In-BoXBART: Get Instructions into Biomedical Multi-Task Learning

Mihir Parmar[1] **Swaroop Mishra**[1] **Mirali Purohit** **Man Luo**[1]
M. Hassan Murad[2] **Chitta Baral**[1]

[1]Arizona State University, USA
[2]Mayo Clinic, USA
[1]{mparmar3, srmishr1, mpurohi3, mluo26, chitta}@asu.edu
[2]murad.mohammad@mayo.edu

Abstract

Single-task models have proven pivotal in solving specific tasks; however, they have limitations in real-world applications where multi-tasking is necessary and domain shifts are exhibited. Recently, instructional prompts have shown significant improvement towards multi-task generalization; however, the effect of instructional prompts and Multi-Task Learning (MTL) has not been systematically studied in the biomedical domain. Motivated by this, this paper explores the impact of instructional prompts for biomedical MTL. We introduce the BoX, a collection of 32 instruction tasks for **Bio**medical NLP across (**X**) various categories. Using this meta-dataset, we propose a unified model termed as In-BoXBART, that can jointly learn all tasks of the BoX without any task-specific modules. To the best of our knowledge, this is the first attempt to propose a unified model in the biomedical domain and use instructions to achieve generalization across several biomedical tasks. Experimental results indicate that the proposed model: 1) outperforms single-task baseline by ∼3% and multi-task (without instruction) baseline by ∼18% on an average, and 2) shows ∼23% improvement compared to single-task baseline in few-shot learning (i.e., 32 instances per task) on an average. Our analysis indicates that there is significant room for improvement across tasks in the BoX, implying the scope for future research direction.[1]

1 Introduction

For long, task-specific models have played a central role in achieving state-of-the-art performance in both general and biomedical NLP (Wang et al., 2021a; Banerjee et al., 2021). During 2017-2019, pre-train and fine-tune paradigm (Liu et al., 2021) became the prevalent approach in NLP. Due to success of Language Models (LMs) in the biomedical domain such as BioBERT (Lee et al., 2020), ClinicalXLNET (Huang et al., 2020), and others (Alrowili and Shanker, 2021; Kraljevic et al., 2021; Phan et al., 2021), this paradigm is widely used for creating many task-specific models (Wang et al., 2021a; Banerjee et al., 2021). However, task-specific models have limitations to real-world applications because this approach is computationally expensive (i.e., require large computational resources) and time-consuming (Strubell et al., 2019; Schwartz et al., 2020). Hence, there is a need for generalization where a single model can perform various tasks leading to a computationally efficient approach. Past attempts have been made in general-domain NLP to achieve generalization across tasks such as MQAN (McCann et al., 2018), UNICORN (Lourie et al., 2021), and UnifiedQA (Khashabi et al., 2020). However, approaches to achieve generalization across various biomedical NLP tasks have not been systematically studied. Hence, this paper studies the multi-tasking approach that can generalize over different biomedical NLP tasks. Figure 1 shows the overview of our proposed multi-tasking approach for various biomedical NLP tasks.

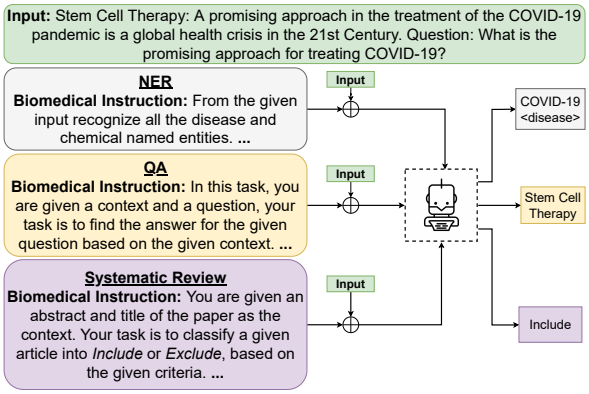

Figure 1: Schematic representation of multi-tasking in biomedical domain using instructional prompts. In this approach, a model is allowed to utilize tasks to get familiar with instructions and use them to map a given input to its corresponding output.

[1]https://github.com/Mihir3009/In-BoXBART

Recently, prompt-based models have been widely used because of their ability to achieve generalization instead of task-specific models (Liu et al., 2021). Mishra et al. (2021b); Wei et al. (2021) and Sanh et al. (2021) show the effectiveness of instructional prompts in generalizing on seen as well as unseen general-domain NLP tasks. In this paper, we adapt this instructional prompt-based approach for the first time to achieve generalization across various biomedical NLP tasks. To this extent, this paper introduces a collection of 32 instruction tasks for **Bio**medical NLP across (**X**) various categories (BoX) and proposes a unified model that can generalize over 32 different biomedical NLP tasks. The proposed unified model (i.e., In-BoXBART) is trained on the instruction-based meta-dataset (i.e., BoX) and evaluated on each task individually from the BoX.

To evaluate the proposed approach, we compare our model (i.e., In-BoXBART) with two baselines: (1) single-task models (i.e., models trained on one task and evaluated on the same task), and (2) multi-task model (i.e., a single model trained on a combination of all tasks) without instructions. Experimental results show that In-BoXBART outperforms single-task baseline by $\sim$3%, and multi-task baseline by $\sim$18%. We also analyze few-shot learning scenario using In-BoXBART since obtaining annotated data in the biomedical domain is costly and time-consuming (Luo et al., 2022b). In the few-shot setting (i.e., 32 instances per task), In-BoXBART outperforms the single-task baseline by 23.33%. This indicates that Multi-Task Learning (MTL) and instruction-tuning have an advantage in the low resources settings. Although the performance of the In-BoxBART is promising, our analysis reveals that there is still room for improvement on some tasks, implying the scope for future research direction. Concisely, our contributions can be summarized in three folds:

1. This paper introduces the first benchmark meta-dataset in biomedical domain, i.e., BoX: a collection of 32 instruction tasks for Biomedical NLP across (X) various categories. Each task is processed in a unified format and equipped with instructions that can be used to train sequence-to-sequence models.
2. Using this meta-dataset, we propose an instruction-tuned Bidirectional and Auto-Regressive Transformer (BART) model, termed as In-BoXBART. The comparison of In-BoxBART and two baselines shows that In-BoXBART outperforms single-task baseline by $\sim$ 3% and multi-task (without instruction) baseline by $\sim$ 18%.
3. In the few-shot setting, we show that In-BoxBART significantly outperforms the single-task baseline by $\sim$ 23%. This indicates the potential application of instruction-tuning in the biomedical domain where annotated data is difficult to obtain.

2 Related Work

Multi-task Learning Owing to the problems associated with single-task learning in terms of their space and time requirements, several multi-task learning approaches have been proposed over the years. DecaNLP (McCann et al., 2018) built a multi-tasking model by converting format of each tasks to question answering format. Several other works have followed similar approach, for example, by converting tasks to reading comprehension (Mishra et al., 2022) and textual entailment format (Wang et al., 2021b). The multitasking model T5 (Raffel et al., 2020) was built with the help of a unified framework that converts all text-based language problems into a text-to-text format. SCIFIVE (Phan et al., 2021) involved building a text-to-text model for the biomedical literature. Aghajanyan et al. (2021) introduced pre-finetuning, an additional large-scale learning stage between language model pre-training and fine-tuning to improve multitask learning performance. Models empowered by multi-task learning have achieved SOTA in many different tasks, e.g., Question Answering (QA) (Khashabi et al., 2020), commonsense reasoning (Lourie et al., 2021) and structured knowledge grounding tasks (Xie et al., 2022).

