BioTABQA: Instruction Learning for Biomedical Table Question Answering

Man Luo, Sharad Saxena, Swaroop Mishra, Mihir Parmar and Chitta Baral

Arizona State University, Tempe, Arizona, 85281, United State

Abstract

Table Question Answering (TQA) is an important but under-explored task. Most of the existing QA datasets are in unstructured text format and only few of them use tables as the context. To the best of our knowledge, none of TQA datasets exist in the biomedical domain where tables are frequently used to present information. In this paper, we first curate a table question answering dataset, BioTabQA, using 22 templates and the context from a biomedical textbook on differential diagnosis. BioTabQA can not only be used to teach a model how to answer questions from tables but also evaluate how a model generalizes to unseen questions, an important scenario for biomedical applications. To achieve the generalization evaluation, we divide the templates into 17 training and 5 cross-task evaluations. Then, we develop two baselines using single and multi-tasks learning on BioTabQA. Furthermore, we explore instructional learning, a recent technique showing impressive generalizing performance. Experimental results show that our instruction-tuned model outperforms single and multi task baselines on an average by $\sim 23\%$ and $\sim 6\%$ across various evaluation settings, and more importantly, instruction-tuned model outperforms baselines by $\sim 5\%$ on cross-tasks.

Keywords

Table question answering, biomedical question answering, instruction learning, prompt learning

1. Introduction

Neural language models have achieved state-of-the-art performance in popular reading comprehension (RC) tasks such as SQuAD [1, 2], DROP [3] and ROPES [4]. Unlike in popular RC where the context contains information in natural language, a significant amount of real-world information is stored in unstructured or semi-structured web tables [5]. In particular, many clinical information is provided in tabular format [6]. Past attempts have been made for TQA in the general domain Natural Language Processing (NLP) [7, 8, 9], however, this task has not been well-studied in the biomedical domain.

This work takes the first step toward studying the TQA task in biomedical domain. To this extent, we first curate a table question answering dataset, BioTabQA, using 22 templates without heavy and expensive human annotation. This dataset also serves to evaluate the generalization of a model, a well-known issue that many language models have failed even though they outperform humans in many popular benchmarks [10, 11]. Recently, instruction-learning [12, 13, 14] have improved model's performance to unseen tasks. Inspired by this, we leverage

CLEF 2022: Conference and Labs of the Evaluation Forum, September 5–8, 2022, Bologna, Italy

mluo26@asu.edu (M. Luo); ssaxen18@asu.edu (S. Saxena); smishr1@asu.edu (S. Mishra); mparmar3@asu.edu (M. Parmar); chitta@asu.edu (C. Baral)

CEUR Workshop Proceedings (CEUR-WS.org)

instruction-tuning to build a model and verify whether instruction learning also show stronger generalization on BioTabQA.

Our contribution can be summarized as: (1) to the best of our knowledge, this is the first attempt to study biomedical TQA and this is also the first attempt to incorporate instructional learning in this task, (2) we reformulate differential diagnosis as a TQA problem and introduce a new dataset BioTabQA, and (3) experimental results show that our instruction-tuned model outperforms single and multi task baselines by 23%, 6%, and outperforms multitask model by 5% in cross-task (generalization to unseen task) setting. Finally, our analysis shows that instruction is more important and useful in cross tasks compared to in-domain tasks in inference time.

2. Related Work

Table Question Answering Past attempts have been made for TQA such as TabMCQ [7], WikiTableQuestions [8], Sequential Q&A [9], Spider [15], WikiSQL [16]. These approaches can handle the large-scale tables from Wikipedia efficiently. However, these QA systems can only answer the question when a strong signal needed for identifying the type of answers is provided explicitly in the table. To overcome this limitation, TabFact [17] is proposed which enables TQA when the answer is not explicitly available in the table. However, none of the above datasets are in the biomedical domain, a domain which is not only essential in human life but also in which tables have wide applications (e.g. many biomedical information are presented by tables). There exists some table datasets in biomedical domain, such as PubTabNet [18], a medical table datasets which are widely used in information retrieval tasks. Some other datasets are designed for biomedical question answering task such as [19, 20]. Nevertheless, these are not biomedical TQA dataset, leaving the biomedical TQA as an under-explored task. This work aims to take the first step to Table question answering in biomedical domain, which is known to be different from the general domain [21, 22].

