`TOP-Training`: Target-Oriented Pretraining for Medical Extractive Question Answering

Saptarshi Sengupta [1], Connor Heaton [1], Shreya Ghosh [2],
Wenpeng Yin [1], Preslav Nakov [3], Suhang Wang[1]
[1]Pennsylvania State University, USA
[2]Indian Institute of Technology (IIT) Bhubaneswar, India
[3]Mohamed bin Zayed University of Artificial Intelligence, UAE
{sks6765, czh5372, wenpeng, szw494}@psu.edu
shreya@iitbbs.ac.in, preslav.nakov@mbzuai.ac.ae

Abstract

We study extractive question-answering in the medical domain (`Medical-EQA`). This problem has two main challenges: (i) domain specificity, as most AI models lack necessary domain knowledge, and (ii) extraction-based answering style, which restricts most autoregressive LLMs due to potential hallucinations. To handle those challenges, we propose `TOP-Training`, a target-oriented pretraining paradigm that stands out among all domain adaptation techniques with two desirable features: (i) `TOP-Training` moves one step further than popular domain-oriented fine-tuning since it not only moves closer to the target domain, but also familiarizes itself with the target dataset, and (ii) it does not assume the existence of a large set of unlabeled instances from the target domain. Specifically, for a target `Medical-EQA` dataset, we extract its entities and leverage large language models (LLMs) to generate synthetic texts containing those entities; we then demonstrate that pretraining on this synthetic text data yields better performance on the target `Medical-EQA` benchmarks. Overall, our contributions are threefold: (i) `TOP-Training`, a new pretraining technique to effectively adapt LLMs to better solve a target problem, (ii) `TOP-Training` has a wide application scope because it does not require the target problem to have a large set of unlabeled data, and (iii) our experiments highlight the limitations of autoregressive LLMs, emphasizing `TOP-Training` as a means to unlock the true potential of bidirectional LLMs.[1]

1 Introduction

The escalating volume of electronic health records (EHR) underscores the growing significance of information extraction (IE) from these datasets. This includes tasks like identifying the medications a patient is currently taking and uncovering recorded drug allergies or adverse reactions. Extractive Question Answering (EQA) plays a central role in EHR-based IE, wherein the system must provide a relevant textual excerpt from medical records based on a query. This is commonly referred to as Medical Extractive Question Answering (`Medical-EQA`) (Tian et al., 2023). The primary challenges in `Medical-EQA` stem from i) limited data availability, especially expert-labeled data; ii) the presence of rare medical terminologies, which many AI models struggle to recognize and interpret; and iii) privacy concerns that restrict the unfettered use or scraping of open resources.

Despite considerable research efforts and advancements in `Medical-EQA`, prior work grapples with two main issues: i) Some approaches rely on the assumption of having access to extensive unlabeled data for domain adaptation, which is often impractical due to privacy constraints (Brown et al., 2022); ii) The recent adoption of decoder-only generative large language models (LLMs) pre-trained on vast datasets holds promise but suffers from *hallucination* issues (Ji et al., 2023; Pal et al., 2023), limiting their reliability in medical applications. This raises critical questions: firstly, how can models be trained effectively when the target medical dataset is severely limited, with scarce training data and unavailable unlabeled data? Secondly, how can we harness the rich pre-trained knowledge in LLMs while mitigating hallucination problems?

In this study, we propose `TOP-Training`, a target-oriented pretraining technique as a solution to these challenges, addressing them from two angles: i) For target datasets with constrained scales, `TOP-Training` extracts medical entities and prompts LLMs to generate large-scale synthetic data tailored to the target domain, thereby enhancing the model's comprehension of rare medical terminologies; ii) `TOP-Training` reconfigures

---

LLMs as bidirectional encoders to ensure output consistency, alleviating hallucination issues. The fundamental rationale behind `TOP-Training` is to utilize the resource-limited target data as a seed, without accessing gold outputs, and to prompt generative LLMs to produce substantial unlabeled data, thus bolstering extractive LLMs for improved `Medical-EQA` performance.

We evaluate `TOP-Training` on two prototypical `Medical-EQA` datasets: COVID-QA (Möller et al., 2020) and RadQA (Soni et al., 2022). Our experiments reveal that: i) generative LLMs underperform significantly on these datasets; ii) `TOP-Training` achieves state-of-the-art performance with moderate-sized synthetic data, offering advantages over baselines that rely on billion-level pre-trained data, often inaccessible in practical settings; iii) `TOP-Training` demonstrates robustness across different encoder-based LLMs, such as BERT and RoBERTa, as well as varying sizes and lengths of synthetic data. In summary, our contributions are threefold:

- We propose a novel LLM-based synthetic data generation approach that leverages medical domain-specific entities to bridge knowledge between target data and synthetic data, facilitating a pretraining stage closely aligned with the target problem and boasting broad application potential.

- We offer insights into why generative LLMs struggle in `Medical-EQA` and why extractive LLMs are better suited for this task.

- We present a fresh perspective on domain adaptation, suggesting a redefinition of "domain" to incorporate not only textual genre but also dataset-specific characteristics, thereby providing a more nuanced understanding of adaptation challenges.

2 Related Work

In this section, we discuss prior work on domain-specific pre-training, decoder-based Foundation Models (FM) for QA, highlighting their limitations, and, efficient methods for `Medical-EQA`.

Autoregressive modelling for QA   On release, GPT-4 (Achiam et al., 2023) became *the* benchmark FM on almost all canonical NLP tasks. Remarkably, Nori et al. (2023) showed that just by careful prompting, GPT-4 can achieve SOTA on various medical QA datasets. Initiatives such as Meditron-70B (Chen et al., 2023) came next to parallel GPT-4's medical expertise. On closer inspection, however, we see that the datasets on which these two models are applied are all multiple-choice style. Thus, their capability of handling EQA datasets, especially in the medical domain, remains unexplored (cf. App. C).

Xu et al. (2021) train BART (Lewis et al., 2020a) for span extraction by considering cross-attention weights as start/end token probabilities to align the generated and true spans while Mallick et al. (2023) reframe the problem as generating *numeric indices* indicating either token or sentence-level spans. However, we see two issues, i) marginal performance gains, and ii) only Mallick et al. (2023) test a `Medical-EQA` dataset, that too without much success as compared to their encoder-based SOTA indicating their ineffectiveness. Luo et al. (2022) show that *encoders outperform decoders* in span-detection with the added advantage of being better at *out-of-domain generalization*. While they do make a case for autoregressive FMs, Liu et al. (2024) demonstrate how newer instruction-tuned FMs are *sensitive to the location of the gold span*.

Domain Specific Pre-Training   Gururangan et al. (2020) introduced DAPT (Domain-Adaptive Pretraining) and TAPT (Task-Adaptive Pretraining), which share similarities with our work. DAPT involves extended pretraining on domain-specific corpora without labels, while TAPT focuses on pretraining on the unlabelled training set of the downstream task. Although they demonstrate the effectiveness of TAPT compared to DAPT, closed-domain datasets like COVID-QA typically lack a separate unlabelled training set and may not even have train/dev/test splits. Further, DAPT considers knowledge beyond what is necessary to the task data, whereas our approach confines training to only required concepts.

DAPT/TAPT requires high quantities of unlabelled corpora to yield useful results, thus raising the question: *What happens when we do not have enough "relevant" domain data, either in style or volume*? This inspires `TOP-Training`, which focuses on a subset of a domain, tailor-made for the downstream dataset, similar in motivation with Gunasekar et al. (2023); Zhou et al. (2023), who make use of smaller yet *better quality* corpora.

Efficient Methods in `Medical-EQA`   We identify seven relevant works on COVID-QA. First off,
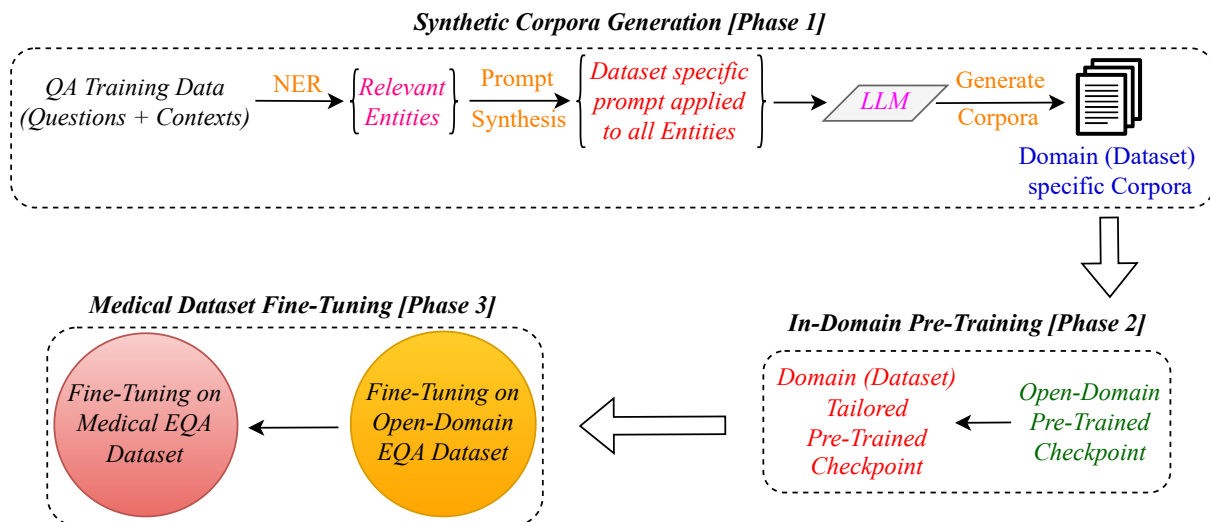
Figure 1: `TOP-Training`. First, we extract relevant entities from the target dataset to generate our synthetic pre-training corpora. Next, we train an open-domain model on this corpus followed by two rounds of EQA fine-tuning, i.e., first on an open-domain dataset to learn what EQA is as a task and then on the target `Medical-EQA` dataset.

Haddouche et al. (2023) perform a simple evaluation of BERT and RoBERTa and achieve scores similar to ours, thus serving as a sanity check. Zafar et al. (2024a,b); Poerner et al. (2020) incorporate medical information either via knowledge graphs or training external embeddings on biomedical corpora ranging from 2 to 94 GB. While the former method is ineffective (as showcased by the results), the latter uses much more resources than our method and is still inferior to us. Levy et al. (2021) create a pipeline to provide high-quality references for user queries and shows strong performance on COVID-QA. However, there is a caveat to these results. Apart from Poerner et al. (2020), Zafar et al. (2024a,b); Levy et al. (2021) all use a **subset of COVID-QA.** Samuel et al. (2023) train on synthetic data from GPT-4 but show weak overall performance as GPT might be unable to create text in the style of COVID-QA. Finally, Seo et al. (2024) generates training data by retrieving samples from a related dataset and prompting an LLM using the original and retrieved data. Although they use fewer samples than us, they generate questions, contexts and answers while we only generate contexts leading to an overall simpler framework capable of outperforming theirs.

For RadQA, we identify three relevant papers. Ghosh et al. (2023) train `RadLing`, an ELECTRA (Clark et al., 2020) model on a collection of 500K radiology reports which we outperform using far less data (c.f. Table 2). Hernandez et al. (2023) conduct a comprehensive analysis of clinical FMs in the radiology domain and show SOTA performance by BioClinicalRoBERTa (Lewis et al., 2020b) once more reinforcing encoder models superiority to causal models for EQA. Finally, Lu et al. (2024) propose a method to encode input text once and have it be shared across each decoder prompt and apply their method to RadQA to obtain similar EM as the best T5 model shown in Hernandez et al. (2023). Both Lu et al. (2024) and Hernandez et al. (2023) show strong performance using Clinical-T5$_{\text{LARGE}}$ which uses much more data and is a larger model than ours. Despite this, we show how competitive our method is at a fraction of the cost.

3 Formulation of `Medical-EQA`

Each labelled data point consists of three elements, I) CONTEXT ($\mathcal{C}$): A piece of text in the medical domain that introduces the necessary information about a topic; II) QUESTION ($\mathcal{Q}$): A question sentence acquiring the information from CONTEXT. *It can be answerable or unanswerable* and III) ANSWER ($\mathcal{A}$): A consecutive span in CONTEXT that acts as the answer of QUESTION.

Typical datasets for `Medical-EQA` are split into $train$, $test$ and optional dev sets. There are two main challenges for `Medical-EQA` i) domain specificity; most AI models lack the necessary knowledge; ii) extraction-based answering style, which restricts most autoregressive LLMs due to hallucination concerns. These challenges motivate us to propose `TOP-Training` accompanied with bi-directional LLMs for this particular problem.

