*, 6:e26726, 2017.

[16] A. Q. Jiang, A. Sablayrolles, A. Mensch, C. Bamford, D. S. Chaplot, D. d. l. Casas, F. Bressand, G. Lengyel, G. Lample, L. Saulnier, et al. Mistral 7b. *arXiv preprint arXiv:2310.06825*, 2023.

[17] P. Ke, H. Ji, Y. Ran, X. Cui, L. Wang, L. Song, X. Zhu, and M. Huang. JointGT: Graph-text joint representation learning for text generation from knowledge graphs. In C. Zong, F. Xia, W. Li, and R. Navigli, editors, *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 2526–2538, Online, Aug. 2021. Association for Computational Linguistics. doi: 10.18653/v1/2021.findings-acl.223.

[18] C. Knox, M. Wilson, C. M. Klinger, M. Franklin, E. Oler, A. Wilson, A. Pon, J. Cox, N. E. Chin, S. A. Strawbridge, et al. DrugBank 6.0: the DrugBank knowledgebase for 2024. *Nucleic Acids Research*, 52(D1): D1265–D1275, 2024.

[19] V. Kumar, Y. Hua, G. Ramakrishnan, G. Qi, L. Gao, and Y.-F. Li. Difficulty-controllable multi-hop question generation from knowledge graphs. In *The Semantic Web–ISWC 2019: 18th International Semantic Web Conference, Auckland, New Zealand, October 26–30, 2019, Proceedings, Part I 18*, pages 382–398. Springer, 2019.

[20] M. Lewis, Y. Liu, N. Goyal, M. Ghazvininejad, A. Mohamed, O. Levy, V. Stoyanov, and L. Zettlemoyer. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. *arXiv preprint arXiv:1910.13461*, 2019.

[21] C.-Y. Lin. ROUGE: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81, 2004.

[22] X. Lin, S. Ma, J. Shan, X. Zhang, S. X. Hu, T. Guo, S. Z. Li, and K. Yu. BioKGBench: A knowledge graph checking benchmark of AI agent for biomedical science. *arXiv preprint arXiv:2407.00466*, 2024.

[23] F. Mahdisoltani, J. Biega, and F. M. Suchanek. YAGO3: A knowledge base from multilingual Wikipedias. In *Seventh Biennial Conference on Innovative Data Systems Research, CIDR 2015, Asilomar, CA, USA, January 4-7, 2015, Online Proceedings*. www.cidrdb.org, 2015.

[24] R. Milo, S. Shen-Orr, S. Itzkovitz, N. Kashtan, D. Chklovskii, and U. Alon. Network motifs: Simple building blocks of complex networks. *Science*, 298(5594):824–827, 2002. doi: 10.1126/science.298.5594.824.

[25] R. Nedelchev, J. Lehmann, and R. Usbeck. Language model transformers as evaluators for open-domain dialogues. In D. Scott, N. Bel, and C. Zong, editors, *Proceedings of the 28th International Conference on Computational Linguistics, COLING 2020, Barcelona, Spain (Online), December 8-13, 2020*, pages 6797–6808. International Committee on Computational Linguistics, 2020. doi: 10.18653/V1/2020.COLING-MAIN.599.

[26] C. C. Osuji, T. C. Ferreira, and B. Davis. A systematic review of data-to-text NLG. *arXiv preprint arXiv:2402.08496*, 2024.

[27] K. Papineni, S. Roukos, T. Ward, and W.-J. Zhu. BLEU: a method for automatic evaluation of machine translation. In *Proceedings of the 40th annual meeting of the Association for Computational Linguistics*, pages 311–318, 2002.

[28] J. Piñero, À. Bravo, N. Queralt-Rosinach, A. Gutiérrez-Sacristán, J. Deu-Pons, E. Centeno, J. García-García, F. Sanz, and L. I. Furlong. DisGeNET: a comprehensive platform integrating information on human disease-associated genes and variants. *Nucleic acids research*, page gkw943, 2016.

[29] C. Raffel, N. Shazeer, A. Roberts, K. Lee, S. Narang, M. Matena, Y. Zhou, W. Li, and P. J. Liu. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of machine learning research*, 21(140):1–67, 2020.

[30] J. C. Rangel, T. M. de Farias, A. C. Sima, and N. Kobayashi. SPARQL generation: an analysis on fine-tuning OpenLLaMA for Question Answering over a life science knowledge graph, 2024.