Instruction Learning The turking test (Efrat and Levy, 2020) was proposed to measure the efficacy of models to follow instructions. Studies have been made to investigate the effect of natural language instructions on model performance (Hase and Bansal, 2021; Ye and Ren, 2021b; Zhong et al., 2021; Weller et al., 2020). Moreover, Mishra et al. (2021b) proposed Natural Instructions which break down each task to multiple sub-tasks that help models in following instructions and subsequently generalize to unseen tasks (i.e., cross-task generalization). FLAN (Wei et al., 2021) and T0 (Sanh et al., 2021) models were built by leveraging instruction/prompt-tuning on diverse range

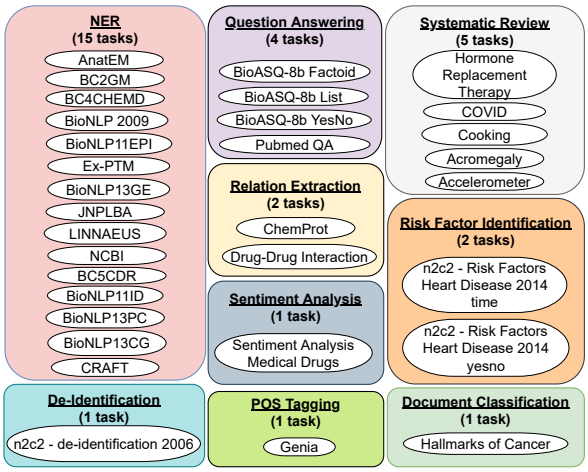

Figure 2: Schematic representation of 9 categories of tasks: each block represents one category with various tasks equipped with instruction.

Category	# of training samples
NER	82503
De-identification	106
POS Tagging	16323
QA	5778
RE	23359
Sentiment Analysis	2860
Systematic Review	5761
Document Classification	3119
Risk Factor Identification	986
Total	140795

Table 1: Size of training samples in each category

of tasks and achieving zero-shot generalization on target unseen tasks. Task reframing (Mishra et al., 2021a) proposed several guidelines to reframe task instructions to improve model response to follow instructions. Analysis introduced to understand in-context learning better on a large set of training tasks (Min et al., 2021, 2022). InstructGPT model (Ouyang et al., 2022) is proposed, which is fine-tuned with human feedback to follow natural instructions. Furthermore, many works focused on investigating whether LMs understands meaning of natural language and prompts (Webson and Pavlick, 2021; Zhao et al., 2021). Weller et al. (2020) and Ye and Ren (2021a) use task descriptions to achieve generalization to new tasks. Puri et al. (2022) introduced instruction augmentation to improve model performance and sample complexity. Wang et al. (2022) has developed instruction-based multi-task framework for few-shot Named Entity Recognition (NER) task. Prasad et al. (2022) introduced Gradient-free Instructional Prompt Search (GrIPS) for improving task instructions for large LMs. Recently, many approaches have been proposed to improve model performance using instructions (Wu et al., 2021, 2022; Lin et al., 2021; Kuznia et al., 2022).

3 BoX

We use 29 existing, widely adopted biomedical NLP datasets collected from various challenges, platforms and organizations to create BoX. We define the BoX as a benchmark dataset for biomedical MTL across 9 different categories. In the BoX, we reframed all the datasets as text generation tasks (see examples in Appendix B) and created 32 instruction tasks. BoX consists of high-quality human-authored Biomedical Instructions (BIs) for all 32 tasks. Figure 2 shows the 9 different categories and corresponding generated tasks. Each category is defined as colored box and each box contains instruction tasks re-purposed from original datasets.

3.1 Tasks

Table 1 shows the number of training samples we have used for each category. Further details of each instruction task statistics is shown in Appendix A. Each category and corresponding tasks from the BoX are defined as below:

Named Entity Recognition (NER) NER has been considered a necessary first step in processing literature for biomedical text mining where the model helps in identifying named entities such as protein, gene, chemical, disease, treatment. We use fifteen publicly available biomedical NER datasets (Crichton et al., 2017) to create instruction tasks.

De-Identification (DI) In this task, the model takes medical discharge records of a patient as input and identify Private Health Information (PHI) such as organizations, persons, locations, dates. We use n2c2 2006 de-identification challenge dataset (Uzuner et al., 2007) to perform this task.

Part-Of-Speech (POS) Tagging The goal of this task is to identify various POS tags from the biomedical text. We use GENIA corpus (Tateisi et al., 2005) built from MEDLINE abstracts for the POS tagging task.

Question-Answering (QA) QA models receive a question and a corresponding context as input and output the relevant answer from the given context. To execute this task, we used the BioASQ-8b dataset (Nentidis et al., 2020) for different question types, i.e., yes/no, factoid, and list type questions. We created three different tasks from this dataset. Also, we use PubMedQA dataset (Jin et al., 2019) for this task.

Relation Extraction (RE) We used two datasets for this task: (1) CHEMPROT corpus from biocreative VI precision medicine track (Islamaj Doğan et al., 2019), and (2) Drug-Drug Interaction (DDI) corpus from SemEval 2013 DDI Extraction challenge (Herrero-Zazo et al., 2013). Here, we only consider binary RE tasks without any label describing the type of the relation.

Systematic Review (SR) We have included data from the following five Systematic Reviews (SRs) that were conducted using the traditional (manual) process and published in relevant venues by Mayo Clinic physicians: (1) Hormone Replacement Therapy (HRT), (2) Cooking, (3) Accelerometer, (4) Acromegaly, and (5) COVID for this task (Parmar, 2021). More details about these datasets creation and statistics are given in Appendix C.

Sentiment Analysis (SA) Analyzing the sentiment of people towards medical drugs is an essential task in the biomedical domain. To that effect, we use medical drug sentiment analysis dataset[2] to identify one of three sentiments: (1) positive, (2) negative, and (3) neutral.

Document Classification (DC) We have used the Hallmarks of Cancer (HoC) dataset (Baker et al., 2016) for this task.

Risk Factor Identification (RFI) The goal of this task is to identify risk factors for Coronary Artery Disease (CAD) in diabetic patients over time. For this, we used n2c2 2014 shared task track 2 dataset (Kumar et al., 2015) with two different purposes: (1) identify if the risk factor is presented in the medical discharge summary and (2) time of risk factor present in the discharge records.

3.2 Biomedical Instructions

Motivated by Mishra et al. (2021b), we have used a similar approach to create Biomedical Instructions (BIs). BI consists of natural language instructions that describe a task and contain instances of that task. Here, we introduce a unified schema to present BI and described how we can construct BI for each task given in the BoX. Figure 3 illustrates the schematic representation of the schema, and Figure 4 shows an example of BI that describes a "Named Entity Recognition (NER)" task accompanied with a few positive examples.

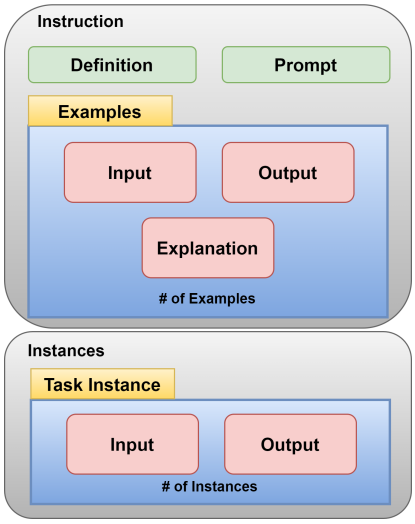

Figure 3: Unified schema used to create a Biomedical Instruction (BI).