4 `TOP-Training`

`TOP-Training`, shown in Figure 1, first extracts entities from the target `Medical-EQA` dataset, then leverages an existing LLM to generate entity-related CONTEXT (Section 4.1), which acts as the data for further tuning a bi-directional LLM for the target problem (Section 4.2).

4.1 LLM generates target-oriented CONTEXT

Entity collection from the target problem. In this work, we use entities in the target dataset as the connection between it with the newly generated synthetic data. To extract entities from the target EQA dataset, first, we combine all the QUESTIONS and CONTEXTS from the *train* split of the `Medical-EQA` dataset. Next, we extract entities through Named Entity Recognition (NER) using spaCy[2]. This step identifies roughly 47k and 11k entities in COVID-QA and RadQA. These entities are aligned with the medical space such as `DC-SIGNR`, `MTCT`, `C-terminal domain`, etc. which are quite different from general-domain entities (person/place/thing) such as *New York*, *John Doe*, etc.

Synthetic CONTEXT generation. Next, we create prompts for the identified entities to generate CONTEXTS mimicking the target datasets. This required studying the characteristics of the target datasets such as the text genre (full research articles in COVID-QA and radiology reports in RadQA), lengths, and relevant keywords. Galactica (Taylor et al., 2022), a generative LLM pre-trained on a collection of text encompassing research articles, knowledge bases, code and LaTeX markup, is used to generate our synthetic data[3]. We choose Galactica over other LLMs for two reasons, a) It possesses a built-in ability to generate research paper-like content (see 4.1) b) Galactica is a general-purpose science model that allows future work to extend our framework to domains beyond biomedical.

The prompts used to generate aligned data [4] for each dataset are described below.

• **Prompt for COVID-QA.** Since COVID-QA comprises research papers, based on this characteristic, we develop the following prompt for

Galactica to generate pseudo-research articles based on retrieved entities. Note, `Title` is the prompt handle/keyword and `entity` is the entity identified in the above stage.

> *Title*: *Entity*

• **Prompt for RadQA.** RadQA's CONTEXTS are redacted radiology reports without any consistent format (Hartung et al., 2020). The *Findings* and *Impressions* sections are the most vital in a patient's report (akin to the experiment and results section in a research paper). Inspired, we propose the following prompt:

> *Patient has* *Entity*. *FINDINGS* *AND* *IMPRESSION*:

It is worth noting that **Galactica had not been trained on radiology reports. Through this prompt, we synthesize *pseudo-reports* bypassing any privacy concerns**. To maintain size parity between the two target datasets, five CONTEXTS are generated for each entity identified in RadQA, yielding around 55k (11k*5) total CONTEXTS, and one CONTEXT for each identified entity in COVID-QA, resulting a set of size 47K.

4.2 System pre-training on synthetic data

After generating CONTEXTS for each target data, we perform `TOP-Training` i.e., extended pre-training of BERT/RoBERTa on our generated corpus. In addition, `TOP-Training` is followed by two rounds of fine-tuning where a) the model will first be fine-tuned on the SQuAD dataset (Rajpurkar et al., 2016, 2018) to learn what EQA is in a general domain and b) it is further fine-tuned on the *train* of each target data (either COVID-QA or RadQA), to solve the *test*.

5 Experiments

We focus on two datasets: COVID-QA, comprising 2,019 answerable QA pairs (no train/dev/test splits) sourced from CORD-19 (Wang et al., 2020), and RadQA, consisting of 6,148 QA pairs from radiology reports, with a train/dev/test split of 4,878/656/614. We experiment in two areas: baselines and `TOP-Training`, and provide mean and standard deviation scores over three random seeds.

---

[2]With "en_core_sci_sm" (https://allenai.github.io/scispacy/).

[3]Other generative models such as BLOOM (Scao et al., 2022) and PubMedGPT performed worse in our experiments.

[4]See Appendix. B

5.1 Baselines & influencing factors

Apart from comparison with existing works (c.f. 2), we consider 13 encoder models in total taking into account which model made the most sense to apply to either dataset. On COVID-QA, we applied models from checkpoints fine-tuned on SQuAD v1 (Rajpurkar et al., 2016) while RadQA, containing unanswerable questions, was tackled with those fine-tuned on SQuAD v2 (Rajpurkar et al., 2018). We begin with SQuAD-trained checkpoints as it has been shown (Soni et al., 2022; Castelli et al., 2020) that models benefit from a first round of training on SQuAD. For consistency, we use the *cased*, *base* version of each architecture when available. We use five-fold cross-validation for fine-tuning and report the results in Table 1 (COVID-QA). Results of models applied to the prescribed splits (RadQA) are presented in Table 2. The metrics used are exact match (EM), a binary measure of whether the prediction and gold-standard spans are identical and F1, the harmonic mean of the number of shared words in the two spans with respect to the number of words in the prediction (precision) and with respect to the number of words in the gold-standard span (recall).

We use Galactica-1.3B for consistency with our corpus generation experiments, MedLLaMA(13B) and MedAlpaca(13B) as strong open-source medical checkpoints. We measure the ability of the three decoder models to generate answers without fine-tuning, considering that decoders do not extract spans, but generate answers, for comparison to the gold-standard spans. Following Yue et al. (2021), each sample is formatted as,

```
Question:<question_text>
Context:<part_of_context>
Answer:
```

Due to the large size of COVID-QA contexts, they were segmented as they exceeded the maximum sequence length of each model (2,048 tokens). We report overall EM/F1 on each dataset and average best EM/F1 (parenthesis in Table 1) from each Q+C+A chunk for COVID-QA (N/A for RadQA since the context size was much smaller than the models' maximum input length).

Corpus Size. We investigated the impact of synthetic dataset size on downstream performance in COVID-QA. We examined the effects of generating one and 10 contexts per entity.

Context Length. The average context length for COVID-QA is 6K tokens, and Galactica has a maximum window of 2K, resulting in a misalignment between the synthetic corpus and the target dataset. We cannot increase the context size of Galactica. Training it from scratch with architectural changes is infeasible for us. Thus, we explore the impact of sequence length in the synthetic corpus by limiting the records to only 1k tokens. We cannot determine if *longer* sequences are *beneficial*; we can see if *shorter* ones are *detrimental*.

Prompting Style. We explore the use of two different prompts when encouraging Galactica to generate *pseudo* radiology reports - as defined in section 4.1 which we call *Fancy prompt* and *Normal prompt* as simply "`[entity]`".

Human-Generated Contexts. We establish a *Wikipedia* baseline alongside our domain-specific models to assess the influence of content and text structure during domain adaptation. Additionally, Micallef et al. (2022) have shown that mBERTu Wiki, pre-trained on Maltese Wikipedia data, surpassed the performance of mBERT, thereby proving to be a competitive baseline. For each entity, we query Wikipedia and retrieve the complete page associated with the top search result. The number of entities available for this baseline is much smaller than that in our approach since most of the entities do not exist in Wikipedia due to either being extremely esoteric, e.g., `pulmonary parenchymal infiltrate` or improperly formed, e.g., `Bao &`.

6 Results & Discussion

Results for COVID-QA and RadQA are presented in Table 1 and 2. Each table is organized as scores from related work, our benchmarks and `TOP-Training`. The last 3 rows of the benchmark section provide the zero-shot performance of our chosen decoder models on each dataset. We do not perform multiple trials here as the extremely poor performance would not benefit from additional runs. Overall, we see that MedAlpaca is the best among the three for RadQA and only marginally poorer in terms of F1 for COVID-QA. On COVID-QA, no model generates text w.r.t the gold standard and only shows positive F1.

6.1 COVID-QA

We now discuss the results of the benchmarking trials on COVID-QA contrasting the methods.

	Approach/Model	Training Corpus	Corpus Size	Time#	EM	F1
Related Work	Haddouche et al. (2023)				38.61	64.87
	Möller et al. (2020)				25.9	59.53
	Levy et al. (2021)				39.16	72.03
	Poerner et al. (2020)	N/A	N/A	N/A	34.62	60.23
	Zafar et al. (2024a)				31.92	59.57
	Zafar et al. (2024b)				35.3	58.64
	Samuel et al. (2023)				31.9	58.66
	Seo et al. (2024)				-	**67.95*** (68.36)
Our Benchmarks	BioBERT	PubMed	4.5B words		37.62 ± 0.18	65.73 ± 0.35
	SciBERT	Semantic Scholar	3.2B words		37.52 ± 0.23	65.58 ± 0.18
	+CORD-19	+ CORD-19	3.2B words + 20GB		35.61 ± 0.30	63.60 ± 0.59
	PubMedBERT	PubMed	3.1B words / 21GB		39.87 ± 0.74	68.47 ± 0.13
	BlueBERT [1]	PubMed + MIMIC	4.5B words		27.35 ± 0.30	52.18 ± 0.40
	CODER [2]	UMLS	N/A		39.33 ± 0.47	67.01 ± 0.30
	Longformer [3]	General Domain	6.5B tokens	N/A	37.79 ± 0.39	66.58 ± 0.22
	BigBird [4]	General Domain	160GB		32.79 ± 0.13	60.06 ± 0.41
	LUKE [5]	Wikipedia	3.5B words		41.01 ± 0.30	68.23 ± 0.19
	XLNET	General Domain	32.89B words		2.45 ± 0.08	8.64 ± 0.19
	Galactica	c.f. section 4	106B tokens		0 (0)	5.01 (11.11)
	MedLLaMA	Medical Corpora	N/A		0 (0)	**5.81 (12.79)**
	MedAlpaca	Medical Meadow	N/A		**0.03 (0.2)**	5.21 (12.73)
Proposed Method	BERT	Vanilla Fine-Tuning	N/A	N/A	33.62 ± 0.59	60.01 ± 0.36
	RoBERTa				38.89 ± 0.52	67.44 ± 0.47
	BERT	Wikipedia	139.6 MB	≈ 2.5 hrs	33.95 ± 0.13	60.76 ± 0.78
	RoBERTa				40.33 ± 0.60	68.30 ± 0.54
	BERT	Gal(47k)	67.4 MB	≈ 6.5 hrs	34.97 ± 0.18	62.11 ± 0.32
	RoBERTa				$\mathbf{41.51 \pm 0.48}$	$\mathbf{69.10 \pm 0.27}$
	BERT	Gal(470k) [10x]	558.2 MB	≈ 2.5 days	$\mathbf{36.39 \pm 0.27}$	$\mathbf{63.84 \pm 1.16}$
	RoBERTa				41.31 ± 0.22	68.84 ± 0.28
	BERT	Gal(25k*2 = 50k)♣	64.0 MB	≈ 6.5 hrs	35.03 ± 0.38	62.14 ± 0.48
	RoBERTa				41.36 ± 0.35	69.00 ± 0.52
	BERT	Gal(47k)♠	44.8 MB	≈ 2.5 hrs	34.90 ± 0.14	62.02 ± 0.95
	RoBERTa				41.57 ± 0.33	68.98 ± 0.31
	Existing Best				41.01 (LUKE)	67.95* (68.36) ((Seo et al., 2024))
	Our Best				**41.51**	**69.10**

Table 1: Results for COVID-QA. Top: Scores from Existing Works. *: Seo et al. (2024) claims this score as statistically significant. | Middle: Our Benchmarks. [1]Peng et al. (2019); [2]Yuan et al. (2022); [3]Beltagy et al. (2020); [4]Zaheer et al. (2020); [5]Yamada et al. (2020); Blue = best/red = worst scores overall; **bold = best decoder** | Bottom: Scores from TOP-Training. Time#: Time to generate the corpus; ♣: entity filter; ♠: maximum context length = 1k tokens; Gal = Galactica; Blue = best/red = worst scores overall; **bold = best BERT/RoBERTa setup.**

6.1.1 Baselines

Our experiments demonstrate that a one-size-fits-all approach does not work always for domain adaptation. BioBERT and PubMedBERT were trained on similar corpora and yet scored in the same range, indicating no clear winner. PubMedBERT which is trained from scratch using a custom vocabulary covering a range of medical jargon performs better.

SciBERT (+CORD-19), trained on CORD-19 articles, performs worse than regular SciBERT, suggesting potential issues in training choices or noisiness in the data. Notably, LUKE, trained solely on Wikipedia data, emerges as the best baseline model, possibly due to its entity-recognition pre-training objective, which aids in identifying relevant entities for QA tasks (van Aken et al., 2019) and highlighting the need for entity representations in

closed domains. Models capable of handling longer context, i.e., Longformer, BigBird and XLNet do not show marked improvements. XLNet degrades completely on COVID-QA potentially due to token permutation hindering its reasoning across large and conceptually dense contexts.

Concerning prior work, we analyze Poerner et al. (2020) and Seo et al. (2024) as these are the only two directly comparable works. Poerner et al. (2020) equips BERT with Word2Vec embeddings trained on corpora ranging from 2-94 GB whereas we use a 67.4 MB corpus and obtain slightly better EM and ∼3% more F1 (BERT: proposed method, row 3, Table 1). Additionally, RoBERTa with TOP-Training shows 1.7% more F1 (proposed method, row 3, Table 1) compared to Seo et al. (2024) who use a much larger model (T5) for their approach. Both instances show the resource bene-

fits coming from `TOP-Training`.