Instruction Learning Recently, the paradigm in ML/DL shifted to prompt-based learning. Liu et al. [23] provides a comprehensive survey on prompt-based methods for various tasks. Prompts enable the generalization across tasks as well as achieves considerable performance on zero-shot learning. T0 model Sanh et al. [24] shows effective performance on multi-tasking and zero-task generalization using a prompt-based approach. [12] introduced natural language instructions to improve the performance of LMs such as BART, GPT-3 for cross-task. Followed by this, FLAN [13] has been proposed which uses instructions to achieve generalization across unseen tasks. Recently, Parmar et al. [14] proposed instruction learning for biomedical multi-task. Along with that, Mishra et al. [25] shows reframing instructional prompts can boost both few-shot and zero-shot model performance. Min et al. [26] shows performance of in-context learning on a large set of training tasks. InstructGPT model is proposed which is fine-tuned with human feedback [27]. Instruction-based multi-task framework for few-shot Named Entity Recognition (NER) has been developed by Wang et al. [28]. Puri et al. [29] introduced instruction augmentation and Prasad et al. [30] introduced Gradient-free Instructional Prompt Search (GrIPS) for improving model performance. Recently, Parmar et al. [31] believe that instruction bias in existing Natural Language Understanding (NLU) datasets can impact the instruction learning, however, many

ID	Question Template	Prompt
1	I have symptom A, what disease do I have?	If symptom A is in symptom list, report corresponding disease.
2	I have symptom A and sign A, what is my diagnosis?	If symptom A is in symptom list, and sign A is in sign list, report corresponding disease.
11	The patient has symptom A, symptom B and symptom C, what disease can cause these symptoms?	If symptom A, symptom B and symptom C are in symptom list, report corresponding disease.
22	I have symptom A, symptom B, symptom C but no symptom D, what is causing this?	If symptom A, symptom B and symptom C are in symptom list, but symptom D is not in symptom list, report corresponding disease.

Table 1
Examples of four question templates for BioTabQA dataset creation and the corresponding prompts

approaches have been proposed recently using instructions to improve model performance [32, 33, 34, 35, 36]. Motivated by the effectiveness of instruction learning, in this work, we explore the potential application of instructional prompts for the biomedical TQA.

3. Task and Dataset

Task Formulation Each data point is a tuple <T, Q, A>, where T is a table, Q is a question, and A is the answer to Q in T. In particular, Q exhibits some symptoms/signs and asks about what potential disease it is, e.g., “I have joint pain and swelling on my face, what’s wrong with me”. A is the corresponding disease (or diagnosis) in T. The task is to predict A given <T, Q> as input.

Dataset Source We use the medical textbook “Differential Diagnosis in Primary Care” [37] as the source of our dataset, which contains information on how to diagnose a patient by observing their disease symptoms. This book is in the tabular format with five columns: (1) diagnosis, (2) key symptoms, (3) key signs[1], (4) background, and (5) additional information. We only use the first three columns to create the dataset. We divide the textbook into 513 tables.

Dataset Creation To create large scale training/evaluation datasets (i.e., BioTabQA) without laborious human annotation, we design a wide range of templates to semi-automate the process of dataset generation. We use key symptoms and/or key signs of a diagnosis in the question templates, and the diagnosis as the answer. In addition, we design the corresponding prompt to enable instruction learning for each template. In total, we design 22 templates. Table 1 shows four templates as an example and the corresponding prompts (all templates and prompts are given in Appendix A. Specifically, some templates have one, two, three or four symptoms/sign (e.g. ID 1, 2, 11, 22), and some have negation (e.g. ID 22). Once the templates are pre-defined, given a table, and a template, for each row, we randomly select the symptoms/signs based on the template and replace the placeholder in the template with the chosen symptoms/signs.

[1] According to JAMA Network, a symptom is a manifestation of disease appears to the patient himself, while a sign is a manifestation of the disease that the physician perceives.

Statistic	Train	IID Test	Cross Task Test
# of Samples	9,126	19,590	2,463
Question Length	240	20	16
Table Length	255	256	246
Prompt Length	18	18	14
# Tasks with 1 sym/sign	0	0	3
# Tasks with 2 sym/sign	9	9	2
# Tasks with 3 sym/sign	7	7	0
# Tasks with 4 sym/sign	1	1	0
# Tasks with negation	2	2	0

Table 2
Statistic of BioTabQA Split 1 for training (Train), in-domain testing (IID Test) and cross task testing (Cross Task Test) sets.

Three Splits in BioTabQA For experimental purposes, we created 3 training/testing/cross-task splits of data. Each split includes 17 templates for in-domain training and testing, where there are non-overlap tables for training and testing. The rest 5 templates are used for cross-task evaluation. For each split, the templates are similar to each other in the training set and less similar to the templates in the evaluation set (cross-task setting) to show the generalization capability of a model. The similarity is defined as either the same number of symptoms/signs presented in the templates or similar phrases in the templates. Table 2 shows the statistics of Split 1 and other Splits as well as the division of the Splits are given in Appendix A.