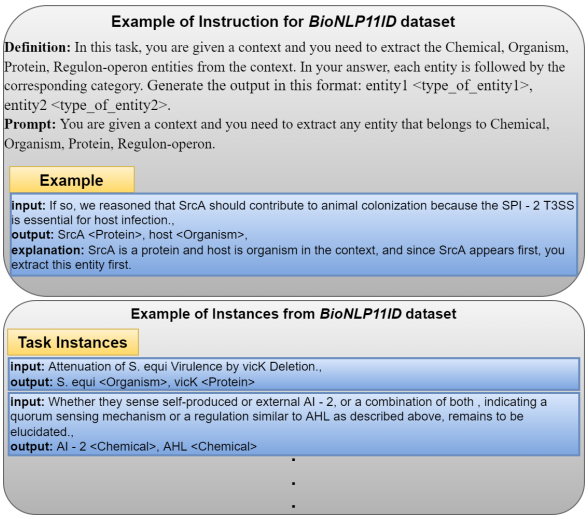

Figure 4: Example of Biomedical Instruction (BI) and task instances from ***BioNLP11ID*** (NER) dataset.

3.2.1 Unified Schema

All BIs are mapped to the unified schema. As shown in Figure 3, unified schema consists of a definition, prompt, and positive examples. This schema helps in organizing each BI. Each of the elements of the schema is explained below:

[2] https://www.kaggle.com/arbazkhan971/analyticvidhyadatasetsentiment

Definition contains the core explanation about the task and detailed instruction to the model that what needs to be done in the given task.

Prompt is the short explanation of the task that needs to be done.

Examples contain the input/output pairs of the task instance along with the explanation of how the output is generated. Generally, we provide 2-3 examples for each task.

Instances contain the input/output pairs of training samples from the task datasets.

3.2.2 Construction of BI

We have created a BI for each dataset given in the BoX. To create BI, we manually fill in the fields of unified instruction schema (Figure 3). For each dataset, the BI is created by one author and were verified by other authors.

Quality of BIs In the instruction verification process, we edit BIs if needed in terms of grammar, typos, ambiguity, etc. to improve the quality. According to (Beltagy et al., 2020), concise instructions are more beneficial compared to repetition, hence, we also redact repetition from BIs. As a result, our BIs consists of high-quality, short, and meaningful task definition, and prompts.

Positive examples and its explanation For each dataset, we have provided 2-3 positive examples and corresponding explanations to give an idea of how to perform the given task. As we know, the selection of examples has an impact on model performance (Lu et al., 2021). To that extent, we have been careful in selecting examples for text generation and classification tasks. For text generation, we have provided 2-3 examples with a detailed explanation about how the output is generated. For text classification tasks, we have included examples corresponding to each class with an explanation of why the particular class is assigned to a given input instance. All positive examples are drawn from training instances and have been removed from training in order to avoid repetition. All the explanations of examples pass through the verification process to maintain high quality.

Collection of input/output instances Since each biomedical NLP dataset included in the BoX has its own annotated input/output instances, we converted them into text-to-text format (Lourie et al., 2021). Example of instances converted for each task is given in Appendix B. After this, we appended all instances tuple (i.e., <input, output>) with instruction schema (as shown in Figure 3).

4 Problem Setup and Models

4.1 Problem setup

Let us assume, we have input/output instances pair (X_t, Y_t) for given task t. Along with that, each task is described in terms of its instruction BI_t.

Single-task models Traditional supervised models learn a mapping function (f_M) between input (x) and output (y), where $(x, y) \in ({X_t}^{\text{train}}, {Y_t}^{\text{train}})$ and are evaluated on the same task $({X_t}^{\text{test}}, {Y_t}^{\text{test}})$. We refer this setup as single-task learning.

Multi-task models In this setup, we combined training data and corresponding biomedical instruction of all tasks together. The goal of multi-task learning models is to learn mapping function (f_M) between input (x), output (y) and biomedical instruction BI_t, i.e., $f_M(BI_t, x) = y$, where $(x, y) \in (X_t, Y_t)$. This model is evaluated on task-specific instances $(x, y) \in ({X_t}^{\text{test}}, {Y_t}^{\text{test}})$ In contrast to single-task models, a single model is used here to solve various tasks, hence, achieving generalization. We refer this setup as MTL.

4.2 Models

We propose an instruction-based model to achieve multi-tasking and compare it with two baselines: (1) single-task models, and (2) multi-task models without instructions. We have fine-tuned the BART (base) model (Lewis et al., 2020) to build baselines as well as the proposed model.

4.2.1 Baselines

Single-Task models As formulated in the single-task problem setup, we have trained the BART model on each task from the BoX and evaluated it on the same task.

Multi-task without instruction To build this baseline, we have combined training data of each task from the BoX together without appending BIs and trained a single model on the combined data. We refer this model as Vanilla-BoXBART. This model is evaluated on each task of the BoX.

4.2.2 Proposed Model

As formulated in the multi-task problem setup, we have combined training data and the corresponding BI of each task. To combine instruction with

input instances, we map a BI and an input (x) into the textual format and obtain $enc(BI_t, x)$. After that, BART model is used to predict an output (y) using a mapping function $f_M : enc(BI_t, x) \rightarrow y$. To perform encoding, a standard NLP paradigm of mapping is used, i.e., mapping an input to text. Here, we map each element of BI (i.e., definition and positive examples as shown in the schema) to a textual format and append it before the input instances. After appending BI of each task to instances, we combined all training data of each task. Now, we fine-tuned the BART model with this combined instruction meta-dataset. We refer this instruction-tuned model as In-BoXBART.

5 Experiments and Analysis

5.1 Experimental Setup

We have used BART (base) model to build all baselines and proposed model. All the experiments are performed using *Quadro RTX 8000* GPU. All models are trained for 3 epochs. In particular, we have used *huggingface implementation* (Wolf et al., 2020) of the BART and its pre-defined functions for the training and evaluation with default parameters.

Instance Selection As we know, BART (base) can accept the input of a maximum 1024 token length. Since there are few instances in some datasets that exceed this limit (after including instructions), we have discarded those instances while creating instruction tasks. We have also removed the same instances while training two baselines to do a fair comparison. We have discarded long samples (>1024 token length) from validation and testing data as well.

Example Selection As discussed in Lu et al. (2021), the selection and order of the examples included in instructions matters for mainly classification tasks and affects the performance of the model. We empirically conclude that the proposed model benefits from ignoring examples from biomedical instructions for classification tasks during training and evaluation. Hence, we have discarded all examples from the BIs associated with the classification instruction tasks.

Instance Sampling Some classification datasets used to create the BoX are imbalanced. To balance these datasets, we have applied the sampling techniques (Poolsawad et al., 2014) before using datasets to create BoX. In particular, we have analyzed three sampling techniques: (1) under-sampling, (2) average-sampling, and (3) over-sampling. In under-sampling, we have reduced instances for all the classes to the class with the lowest number of instances. In contrast, we have over-sampled instances via replication of random instances to the class with the highest number of instances to achieve over-sampling. In average sampling, we calculated mean of number of instances across all the classes and over-sampled or under-sampled instances accordingly for each class.

Few-shot setting Similar to the (Schick and Schütze, 2021), we have started with 32 randomly selected instances for each instruction task from the BoX to exhibit few-shot learning. After that, we have increased randomly selected instance instances per task to 100/1k/4k. If any task have already less number of instances than the threshold (i.e., 100/1k/4k), we keep all the instances from that task. While selecting the instances, we made sure that we select balanced data for the classification tasks. Moreover, the BoX contains an average 6k instances per task.

Evaluation Metric We use Rouge-L (Lin, 2004) as our evaluation metric since we treat all the tasks as text generation problems. We also use F_1-Score for evaluations.

5.2 Results and Findings

Effect of Sampling As mentioned above, we conduct three experiments to analyze the effect of sampling on In-BoXBART. We train our model using training data obtained from (1) under-sampling, (2) average-sampling, and (3) over-sampling. We achieve on an average (across all instruction tasks) 69.62, 70.23 and 73.49 Rouge-L for under-, average- and over-sampling, respectively. Here, we observe from the experimental results that over-sampling gives better performance compared to under- and average-sampling since there is a loss of training data samples for under- and average-sampling. Hence, we report results of over-sampling as the main result in Table 2.