6.1.2 `TOP-Training`

Wiki Baseline: Fine-Tuning BERT on Wikipedia yields marginal improvement of 0.9% EM and 1.2% F1; RoBERTa shows a 3.7% increase in EM and a 1.3% increase in F1. Our Wikipedia corpus, while small, contains relevant information about COVID literature, which in turn aids in answering related questions.

47k corpus: With `TOP-Training`, BERT achieves a 4.01% increase in EM and a 3.5% increase in F1, while RoBERTa shows a 6.7% increase in EM and a 2.5% increase in F1, **setting a new SOTA on COVID-QA**. RoBERTa even outperforms the previous SOTA model (LUKE) by 1.2% in EM and 1.3% in F1, despite using a training corpus significantly smaller (67.4 MB/0.032B words) than LUKE's 3.5B-word corpus (0.9% of the size). Moreover, any variation of our approach, ablation or otherwise, improves performance for both models over the Wikipedia baseline. These results indicate that our models benefit from additional training on corpora aligned with entity information from the downstream dataset.

470k [10x] corpus: Training with a 10x corpus (10 contexts per entity) led to the most improvements for BERT with EM increasing by 8.2% and F1 by 6.4%. This is consistent with Liu et al. (2019) who argue that BERT was *significantly under-trained*. Though this improvement does not achieve RoBERTa performance, it demonstrates the scalability of our approach. RoBERTa improves too, but not as much as when using the base 47k corpus as compared to BERT, which, being undertrained, is more *malleable* to learning new concepts while RoBERTa seems to have hit its *ceiling for learning* in this domain.

Filtration - 50k corpus: Surprisingly, on removing ill-formed entities (proposed method, row 5, Table 1) the performance declined with respect to the best BERT (proposed method, 10x corpus - row 4) and best RoBERTa (proposed method, base 47k corpus - row 3) model. We attribute this to our regex rules which mistakenly (as they cannot distinguish between true/false patterns) removed entities relevant to research articles such as author names or URLs, leading to the decline in performance.

Reduced context length - 47k (at most 1k context tokens) corpus: Galactica can process at most 2048 input tokens which meant that we could not generate text beyond this limit. Instead, we wanted to see if a lower number of tokens, i.e., smaller context size impacted performance (last row of the proposed method, Table 1). Inevitably, both models perform worse on both metrics as Galactica is unable to generate content matching the style of COVID-QA contexts, underscoring the importance of domain-aware writing styles for adaptation pipelines.

6.2 RadQA

Results on RadQA are presented for both its validation and test splits. We consider various combinations of contexts (prompts) and entity filtration. Despite finding no information leakage, higher test scores on average are observed as compared to validation scores, which we attribute to fewer unanswerable questions in the test set (154 vs. 231) and slightly shorter contexts (73.82 vs. 78.1 tokens). As such, while we report scores for both, we mainly focus our analysis on the validation set.

6.2.1 Baselines

On the RadQA benchmark dev set, CODER, a PubMEDBERT checkpoint, has the best EM and F1 but suffers slightly v/s PubMedBERT on only answerable questions presumably because CODER learned clinical embeddings from the UMLS knowledge graph with radiology terms.

Surprisingly, PubMed/Blue-BERT performed similarly on both the dev and test sets. Theoretically, BlueBERT should have performed better being pre-trained on MIMIC clinical notes. RadBERT, which is a superior RoBERTa architecture, and specifically trained on radiology reports did not perform well overall. Although it marginally improved over PubMed/Blue-BERT, it comes at the cost of a fraction of the training data. This again indicates the importance of proper domain alignment, i.e., *what* data the models are trained on.

Unfortunately, LUKE performed poorly as compared to Bio/Sci-BERT, showing little, v/s both on dev, to no gain, v/s SciBERT on test, during evaluation. The impact of writing styles in the training corpora is evident in the performance gap between Clinical/Rad-BERT. While the former was trained on more clinical data, it was not the *right* type of data, i.e., radiology reports, leading RadBERT to outperform it on both splits.

Contrasting `TOP-Training` with prior work, we see that by using our corpus with fancy prompts on unfiltered entities (proposed method, row 5, Ta-

	Model	Train Dataset / Corpus Size	Time#	Dev				Test			
				EM	F1	H(EM)	H(F1)	EM	F1	H(EM)	H(F1)
Related Work	Ghosh et al. (2023)							–	62.55		
	Lu et al. (2024)	N/A	N/A					54.6	–		
	Hernandez et al. (2023)							55.0	74.5		
	Lewis et al. (2020b)							60.4	75.9		
	TOP-Training (best)							52.39	65.57		
Our Benchmarks	BioBERT	PubMed/4.5B words		26.42 ± 0.49	44.26 ± 0.09	40.79 ± 0.76	68.31 ± 0.14	49.95 ± 1.08	63.32 ± 0.40	45.65 ± 1.21	63.50 ± 0.57
	SciBERT	Semantic Scholar/3.2B words		27.03 ± 0.32	44.40 ± 0.06	41.65 ± 0.62	68.45 ± 0.22	53.04 ± 0.38	67.17 ± 0.73	48.62 ± 0.70	67.49 ± 1.02
	PubMedBERT	PubMed/3.1B words(21GB)		31.45 ± 0.17	47.89 ± 0.46	48.40 ± 0.27	73.77 ± 0.62	54.07 ± 0.71	68.76 ± 0.22	49.49 ± 0.87	69.09 ± 0.80
	BlueBERT	PubMed + MIMIC/4.5B words		30.08 ± 1.33	47.14 ± 0.81	46.12 ± 2.01	73.44 ± 2.78	54.99 ± 1.91	68.11 ± 1.41	48.55 ± 1.66	66.06 ± 1.40
	CODER	UMLS/N/A		40.50 ± 1.31	57.32 ± 1.74	47.37 ± 1.70	73.34 ± 1.29	53.74 ± 0.71	68.36 ± 0.36	49.86 ± 0.50	69.36 ± 1.00
	LUKE	Wikipedia/3.5B words	N/A	27.44 ± 0.70	44.77 ± 0.40	42.35 ± 1.08	69.10 ± 0.62	50.92 ± 1.26	64.47 ± 1.75	46.16 ± 0.25	64.25 ± 1.28
	RadBERT[1]	Radiology reports/2.6 GB		30.34 ± 1.50	48.00 ± 1.43	45.73 ± 0.68	73.00 ± 0.73	54.40 ± 2.84	67.34 ± 1.74	51.52 ± 0.87	68.80 ± 0.80
	ClinicalBERT[2]	MIMIC/0.5B words(3.7GB)		27.18 ± 1.89	44.69 ± 0.54	41.88 ± 2.86	68.90 ± 0.71	50.27 ± 1.63	63.40 ± 1.52	46.89 ± 0.13	64.41 ± 0.16
	BioMed-RoBERTa[3]	S2ORC/7.55B tokens(47GB)		27.44 ± 1.10	45.44 ± 0.64	42.35 ± 1.70	70.14 ± 0.99	52.82 ± 0.57	66.52 ± 0.32	48.62 ± 0.33	66.91 ± 0.82
	Galactica	c.f. section 4/106B tokens		1.37	8.5	1.37	8.5	0.49	10.23	0.49	10.23
	MedLLaMA	Medical Corpora/ N/A		0.3	10.63	0.3	10.63	0.16	12.14	0.16	12.14
	MedAlpaca	Medical Meadow/ N/A		1.68	15.18	1.68	15.18	1.3	16.95	1.3	16.95
Proposed Method	BERT	N/A*	N/A	23.83 ± 0.49	42.91 ± 0.46	36.79 ± 0.76	66.23 ± 0.70	46.20 ± 1.96	59.42 ± 1.10	40.65 ± 1.15	58.30 ± 0.74
	RoBERTa			26.12 ± 0.69	43.83 ± 0.44	40.31 ± 1.06	67.65 ± 0.68	51.68 ± 0.73	64.94 ± 0.62	46.45 ± 0.66	64.14 ± 0.42
	BERT	Wikipedia/18.4 MB	≈30 mins	24.49 ± 0.38	42.62 ± 0.08	37.80 ± 0.59	65.79 ± 0.13	47.40 ± 1.85	60.11 ± 1.59	41.95 ± 1.73	58.92 ± 1.56
	RoBERTa			27.19 ± 0.32	44.49 ± 0.31	41.96 ± 0.49	68.68 ± 0.47	50.54 ± 0.62	63.43 ± 0.75	45.72 ± 0.50	62.92 ± 0.67
	BERT	Gal(≈55k) †/ 81.6 MB	≈11 hrs	24.70 ± 0.46	42.88 ± 0.30	38.12 ± 0.71	66.19 ± 0.46	46.74 ± 1.84	59.69 ± 0.92	41.60 ± 2.31	58.88 ± 1.64
	RoBERTa			27.09 ± 0.70	44.42 ± 0.55	41.81 ± 1.09	68.56 ± 0.86	51.25 ± 0.41	64.41 ± 0.95	46.45 ± 0.66	64.01 ± 0.71
	BERT	Gal(≈55k) †♣/ 80.3 MB	≈11 hrs	25.51 ± 0.38	43.11 ± 0.23	40.38 ± 2.31	64.76 ± 2.75	46.36 ± 1.79	59.41 ± 1.67	46.59 ± 8.06	57.83 ± 1.81
	RoBERTa			27.69 ± 0.23	45.16 ± 0.35	42.74 ± 0.36	69.70 ± 0.54	51.14 ± 0.49	64.29 ± 0.38	46.31 ± 1.36	63.85 ± 0.76
	BERT	Gal(≈55k) ‡/ 38.1 MB	≈11 hrs	25.10 ± 0.32	42.78 ± 0.55	38.74 ± 0.49	66.03 ± 0.84	46.85 ± 1.74	59.54 ± 1.19	41.66 ± 0.45	58.60 ± 0.37
	RoBERTa			27.64 ± 0.49	44.99 ± 0.11	42.67 ± 0.76	69.45 ± 0.91	52.39 ± 0.80	65.57 ± 0.93	47.76 ± 1.85	65.34 ± 1.97
	BERT	Gal(≈55k) ‡♣/ 34.3 MB	≈11 hrs	25.76 ± 0.66	43.10 ± 0.27	39.68 ± 1.11	66.44 ± 0.55	46.52 ± 1.20	58.98 ± 1.09	40.44 ± 0.95	57.06 ± 0.94
	RoBERTa			27.08 ± 0.46	44.67 ± 0.23	41.81 ± 0.72	68.95 ± 0.35	51.30 ± 1.17	64.14 ± 0.41	47.39 ± 1.64	64.53 ± 0.26
	BERT	Gal(≈100k) ††/ 120.8 MB	≈22 hrs	25.10 ± 0.54	42.74 ± 0.06	38.74 ± 0.83	65.97 ± 0.09	47.07 ± 1.23	60.04 ± 1.19	42.46 ± 0.13	59.78 ± 0.41
	RoBERTa			27.49 ± 0.49	44.82 ± 0.38	42.43 ± 0.75	69.17 ± 0.58	51.79 ± 1.98	64.99 ± 1.80	47.03 ± 1.45	64.45 ± 1.21
	BERT	Gal(≈100k) † ‡ ♣/ 115.6 MB	≈22 hrs	24.75 ± 0.23	42.96 ± 0.10	38.20 ± 0.36	66.31 ± 0.16	47.50 ± 1.06	60.38 ± 0.48	41.96 ± 1.31	59.14 ± 1.97
	RoBERTa			**27.85 ± 0.09**	**45.16 ± 0.11**	**42.98 ± 0.14**	**69.71 ± 0.18**	51.79 ± 0.59	64.77 ± 0.40	46.88 ± 1.28	64.21 ± 0.75

Table 2: Results for RadQA. Top: Comparison with Existing Works. | Middle: Our Baselines. H(F1)/H(EM): Has_F1/Has_EM i.e F1/EM for answerable questions; Blue = best/red = worst scores overall; **bold = best decoder** | Bottom: TOP-Training Results. *: [Vanilla Fine-Tuning]. [†: normal, ‡: fancy] prompt, ♣: entity filter. Time#: Time to generate the corpus. Gal = Galactica; Blue = best/red = worst overall; **bold = best BERT/RoBERTa setup**.

ble 2), RoBERTa outperforms RadLing (Ghosh et al., 2023) in test F1 by 4.8%, the latter using a much larger corpus of radiology reports. Although we are unable to reach SOTA on RadQA, our best RoBERTa, on the test set (proposed method, row 5, Table 2) comes within 2 points absolute EM of Clinical-T5$_{LARGE}$ (Lu et al., 2024) which again shows the competitiveness of our proposed method as all our models are BASE versions.

