Self-Specialization:
Uncovering Latent Expertise within Large Language Models

Anonymous ACL submission

Abstract

Recent works have demonstrated the effectiveness of self-alignment in which a large language model is aligned to follow general instructions using instructional data generated from the model itself starting from a handful of human-written seeds. Instead of general alignment, in this work, we focus on self-alignment for expert domain specialization (e.g., biomedicine, finance). As a preliminary, we quantitively show the marginal effect that generic instruction-following training has on downstream expert domains' performance. To remedy this, we propose **self-specialization** - allowing for effective model specialization while achieving cross-task generalization by leveraging only a few labeled seeds. Self-specialization offers a data- and parameter-efficient way of "carving out" an expert model out of a generalist pre-trained LLM. Exploring a variety of popular open large models as a base for specialization, our experimental results in both biomedical and financial domains show that our self-specialized models outperform their base models by a large margin, and even larger models that are generally instruction-tuned or that have been adapted to the target domain by other means. Our code will be released upon acceptance.[1]

1 Introduction

Instruction-tuning (Ouyang et al., 2022; Wei et al., 2022; Mishra et al., 2022; Su et al., 2022) of large language models (LLMs) offers a mechanism to adeptly guide models using specific directives, thereby enhancing their versatility across diverse tasks. However, as promising as this concept might seem, it poses an inherent challenge: the substantial need for quality data (Chung et al., 2022; Wan et al., 2023; Köpf et al., 2023). The very premise of instruction-tuning hinges on the availability of well-crafted, human-annotated data, a resource that is both time-consuming and challenging to scale efficiently (Honovich et al., 2022; Kang et al., 2023).

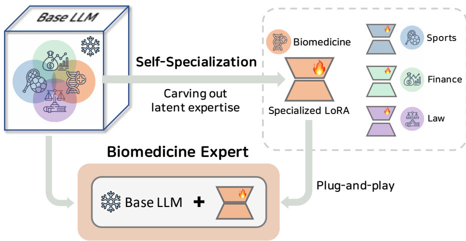


Figure 1: Self-specialization concept. Expertise in various domains is mixed and latent within base LLMs, and can be carved out through self-specialization.

When it comes to specialized domains, such as biomedicine, it is more challenging to acquire human labels, due to the need for expert annotators (Wang et al., 2023b). While adaptation through in-domain pre-training (Gururangan et al., 2020; Wu et al., 2023) has been shown to be effective, this approach requires extensive (unlabeled) target-domain data, in addition to significant computational resources. Moreover, prior work has shown the benefits of adaptive pre-training can be less than those achieved by moderate amounts of fine-tuning data from the target domain (Bai et al., 2021).

Emerging as a promising solution to this data-intensive challenge in the context of instruction-tuning is the approach of self-alignment (Wang et al., 2022a; Sun et al., 2023). By allowing LLMs to automatically generate instructional data from minimal human-authored seeds, self-alignment presents a means to harness the internal general knowledge of models, which results from extensive pre-training on internet corpora (Devlin et al., 2019; Raffel et al., 2020; Brown et al., 2020), without extensive human annotations.

However, a pertinent question remains: How effective are the self-aligned models when applied to more niche domains, such as biomedicine? Given that both the initial pre-training and subsequent self-alignment are general, the knowledge embedded

[1] Major updates are highlighted in color in this revision.

in LLM parameters may be a mixture of semantics and various domains. This raises questions about their effectiveness in specialized domains, despite the aims of instruction-tuning and self-alignment for cross-task generalization. In our preliminary study, however, we find that existing models such as Alpaca (Taori et al., 2023) and Dromedary (Sun et al., 2023), although aligned, exhibit only a modest degree of improvement within the specialized domains. These observations underline the need for focused approaches that can leverage the domain expertise existing in the base models, to ensure the self-generated instruction-tuning data remains both contextually appropriate and accurate.

In this work, we explore the possibility of **self-specialization** (Fig. 1). Drawing inspiration from the foundational principles of self-alignment, self-specialization goes a step further by incorporating domain-specific seed instructions and is further bolstered by parameter-efficient fine-tuning, as well as optional iterative refinement and retrieval components. Our goal is to guide models beyond generic alignment, directing them to generate data that are not just contextually fitting for a specialized domain but also maintain high accuracy.