Performance comparison Table 2 presents the results for single-task model, Vanilla-BoXBART and In-BoXBART. We can see from Table 2 that the single-task model, Vanilla-BoXBART, and In-BoXBART achieve on an average (across all tasks) Rouge-L of 70.51, 55.55, and 73.49, respectively. They achieve 70.15%, 55.21%, and 73.01%

Category	Task	Rouge-L			F_1-Score		
		Single Task	V-BB	I-BB	Single Task	V-BB	I-BB
NER	AnatEM	**84.88**	32.30	83.93	**85.55**	33.50	84.61
	BC2GM	**77.66**	50.87	74.10	**78.56**	50.86	75.03
	BC4CHEMD	**88.85**	71.05	86.50	**89.06**	71.44	86.97
	BC5CDR	**74.83**	69.81	74.76	75.13	70.11	**75.24**
	BioNLP11EPI	84.64	50.10	**87.60**	84.95	52.85	**88.04**
	BioNLP11ID	71.08	59.12	**72.64**	71.64	60.15	**73.39**
	BioNLP13CG	64.19	55.18	**67.72**	61.68	53.88	**65.09**
	BioNLP13GE	83.74	49.30	**86.71**	84.08	51.78	**87.39**
	BioNLP13PC	70.42	53.06	**72.46**	66.89	51.61	**67.77**
	BioNLP09	85.16	51.54	**88.09**	85.54	54.31	**88.48**
	CRAFT	63.72	51.85	**64.10**	63.92	52.31	**64.30**
	Ex-PTM	82.32	49.61	**83.73**	82.38	52.07	**84.49**
	JNLPBA	**71.65**	69.37	71.54	**70.79**	68.60	70.26
	NCBI	**89.51**	74.46	86.11	**89.81**	75.55	80.91
	linnaeus	**94.43**	44.99	93.46	93.21	44.59	**93.77**
	Average	79.14	55.51	**79.54**	78.88	56.24	**79.45**
DI	DI 2006	12.60	46.38	**50.82**	10.60	43.28	**47.45**
POS	Genia	**71.45**	27.94	71.26	70.48	27.50	**71.99**
QA	BioASQ8b (factoid)	**52.95**	51.14	47.28	**54.67**	53.52	49.51
	BioASQ8b (list)	**38.96**	19.87	36.11	-	17.74	35.59
	BioASQ8b (yesno)	61.74	62.61	**68.25**	63.48	62.61	**68.25**
	PubMedQA	**27.12**	25.48	24.49	**31.44**	30.74	29.58
	Average	**45.19**	39.78	44.03	**46.39**	41.15	45.73
RE	ChemProt	76.08	76.00	**81.61**	**63.89**	52.17	63.22
	DDI	**91.78**	82.97	89.35	**94.10**	82.97	89.35
	Average	83.04	79.48	**85.48**	**79.00**	67.57	76.28
SA	Medical Drugs	**47.51**	46.39	47.37	**47.51**	46.39	47.37
SR	Accelerometer	74.65	72.54	**81.25**	74.65	72.54	**81.25**
	Acromegaly	80.21	**81.77**	80.71	80.21	**81.77**	80.71
	COVID	74.81	76.30	**77.28**	74.81	76.30	**77.28**
	Cooking	71.71	82.93	**83.25**	71.71	82.93	**83.25**
	HRT	75.68	77.17	**82.70**	75.68	77.17	**82.70**
	Average	75.41	78.14	**81.04**	75.41	78.14	**81.04**
DC	HoC	**88.53**	49.64	82.53	**88.53**	49.51	82.53
RFI	RFHD 2014 (yesno)	57.21	64.97	**69.17**	57.21	64.97	**69.17**
	RFHD 2014 (time-riskfactor)	66.18	0.97	**85.24**	66.18	0.97	**85.28**
	Average	72.87	57.30	**77.21**	61.69	32.97	**77.22**
Average	-	70.51	55.55	**73.49**	70.15	55.21	**73.01**

Table 2: Results comparison between single-task baseline, Vanilla-BoXBART and In-BoXBART in terms of Rouge-L and F_1-Score. All the results for F_1-Score are presented in %. V-BB: Vanilla-BoXBART, I-BB: In-BoXBART, RFHD: Risk Factor for Heart Disease.

F_1-Score, respectively, exhibiting the same performance behaviour as Rouge-L. Hence, we use Rouge-L for further comparisons. From the result, we can observe that Vanilla-BoXBART reduces the complexity compared to the single-task model (i.e., 110 million parameters *vs.* 32x110 million parameters), however, on an average the performance drops by 14.96% in terms of Rouge-L, and compared to single-task models. This indicates that multi-task learning in the biomedical domain is more difficult than general domain NLP since many previous works have shown that the multi-task

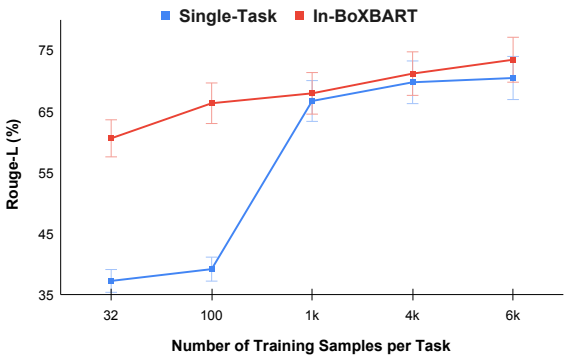

Figure 5: Comparison of on an average Rouge-L across all instruction tasks between single-task and In-BoXBART based on the average number of training instances per task.

model outperforms the single-task model (Lourie et al., 2021; McCann et al., 2018). On the other hand, In-BoXBART, which has the same complexity as Vanilla-BoXBART, significantly outperforms Vanilla-BoXBART by on average 17.94%, and also outperforms the single-task model by a 2.98% margin, precisely. This indicates the benefit of using instructions to achieve the MTL in the biomedical domain.

Effect of instruction in few-shot learning We have compared the average Rouge-L of In-BoXBART with a single-task baseline for few-shot setting. Figure 5 shows the relative performance of In-BoXBART compared to single-task baseline. We have shown results for all few-shot learning experiments in Appendix D. From the results, we see that In-BoXBART achieves on an average 60.64% Rouge-L and the single-task model achieves 37.31% for 32 instances per task. In-BoxBART significantly outperforms the single-task baseline by 23.33%. From Figure 5, we can see that In-BoXBART consistently perform better compared to the baseline. As we know, obtaining a large annotated dataset in the biomedical domain is difficult, time-consuming and costly. From few-shot learning, we can see that instructions are beneficial in achieving high performance compared to task-specific models.

5.3 Analysis

For which tasks, instruction is helpful? From Table 2, we can see that In-BoXBART outperforms baselines for 5 categories, i.e., NER, de-identification, RE, SR and risk factor identification. From this, we can see that instructions are more helpful in these five categories. However, In-BoXBART achieves performance lower or par with the single-task baseline for the tasks from QA, POS tagging, sentiment analysis and document classification which indicates room for improvement in this direction.

Which are harder tasks to solve using instructions? Although instructions help in achieving better performance for some tasks compared to the single-task model, the overall performance is still lower. For example, instruction improves performance for de-identification, but overall performance on this task is only 50.82% which can be improved. A similar pattern we can see for BioNLP12CG and CRAFT from NER; BioASQ-8b (factoid, list) and PubmedQA from QA; and Medical Drug from the sentiment analysis category. In general, we can observe that tasks that include either multi-class scenario or answer generation from the context are most likely to be harder to solve using instructions. For example, CRAFT and BioNLP13CG have 6 entity types which are higher than any other tasks from NER, and we can see that the performance for these two tasks is lower compared to other tasks of NER.