6.2.2 TOP-Training

Wiki Baseline: Training on the Wikipedia corpus, BERT shows an increase in EM only (2.8%) on the dev set, but an overall improvement in all measures on the test set compared to vanilla fine-tuning. RoBERTa showed overall improvement on the dev set while test scores suffered compared to the vanilla baseline.

Normal Prompting: Training on the unfiltered corpus with normal prompts (proposed method, row 3, Table 2) improved BERT and RoBERTa 3.7% EM, and 1.4% F1 for RoBERTa, over the vanilla baseline, which is greater than that of our Wiki baseline. However, **when the filter is applied (proposed method, row 4, Table 2)** BERT shows

~7% EM & 0.5% F1 increase while RoBERTa shows 6̃% EM & 3̃% F1 increase over regular fine-tuning indicating the benefits of data filtration.

Best BERT setting: When both filtered entities and the corpus from the fancy prompt are used (proposed method, row 6, Table 2), improvements over basic fine-tuning (8.1% EM, 0.4% F1 on dev) and the Wikipedia baseline (5.2% EM, 1.1%F1 on dev) occur. Note: a) the benefit of studying RadQA and designing targeted prompts, and b) that BERT reaches these scores with a modest 34.3 MB corpus, much smaller than the benchmarked models.

Best RoBERTa setting: RoBERTa demonstrates improvements across different combinations of filtration methods, prompt styles and using the Wikipedia corpus. However, the improvements are inconsistent with respect to a specific approach. Excluding the combined corpora settings, we see it achieve the best performance, on validation, using the corpus obtained from the filtered entities and normal prompting (proposed method, row 4, Table 2) and on the test, using unfiltered entities and fancy prompting (proposed method, row 5, Table 2).

RoBERTa v/s Benchmarks: In row 4 (proposed method) of Table 2, RoBERTa, outperforms Bio/Sci-BERT and LUKE on all metrics along with long-context models BigBird and Longformer. Interestingly, RoBERTa even beats BioClinicalBERT by 1.9% EM and 1.1% F1, which was trained using much higher quality clinical notes.

Combined Prompting Styles: Here, we merge the contexts from both prompt styles (proposed method, rows 7 and 8, Table 2) for the filtered and the unfiltered entities separately. BERT shows better performance in row 7 i.e. using the unfiltered corpus (1.7% EM increase over the filtered variant and roughly the same F1) and RoBERTa in row 8 i.e. using the filtered corpus (1.3% EM and 0.8%F1 over the unfiltered variant). Overall, this version of RoBERTa resulted in our best overall model on the validation data suggesting that incorporating a mixture of prompt styles will create more diverse corpora, enhancing domain alignment.

6.3 Investigating Information Leakage
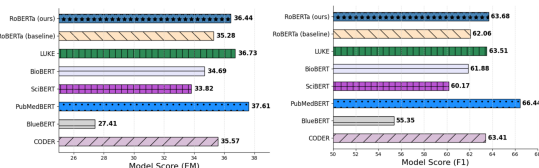
Figure 2: Information Leakage Validation Trials (Left - EM | Right - F1): RoBERTa (ours) was trained on a subset of the 47k corpus with entities only from the 80% train set. All of the models were fine-tuned in the usual manner i.e. SQuAD→COVID-QA (80% train set) and evaluated on the 20% test set.

Given that the synthetic corpus generated for COVID-QA in §5 contains entities identified in the *entire* COVID-QA dataset - not from the *train* split within each *fold* - we explore if the performance gains from `TOP-Training` are a result of information leak. To this end, we construct a roughly 80%/20% train/test split (1,676/343 records), ensuring no context overlap, and apply a suite of models to this new split. When applying our `TOP-Training`, a synthetic corpus is generated *only* from entities identified in the train split.

RoBERTa subjected to `TOP-Training` still yields strong performance in this restricted scenario, only surpassed by PubMedBERT (and marginally by LUKE in EM) (Fig. 2) demonstrating that the improved performance on COVID-QA cannot be attributed to information leak from the

test set. Although the scores are lower than Table 1, the relative scores produced by each model lead to a similar conclusion that `TOP-Training` yields optimal results.

7 Conclusion

We introduce `TOP-Training`, an innovative pre-training solution to enhance the alignment of an LLM with rare-domain target problems. The distinctive feature of `TOP-Training` lies in its automated synthesis of target-oriented data with the assistance of an LLM. Our experiments on `Medical-EQA` demonstrate the effectiveness of `TOP-Training`, shedding light on the limitations of widely used autoregressive LLMs.

Limitations

We recognize that hallucination is still a concern with our method i.e. we do not provide a way to alleviate it. This is one area which our future work will focus on, i.e., developing ways to detect and remove hallucinated text from the generated corpus. In our analysis, we do not examine the *quality* of the generated corpora and only rely on final task performance as an indicator of how good our generated corpus is. While we recognize the insights that we can gain from this analysis are interesting, we do not believe that its absence seriously undermines the proposed pipeline.

Ethics Statement

As our work relied on publicly available datasets, we believe that the ethical ramifications here are limited. That being said, we recognize that to use RadQA, we had to acquire certifications to access it. This shows that even though the data in it is redacted, loosely disseminated patient reports are a threat to privacy. Moreover, we had to make sure that when generating our synthetic reports, we did not mention any patient names which even with a small probability might bear resemblance to an actual person.

References

Josh Achiam, Steven Adler, Sandhini Agarwal, Lama Ahmad, Ilge Akkaya, Florencia Leoni Aleman, Diogo Almeida, Janko Altenschmidt, Sam Altman, Shyamal Anadkat, et al. 2023. Gpt-4 technical report. *arXiv preprint arXiv:2303.08774*.

Helena Balabin, Charles Tapley Hoyt, Colin Birkenbihl, Benjamin M Gyori, John Bachman, Alpha Tom Kodamullil, Paul G Plöger, Martin Hofmann-Apitius, and Daniel Domingo-Fernández. 2022. Stonkgs: a sophisticated transformer trained on biomedical text and knowledge graphs. *Bioinformatics*, 38(6):1648–1656.

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

Hannah Brown, Katherine Lee, Fatemehsadat Mireshghallah, Reza Shokri, and Florian Tramèr. 2022. What does it mean for a language model to preserve privacy? In *Proceedings of the 2022 ACM Conference on Fairness, Accountability, and Transparency*, pages 2280–2292.

Vittorio Castelli, Rishav Chakravarti, Saswati Dana, Anthony Ferritto, Radu Florian, Martin Franz, Dinesh Garg, Dinesh Khandelwal, J Scott McCarley, Michael McCawley, et al. 2020. The techqa dataset. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 1269–1278, "Online". "Association for Computational Linguistics".

Zeming Chen, Alejandro Hernández Cano, Angelika Romanou, Antoine Bonnet, Kyle Matoba, Francesco Salvi, Matteo Pagliardini, Simin Fan, Andreas Köpf, Amirkeivan Mohtashami, Alexandre Sallinen, Alireza Sakhaeirad, Vinitra Swamy, Igor Krawczuk, Deniz Bayazit, Axel Marmet, Syrielle Montariol, Mary-Anne Hartley, Martin Jaggi, and Antoine Bosselut. 2023. Meditron-70b: Scaling medical pretraining for large language models. *Preprint*, arXiv:2311.16079.

Kevin Clark, Minh-Thang Luong, Quoc V. Le, and Christopher D. Manning. 2020. ELECTRA: Pretraining text encoders as discriminators rather than generators. In *ICLR*.

Rikhiya Ghosh, Oladimeji Farri, Sanjeev Kumar Karn, Manuela Danu, Ramya Vunikili, and Larisa Micu. 2023. RadLing: Towards efficient radiology report understanding. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 5: Industry Track)*, pages 640–651, Toronto, Canada. Association for Computational Linguistics.

Suriya Gunasekar, Yi Zhang, Jyoti Aneja, Caio César Teodoro Mendes, Allie Del Giorno, Sivakanth Gopi, Mojan Javaheripi, Piero Kauffmann, Gustavo de Rosa, Olli Saarikivi, Adil Salim, Shital Shah, Harkirat Singh Behl, Xin Wang, Sébastien Bubeck, Ronen Eldan, Adam Tauman Kalai, Yin Tat Lee, and Yuanzhi Li. 2023. Textbooks are all you need. *Preprint*, arXiv:2306.11644.

Suchin Gururangan, Ana Marasović, Swabha Swayamdipta, Kyle Lo, Iz Beltagy, Doug Downey, and Noah A. Smith. 2020. Don't stop pretraining: Adapt language models to domains and tasks. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics*, pages 8342–8360, Online. Association for Computational Linguistics.

Ahcene Haddouche, Ikram Rabia, and Aicha Aid. 2023. Transformer-based question answering model for the biomedical domain. In *2023 5th International Conference on Pattern Analysis and Intelligent Systems (PAIS)*, pages 1–6. IEEE.

Michael P Hartung, Ian C Bickle, Frank Gaillard, and Jeffrey P Kanne. 2020. How to create a great radiology report. *RadioGraphics*, 40(6):1658–1670.

Evan Hernandez, Diwakar Mahajan, Jonas Wulff, Micah J Smith, Zachary Ziegler, Daniel Nadler, Peter Szolovits, Alistair Johnson, Emily Alsentzer, et al. 2023. Do we still need clinical language models? In *Conference on Health, Inference, and Learning*, pages 578–597. PMLR.

Ziwei Ji, Nayeon Lee, Rita Frieske, Tiezheng Yu, Dan Su, Yan Xu, Etsuko Ishii, Ye Jin Bang, Andrea Madotto, and Pascale Fung. 2023. Survey of hallucination in natural language generation. *ACM Computing Surveys*, 55(12):1–38.

Sharon Levy, Kevin Mo, Wenhan Xiong, and William Yang Wang. 2021. Open-Domain question-Answering for COVID-19 and other emergent domains. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing: System Demonstrations*, pages 259–266, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Mike Lewis, Yinhan Liu, Naman Goyal, Marjan Ghazvininejad, Abdelrahman Mohamed, Omer Levy, Veselin Stoyanov, and Luke Zettlemoyer. 2020a. 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.

Patrick Lewis, Myle Ott, Jingfei Du, and Veselin Stoyanov. 2020b. Pretrained language models for biomedical and clinical tasks: Understanding and extending the state-of-the-art. In *Proceedings of the 3rd Clinical Natural Language Processing Workshop*, pages 146–157, Online. Association for Computational Linguistics.

Nelson F Liu, Kevin Lin, John Hewitt, Ashwin Paranjape, Michele Bevilacqua, Fabio Petroni, and Percy Liang. 2024. Lost in the middle: How language models use long contexts. *Transactions of the Association for Computational Linguistics*, 12:157–173.

Yinhan Liu, Myle Ott, Naman Goyal, Jingfei Du, Mandar Joshi, Danqi Chen, Omer Levy, Mike Lewis, Luke Zettlemoyer, and Veselin Stoyanov. 2019. Roberta: A robustly optimized bert pretraining approach. *Preprint*, arXiv:1907.11692.

Bo-Ru Lu, Nikita Haduong, Chien-Yu Lin, Hao Cheng, Noah A Smith, and Mari Ostendorf. 2024. Encode once and decode in parallel: Efficient transformer decoding. *arXiv preprint arXiv:2403.13112*.

Man Luo, Kazuma Hashimoto, Semih Yavuz, Zhiwei Liu, Chitta Baral, and Yingbo Zhou. 2022. Choose your QA model wisely: A systematic study of generative and extractive readers for question answering. In *Proceedings of the 1st Workshop on Semiparametric Methods in NLP: Decoupling Logic from Knowledge*, pages 7–22, Dublin, Ireland and Online. Association for Computational Linguistics.

Prabir Mallick, Tapas Nayak, and Indrajit Bhattacharya. 2023. Adapting pre-trained generative models for extractive question answering. In *Proceedings of the Third Workshop on Natural Language Generation, Evaluation, and Metrics (GEM)*, pages 128–137, Singapore. Association for Computational Linguistics.

Kurt Micallef, Albert Gatt, Marc Tanti, Lonneke van der Plas, and Claudia Borg. 2022. Pre-training data quality and quantity for a low-resource language: New corpus and BERT models for Maltese. In *Proceedings of the Third Workshop on Deep Learning for Low-Resource Natural Language Processing*, pages 90–101, Hybrid. Association for Computational Linguistics.

Timo Möller, Anthony Reina, Raghavan Jayakumar, and Malte Pietsch. 2020. COVID-QA: A question answering dataset for COVID-19. In *Proceedings of the 1st Workshop on NLP for COVID-19 at ACL 2020*, Online. Association for Computational Linguistics.