[31] L. M. Schriml, C. Arze, S. Nadendla, Y.-W. W. Chang, M. Mazaitis, V. Felix, G. Feng, and W. A. Kibbe. Disease Ontology: a backbone for disease semantic integration. *Nucleic acids research*, 40(D1):D940–D946, 2012.

[32] T. Sellam, D. Das, and A. Parikh. BLEURT: Learning robust metrics for text generation. In D. Jurafsky, J. Chai, N. Schluter, and J. Tetreault, editors, *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7881–7892, Online, July 2020. Association for Computational Linguistics. doi: 10.18653/v1/2020.acl-main.704.

[33] Y. Shu and Z. Yu. Distribution shifts are bottlenecks: Extensive evaluation for grounding language models to knowledge bases. In N. Falk, S. Papi, and M. Zhang, editors, *Proceedings of the 18th Conference of the European Chapter of the Association for Computational Linguistics: Student Research Workshop*, pages 71–88, St. Julian's, Malta, Mar. 2024. Association for Computational Linguistics.

[34] A. C. Sima, T. Mendes de Farias, M. Anisimova, C. Dessimoz, M. Robinson-Rechavi, E. Zbinden, and K. Stockinger. Bio-SODA: Enabling natural language question answering over knowledge graphs without training data. In *Proceedings of the 33rd International Conference on Scientific and Statistical Database Management*, SSDBM '21, page 61–72, New York, NY, USA, 2021. Association for Computing Machinery. ISBN 9781450384131. doi: 10.1145/3468791.3469119.

[35] G. Tsatsaronis, M. Schroeder, G. Paliouras, Y. Almirantis, I. Androutsopoulos, E. Gaussier, P. Gallinari, T. Artieres, M. R. Alvers, M. Zschunke, et al. BioASQ: A challenge on large-scale biomedical semantic indexing and Question Answering. In *2012 AAAI Fall Symposium Series*, 2012.

[36] C. Unger, C. Forascu, V. Lopez, A.-C. N. Ngomo, E. Cabrio, P. Cimiano, and S. Walter. Question Answering over Linked Data (QALD-4). In *Working notes for CLEF 2014 conference*, 2014.

[37] O. Ursu, J. Holmes, J. Knockel, C. Bologa, J. J. Yang, S. L. Mathias, S. J. Nelson, and T. I. Oprea. DrugCentral: online drug compendium. *Nucleic Acids Res.*, 45(Database-Issue):D932–D939, 2017. doi: 10.1093/NAR/GKW993.

[38] Y. Wang, Q. Yao, J. T. Kwok, and L. M. Ni. Generalizing from a few examples: A survey on few-shot learning. *ACM Comput. Surv.*, 53(3), jun 2020. ISSN 0360-0300. doi: 10.1145/3386252.

[39] J. Wei, X. Wang, D. Schuurmans, M. Bosma, F. Xia, E. Chi, Q. V. Le, D. Zhou, et al. Chain-of-thought prompting elicits reasoning in large language models. *Advances in neural information processing systems*, 35:24824–24837, 2022.

[40] P. Wu, S. Huang, R. Weng, Z. Zheng, J. Zhang, X. Yan, and J. Chen. Learning representation mapping for relation detection in knowledge base question answering. In A. Korhonen, D. Traum, and L. Màrquez, editors, *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 6130–6139, Florence, Italy, July 2019. Association for Computational Linguistics. doi: 10.18653/v1/P19-1616.

[41] W. Yih, M. Richardson, C. Meek, M. Chang, and J. Suh. The value of semantic parse labeling for knowledge base question answering. In *Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics, ACL 2016, August 7-12, 2016, Berlin, Germany, Volume 2: Short Papers*. The Association for Computer Linguistics, 2016. doi: 10.18653/V1/P16-2033.

[42] T. Zhang, V. Kishore, F. Wu, K. Q. Weinberger, and Y. Artzi. BERTScore: Evaluating text generation with BERT. In *8th International Conference on Learning Representations, ICLR 2020, Addis Ababa, Ethiopia, April 26-30, 2020*, 2020.

[43] Y. Zhang, X. Sui, F. Pan, K. Yu, K. Li, S. Tian, A. Erdengasileng, Q. Han, W. Wang, J. Wang, et al. BioKG: a comprehensive, large-scale biomedical knowledge graph for AI-powered, data-driven biomedical research. *bioRxiv*, 2023.