For which tasks, instruction is the most beneficial in few shot setting? From the results shown in Appendix D, tasks from the NER, de-identification, QA, sentiment analysis and risk factor identification shows on average larger improvement compared to baselines for the few-shot settings (i.e., 32 and 100 instances per task). This indicates that instructions are beneficial for the tasks from the above categories.

6 Discussion

Can we design better instructions? Since instruction teach the model how to solve a given task, domain specific information rich instructions can improve model performance. One potential way is to use the knowledge of domain experts. However, designing a good biomedical instruction can be one research direction.

How to handle long-context input? Training instances of many biomedical datasets consist Electronic Health Records (EHRs) or discharge summaries of patients. Because of this, these instances are long and exceed the maximum input length of LMs such as BERT, BART. In this scenario, encoding extra information in terms of prompts or

instructions becomes difficult. One potential solution is to use Longformer (Beltagy et al., 2020), and another solution is to use T5 kind of models which use relative position embeddings so that the inference length can be longer (Luo et al., 2022a).

How to handle multi-class classification tasks? Multiple classes cause an issue while creating biomedical instructions because we can not present one example per class. If we do that, the encoding of BI and input will exceed the maximum length of LMs. A naive solution is to select examples of a few labels or remove the examples. However, this will cause a label bias issue or performance degradation. Potential future research direction can be designing a methodology to handle multi-class classification tasks.

How far we are from the SOTA? We have presented preliminary comparison of our results w.r.t. state-of-the-art (SOTA) single-task systems for 21 instruction tasks from the BoX as shown in Appendix E. Form the results, we can see that the performance of the proposed model remains far from the SOTA for some tasks, indicating significant room for further research in this domain.

7 Summary and Conclusions

This research shows the impact of instructions in MTL for the first time in the biomedical domain. To this extent, we introduced the BoX, a first benchmark dataset consisting of 32 instruction tasks across various biomedical NLP domains. Using this meta-dataset, we proposed a unified model, i.e., In-BoXBART which outperforms single-task baseline and Vanilla-BoxBART by $\sim 3\%$ and $\sim 18\%$, respectively. Our proposed approach also shows an effective performance for a few-shot setting which is more beneficial in the biomedical domain where obtaining large annotated datasets is difficult. We hope that the BoX benchmark, In-BoXBART, and experimental results encourage future research into more unified models for biomedical NLP.

Acknowledgement

Authors acknowledge support from ONR award number N00014-20-1-2332 for this project and thank the anonymous reviewers for feedback.

References

Armen Aghajanyan, Anchit Gupta, Akshat Shrivastava, Xilun Chen, Luke Zettlemoyer, and Sonal Gupta. 2021. Muppet: Massive multi-task representations with pre-finetuning. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5799–5811.

Sultan Alrowili and Vijay Shanker. 2021. BioM-transformers: Building large biomedical language models with BERT, ALBERT and ELECTRA. In *Proceedings of the 20th Workshop on Biomedical Language Processing*, pages 221–227, Online. Association for Computational Linguistics.

Simon Baker, Ilona Silins, Yufan Guo, Imran Ali, Johan Högberg, Ulla Stenius, and Anna Korhonen. 2016. Automatic semantic classification of scientific literature according to the hallmarks of cancer. *Bioinformatics*, 32(3):432–440.

Pratyay Banerjee, Kuntal Kumar Pal, Murthy Devarakonda, and Chitta Baral. 2021. Biomedical named entity recognition via knowledge guidance and question answering. *ACM Transactions on Computing for Healthcare*, 2(4):1–24.

Iz Beltagy, Matthew E Peters, and Arman Cohan. 2020. Longformer: The long-document transformer. *arXiv preprint arXiv:2004.05150*.

Gamal Crichton, Sampo Pyysalo, Billy Chiu, and Anna Korhonen. 2017. A neural network multi-task learning approach to biomedical named entity recognition. *BMC bioinformatics*, 18(1):1–14.

Avia Efrat and Omer Levy. 2020. The turking test: Can language models understand instructions? *arXiv preprint arXiv:2010.11982*.

Peter Hase and Mohit Bansal. 2021. When can models learn from explanations? a formal framework for understanding the roles of explanation data. *arXiv preprint arXiv:2102.02201*.

María Herrero-Zazo, Isabel Segura-Bedmar, Paloma Martínez, and Thierry Declerck. 2013. The ddi corpus: An annotated corpus with pharmacological substances and drug–drug interactions. *Journal of biomedical informatics*, 46(5):914–920.

Kexin Huang, Abhishek Singh, Sitong Chen, Edward Moseley, Chih-Ying Deng, Naomi George, and Charolotta Lindvall. 2020. Clinical XLNet: Modeling sequential clinical notes and predicting prolonged mechanical ventilation. In *Proceedings of the 3rd Clinical Natural Language Processing Workshop*, pages 94–100, Online. Association for Computational Linguistics.

Rezarta Islamaj Doğan, Sun Kim, Andrew Chatr-Aryamontri, Chih-Hsuan Wei, Donald C Comeau, Rui Antunes, Sérgio Matos, Qingyu Chen, Aparna Elangovan, Nagesh C Panyam, et al. 2019. Overview

of the biocreative vi precision medicine track: mining protein interactions and mutations for precision medicine. *Database*, 2019.

Qiao Jin, Bhuwan Dhingra, Zhengping Liu, William Cohen, and Xinghua Lu. 2019. 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)*, pages 2567–2577, Hong Kong, China. Association for Computational Linguistics.

Daniel Khashabi, Sewon Min, Tushar Khot, Ashish Sabharwal, Oyvind Tafjord, Peter Clark, and Hannaneh Hajishirzi. 2020. UNIFIEDQA: Crossing format boundaries with a single QA system. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1896–1907, Online. Association for Computational Linguistics.

Zeljko Kraljevic, Anthony Shek, Daniel Bean, Rebecca Bendayan, James Teo, and Richard Dobson. 2021. MedGPT: Medical Concept Prediction from Clinical Narratives. *arXiv preprint arXiv:2107.03134*.

Vishesh Kumar, Amber Stubbs, Stanley Shaw, and Özlem Uzuner. 2015. Creation of a new longitudinal corpus of clinical narratives. *Journal of biomedical informatics*, 58:S6–S10.

Kirby Kuznia, Swaroop Mishra, Mihir Parmar, and Chitta Baral. 2022. Less is more: Summary of long instructions is better for program synthesis. *arXiv preprint arXiv:2203.08597*.

Jinhyuk Lee, Wonjin Yoon, Sungdong Kim, Donghyeon Kim, Sunkyu Kim, Chan Ho So, and Jaewoo Kang. 2020. Biobert: a pre-trained biomedical language representation model for biomedical text mining. *Bioinformatics*, 36(4):1234–1240.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020. BART: Denoising sequence-to-sequence pre-training for natural language generation, translation, and comprehension. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 7871–7880, Online. Association for Computational Linguistics.

Chin-Yew Lin. 2004. Rouge: A package for automatic evaluation of summaries. In *Text summarization branches out*, pages 74–81.

Xi Victoria Lin, Todor Mihaylov, Mikel Artetxe, Tianlu Wang, Shuohui Chen, Daniel Simig, Myle Ott, Naman Goyal, Shruti Bhosale, Jingfei Du, et al. 2021. Few-shot learning with multilingual language models. *arXiv preprint arXiv:2112.10668*.

Pengfei Liu, Weizhe Yuan, Jinlan Fu, Zhengbao Jiang, Hiroaki Hayashi, and Graham Neubig. 2021. Pre-train, prompt, and predict: A systematic survey of prompting methods in natural language processing. *arXiv preprint arXiv:2107.13586*.