Harsha Nori, Yin Tat Lee, Sheng Zhang, Dean Carignan, Richard Edgar, Nicolo Fusi, Nicholas King, Jonathan Larson, Yuanzhi Li, Weishung Liu, et al. 2023. Can generalist foundation models outcompete special-purpose tuning? case study in medicine. *Medicine*, 84(88.3):77–3.

Ankit Pal, Logesh Kumar Umapathi, and Malaikannan Sankarasubbu. 2023. Med-HALT: Medical domain hallucination test for large language models. In *Proceedings of the 27th Conference on Computational Natural Language Learning (CoNLL)*, pages 314–334, Singapore. Association for Computational Linguistics.

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.

Nina Poerner, Ulli Waltinger, and Hinrich Schütze. 2020. Inexpensive domain adaptation of pretrained language models: Case studies on biomedical NER and covid-19 QA. In *Findings of the Association for Computational Linguistics: EMNLP 2020*, pages 1482–1490, Online. Association for Computational Linguistics.

Peng Qi, Yuhao Zhang, Yuhui Zhang, Jason Bolton, and Christopher D. Manning. 2020. Stanza: A python natural language processing toolkit for many human languages. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics: System Demonstrations*, pages 101–108, Online. Association for Computational Linguistics.

Pranav Rajpurkar, Robin Jia, and Percy Liang. 2018. Know what you don't know: Unanswerable questions for SQuAD. In *Proceedings of the 56th Annual Meeting of the Association for Computational Linguistics (Volume 2: Short Papers)*, pages 784–789, Melbourne, Australia. Association for Computational Linguistics.

Pranav Rajpurkar, Jian Zhang, Konstantin Lopyrev, and Percy Liang. 2016. SQuAD: 100,000+ questions for machine comprehension of text. In *Proceedings of the 2016 Conference on Empirical Methods in Natural Language Processing*, pages 2383–2392, Austin, Texas. Association for Computational Linguistics.

Vinay Samuel, Houda Aynaou, Arijit Ghosh Chowdhury, Karthik Venkat Ramanan, and Aman Chadha. 2023. Can llms augment low-resource reading comprehension datasets? opportunities and challenges. *arXiv preprint arXiv:2309.12426*.

Teven Le Scao, Angela Fan, Christopher Akiki, Ellie Pavlick, Suzana Ilić, Daniel Hesslow, Roman Castagné, Alexandra Sasha Luccioni, François Yvon, Matthias Gallé, et al. 2022. Bloom: A 176b-parameter open-access multilingual language model. *arXiv preprint arXiv:2211.05100*.

Minju Seo, Jinheon Baek, James Thorne, and Sung Ju Hwang. 2024. Retrieval-augmented data augmentation for low-resource domain tasks. *arXiv preprint arXiv:2402.13482*.

Sarvesh Soni, Meghana Gudala, Atieh Pajouhi, and Kirk Roberts. 2022. RadQA: A question answering dataset to improve comprehension of radiology reports. In *Proceedings of the Thirteenth Language Resources and Evaluation Conference*, pages 6250–6259, Marseille, France. European Language Resources Association.

Ross Taylor, Marcin Kardas, Guillem Cucurull, Thomas Scialom, Anthony Hartshorn, Elvis Saravia, Andrew Poulton, Viktor Kerkez, and Robert Stojnic. 2022. Galactica: A large language model for science. *arXiv preprint arXiv:2211.09085*.

Y. Tian, Z. Chen, J. Yang, B. Xu, Z. Guo, X. Zhang, R. Hao, Q. Li, and M. Sun. 2023. Medical extractive question-answering based on fusion of hierarchical features. In *2023 IEEE International Conference on Bioinformatics and Biomedicine (BIBM)*, pages 3938–3945, Los Alamitos, CA, USA. IEEE Computer Society.

Betty van Aken, Benjamin Winter, Alexander Löser, and Felix A. Gers. 2019. How does bert answer questions? a layer-wise analysis of transformer representations. In *Proceedings of the 28th ACM International*

Conference on Information and Knowledge Management, CIKM '19, page 1823–1832, New York, NY, USA. Association for Computing Machinery.

Lucy Lu Wang, Kyle Lo, Yoganand Chandrasekhar, Russell Reas, Jiangjiang Yang, Doug Burdick, Darrin Eide, Kathryn Funk, Yannis Katsis, Rodney Michael Kinney, Yunyao Li, Ziyang Liu, William Merrill, Paul Mooney, Dewey A. Murdick, Devvret Rishi, Jerry Sheehan, Zhihong Shen, Brandon Stilson, Alex D. Wade, Kuansan Wang, Nancy Xin Ru Wang, Christopher Wilhelm, Boya Xie, Douglas M. Raymond, Daniel S. Weld, Oren Etzioni, and Sebastian Kohlmeier. 2020. CORD-19: The COVID-19 open research dataset. In *Proceedings of the 1st Workshop on NLP for COVID-19 at ACL 2020*, Online. Association for Computational Linguistics.

Peng Xu, Davis Liang, Zhiheng Huang, and Bing Xiang. 2021. Attention-guided generative models for extractive question answering. *Preprint*, arXiv:2110.06393.

Ikuya Yamada, Akari Asai, Hiroyuki Shindo, Hideaki Takeda, and Yuji Matsumoto. 2020. LUKE: Deep contextualized entity representations with entity-aware self-attention. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 6442–6454, Online. Association for Computational Linguistics.

Zheng Yuan, Zhengyun Zhao, Haixia Sun, Jiao Li, Fei Wang, and Sheng Yu. 2022. Coder: Knowledge-infused cross-lingual medical term embedding for term normalization. *Journal of biomedical informatics*, 126:103983.

Zhenrui Yue, Bernhard Kratzwald, and Stefan Feuerriegel. 2021. Contrastive domain adaptation for question answering using limited text corpora. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 9575–9593, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Aizan Zafar, Sovan Kumar Sahoo, Deeksha Varshney, Amitava Das, and Asif Ekbal. 2024a. Kimedqa: towards building knowledge-enhanced medical qa models. *Journal of Intelligent Information Systems*, pages 1–26.

Aizan Zafar, Deeksha Varshney, Sovan Kumar Sahoo, Amitava Das, and Asif Ekbal. 2024b. Are my answers medically accurate? exploiting medical knowledge graphs for medical question answering. *Applied Intelligence*, 54(2):2172–2187.

Manzil Zaheer, Guru Guruganesh, Avinava Dubey, Joshua Ainslie, Chris Alberti, Santiago Ontanon, Philip Pham, Anirudh Ravula, Qifan Wang, Li Yang, and Amr Ahmed. 2020. Big bird: transformers for longer sequences. In *Proceedings of the 34th International Conference on Neural Information Processing Systems*, NIPS '20, Red Hook, NY, USA. Curran Associates Inc.

Chunting Zhou, Pengfei Liu, Puxin Xu, Srinivasan Iyer, Jiao Sun, Yuning Mao, Xuezhe Ma, Avia Efrat, Ping Yu, LILI YU, Susan Zhang, Gargi Ghosh, Mike Lewis, Luke Zettlemoyer, and Omer Levy. 2023. Lima: Less is more for alignment. In *Advances in Neural Information Processing Systems*, volume 36, pages 55006–55021. Curran Associates, Inc.

A Token Filtering

We performed entity filtering as a common ablation technique for both datasets. We used regular expressions to remove entities with special characters such as *, !, etc., as well as specific text patterns like `https*` and `baby`. We implemented a length-based filter, retaining only entities longer than a certain number of characters. Additionally, for COVID-QA, we applied a second round of filtration using TF-IDF, considering the questions + context as the corpus and retaining the top 25k entities with the highest IDF scores. However, as this approach did not lead to substantial gains, we decided not to use it for RadQA. Due to the large number of possible combinations, we did not extensively explore these settings in our experiments.

B Synthetic Corpora Samples

We provide samples from our generated corpus. In Figure 3 we show two positive results for our COVID-QA-directed corpus. The top one is in the style of a research paper while the bottom one, though shorter in length, details useful information on the required entity. We show negative examples of the same in Figure 4. The top one completely degrades into noise while the bottom one although coherent is talking about an unrelated topic.

For RadQA, we show similar examples. In Figure 5 we show positive (top) and negative (bottom) samples from normal prompting. Note that since programming languages were a part of Galactica's training corpora, it sometimes generates the same during prompting. Finally, Figure 6 shows positive (top) and negative (bottom) examples of fancy prompting. While the positive sample shows a strong correlation with an actual radiology report, the bottom one completely breaks down.

C ChatGPT trials

We were curious to see if ChatGPT (with GPT3.5) would be able to answer a few questions from COVID-QA. Only the questions were provided to it without the context since they were much longer than its input capacity. We did not test RadQA

Title: chest pain *Title: an unusual cause of chest pain in an elderly patient*
Authors: Sushila Shrivastava, Vinay Kumar, Pramod Kumar
Keywords: Chest pain, Eosinophilic pneumonia, Chest pain

 # Abstract

 # Background

Pulmonary eosinophilic infiltration is one of the rare causes of non-specific chest pain in the elderly patients. We present a case of elderly patient who presented with a non-specific chest pain of 1 day duration with eosinophilia and chest CT scan showing ground glass opacities.

 # Case presentation

A 74-year-old lady presented with a 1-day history of chest pain, nausea and vomiting. The patient had a history of cough for 1 year, and she was on regular medication for cough. The patient had a significant history of hypertension, diabetes mellitus and COPD. On examination, the patient was afebrile with an oxygen saturation of 98% on room air. She had a heart rate of 90 beats per minute and blood pressure of 115/70 mmHg. Chest auscultation revealed a crackles in the lower lung fields. Her chest X-ray showed bilateral pulmonary infiltrates. A diagnosis of eosinophilic pneumonia was made and she was started on oral corticosteroids. She recovered without any further complications after two weeks of treatment.

 # Conclusion

Chest pain in the elderly patient is not always indicative of a life-threatening condition. A high index of suspicion for this rare entity should be kept in mind when a non-specific chest pain in elderly patients is seen.

 # Background

Pulmonary eosinophilic infiltration is a rare cause of non-specific chest pain in elderly patients. The clinical presentation of the disease is similar to other lung diseases with non-specific symptoms, and a high index of suspicion for this rare entity is required in elderly patients presenting with non-specific chest pain. We present a case of a 74-year-old lady with a history of cough who presented with a non-specific chest pain of 1 day duration with eosinophilia.

 # Case presentation

A 74-year-old lady was admitted in a state of mild depression. The patient had a significant history of hypertension, diabetes mellitus and COPD. She had a history of chronic cough for 1 year and was on regular medication for the cough. On examination, the patient was afebrile with an oxygen saturation of 98% on room air. She had a heart rate of 90 beats per minute and blood pressure of 115/70 mmHg. Chest auscultation revealed a crackles in the lower lung fields. She was investigated with a chest X-ray which showed bilateral pulmonary infiltrates. An ECG showed no sign of ST-T segment elevation. Her complete blood count (CBC) was as follows: haemoglobin 12.4 gm%, leucocytes 10,300/cu mm, platelets 253,000/cu mm, red blood cells (RBCs) 2.28 million/cu mm, and haemoglobin percentage (Hb%) 24.9%. Her electrolytes were normal. Her C-reactive protein (CRP) was 37.1 mg/L, D-dimers 0.24 μg/ml, and erythrocyte sedimentation rate (ESR) was 66 mm/1st hour. Her sputum was clear. The chest X-ray showed bilateral pulmonary infiltrates (Fig. 1). A diagnosis of eosinophilic pneumonia was made, and she was started on oral corticosteroids. She recovered without any further complications after two weeks of treatment.Fig. 1: Chest X-ray shows bilateral pulmonary infiltrates

 # Discussion

Eosinophilic pneumonia is an uncommon cause of non-specific chest pain. The incidence of this condition is approximately 0.1% of all pulmonary infiltrations [Idiopathic eosinophilic pneumonia: clinical features, radiological appearances and long-term outcome., Brown, Eosinophilic pneumonia: imaging findings., Bongiovanni]. The most common age group affected is between 40 and 60 years [Idiopathic eosinophilic pneumonia: clinical features, radiological appearances and long-term outcome., Brown]. However, there have been several case reports of eosinophilic pneumonia in the elderly, with the peak age between 60 and 70 years [Eosinophilic pneumonia in elderly patients: a case series and review of the literature, Kara, Eosinophilic pneumonia in the elderly, De Luca]. Most of the patients with eosinophilic pneumonia are asymptomatic at presentation, but it can also present with fever, dyspnea, cough, and chest pain [Idiopathic eosinophilic pneumonia: clinical features, radiological appearances and long-term outcome., Brown, Eosinophilic pneumonia in elderly patients: a case series and review of the literature, Kara, Eosinophilic pneumonia in the elderly, De Luca]. The chest pain in patients with eosinophilic pneumonia can be localized or generalized and can be of mild, moderate or severe intensity. It can be persistent or acute in nature [Idiopathic eosinophilic pneumonia: clinical features, radiological appearances and long-term outcome., Brown, Eosinophilic pneumonia in the elderly, De Luca]. The patient can be admitted to hospital with a cough, dyspnea, chest pain and fever. However, the patient may not show any of these symptoms at presentation [Idiopathic eosinophilic pneumonia: clinical features, radiological appearances and long-term outcome., Brown, Eosinophilic pneumonia in the elderly, De Luca].