4. Experiments and Results

From our dataset, each question type (i.e., template) is considered an individual task. Hence, we have 22 different tasks in total. We design two baselines, the single-task model (STM) and the multi-tasks model (MTM). We compare the performance of the instruction-tuned model (In-MTM) with these two baselines on the in-domain test set, cross-task, and robustness [38, 39]. We use DistilBert [40] as the backbone model for all experiments. Exact Match (EM) score is used as an evaluation metric. Other experimental setup can be found in Appendix B. In the following, we describe the table linearization technique followed by our instructional multi-task learning model. We present the results and analysis at the end of this section.

4.1. Table Linearization

Since input of the language model is text, we need to linearize the table context from BioTabQA. We use a simple yet effective linearization method suggested by [17] to convert the table context into a string of text. We pre-define the format "Row 1 is: Diagnosis is _, Key symptoms are _, Key signs are _;..., Row N is Diagnosis is _, Key symptoms are X, Key signs are XX".

4.2. Instructional Multi-task Learning Model

Apart from the prompt designed for each template (see §3), one additional example is also given in the instruction. The example consists of a question and the answer without the context table due to the input length restriction of the language model. We also use special words to denote the beginning of the prompt, and question and answer. In particular, the instruction set of {Prompt: p. Question: q. Answer: a}. The input to our instruction learning model is {[CLS] Question: Q, Context: C, Instruction: I}, where [CLS] is the special token of the DistilBERT model, Q is the input question, C is the input table after linearization. As mention in §3, we create multiple templates and we term the data created by individual template as task. A single task model (STM) is trained by one task, and a multitask model (MTM) is trained by multiple tasks.

4.3. Main Results

We evaluated our proposed model In-MTM in terms of various aspects including in-domain testing, cross-task setting and robustness. All the results are presented in Table 3. In the following, we present insightful results and findings based on our experiments. The performance of MTM and In-MTM varies for different split since each split consists of different tasks.

Finding 1: Multitask Model performs better than Single-task Model From Figure 1, we can observe that MTM outperforms STM in majority cases, leading to on an average 14%, 21%, and 18% improvement on split 1, 2, and 3, respectively. Also, we observe that multi-task learning is significantly helpful on the tasks where the training data is less. Hence, we observe tasks 1, 15, and 21 which have only 667 training examples (see results in the first block of Table 4). We can see that the STM model achieve less than 0.60 EM score; while the MTM trained on split 2 achieves at least 0.85 EM score[2]. For split 1 where the MTM does not train on tasks 1, 15 and 21 tasks, it shows superior performance compared to the STM. This indicates that multi-task learning is effective in a low-resource setting for TQA. Moreover, for tasks 5 and 13 which have more than $10k$ instances, the STM can obtain a 0.90 EM score; while the MTM trained on the split 2 achieves similar performance. This finding is aligned with the literature that multitask learning model improves single task learning model [41, 42, 43]

Finding 2: Instruction further improve Multi-task learning Model We can observe from Figure 1 that In-MTM further improves the performance of the MTM, yielding on an average 6%, 6% and 5% improvement on split 1,2, and 3, respectively. These results indicate the use of instructional prompts increases the question-answering performance both consistently and significantly.

Finding 3: Multi-task learning and Instruction Learning improve generalization capacity of model For each split, we hold out 5 different tasks for the cross-task evaluation. This is similar to out-of-domain evaluation where a model has not seen such types of questions in the

[2] We compare STM with MTM only on split 2 results in this scenario because task 1, 15, and 21 are used for training in split 2.

Task ID	# Training	STM	Split 1		Split 2		Split 3	
			MTM	In-MTM	MTM	In-MTM	MTM	In-MTM
1	667	0.53	0.84	0.85	0.88	**0.91**	0.88	0.88
2	3023	0.55	0.83	0.93	0.88	**0.94**	0.87	0.93
3	3082	0.62	0.87	0.93	0.89	**0.96**	0.90	0.95
4	3170	0.64	0.80	0.92	0.88	0.93	0.87	**0.94**
5	47561	0.90	0.88	0.93	0.92	**0.95**	0.90	0.93
6	10991	0.86	0.86	0.94	0.89	**0.98**	0.91	0.96
7	3082	0.60	0.87	0.93	0.89	**0.97**	0.89	0.95
8	3082	0.60	0.87	0.92	0.89	0.92	0.89	**0.95**
9	10324	0.82	0.80	0.95	0.83	**0.96**	0.83	**0.96**
10	3082	0.63	0.87	0.93	0.89	**0.96**	0.90	0.95
11	10991	0.88	0.86	0.94	0.89	**0.97**	0.91	0.95
12	3082	0.60	0.87	0.92	0.89	**0.95**	0.89	0.94
13	10991	0.90	0.86	0.93	0.89	**0.98**	0.90	0.95
14	10991	0.71	0.86	0.93	0.89	**0.98**	0.90	0.96
15	667	0.51	0.84	0.85	0.89	**0.90**	0.87	**0.90**
16	3082	0.63	0.87	0.92	0.89	**0.96**	0.89	0.92
17	10991	0.80	0.87	0.94	0.89	**0.98**	0.90	0.95
18	3082	0.68	0.88	0.93	0.89	**0.96**	0.89	0.94
19	3082	0.60	0.87	0.93	0.89	**0.96**	0.90	0.95
20	3082	0.61	0.87	**0.91**	0.89	**0.95**	0.90	0.94
21	667	0.54	0.83	0.85	0.87	0.89	0.85	**0.90**
22	14639	0.88	0.89	0.93	0.93	**0.97**	0.92	0.94
Avg. Split 1	9127	0.72	0.86	**0.93**				
Avg. Split 2	7349	0.68	-	-	0.89	**0.95**		
Avg. Split 3	8525	0.71	-	-	-	-	0.89	**0.94**
Avg. cross Split 1	-	-	0.84	**0.88**				
Avg. cross Split 2	-	-	-	-	0.88	**0.97**	-	-
Avg. cross Split 3	-	-	-	-	-	-	0.89	**0.92**