Nicholas Lourie, Ronan Le Bras, Chandra Bhagavatula, and Yejin Choi. 2021. UNICORN on RAINBOW: A Universal Commonsense Reasoning Model on a New Multitask Benchmark. In *Proceedings of the AAAI Conference on Artificial Intelligence*, volume 35, pages 13480–13488.

Yao Lu, Max Bartolo, Alastair Moore, Sebastian Riedel, and Pontus Stenetorp. 2021. Fantastically ordered prompts and where to find them: Overcoming few-shot prompt order sensitivity. *arXiv preprint arXiv:2104.08786*.

Man Luo, Kazuma Hashimoto, Semih Yavuz, Zhiwei Liu, Chitta Baral, and Yingbo Zhou. 2022a. Choose your qa model wisely: A systematic study of generative and extractive readers for question answering. *arXiv preprint arXiv:2203.07522*.

Man Luo, Arindam Mitra, Tejas Gokhale, and Chitta Baral. 2022b. Improving biomedical information retrieval with neural retrievers. *arXiv preprint arXiv:2201.07745*.

Bryan McCann, Nitish Shirish Keskar, Caiming Xiong, and Richard Socher. 2018. The natural language decathlon: Multitask learning as question answering. *arXiv preprint arXiv:1806.08730*.

Sewon Min, Mike Lewis, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2021. Metaicl: Learning to learn in context. *arXiv preprint arXiv:2110.15943*.

Sewon Min, Xinxi Lyu, Ari Holtzman, Mikel Artetxe, Mike Lewis, Hannaneh Hajishirzi, and Luke Zettlemoyer. 2022. Rethinking the role of demonstrations: What makes in-context learning work? *arXiv preprint arXiv:2202.12837*.

Swaroop Mishra, Daniel Khashabi, Chitta Baral, Yejin Choi, and Hannaneh Hajishirzi. 2021a. Reframing instructional prompts to gptk's language. *ACL Findings*.

Swaroop Mishra, Daniel Khashabi, Chitta Baral, and Hannaneh Hajishirzi. 2021b. Cross-task generalization via natural language crowdsourcing instructions. *ACL*.

Swaroop Mishra, Arindam Mitra, Neeraj Varshney, Bhavdeep Sachdeva, Peter Clark, Chitta Baral, and Ashwin Kalyan. 2022. NumGLUE: A Suite of Fundamental yet Challenging Mathematical Reasoning Tasks. *ACL*.

Anastasios Nentidis, Anastasia Krithara, Konstantinos Bougiatiotis, and Georgios Paliouras. 2020. Overview of bioasq 8a and 8b: Results of the eighth edition of the bioasq tasks a and b. In *CLEF (Working Notes)*.

Long Ouyang, Jeff Wu, Xu Jiang, Diogo Almeida, Carroll L Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Preprint*.

Mihir Parmar. 2021. Automation of Title and Abstract Screening for Clinical Systematic Reviews. Master's thesis, Arizona State University, Tempe, AZ, USA.

Yifan Peng, Shankai Yan, and Zhiyong Lu. 2019. Transfer learning in biomedical natural language processing: An evaluation of BERT and ELMo on ten benchmarking datasets. In *Proceedings of the 18th BioNLP Workshop and Shared Task*, pages 58–65, Florence, Italy. Association for Computational Linguistics.

Long N Phan, James T Anibal, Hieu Tran, Shaurya Chanana, Erol Bahadroglu, Alec Peltekian, and Grégoire Altan-Bonnet. 2021. Scifive: a text-to-text transformer model for biomedical literature. *arXiv preprint arXiv:2106.03598*.

N Poolsawad, C Kambhampati, and JGF Cleland. 2014. Balancing class for performance of classification with a clinical dataset. In *proceedings of the World Congress on Engineering*, volume 1, pages 1–6.

Archiki Prasad, Peter Hase, Xiang Zhou, and Mohit Bansal. 2022. Grips: Gradient-free, edit-based instruction search for prompting large language models. *arXiv preprint arXiv:2203.07281*.

Ravsehaj Singh Puri, Swaroop Mishra, Mihir Parmar, and Chitta Baral. 2022. How many data samples is an additional instruction worth? *arXiv preprint arXiv:2203.09161*.

Colin Raffel, Noam Shazeer, Adam Roberts, Katherine Lee, Sharan Narang, Michael Matena, Yanqi Zhou, Wei Li, and Peter J Liu. 2020. Exploring the limits of transfer learning with a unified text-to-text transformer. *Journal of Machine Learning Research*, 21(140):1–67.

Victor Sanh, Albert Webson, Colin Raffel, Stephen H Bach, Lintang Sutawika, Zaid Alyafeai, Antoine Chaffin, Arnaud Stiegler, Teven Le Scao, Arun Raja, et al. 2021. Multitask prompted training enables zero-shot task generalization. *arXiv preprint arXiv:2110.08207*.

Timo Schick and Hinrich Schütze. 2021. It's not just size that matters: Small language models are also few-shot learners. In *Proceedings of the 2021 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2339–2352, Online. Association for Computational Linguistics.

Roy Schwartz, Jesse Dodge, Noah A Smith, and Oren Etzioni. 2020. Green ai. *Communications of the ACM*, 63(12):54–63.

Emma Strubell, Ananya Ganesh, and Andrew McCallum. 2019. Energy and policy considerations for deep learning in NLP. In *Proceedings of the 57th Annual Meeting of the Association for Computational Linguistics*, pages 3645–3650, Florence, Italy. Association for Computational Linguistics.

Amber Stubbs, Christopher Kotfila, Hua Xu, and Özlem Uzuner. 2015. Identifying risk factors for heart disease over time: Overview of 2014 i2b2/uthealth shared task track 2. *Journal of biomedical informatics*, 58:S67–S77.

Yuka Tateisi, Akane Yakushiji, Tomoko Ohta, and Jun'ichi Tsujii. 2005. Syntax annotation for the genia corpus. In *Companion Volume to the Proceedings of Conference including Posters/Demos and tutorial abstracts*.

Özlem Uzuner, Yuan Luo, and Peter Szolovits. 2007. Evaluating the state-of-the-art in automatic de-identification. *Journal of the American Medical Informatics Association*, 14(5):550–563.

Benyou Wang, Qianqian Xie, Jiahuan Pei, Prayag Tiwari, Zhao Li, et al. 2021a. Pre-trained language models in biomedical domain: A systematic survey. *arXiv preprint arXiv:2110.05006*.

Liwen Wang, Rumei Li, Yang Yan, Yuanmeng Yan, Sirui Wang, Wei Wu, and Weiran Xu. 2022. Instructionner: A multi-task instruction-based generative framework for few-shot ner. *arXiv preprint arXiv:2203.03903*.

Sinong Wang, Han Fang, Madian Khabsa, Hanzi Mao, and Hao Ma. 2021b. Entailment as few-shot learner. *arXiv preprint arXiv:2104.14690*.

Albert Webson and Ellie Pavlick. 2021. Do prompt-based models really understand the meaning of their prompts? *arXiv preprint arXiv:2109.01247*.

Jason Wei, Maarten Bosma, Vincent Y Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M Dai, and Quoc V Le. 2021. Finetuned language models are zero-shot learners. *arXiv preprint arXiv:2109.01652*.

Orion Weller, Nicholas Lourie, Matt Gardner, and Matthew E. Peters. 2020. Learning from task descriptions. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 1361–1375, Online. Association for Computational Linguistics.

Thomas Wolf, Lysandre Debut, Victor Sanh, Julien Chaumond, Clement Delangue, Anthony Moi, Pierric Cistac, Tim Rault, Remi Louf, Morgan Funtowicz, Joe Davison, Sam Shleifer, Patrick von Platen, Clara Ma, Yacine Jernite, Julien Plu, Canwen Xu, Teven Le Scao, Sylvain Gugger, Mariama Drame, Quentin Lhoest, and Alexander Rush. 2020. Transformers: State-of-the-art natural language processing. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing: System*

Demonstrations, pages 38–45, Online. Association for Computational Linguistics.