The most common clinical feature of eosinophilic pneumonia is cough, which is present in 85% of patients [Idiopathic eosinophilic pneumonia: clinical features, radiological appearances and long-term outcome., Brown]. Fever is present in 50% to 75% of patients [Idiopathic eosinophilic pneumonia: clinical features, radiological appearances and long-term outcome., Brown, Eosinophilic pneumonia in the elderly, De Luca]. Dyspnea and chest pain are seen in less than 10% of patients [Idiopathic eosinophilic pneumonia: clinical features, radiological appearances and long-term outcome., Brown, Eosinophilic pneumonia in the elderly, De Luca]. The most common respiratory symptoms are cough and dyspnea [Eosinophilic pneumonia in the elderly, De Luca]. The majority of the patients with eosinophilic pneumonia are asymptomatic, while a minority of patients present with non-specific symptoms of cough, fever, dyspnea and chest pain [Idiopathic eosinophilic pneumonia: clinical features, radiological appearances and long-term outcome., Brown, Eosinophilic pneumonia in the elderly, De Luca]. Patients with pulmonary eosinophilia and elevated ESR or CRP have been reported to have a poorer prognosis [Idiopathic eosinophilic pneumonia: clinical features, radiological appearances and long-term outcome., Brown]. Chest radiography can be nonspecific and shows bilateral or unilateral infiltrates [Idiopathic eosinophilic pneumonia: clinical features, radiological appearances and long-term outcome., Brown, Eosinophilic pneumonia in the elderly, De Luca]. A chest CT scan is often required to diagnose eosinophilic pneumonia and show a variety of appearances. These include ground glass opacities (GGO) and reticular opacities [Idiopathic eosinophilic pneumonia: clinical features, radiological appearances and long-term outcome., Brown, Eosinophilic pneumonia in the elderly, De Luca]. In our case, the patient had a history of chronic cough and had no fever on presentation. A chest X-ray showed bilateral infiltrates. A chest CT scan showed GGO and reticular opacities.

The etiology of eosinophilic pneumonia is unknown, but it has been proposed that it is a T-helper 2-mediated inflammatory disorder [Idiopathic eosinophilic pneumonia: clinical features, radiological appearances and long-term outcome., Brown]. The diagnosis of eosinophilic pneumonia can be difficult because of the absence of typical radiological findings. It can be difficult to distinguish eosinophilic pneumonia from other inflammatory lung diseases. Treatment consists of corticosteroids, and the use of immunosuppressive drugs, such as azathioprine, hydroxychloroquine, and methotrexate, has been reported to improve the symptoms of eosinophilic pneumonia [Idiopathic eosinophilic pneumonia: clinical features, radiological appearances and long-term outcome., Brown, Eosinophilic pneumonia in the elderly, De Luca].

 # Conclusion

Eosinophilic pneumonia is a rare cause of non-specific chest pain in elderly patients. In these patients, the diagnosis can be difficult and a high index of suspicion should be kept in mind in such patients. Early diagnosis and prompt treatment can improve the prognosis.

 # Consent

Written informed consent was obtained from the patient for publication of this case report and any accompanying images. A copy of the written consent is available for review by the Editor-in-Chief of this journal.

Footnote: Competing interests

The authors declare that they have no competing interests.

Footnote: Authors' contributions

SS, KK and PK were involved in the management of the patient. SS drafted the manuscript. All authors read and approved the final manuscript.

- -

Title: Polymerase chain reaction *#1: a new diagnostic tool in the management of tuberculosis.*

Abstract: Polymerase chain reaction (PCR) is an advanced molecular diagnostic technique. The potential of PCR in the diagnosis of tuberculosis (TB) has been evaluated by several investigators. PCR techniques have been used in the diagnosis of extra-pulmonary TB as well as the diagnosis of drug resistance. This article reviews the current status of PCR in the diagnosis of TB.

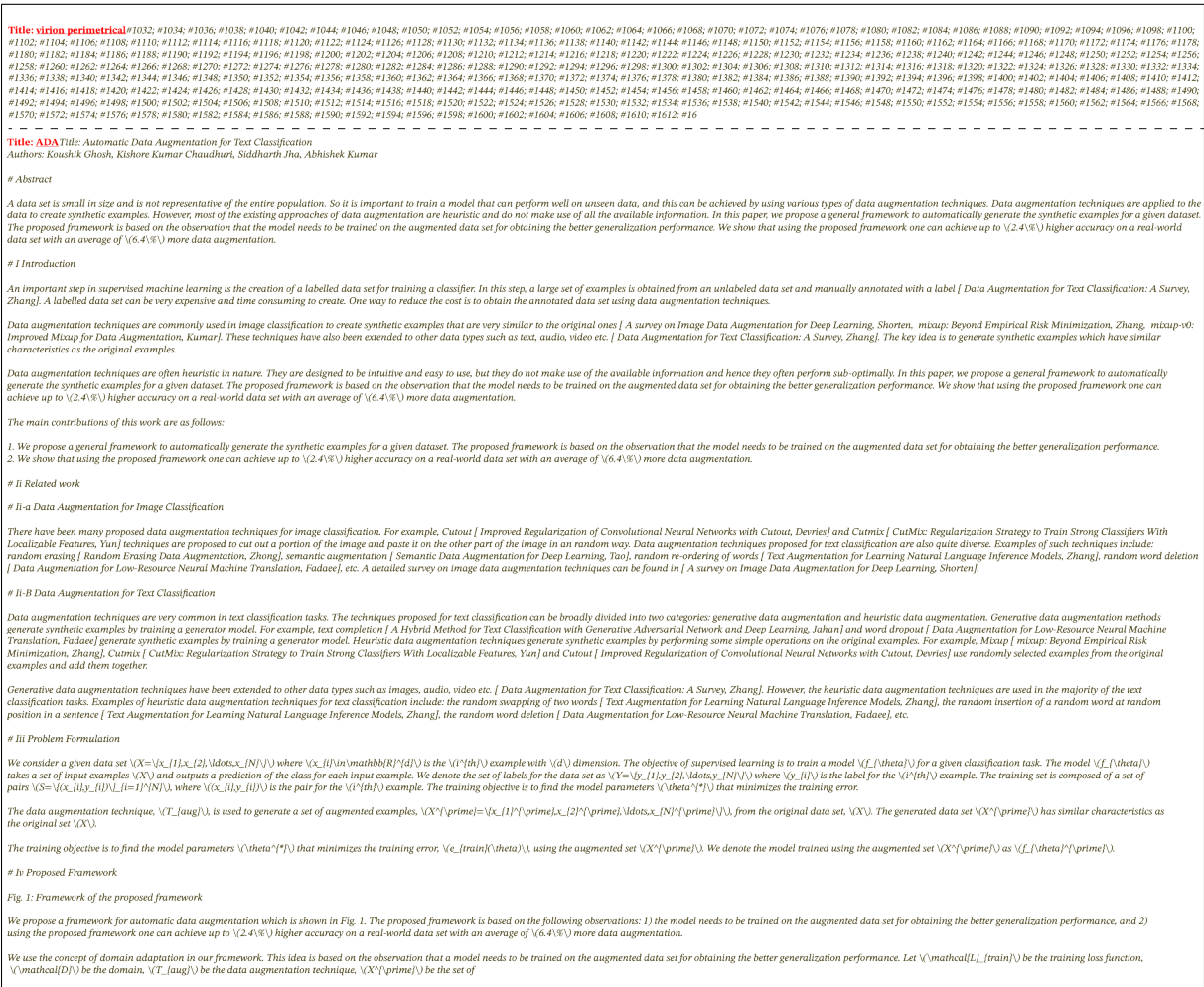Figure 3: **COVID-QA positive** examples (underline = entity | red = prompt) | olive = generation

Title: ADA Title: Automatic Data Augmentation for Text Classification
Authors: Koushik Ghosh, Kishore Kumar Chaudhuri, Siddharth Jha, Abhishek Kumar

Abstract

A data set is small in size and is not representative of the entire population. So it is important to train a model that can perform well on unseen data, and this can be achieved by using various types of data augmentation techniques. Data augmentation techniques are applied to the data to create synthetic examples. However, most of the existing approaches of data augmentation are heuristic and do not make use of all the available information. In this paper, we propose a general framework to automatically generate the synthetic examples for a given dataset. The proposed framework is based on the observation that the model needs to be trained on the augmented data set for obtaining the better generalization performance. We show that using the proposed framework one can achieve up to 2.4% higher accuracy on a real-world data set with an average of 6.4% more data augmentation.

I Introduction

An important step in supervised machine learning is the creation of a labelled data set for training a classifier. In this step, a large set of examples is obtained from an unlabeled data set and manually annotated with a label [Data Augmentation for Text Classification: A Survey, Zhang]. A labelled data set can be very expensive and time consuming to create. One way to reduce the cost is to obtain the annotated data set using data augmentation techniques.

Data augmentation techniques are commonly used in image classification to create synthetic examples that are very similar to the original ones [A survey on Image Data Augmentation for Deep Learning, Shorten, mixup: Beyond Empirical Risk Minimization, Zhang, mixup-v0: Improved Mixup for Data Augmentation, Kumar]. These techniques have also been extended to other data types such as text, audio, video etc. [Data Augmentation for Text Classification: A Survey, Zhang]. The key idea is to generate synthetic examples which have similar characteristics as the original examples.

Data augmentation techniques are often heuristic in nature. They are designed to be intuitive and easy to use, but they do not make use of the available information and hence they often perform sub-optimally. In this paper, we propose a general framework to automatically generate the synthetic examples for a given dataset. The proposed framework is based on the observation that the model needs to be trained on the augmented data set for obtaining the better generalization performance. We show that using the proposed framework one can achieve up to 2.4% higher accuracy on a real-world data set with an average of 6.4% more data augmentation.

The main contributions of this work are as follows:

1. We propose a general framework to automatically generate the synthetic examples for a given dataset. The proposed framework is based on the observation that the model needs to be trained on the augmented data set for obtaining the better generalization performance.
2. We show that using the proposed framework one can achieve up to 2.4% higher accuracy on a real-world data set with an average of 6.4% more data augmentation.

Ii Related work

Ii-a Data Augmentation for Image Classification

There have been many proposed data augmentation techniques for image classification. For example, Cutout [Improved Regularization of Convolutional Neural Networks with Cutout, Devries] and Cutmix [CutMix: Regularization Strategy to Train Strong Classifiers With Localizable Features, Yun] techniques are proposed to cut out a portion of the image and paste it on the other part of the image in an random way. Data augmentation techniques proposed for text classification are also quite diverse. Examples of such techniques include: random erasing [Random Erasing Data Augmentation, Zhong], semantic augmentation [Semantic Data Augmentation for Deep Learning, Tao], random re-ordering of words [Text Augmentation for Learning Natural Language Inference Models, Zhang], random word deletion [Data Augmentation for Low-Resource Neural Machine Translation, Fadaee], etc. A detailed survey on image data augmentation techniques can be found in [A survey on Image Data Augmentation for Deep Learning, Shorten].

Ii-B Data Augmentation for Text Classification

Data augmentation techniques are very common in text classification tasks. The techniques proposed for text classification can be broadly divided into two categories: generative data augmentation and heuristic data augmentation. Generative data augmentation methods generate synthetic examples by training a generator model. For example, text completion [A Hybrid Method for Text Classification with Generative Adversarial Network and Deep Learning, Jahan] and word dropout [Data Augmentation for Low-Resource Neural Machine Translation, Fadaee] generate synthetic examples by training a generator model. Heuristic data augmentation techniques generate synthetic examples by performing some simple operations on the original examples. For example, Mixup [mixup: Beyond Empirical Risk Minimization, Zhang], Cutmix [CutMix: Regularization Strategy to Train Strong Classifiers With Localizable Features, Yun] and Cutout [Improved Regularization of Convolutional Neural Networks with Cutout, Devries] use randomly selected examples from the original examples and add them together.

Generative data augmentation techniques have been extended to other data types such as images, audio, video etc. [Data Augmentation for Text Classification: A Survey, Zhang]. However, the heuristic data augmentation techniques are used in the majority of the text classification tasks. Examples of heuristic data augmentation techniques for text classification include: the random swapping of two words [Text Augmentation for Learning Natural Language Inference Models, Zhang], the random insertion of a random word at random position in a sentence [Text Augmentation for Learning Natural Language Inference Models, Zhang], the random word deletion [Data Augmentation for Low-Resource Neural Machine Translation, Fadaee] etc.