Table 3
The EM (exact matching) scores of three Models on BioTabQA. Green denotes cross task performance. **Bold number** denotes the best performance for each task.

training time, thus the performance on the cross tasks demonstrates the generalization capacity of a model. From the results shown in Figure 2, we have two observations. First, we can see that both MTM and In-MTM show sufficient performance. In split 3, In-MTM achieves average 0.94 EM on in-domain tasks (see Figure 1) and average 0.92 EM on cross-tasks (see Figure 2), a marginal drop ($\sim 2\%$). On the same split, MTM achieves the same performance on in-domain tasks and cross-tasks. More importantly, both MTM and In-MTM achieve higher performance than the STM on every task even though the former two models do not train on these tasks. This demonstrates the benefits of multi-task learning. Second, for each Split, In-MTM achieves better performance than MTM on every cross-task. This shows that instruction learning can further improve generalization.

Task ID	STM	Split 1		Split 2	
		MTM	In-MTM	MTM	In-MTM
1	0.53	0.84	0.85	0.88	0.91
15	0.51	0.84	0.85	0.89	0.90
21	0.54	0.83	0.85	0.87	0.89
5	0.90	0.88	0.93	0.92	0.95
13	0.90	0.86	0.93	0.89	0.98

Table 4
The EM (exact matching) scores of STM, MTM and In-MTM on the low resources tasks (first block) and high resources tasks (second block).

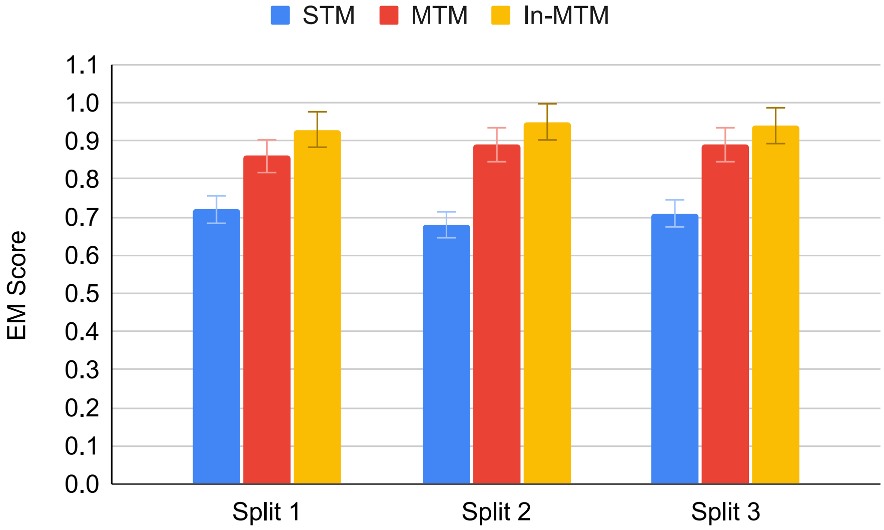

Figure 1: The average performance of three models on the in-domain testing sets of different Splits.