Tongshuang Wu, Ellen Jiang, Aaron Donsbach, Jeff Gray, Alejandra Molina, Michael Terry, and Carrie J Cai. 2022. Promptchainer: Chaining large language model prompts through visual programming. *arXiv preprint arXiv:2203.06566*.

Tongshuang Wu, Michael Terry, and Carrie J Cai. 2021. Ai chains: Transparent and controllable human-ai interaction by chaining large language model prompts. *arXiv preprint arXiv:2110.01691*.

Tianbao Xie, Chen Henry Wu, Peng Shi, Ruiqi Zhong, Torsten Scholak, Michihiro Yasunaga, Chien-Sheng Wu, Ming Zhong, Pengcheng Yin, Sida I Wang, et al. 2022. Unifiedskg: Unifying and multi-tasking structured knowledge grounding with text-to-text language models. *arXiv preprint arXiv:2201.05966*.

Qinyuan Ye and Xiang Ren. 2021a. Learning to generate task-specific adapters from task description. In *Proceedings of the 59th Annual Meeting of the Association for Computational Linguistics and the 11th International Joint Conference on Natural Language Processing (Volume 2: Short Papers)*, pages 646–653, Online. Association for Computational Linguistics.

Qinyuan Ye and Xiang Ren. 2021b. Zero-shot learning by generating task-specific adapters. *arXiv e-prints*, pages arXiv–2101.

Jieyu Zhao, Daniel Khashabi, Tushar Khot, Ashish Sabharwal, and Kai-Wei Chang. 2021. Ethical-advice taker: Do language models understand natural language interventions? In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 4158–4164, Online. Association for Computational Linguistics.

Ruiqi Zhong, Kristy Lee, Zheng Zhang, and Dan Klein. 2021. Adapting language models for zero-shot learning by meta-tuning on dataset and prompt collections. In *Findings of the Association for Computational Linguistics: EMNLP 2021*, pages 2856–2878, Punta Cana, Dominican Republic. Association for Computational Linguistics.

A Statistics of Instruction Tasks

This section provides all the statistics of training, validation and inference data used for experiments in Table 3. All the number of instances provided in Table 3 are calculated after discarding the instances with more than 1024 token length as described in the section 5.1. We have divided the dataset into standard 70/10/20 splits for train/validation/test if there is no separate validation and testing set provided in the dataset.

B Instruction Tasks and Examples

To build all the models (baselines, proposed model and few-shot learning), we adapt the unified format for all the tasks of BoX. We converted all the tasks into the text-to-text format, including the classification tasks. Table 4 shows an example of input and output from each category. Moreover, we have also re-purposed some biomedical datasets to create more than one task as described in the section 3.1.

C Systematic Review Datasets

This section describes the brief data creation process for Systematic Reviews (SRs) that are used in this study. The relentless growth in clinical research and published articles have created a need for automation to expedite the process of SRs and to enable Living Systematic Reviews (LSRs). A crucial step in both SRs and LSRs is the title and abstract-based screening of the articles. A new dataset was developed from six SRs in the clinical domain by Mayo clinic physicians. In this study, we used data from the following five SRs that were conducted using the traditional (manual) process and published in relevant venues: (1) Hormone Replacement Therapy (HRT), (2) Cooking, (3) Accelerometer, (4) Acromegaly, and (5) COVID. The initial bibliographic search was designed and conducted by an experienced librarian with guidance from the principal investigators for the respective studies. The search was conducted in different bibliographic databases like PubMed, PubMed Central (PMC), Embase, EBM Reviews, and Ovid MEDLINE(R). Each article in the bibliographic search results was categorized by two physicians with domain expertise as "Include" or "Exclude", by reading the title and abstract of the article. When there was a disagreement between two annotators, a positive class (i.e., "Include") was preferred.

D Few-Shot Learning results

This section presents the results of few-shot learning for all instruction tasks in Table 5.

E State-of-the-art results

In Table 6, we present State-Of-The-Art (SOTA) results for 21 tasks. To compare the SOTA results with the proposed model, we calculate the corresponding metric used in particular research from our model predictions. For each task, we gather the best performance, and specifically, they are BioASQ-8b (Nentidis et al., 2020), Chemprot (Peng et al., 2019), DDI (Peng et al., 2019). In Chemprot and DDI, we compare results with the base LMs instead of large for a fair comparison. SOTA results for all 15 NER datasets are obtained from (Banerjee et al., 2021). Best performance for the HoC dataset is obtained from (Peng et al., 2019). Here, we have considered the result of the best system submitted to (Stubbs et al., 2015) as SOTA result.

Category	Tasks	# of Instances		
		Train	Dev	Test
NER	AnatEM	3507	1121	2303
	BC2GM	6427	1291	2570
	BC4CHEMD	14466	14568	12397
	BC5CDR	4940	4940	5158
	BioNLP11EPI	3796	1242	2836
	BioNLP11ID	2466	780	1869
	BioNLP13CG	4591	1489	2759
	BioNLP13GE	1503	1663	1937
	BioNLP13PC	2945	1070	1997
	BioNLP09	4710	1013	1699
	CRAFT	12839	4423	8882
	Ex-PTM	855	278	1160
	JNLPBA	15124	1533	3152
	NCBI	2922	488	538
	linnaeus	1484	524	993
DI	DI 2006	106	22	27
POS	Genia	16323	2174	2035
QA	BioASQ8b (factoid)	695	16	115
	BioASQ8b (list)	373	8	45
	BioASQ8b (yesno)	543	16	115
	PubMedQA	4167	500	473
RE	ChemProt	3350	2415	2660
	DDI	20009	2780	2660
SA	Medical Drugs	2860	526	804
SR	Accelerometer	499	58	142
	Acromegaly	663	80	192
	COVID	2385	300	675
	Cooking	735	84	205
	HRT	1479	171	410
DC	HoC	3119	445	890
RFI	RFHD 2014 (yesno)	834	360	451
	RFHD 2014 (time-riskfactor)	152	177	69
Total	-	140795	46554	64561

Table 3: Statistics of training (i.e., Train), validation (i.e, Dev) and evaluation (i.e., Test) data for all instruction tasks from the BoX. RFHD: Risk Factor for Heart Disease.