Iii Problem Formulation

We consider a given data set $X=\{x_{1}, x_{2}, \ldots x_{N}\}$ where $x_{i} \in \mathbb{R}^{d}$ is the ith example with d dimension. The objective of supervised learning is to train a model f_{θ} for a given classification task. The model f_{θ} takes a set of input examples X and outputs a prediction of the class for each input example. We denote the set of labels for the data set as $Y=\{y_{1}, y_{2}, \ldots, y_{N}\}$ where y_{i} is the label for the ith example. The training set is composed of a set of pairs $S=\{(x_{i}, y_{i})\}_{i=1}^{N}$, where (x_{i}, y_{i}) is the pair for the ith example. The training objective is to find the model parameters θ^{*} that minimizes the training error.

The data augmentation technique, T_{aug}, is used to generate a set of augmented examples, $X^{\prime}=\{x_{1}^{\prime}, x_{2}^{\prime}, \ldots x_{N}^{\prime}\}$, from the original data set, X. The generated data set $X^{\prime}$ has similar characteristics as the original set X.

The training objective is to find the model parameters θ^{*} that minimizes the training error, $e_{train}(\theta)$, using the augmented set $X^{\prime}$. We denote the model trained using the augmented set $X^{\prime}$ as $f_{\theta}^{\prime}$.

Iv Proposed Framework

Fig. 1: Framework of the proposed framework

We propose a framework for automatic data augmentation which is shown in Fig. 1. The proposed framework is based on the following observations: 1) the model needs to be trained on the augmented data set for obtaining the better generalization performance, and 2) using the proposed framework one can achieve up to 2.4% higher accuracy on a real-world data set with an average of 6.4% more data augmentation.

We use the concept of domain adaptation in our framework. This idea is based on the observation that a model needs to be trained on the augmented data set for obtaining the better generalization performance. Let $\mathcal{L}_{train}$ be the training loss function, $\mathcal{D}$ be the domain, T_{aug} be the data augmentation technique, $X^{\prime}$ be the set of

Figure 4: **COVID-QA negative** examples (underline = entity | red = prompt) | olive = generation

```
displacementWhatIs()){
        EXPECT_EQ(1, dynamic_cast<const Shape&>(shape_->shape()).dimensions());
        EXPECT_EQ(1, shape_->rank());
        EXPECT_TRUE(IsSameRank(shape_->shape(), shape_));
        EXPECT_EQ(1, shape_->dimensions());
    }
}

TEST_P(ShapeOpTest, ShapeNotSameRank_1D){
  Shape shape_;
  shape_.set_rank(1);
  shape_.set_dims({1});
  Shape shape_copy = *shape_;
  shape_.set_rank(0);
  shape_copy.set_rank(1);
  Shape expected(shape_copy);
  TF_EXPECT_OK(
        shape_.AddScalar(2.0, /*add_shape=*/nullptr, /*is_dynamic=*/false));
  TF_EXPECT_OK(shape_.AddScalar(3.0, /*add_shape=*/nullptr, /*is_dynamic=*/false));
  TF_EXPECT_OK(shape_.AddScalar(4.0, /*add_shape=*/nullptr, /*is_dynamic=*/false));
  TF_EXPECT_OK(
        shape_.AddScalar(2.0, /*add_shape=*/nullptr, /*is_dynamic=*/true));
  TF_EXPECT_OK(shape_.AddScalar(3.0, /*add_shape=*/nullptr, /*is_dynamic=*/true));
  TF_EXPECT_OK(shape_.AddScalar(4.0, /*add_shape=*/nullptr, /*is_dynamic=*/true));
  TF_EXPECT_OK(shape_.AddScalar(2.0, /*add_shape=*/nullptr, /*is_dynamic=*/true));
  TF_EXPECT_OK(shape_.AddScalar(3.0, /*add_shape=*/nullptr, /*is_dynamic=*/true));
  TF_EXPECT_OK(shape_.AddScalar(4.0, /*add_shape=*/nullptr, /*is_dynamic=*/true));
  TF_EXPECT_OK(shape_.AddScalar(2.0, /*add_shape=*/nullptr, /*is_dynamic=*/true));
  TF_EXPECT_OK(shape_.AddScalar(3.0, /*add_shape=*/nullptr, /*is_dynamic=*/true));
  TF_EXPECT_OK(shape_.AddScalar(4.0, /*add_shape=*/nullptr, /*is_dynamic=*/true));
  TF_EXPECT_OK(shape_.AddScalar(2.0, /*add_shape=*/nullptr, /*is_dynamic=*/true));
  TF_EXPECT_OK(shape_.AddScalar(3.0, /*add_shape=*/nullptr, /*is_dynamic=*/true));
  TF_EXPECT_OK(shape_.AddScalar(4.0, /*add_shape=*/nullptr, /*is_dynamic=*/true));

  EXPECT_TRUE(ShapeIsSame(shape_, expected));
}

TEST_P(ShapeOpTest, ShapeNotSameRank_2D){
  Shape shape_;
  shape_.set_rank(1);
  shape_.set_dims({1, 2});
  shape_.set_strides({1, 2});
  shape_.set_sizes({1, 2});
  Shape shape_copy = *shape_;
  shape_.set_rank(0);
  shape_copy.set_rank(1);
  shape_copy.set_dims({1, 2});
  shape_copy.set_strides({1, 2});
  shape_copy.set_sizes({1, 2});
  Shape expected(shape_copy);
  TF_EXPECT_OK(
        shape_.AddScalar(2.0, /*add_shape=*/nullptr, /*is_dynamic=*/false));
  TF_EXPECT_OK(shape_.AddScalar(3.0, /*add_shape=*/nullptr, /*is_dynamic=*/false));
  TF_EXPECT_OK(shape_.AddScalar(4.0, /*add_shape=*/nullptr, /*is_dynamic=*/false));
  TF_EXPECT_OK(
        shape_.AddScalar(2.0, /*add_shape=*/nullptr, /*is_dynamic=*/true));
  TF_EXPECT_OK(shape_.AddScalar(3.0, /*add_shape=*/nullptr, /*is_dynamic=*/true));
  TF_EXPECT_OK(shape_.AddScalar(4.0, /*add_shape=*/nullptr, /*is_dynamic=*/true));
  TF_EXPECT_OK(shape_.AddScalar(2.0, /*add_shape=*/nullptr, /*is_dynamic=*/true));
  TF_EXPECT_OK(shape_.AddScalar(3.0, /*add_shape=*/nullptr, /*is_dynamic=*/true));
  TF_EXPECT_OK(shape_.AddScalar(4.0, /*add_shape=*/nullptr, /*is_dynamic=*/true));

  EXPECT_TRUE(ShapeIsSame(shape_, expected));
}

TEST_P(ShapeOpTest, ShapeNotSameRank_3D){
  Shape shape_;
  shape_.set_rank(1);
  shape_.set_dims({1, 2, 3});
  shape_.set_strides({1, 2, 3});
  shape_.set_sizes({1, 2, 3});
  Shape shape_copy = *shape_;
  shape_.set_rank(0);
  shape_copy.set_rank(1);
  shape_copy.set_dims({1, 2, 3});
  shape_copy.set_strides({1, 2, 3});
  shape_copy.set_sizes({1, 2, 3});
  Shape expected(shape_copy);
  TF_EXPECT_OK(
        shape_.AddScalar(2.0, /*add_shape=*/nullptr, /*is_dynamic=*/false));
  TF_EXPECT_OK(shape_.AddScalar(3.0, /*add_shape=*/nullptr, /*is_dynamic=*/false));
  TF_EXPECT_OK(shape_.AddScalar(4.0, /*add_shape=*/nullptr, /*is_dynamic=*/false));
  TF_EXPECT_OK(
        shape_.AddScalar(2.0, /*add_shape=*/nullptr, /*is_dynamic=*/true));
  TF_EXPECT_OK(shape_.AddScalar(3.0, /*add_shape=*/nullptr, /*is_dynamic=*/true));
  TF_EXPECT_OK(shape_.AddScalar(4.0, /*add_shape=*/nullptr, /*is_dynamic=*/true));
  TF_EXPECT_OK(shape_.AddScalar(2.0, /*add_shape=*/nullptr, /*is_dynamic=*/true));
  TF_EXPECT_OK(shape_.AddScalar(3.0,
```

Figure 5: **RadQA - normal prompt** (underline = entity | red = prompt) | olive = generation

Patient has post infectious scarring. FINDINGS AND IMPRESSION: *Our patient had multiple skin lesions on the chest wall and abdomen. Our patient had no evidence of cutaneous lesions elsewhere. CT thorax and abdomen were negative. Our patient had multiple bilateral pleural effusions. CT thorax was negative for any malignancy. Pleural biopsy showed acute and chronic inflammatory lesions. Our patient had acute hepatitis on day 11 of illness. Our patient was treated with ursodeoxycholic acid and oral prednisone. Our patient had a spontaneous improvement of symptoms after discharge from hospital. He was seen in clinic in clinic 2 weeks after discharge and was asymptomatic. He was seen in clinic 4 weeks after discharge and was asymptomatic. He was seen in clinic 6 weeks after discharge and was asymptomatic. He was seen in clinic 8 weeks after discharge and was asymptomatic. He was seen in clinic 10 weeks after discharge and was asymptomatic. He was seen in clinic 14 weeks after discharge and was asymptomatic. He was seen in clinic 28 weeks after discharge and was asymptomatic. DISCUSSION: Our patient developed multiple skin lesions during his illness. Our patient developed acute hepatitis on day 11 of illness. We suspect that our patient developed skin lesions as a result of acute hepatitis. Our patient was given an intravenous steroid and ursodeoxycholic acid. We suggest that steroids are contraindicated in patients with active skin infections. Our patient was asymptomatic at the last follow-up. He was advised to continue the treatment for acute hepatitis. Our patient was lost to follow up after discharge from hospital. We recommend to follow up the patients for a longer period.*

Patient has Questionable tiny right apical pneumothorax. FINDINGS AND IMPRESSION: In this case the patient is suffering from right apical pneumothorax. The patient is suffering from a huge right apical pneumothorax and has a huge right apical pneumothorax. The patient is suffering from a huge right apical pneumothorax.

[text continues repeating "The patient is suffering from a huge right apical pneumothorax." for the remainder of the box]

Figure 6: **RadQA - fancy prompt** (underline = entity | red = prompt) | olive = generation

---

Question: What is DC-GENR and where is it expressed?
ChatGPT: *As an AI language model, I do not have access to up-to-date information or research beyond my training data. Therefore, I am not aware of any current information on "DC-GENR." It is possible that this refers to a specific term or concept within a particular field or context. If you could provide me with more context or information about what you are referring to, I may be able to assist you further.*

Question: What is the main cause of HIV-1 infection in children?
ChatGPT: *The main cause of HIV-1 infection in children is mother-to-child transmission of the virus during pregnancy, childbirth, or breastfeeding. This is also known as vertical transmission. If the mother is HIV-positive and not receiving treatment, there is a high risk of transmitting the virus to her baby. However, with proper medical care and treatment, the risk of mother-to-child transmission can be significantly reduced.*

Question: What plays the crucial role in the Mother to Child Transmission of HIV-1 and what increases the risk?
ChatGPT: *The level of the mother's HIV viral load plays a crucial role in mother-to-child transmission of HIV-1. A higher viral load in the mother's blood and genital secretions increases the risk of transmission to the baby. Factors that can increase the risk of mother-to-child transmission include lack of antiretroviral therapy during pregnancy, vaginal delivery, and breastfeeding. However, with proper medical care and treatment, the risk of mother-to-child transmission can be greatly reduced. Pregnant women living with HIV should seek medical care and treatment to help protect their babies*

Figure 7: Testing ChatGPT

since its questions were heavily linked to the patient report on hand and were not as *general* as the former's. Figure 7 shows a few examples from our trials. As we can see, it seems to answer the questions correctly. However, as mentioned before, these answers need validation from a professional in the field. Also, we see that sometimes it **cannot answer** questions on topics not found in its training data, a problem not faced by our encoder models which rely only on the provided context.

D Hardware Details

To run our corpus generation code, we used a total of 8 40GB NVIDIA A100 or 8 32GB NVIDIA Tesla V100 cards, depending on our institute's job scheduler and availability. Although the former is a superior card, we preferred the latter since it gives a relatively quick turnaround and is a general-purpose GPU, unlike the A100 which is required for even heavier computing. We were requested by our system administrator to use the V100 over the A100, if it was possible, to avoid blocking off even more compute-heavy experiments, which we did not have an issue obeying. That being said, if the A100 cluster was free, we did take advantage of it to run our code. For pre-training and fine-tuning BERT/RoBERTa, we used only 1 A100 80GB GPU as RoBERTa was unable to fit on the 40/32 GB cards owing to the large batch size.