Finding 4: Instruction is more useful in cross tasks compared to in-domain tasks. We evaluate the In-MTM on two cross tasks and in-domain tasks with different types of instructions to analyze the change in model performance. First, we use mismatched instruction, i.e., instruction of one task for other tasks, however, it is still instructive. From Table 5, we can see that model shows similar performance with the mismatch instruction as original instruction on both in-domain and cross tasks. The reason might be that instructions for the different tasks have a set of similar words (see Appendix A), and previous studies [44, 45] have shown that the model can perform well if the instructions have similar words. Moreover, we also construct three types of meaningless instructions which does not have any linguistic meaning such as random strings (e.g., 'ashlksadkl'), random words (e.g., 'hello bye you east'), and repeated characters

Task ID	Correct	Mismatched	Repeat	Random String	Random Words
2	0.93	0.92	0.89	0.90	0.89
4	0.92	0.91	0.85	0.86	0.85
20	0.91	0.92	0.93	0.94	0.93
22	0.93	0.95	0.96	0.95	0.96

Table 5
Test the performance of In-MTM (trained on Split 1) using 4 variants of instructions) on two cross tasks (first block) and two in-domain tasks (second block).

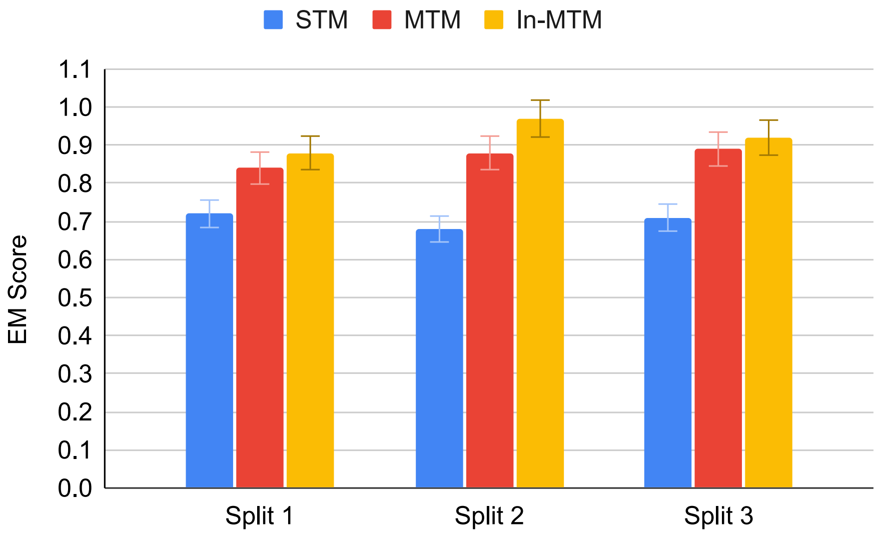

Figure 2: The average performance of three models on the cross-tasks of different Splits.

(e.g., ‘AAAAA’). These meaningless instructions hamper the performance of cross tasks more significantly than in-domain tasks. For in-domain tasks, the model is already exposed to the same type of instances at training time, but for cross-task, these unseen tasks heavily rely on the instructions [12]. In summary, instructions can be more important and helpful in cross-task settings.

5. Future Work and Conclusion

In this work, we take the first step toward studying table question answering in biomedical domain. We firstly create a dataset, BioTabQA, based on templates using a primary care textbook. We then experiment with three models on BioTabQA and find that multi-task learning is better than single task learning especially in low resource scenarios. Furthermore, the instruction learning can significantly improve the model without instructions on both in-domain as well as cross-tasks. This suggest the benefits of instruction learning on table question answering task, and explore the role of instruction learning in other general table question answering datasets is one interesting future work. The questions in current dataset are based on the formal symptoms or signs given in the textbook, which make some questions unnatural. Using more natural terms to generate the question can produce a dataset more close to real-life scenario.

References

[1] P. Rajpurkar, J. Zhang, K. Lopyrev, P. Liang, Squad: 100,000+ questions for machine comprehension of text, arXiv preprint arXiv:1606.05250 (2016).

[2] P. Rajpurkar, R. Jia, P. Liang, Know what you don't know: Unanswerable questions for squad, arXiv preprint arXiv:1806.03822 (2018).

[3] D. Dua, Y. Wang, P. Dasigi, G. Stanovsky, S. Singh, M. Gardner, Drop: A reading comprehension benchmark requiring discrete reasoning over paragraphs, arXiv preprint arXiv:1903.00161 (2019).

[4] K. Lin, O. Tafjord, P. Clark, M. Gardner, Reasoning over paragraph effects in situations, arXiv preprint arXiv:1908.05852 (2019).

[5] W. Chen, M.-W. Chang, E. Schlinger, W. Wang, W. W. Cohen, Open question answering over tables and text, arXiv preprint arXiv:2010.10439 (2020).

[6] C. G. Durbin, Effective use of tables and figures in abstracts, presentations, and papers, Respiratory care 49 (2004) 1233–1237.

[7] S. K. Jauhar, P. Turney, E. Hovy, Tables as semi-structured knowledge for question answering, in: Proceedings of the 54th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 2016, pp. 474–483.

[8] P. Pasupat, P. Liang, Compositional semantic parsing on semi-structured tables, arXiv preprint arXiv:1508.00305 (2015).