Category	Task	Input	Output
NER	BC5CDR	Such interactions may result in serious cardiovascular complications even after cessation of an infusion of ritodrine.	cardiovascular complications <Disease>, ritodrine <Chemical>
de-identification	DI2006	757085252 HLGMC 1228824 18705/6o5b 3/25/1993 12:00:00 AM CONGESTIVE HEART FAILURE . Unsigned DIS Report Status : Unsigned ADMISSION DATE : 3/25/93 DISCHARGE DATE : 4/4/93 PRINCIPAL DIAGNOSIS : congestive heart failure . ASSOCIATED DIAGNOSIS : aortic stenosis ; coronary artery disease , status post multi vessel coronary artery bypass graft surgery , ... , M.D. TR : go / bmot DD : 4/4/93 TD : 04/06/93 CC : [report_end]	3/25 <DATE>, 18705/6o5b <ID>, 757085252 <ID>, go / bmot <DOCTOR>, 4/4 <DATE>, 04/06 <DATE>
POS-Tagging	Genia	Binding sites were mapped for each factor .	Binding <VBG> sites <NNS> were <VBD> mapped <VBN> for <IN> each <DT> factor <NN> . <.>
QA	BioASQ8b (factoid)	Context: Hyperosmia is suspected in pregnancy; however, no empirical study using validated measures of olfactory function has clearly confirmed the anecdotal reports of this phenomenon. subjective hyperosmia is associated with primarily negative odor-related experiences. Hyperosmia is increased olfactory acuity \n Question: What is hyperosmia	Hyperosmia is increased olfactory acuity.
RE	Drug-Drug Interaction	Context: Antacids may interfere with the absorption of LEVSIN. Drug_1: Antacids Drug_2: LEVSIN	true
Sentiment Analysis	Medical Drugs	Why don't more folk opt for Cladribine? \n Drug: cladribine \n Option1: Neutral Option2: Positive Option3: Negative	Positive
Systematic Review	Acromegaly	No greater incidence or worsening of cardiac valve regurgitation with somatostatin analog treatment of acromegaly CONTEXT: Excess GH and IGF-I in acromegaly are associated with reduced life expectancy due to cardiovascular complications. Option_1: Include, Option_2: Exclude.	Include
Document Classification	Hallmarks of Cancer (HoC)	Studies of cell-cycle progression showed that the anti-proliferative effect of Fan was associated with an increase in the G1/S phase of PC3 cells.	Evading growth suppressors, Sustaining proliferative signaling
Risk Factor Identification	n2c2 - Risk Factors Heart Disease 2014 (yesno)	Context: Record date: 2157-08-27 History of Present Illness ID:Admitted from cardiac cath lab. HPI:Mr. Doty is a 80 y.o. male with h/o HTN, DM, PVD, elevated cholesterol who presents with 6 month h/o chest and upper extremity discomfort on exertion along with SOB. He has limited his activities to prevent symptoms. ... \n Risk Factor: Diabetes	Yes

Table 4: Examples of one instruction tasks converted into text-to-text format for each category

Category	Task	32		100		1k		4k	
		S	I-BB	S	I-BB	S	I-BB	S	I-BB
NER	AnatEM	12.74	60.73	20.68	79.34	87.81	86.76	84.88	83.44
	BC2GM	16.92	65.65	21.31	70.39	82.92	77.19	77.66	74.11
	BC4CHEMD	10.55	71.05	14.93	73.85	86.53	83.75	88.85	86.19
	BC5CDR	11.75	60.37	12.58	67.51	69.62	73.66	74.83	74.34
	BioNLP11EPI	31.14	78.64	42.31	81.51	85.71	85.57	84.64	86.68
	BioNLP11ID	11.00	62.38	10.06	68.92	71.41	71.62	71.08	71.96
	BioNLP13CG	12.39	49.15	12.53	52.68	55.23	63.15	64.19	67.23
	BioNLP13GE	26.10	78.80	25.00	81.82	84.77	84.29	83.74	85.58
	BioNLP13PC	12.40	69.29	12.59	71.89	68.11	68.49	70.42	71.97
	BioNLP09	32.51	78.17	30.51	82.71	87.48	86.39	85.16	86.33
	CRAFT	8.07	37.35	8.60	40.38	49.67	51.56	63.72	63.35
	Ex-PTM	16.06	74.32	47.93	76.15	82.92	84.11	82.32	83.81
	JNLPBA	20.15	57.61	19.77	59.54	64.46	63.63	71.65	70.45
	NCBI	38.69	68.82	30.46	79.35	93.02	90.36	89.51	86.46
	linnaeus	28.75	58.69	36.94	67.29	93.81	92.50	94.43	70.57
	Average	19.28	**64.74**	23.08	**70.22**	77.56	77.54	**79.14**	77.50
DI	DI 2006	12.67	**50.19**	13.30	**49.54**	13.54	**55.28**	12.60	**50.10**
POS	Genia	**51.48**	13.41	**48.26**	30.65	**66.27**	61.93	**71.45**	70.57
QA	BioASQ8b (factoid)	36.63	35.99	41.89	40.77	51.96	49.84	52.95	51.72
	BioASQ8b (list)	14.99	20.91	19.66	29.38	40.14	29.59	38.96	34.68
	BioASQ8b (yesno)	43.48	61.11	39.13	57.94	66.96	60.32	56.52	52.17
	PubMedQA	17.32	19.28	25.16	23.26	27.68	25.86	27.12	24.96
	Average	28.11	**34.32**	31.46	**37.84**	**46.68**	41.40	**43.89**	40.88
RE	ChemProt	61.64	72.02	66.07	64.91	66.01	55.22	76.86	77.38
	DDI	85.53	77.37	85.53	81.37	46.99	55.41	87.39	73.04
	Average	73.59	**74.70**	**75.80**	73.14	**56.50**	55.31	**82.12**	75.21
SA	Medical Drugs	33.29	**63.48**	24.51	**63.66**	**43.41**	31.58	37.31	**49.50**
SR	Accelerometer	76.76	77.78	75.35	68.06	83.80	73.61	72.54	70.83
	Acromegaly	80.21	80.71	81.25	75.63	76.56	79.19	76.04	77.66
	COVID	87.85	88.36	87.85	84.85	61.93	86.96	73.93	78.12
	Cooking	88.29	87.08	87.80	87.56	81.95	87.08	80.98	82.78
	HRT	85.86	86.02	85.61	75.12	89.08	81.99	83.87	80.81
	Average	83.79	**83.99**	**83.57**	78.24	78.66	**81.77**	77.47	**78.04**
DC	HoC	17.06	**19.87**	17.98	**27.13**	46.94	**52.36**	**88.53**	81.51
RFI	RFHD 2014 (yesno)	57.21	51.78	57.21	51.50	43.02	66.35	43.86	66.46
	RFHD 2014 (time-riskfactor)	54.51	64.22	52.75	63.37	66.18	59.60	66.18	62.70
	Average	55.86	**58.00**	54.98	**57.43**	54.60	**62.98**	54.93	**64.58**
Average	-	37.31	**60.64**	39.24	**63.38**	66.75	**67.98**	69.81	**70.23**

Table 5: Comparison of few-shot learning results in terms of Rouge-L between single-task models and In-BoXBART for 32/100/1000 training samples per instruction tasks. All results are presented in %. S: Single-task model, I-BB: In-BoxBART, RFHD: Risk Factor for Heart Disease.

Category	Task	Metric	SOTA	Multi-Task	
				V-BB	I-BB
NER	AnatEM	F	91.61	33.50	84.61
	BC2GM	F	83.47	50.86	75.03
	BC4CHEMD	F	92.39	71.44	86.97
	BC5CDR	F	90.50	70.11	75.24
	BioNLP11EPI	F	88.66	52.85	88.04
	BioNLP11ID	F	87.36	60.15	73.39
	BioNLP13CG	F	90.16	53.88	65.09
	BioNLP13GE	F	85.81	51.78	87.39
	BioNLP13PC	F	91.65	51.61	67.77
	BioNLP09	F	91.94	54.31	88.48
	CRAFT	F	90.12	52.31	64.03
	Ex-PTM	F	87.08	52.07	84.49
	JNLPBA	F	79.19	68.60	70.26
	NCBI	F	89.82	75.55	86.91
	linnaeus	F	95.68	44.59	93.77
QA	BioASQ8 (list)	F	52.99	17.74	35.59
	BioASQ8 (yesno)	F	89.95	62.61	68.25
RE	Chemprot	F	74.40	52.17	63.22
	DDI	F	79.40	82.97	89.35
DC	HoC	F	85.30	49.51	82.53
RFI	RFHD 2014 (time-riskfactor)	F	92.76	0.97	85.28
Average	-	-	85.55	50.36	72.24

Table 6: The state-of-the-art (SOTA) results for each task compared with Vanilla-BoXBART and In-BoXBART. All results are in %. F: F_1-score, V-BB: Vanilla-BoXBART, I-BB: In-BoXBART, RFHD: Risk Factor for Heart Disease.