E Prompts Tried

As mentioned before, we knew the prompt to be used for COVID-QA since it was a feature of Galactica. However, we had to experiment with a few prompt styles for RadQA before landing on our final one. We describe the prompts we tried below for the entity `small-bowel injury` and the reasons for not using them. The text in `courier` is the prompt and in *italics* is the generated text.

- `IMPRESSION:` `small-bowel injury` *(SBI) was defined as injury to the small bowel wall, or transmural disruption of the bowel wall, in the absence of penetrating abdominal trauma (The management of small-bowel injuries in the emergency room., Brown). Patients were excluded if the SBIs occurred after admission to the emergency department or were complicated by perforation.   ...* (truncated)

  – Issue - Did not include the `FINDINGS` section.

- `FINDINGS:` `small-bowel injury` *was observed in 11.1% and 21.9% of patients, respectively. Small-bowel injury associated with liver injury occurred in 10.1%. The mortality rate was significantly higher in the small-bowel injury group than in the non-small-bowel injury group (11.6% vs. 2.1%, P = 0.005). This study shows that small-bowel injury is associated with a high mortality rate, and that this complication is an important cause of morbidity and mortality in patients with severe acute pancreatitis.*

  – Issue - Did not include the `IMPRESSIONS` section.

- `Patient has small-bowel injury.` `FINAL REPORT` *This patient was treated with conservative methods. The small-bowel injury was treated surgically with resection of the bowel, primary anastomosis, and end-to-end anastomosis between the ends of the resected bowel. She did not have recurrence of intestinal obstruction after 3 years.*

  – Issue - Did not include either `IMPRESSIONS` or `FINDINGS` section.

- `John Doe has small-bowel injury.` `IMPRESSIONS` *: This case study was made possible by his support by a research grant from the National Institutes of Health (CA 10644).*

  – Issue - Had to make sure that the pronouns used were same throughout the report otherwise it would not be consistent.

- `Jane Doe has small-bowel injury.` `IMPRESSIONS` *: This case study was made possible by the support of a research grant from the Department of Veterans Affairs.*

  – Issue - Same as above.

- `Patient I has small-bowel injury` *He has no history of abdominal trauma or surgery. At first, he has only mild abdominal pain. But with the progress of disease, he has become distressed. At*

7051

the initial examination, the abdominal pain was alleviated, but the patient still has slight distention of abdomen. He was admitted to our hospital on the third day of admission. His vital signs were: body temperature 37.5 °C, blood pressure 110/70 mmHg, pulse 105 beats/min, respiratory rate 26 breaths/min and oxygen saturation 97% on room air. He had mild abdominal distention. The results of routine blood tests, urinalysis and abdominal ultrasound examination were normal. X-ray showed free air in the abdomen. ... (truncated)

- Issue - Galactica could get confused between I (alphabet) and I (roman numeral).

- `Patient A has small-bowel injury` *(Fig. 1). He has no history of abdominal trauma or surgery. At first, an abdominal CT scan was performed to evaluate abdominal pain. Abdominal CT showed a partial small-bowel obstruction at the splenic flexure of the colon (Fig. 2). An oral contrast medium was then administered via a nasogastric tube and an abdominal CT scan was performed. ...* (truncated)

- Issue - Galactica could get confused misinterpret "A" for the beginning of a sentence (it was observed for a few cases)

- `Patient alpha has small-bowel injury` *requiring an open operation and was diagnosed with C. difficile colitis. The patient had a history of severe anemia and leukopenia, which were also observed in patient beta. Patient alpha was treated with a single course of intravenous moxifloxacin 400 mg/day, which resulted in the resolution of the diarrhea. He was followed closely after discharge. In patient beta, he had a similar episode of diarrhea after hospital admission, but the diarrhea did not resolve. He was prescribed with an oral vancomycin 1000 mg twice a day, and the diarrhea was resolved. ...* (truncated)

- Issue - This would have been a good choice. However, as we see above, patient qualifiers such as `A`, `I` and `alpha`, *might* bias the model towards more *male*

patients (He). As such, we decided to drop the qualifier altogether.

F   Hyperparameters Used

Experiment	Hyperparameters
Corpus Generation	random seed: 42 renormalize_logits: True do_sample: True max_length (prompt + generated tokens): 2,048 top_p: 0.9 temperature: 0.9
Pre-Training	batch_size: 40 learning_rate: 5e-5 epochs: 3
Fine-Tuning (SQuAD)	batch_size: 16 max_input_length (question + context): 384 stride: 128 learning_rate: 2e-5 epochs: 3 n_best (top n answer spans): 20 max_answer_length: 30 optimizer_type: AdamW
Fine-Tuning (COVID-QA)	batch_size: 40 max_input_length (question + context): 384 stride: 128 learning_rate: 2e-5 epochs: 1 n_best (top n answer spans): 20 max_answer_length: 1000 optimizer_type: AdamW
Fine-Tuning (RadQA)	batch_size: 16 max_length: 384 stride: 128 learning_rate: 3e-5 epochs: 1 n_best (top n answer spans): 20 max_answer_length: 1000 optimizer_type: AdamW

Table 3: Hyperparameters for each experiment. We use three random seeds during pre-training and fine-tuning, `41,42,43` but only `42` when generating the corpus. This is done since otherwise running the entire pipeline from generation to training across all ablations would take an infeasible amount of time.

Hyperparameters for each experiment are detailed in Table 3. These were selected mostly from preexisting implementations or through minimal exploration of known settings.

G   Note on Stability

All of our experiments were run using `PyTorch 1.13.1` and `Huggingface 4.26.1`. However, we have noticed fluctuations in results when training with other versions of these libraries. Thus, to replicate our scores to the best extent, we recommend installing the aforementioned versions of the packages.

H   Responses to Reviews

In this section, we provide clarifying statements to questions raised during the review process.

- *The generative models used in the evaluation (Galactia, MedLLaMMA, and MedAlpaca) do not appear to be producing valid outputs. One possible cause of this is the lack of instruction following ability in the models, it may be necessary to try instruction tuning or few-shot prompting that are suited to the assumed task or to try more capable recent generative models.*

  - While not being able to produce valid outputs is a legitimate concern, we point out that EQA systems are evaluated using EM and F1, both of which count token overlap with the gold answer. As these models are free to produce text in an unconstrained manner, their answers display poor alignment with the gold reference. We observe that this is more often the case than producing invalid responses.
  - Considering the complexity of the task, it would make sense to try instruction tuning, i.e., instead of span-extraction, make the objective span generation. That said, span generation is typically used for open-ended QA, i.e., abstractive QA instead of EQA. Additionally, we have not found studies in the existing literature that approach EQA in this manner. However, we agree that this will be an interesting experiment for the project. As for trying more capable LLMs, we tried Chat-GPT (3.5) on a few samples as shown in Appendix C. While we did not provide a context in those trials, we estimate using OAI documentation that processing the entire COVID-QA dataset would require upwards of $450, which is quite expensive.

- *In your trial, how much does the proposed method depend on the prompts? Have you noticed any limitations in the prompt design?*

  - This is a key point to the setup. The goal is to generate data that matches the content & style of the downstream dataset. As such, writing prompts to obey this observation is very important. The prompt for COVID-QA was decided based on Galactica's pre-training instructions. RadQA's prompt required data examination to determine important features such as "Findings" and "Impressions". Determining how much the overall method is impacted by prompt choice is challenging as it requires us to rerun the pipeline several times for corpora generated using different prompts. As such, our goal was to first write a prompt to mimic the downstream dataset's contexts as best as possible, before generation and training.

- *I would also have liked to have had even a brief indication of the results that can be obtained in a specialist field other than medicine.*

  - Although in this paper, we have shown a method to develop corpora for the medical domain using dataset-specific entities, there is no reason why the same cannot be applied to other closed domains such as legal or finance. This ideally will involve studying the dataset to gain insights on the linguistic characteristics to be mimicked by the synthetic data and using an associated entity recognizer that can perform better than general domain ones to better identify important entities.

- *One of the first issues addressed in the abstract and the introduction is the challenge of potential hallucinations of autoregressive LLMs. The pre-training framework TOP-Training is proposed "to address these challenges". However, the hallucination issue is neither addressed nor evaluated in the paper. The authors are aware of this problem and mention it as a limitation of the paper. While this is honest and appreciated, the objectives, abstract and introduction need to be more clarified.*

  - The main point we highlight is that "hallucination" is a point of weakness in LLMs for solving EQA. However, we can still utilize their generation capabilities to produce synthetic "pre-training" data. The point on hallucination we mention under "Limitations" is that we do not implement a strategy to detect/filter such instances in the pre-training corpus as opposed to the task generations, when we apply the LLMs directly for the task.

7053

- *Regarding the limitations of autoregressive LLMs for the EQA task, as written in the paper, their capabilities with EQA datasets are unexplored, and while one of the possible contributions of the paper was to show their limitations, autoregressive models are not evaluated in depth, which is normal given the scope of the article. The explanation of why they are not effective should be a bit more explicit.*

  – As we've explained throughout the paper, such as the introduction, related work and benchmarking on both datasets (Table 1, 2), autoregressive LLMs are suboptimal for EQA. Apart from hallucination, another reason is the way EQA systems are evaluated, i.e., EM and F1, both of which require token overlap with the gold reference. As these models generate text in an open-ended manner, their performance drastically takes a hit.

- *TOP-Training is a complex framework involving several important steps such as data argumentation and double fine-tuning with an open and specific domain dataset. This complexity makes the reasons why the method works less clear. Is it the combination of data augmentation and double fine-tuning that is successful, or would just fine-tuning in a specific domain dataset be enough, like is done in (Seo et al., 2024)? How does the use of different NER systems affect the results? etc.*

  – TOP-Training refers to the entire pipeline of entity extraction + data generation + model training. While there are several steps, the complexity of our framework is mitigated by the fact that each step is straightforward and has a reason. First, double pre-training is needed to align the open-domain model with the target domain. Next, the two rounds of fine-tuning are needed to first learn the task on a high-quality dataset with several samples before approaching the target dataset with far less samples (both COVID-QA and RadQA have ∼8K samples << than SQuAD's 100K samples). We also find that directly fine-tuning models on the target dataset, i.e., without the first round of fine-tuning, leads to much worse performance as the ran-

domly initialized span-extraction head is unable to learn from those few samples. The impact of different NER tools is a bit more straightforward in that using a domain-aligned entity recognizer will lead to a higher number of quality identifications. That said, we did initially try Stanford's Stanza (Qi et al., 2020), but found Spacy's extractions to be qualitatively better and thus decided to use it.

I Model Cards

Model	Model Card (URL)
BERT-Base, Cased	bert-base-cased
BERT-Base, Cased, SQuAD v1	batterydata/bert-base-cased-squad-v1
BERT-Base, Cased, SQuAD v2	deepset/bert-base-cased-squad2
RoBERTa-Base	roberta-base
RoBERTa-Base, SQuAD v1	csarron/roberta-base-squad-v1
RoBERTa-Base, SQuAD v2	deepset/roberta-base-squad2
BioBERT	dmis-lab/biobert-base-cased-v1.2
SciBERT	allenai/scibert_scivocab_uncased
SciBERT (+CORD-19)	lordtt13/COVID-SciBERT
PubMedBERT	microsoft/BiomedNLP-PubMedBERT-base-uncased-abstract-fulltext
BlueBERT	bionlp/bluebert_pubmed_mimic_uncased_L-12_H-768_A-12
CODER	GanjinZero/UMLSBert_ENG
LUKE	studio-ousia/luke-base
XLNet, SQuAD v1	arrafmousa/xlnet-base-cased-finetuned-squad
STonKGs *	stonkgs/stonkgs-150k
RadBERT	zzxslp/RadBERT-RoBERTa-4m
Clinical BERT	emilyalsentzer/Bio_ClinicalBERT
BioMed-RoBERTa	allenai/biomed_roberta_base
MedLLaMA	chaoyi-wu/MedLLaMA_13B
MedAlpaca	medalpaca/medalpaca-13b
Galactica	facebook/galactica-1.3b
Longformer, SQuAD v1	valhalla/longformer-base-4096-finetuned-squadv1
Longformer, SQuAD v2	mrm8488/longformer-base-4096-finetuned-squadv2
BigBird, SQuAD v1	FredNajjar/NF-bigbird-squad
BigBird, SQuAD v2	FredNajjar/bigbird-QA-squad_v2

Table 4: Model cards and URLs for all models used in our paper. * We wanted to use STonKGs (Balabin et al., 2022). However, there was no vocabulary file for the model which resulted in errors.

All models used in this study were downloaded from HuggingFace [5]. Each model, along with its model card (name as it appears in the HuggingFace model hub) and URL is listed in Table 4.

---

[5] https://huggingface.co/