[9] M. Iyyer, W.-t. Yih, M.-W. Chang, Search-based neural structured learning for sequential question answering, in: Proceedings of the 55th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers), 2017, pp. 1821–1831.

[10] K. Eykholt, I. Evtimov, E. Fernandes, B. Li, A. Rahmati, C. Xiao, A. Prakash, T. Kohno, D. Song, Robust physical-world attacks on deep learning visual classification, in: Proceedings of the IEEE conference on computer vision and pattern recognition, 2018, pp. 1625–1634.

[11] R. Le Bras, S. Swayamdipta, C. Bhagavatula, R. Zellers, M. Peters, A. Sabharwal, Y. Choi, Adversarial filters of dataset biases, in: International Conference on Machine Learning, PMLR, 2020, pp. 1078–1088.

[12] S. Mishra, D. Khashabi, C. Baral, H. Hajishirzi, Cross-task generalization via natural language crowdsourcing instructions, ACL (2022).

[13] J. Wei, M. Bosma, V. Y. Zhao, K. Guu, A. W. Yu, B. Lester, N. Du, A. M. Dai, Q. V. Le, Finetuned language models are zero-shot learners, arXiv preprint arXiv:2109.01652 (2021).

[14] M. Parmar, S. Mishra, M. Purohit, M. Luo, M. H. Murad, C. Baral, In-BoXBART: Get Instructions into Biomedical Multi-Task Learning, NAACL 2022 Findings (2022).

[15] T. Yu, R. Zhang, K. Yang, M. Yasunaga, D. Wang, Z. Li, J. Ma, I. Li, Q. Yao, S. Roman,

et al., Spider: A large-scale human-labeled dataset for complex and cross-domain semantic parsing and text-to-sql task, arXiv preprint arXiv:1809.08887 (2018).

[16] V. Zhong, C. Xiong, R. Socher, Seq2sql: Generating structured queries from natural language using reinforcement learning, arXiv preprint arXiv:1709.00103 (2017).

[17] W. Chen, H. Wang, J. Chen, Y. Zhang, H. Wang, S. Li, X. Zhou, W. Y. Wang, Tabfact: A large-scale dataset for table-based fact verification, arXiv preprint arXiv:1909.02164 (2019).

[18] X. Zhong, E. ShafieiBavani, A. Jimeno Yepes, Image-based table recognition: data, model, and evaluation, in: Computer Vision–ECCV 2020: 16th European Conference, Glasgow, UK, August 23–28, 2020, Proceedings, Part XXI 16, Springer, 2020, pp. 564–580.

[19] 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: AAAI fall symposium: Information retrieval and knowledge discovery in biomedical text, Citeseer, 2012.

[20] Q. Jin, B. Dhingra, Z. Liu, W. Cohen, X. Lu, Pubmedqa: A dataset for biomedical research question answering, in: Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP), 2019, pp. 2567–2577.

[21] J. Lee, W. Yoon, S. Kim, D. Kim, S. Kim, C. H. So, J. Kang, Biobert: a pre-trained biomedical language representation model for biomedical text mining, Bioinformatics 36 (2020) 1234–1240.

[22] M. Luo, A. Mitra, T. Gokhale, C. Baral, Improving biomedical information retrieval with neural retrievers (2022).

[23] P. Liu, W. Yuan, J. Fu, Z. Jiang, H. Hayashi, G. Neubig, Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing, arXiv preprint arXiv:2107.13586 (2021).

[24] V. Sanh, A. Webson, C. Raffel, S. H. Bach, L. Sutawika, Z. Alyafeai, A. Chaffin, A. Stiegler, T. L. Scao, A. Raja, et al., Multitask prompted training enables zero-shot task generalization, arXiv preprint arXiv:2110.08207 (2021).

[25] S. Mishra, D. Khashabi, C. Baral, Y. Choi, H. Hajishirzi, Reframing instructional prompts to gptk's language, ACL Findings (2022).

[26] S. Min, M. Lewis, L. Zettlemoyer, H. Hajishirzi, Metaicl: Learning to learn in context, arXiv preprint arXiv:2110.15943 (2021).

[27] L. Ouyang, J. Wu, X. Jiang, D. Almeida, C. L. Wainwright, P. Mishkin, C. Zhang, S. Agarwal, K. Slama, A. Ray, et al., Training language models to follow instructions with human feedback, Preprint (2022).

[28] L. Wang, R. Li, Y. Yan, Y. Yan, S. Wang, W. Wu, W. Xu, Instructionner: A multi-task instruction-based generative framework for few-shot ner, arXiv preprint arXiv:2203.03903 (2022).

[29] R. S. Puri, S. Mishra, M. Parmar, C. Baral, How many data samples is an additional instruction worth?, arXiv preprint arXiv:2203.09161 (2022).

[30] A. Prasad, P. Hase, X. Zhou, M. Bansal, Grips: Gradient-free, edit-based instruction search for prompting large language models, arXiv preprint arXiv:2203.07281 (2022).

[31] M. Parmar, S. Mishra, M. Geva, C. Baral, Don't blame the annotator: Bias already starts in the annotation instructions, arXiv preprint arXiv:2205.00415 (2022).

[32] T. Wu, M. Terry, C. J. Cai, Ai chains: Transparent and controllable human-ai interaction by chaining large language model prompts, arXiv preprint arXiv:2110.01691 (2021).

[33] T. Wu, E. Jiang, A. Donsbach, J. Gray, A. Molina, M. Terry, C. J. Cai, Promptchainer: Chaining large language model prompts through visual programming, arXiv preprint arXiv:2203.06566 (2022).

[34] X. V. Lin, T. Mihaylov, M. Artetxe, T. Wang, S. Chen, D. Simig, M. Ott, N. Goyal, S. Bhosale, J. Du, et al., Few-shot learning with multilingual language models, arXiv preprint arXiv:2112.10668 (2021).

[35] K. Kuznia, S. Mishra, M. Parmar, C. Baral, Less is more: Summary of long instructions is better for program synthesis, arXiv preprint arXiv:2203.08597 (2022).

[36] Y. Wang, S. Mishra, P. Alipoormolabashi, Y. Kordi, A. Mirzaei, A. Arunkumar, A. Ashok, A. S. Dhanasekaran, A. Naik, D. Stap, et al., Benchmarking generalization via in-context instructions on 1,600+ language tasks, arXiv preprint arXiv:2204.07705 (2022).

[37] N. Rasul, M. Syed, Differential Diagnosis in Primary Care, Wiley, 2009. URL: https://books.google.com/books?id=r5cTAQAAMAAJ.

[38] H. Kitano, Biological robustness, Nature Reviews Genetics 5 (2004) 826–837.

[39] T. Gokhale, S. Mishra, M. Luo, B. Sachdeva, C. Baral, Generalized but not robust? comparing the effects of data modification methods on out-of-domain generalization and adversarial robustness, in: Findings of the Association for Computational Linguistics: ACL 2022, 2022, pp. 2705–2718.

[40] V. Sanh, L. Debut, J. Chaumond, T. Wolf, Distilbert, a distilled version of bert: smaller, faster, cheaper and lighter, arXiv preprint arXiv:1910.01108 (2019).

[41] B. McCann, N. S. Keskar, C. Xiong, R. Socher, The natural language decathlon: Multitask learning as question answering, arXiv preprint arXiv:1806.08730 (2018).

[42] A. Fisch, A. Talmor, R. Jia, M. Seo, E. Choi, D. Chen, Mrqa 2019 shared task: Evaluating generalization in reading comprehension, in: Proceedings of the 2nd Workshop on Machine Reading for Question Answering, 2019, pp. 1–13.

[43] M. Luo, K. Hashimoto, S. Yavuz, Z. Liu, C. Baral, Y. Zhou, Choose your qa model wisely: A systematic study of generative and extractive readers for question answering, Spa-NLP 2022 (2022) 7.

[44] A. Webson, E. Pavlick, Do prompt-based models really understand the meaning of their prompts?, arXiv preprint arXiv:2109.01247 (2021).

[45] T. Schick, H. Schütze, True few-shot learning with prompts–a real-world perspective, arXiv preprint arXiv:2111.13440 (2021).

[46] T. Wolf, L. Debut, V. Sanh, J. Chaumond, C. Delangue, A. Moi, P. Cistac, T. Rault, R. Louf, M. Funtowicz, J. Davison, S. Shleifer, P. von Platen, C. Ma, Y. Jernite, J. Plu, C. Xu, T. L. Scao, S. Gugger, M. Drame, Q. Lhoest, A. M. Rush, Transformers: State-of-the-art natural language processing, in: Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System Demonstrations, Association for Computational Linguistics, Online, 2020, pp. 38–45. URL: https://www.aclweb.org/anthology/2020.emnlp-demos.6.

A. Details of BioTabQA

Table 6 shows the 22 templates and the corresponding prompts for BioTabQA datasets. Table 9 shows the three Split division including which tasks in the training and cross-task evaluation. Table 7 and 8 show the statistic of Split 1 and 2, respectively.

B. Experimental Setup

We use DistilBERT [40] as the backbone model and load the pretrained model distilbert-base-uncased from Huggingface library [46]. All models are optimized by AdamW with learning rate 5e-5 in 4 epochs, batch size 16. The maximum length input to every model is 512. All models are trained on Tesla V100 machine with one GPU.

ID	Question Template	Prompt
1	I have symptom A, what disease do I have?	If symptom A is in symptom list, report corresponding disease.
2	I have symptom A and sign A, what is my diagnosis?	If symptom A is in symptom list, and sign A is in sign list, report corresponding disease.
3	I have symptom A and symptom B, what is wrong with me?	If symptom A and symptom B are in symptom list, report corresponding disease.
4	I have sign A and sign B, what disease do you think I have?	If sign A is in sign list, and sign B is in sign list, report corresponding disease.
5	I have symptom A and symptom B but not symptom C, what is my potential diagnosis?	If symptom A and symptom B are in symptom list, but symptom C is not in symptom list, report corresponding disease.
6	A patient is showing symptom A , symptom B and symptom C, what could be causing this?	If symptom A, symptom B and symptom C are in symptom list, report corresponding disease.
7	A patient is exhibitng syptom A and sign A, diagnose her	If symptom A is in symptom list, and sign A is in sign list, report corresponding disease.
8	What disease can cause symptom A and symptom B?	If symptom A and symptom B are in symptom list, report corresponding disease.
9	What disease causes symptom A, symptom B and sign A?	If symptom A and symptom B are in symptom list, and sign A is in sign list, report corresponding disease.
10	If my friend has symptom A and symptom B, then what is his potential diagnosis?	If symptom A and symptom B are in symptom list, report corresponding disease.
11	The patient has symptom A,symptom B and symptom C, what disease can cause these symptoms?	If symptom A, symptom B and symptom C are in symptom list, report corresponding disease.
12	Which disease is associated with symptom A and symptom B?	If symptom A and symptom B are in symptom list, report corresponding disease.
13	A patient is complaining about symptom A, symptom B and symptom C, diagnose him.	If symptom A, symptom B and symptom C are in symptom list, report corresponding disease.
14	What disease is responsible for symptom A, symptom B and symptom C?	If symptom A, symptom B and symptom C are in symptom list, report corresponding disease.
15	I am experiencing symptom A, what is wrong with me?	If symptom A is in symptom list, report corresponding disease.
16	Why am I experiencing symptom A and symptom B?	If symptom A and symptom B are in symptom list, report corresponding disease.
17	I have symptom A, symptom B and symptom C, why is this happening?	If symptom A, symptom B and symptom C are in symptom list, report corresponding disease.
18	A patient is showing symptom A and symptom B, what illness is associated with these symptoms?	If symptom A, symptom B and symptom C are in symptom list, report corresponding disease.
19	I have symptom A, and symptom B, what disease may I have?	If symptom A and symptom B are in symptom list, report corresponding disease.
20	I have symptom A and symptom B, what possible disease could I have?	If symptom A and symptom B are in symptom list, report corresponding disease.
21	What is causing my symptom A?	If symptom A is in symptom list, report corresponding disease.
22	I have symptom A, symptom B, symptom C but no symptom D, what is causing this?	If symptom A, symptom B and symptom C are in symptom list, but symptom D is not in symptom list, report corresponding disease.

Table 6

22 types of templates and the corresponding prompts for BioTabQA datasets.

Statistic	Train	IID Test	Cross Task Test
# of Samples	7,349	15,566	16,145
Question Length	19	21	17
Table Length	239	255	259
Prompt Length	17	17	17
# Tasks with 1 sym/sign	3	3	0
# Tasks with 2 sym/sign	9	9	2
# Tasks with 3 sym/sign	4	4	3
# Tasks with 4 sym/sign	1	1	0
# Tasks with negation	2	2	0

Table 7
Statistic of BioTabQA Split 2 for training (Train), in-domain testing (IID Test) and cross task testing (Cross Task Test) sets.

Statistic	Train	IID Test	Cross Task Test
# of Samples	8,524	18,278	6924
Question Length	240	21	254
Table Length	19	256	18
Prompt Length	18	18	14
# Tasks with 1 sym/sign	1	1	2
# Tasks with 2 sym/sign	9	9	2
# Tasks with 3 sym/sign	6	6	1
# Tasks with 4 sym/sign	1	1	0
# Tasks with negation	2	2	0

Table 8
Statistic of BioTabQA Split 3 for training (Train), in-domain testing (IID Test) and cross task testing (Cross Task Test) sets.

Split	Train/Test	Cross-Task Test
1	2,3,5,6,8,9,10,11,12,13,14,16,17,18,19,20,22	1,4,7,15,21
2	1,2,3,4,5,6,7,10,13,15,16,17,18,19,20,21,22	8,9,11,12,14
3	2,4,5,6,7,8,9,10,11,12,13,14,18,19,20,21,22	1,3,15,16,17

Table 9
BioTabQA provides three Splits, each Split has 17 tasks for training, and the rest 5 for cross-task evaluations.