We evaluate our self-specialized models within the biomedical and finance domains (20 datasets in total), and across a variety of base models that we specialize. Surprisingly, despite the simplicity of our approach, our results present a compelling case for self-specialization significantly outperforming the base models, and even larger models that are generally instruction-tuned or specifically pre-trained on the target domain. Notably, our self-specialized one based on MPT-30B (Team, 2023) for biomedicine even surpasses larger models (based on LLaMA-65B (Touvron et al., 2023a)), including the ones improved through self-alignment by leading methods (Wang et al., 2022a; Sun et al., 2023).

2 Preliminaries: Benchmarking Existing Aligned Models

To motivate our exploration of self-specialization, we first begin by addressing a fundamental question: How well do generally aligned models perform on specialized domains? While popular models, such as Alpaca (Taori et al., 2023) and Dromedary (Sun et al., 2023), have demonstrated effectiveness in following general instructions, it remains unclear whether general alignment can also elicit expertise for a certain domain.

	BASE	ALIGNED	
Model	**LLaMA-65B**	**Alpaca-65B**	**Dromedary-65B**
Averaged F_1-SCORE	43.87	46.39 (+2.52)	45.10 (+1.23)

Table 1: Benchmarking results of a base LLaMA-65B and its aligned variants in a biomedical domain. The evaluation covers various NLP tasks such as question answering, information extraction, and classification. 5-shot results averaged across 10 datasets are presented.

Investigating this, we assess the capabilities of Alpaca and Dromedary against their base model, LLaMA-65B (Touvron et al., 2023a), on a collection of benchmarks within the biomedical domain. We evaluate Alpaca as an upper bound, due to its reliance on GPT-3.5-generated datasets (Ouyang et al., 2022) via the self-instruct process (Wang et al., 2022a), unlike Dromedary, which generates instructional data from its base model. We use 10 biomedical NLP datasets (see Section 4 for details), covering a diverse set of tasks to ensure a comprehensive mix of content and also to look at the cross-task generalization, the core of instruction-tuning. Table 1 summarizes the result.

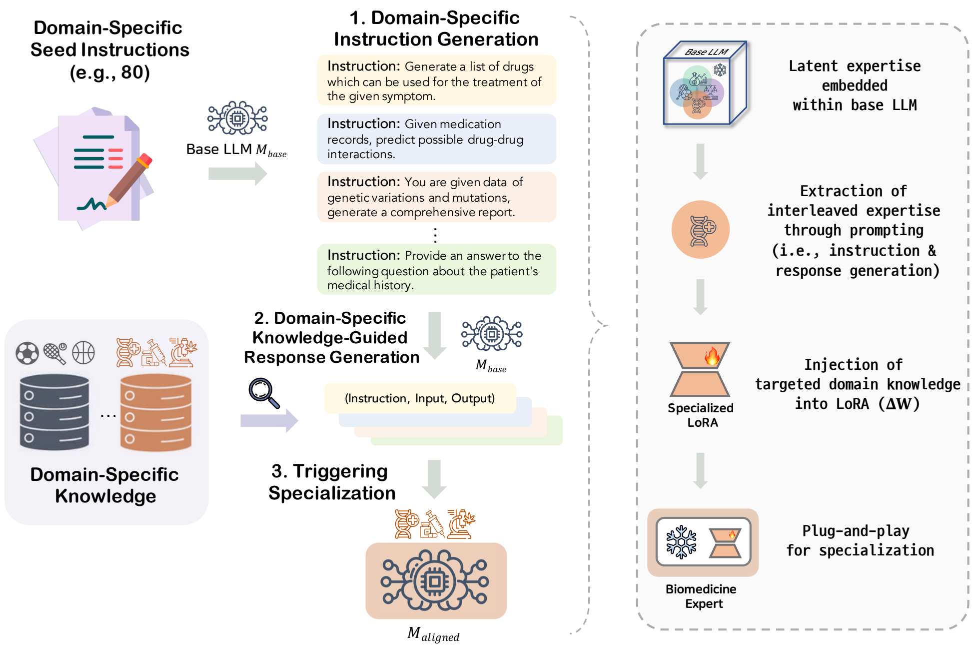
We find that both Alpaca and Dromedary have only a slight (1.2 - 2.5) advantage over LLaMA in biomedicine. While they are aligned to handle a broad set of instructions, they do not seem to effectively improve their specialized domain expertise; intuitively trading their expertise for generality given finite parameters. In light of these findings, it becomes evident that for cases where we are only interested in expert domains for all our downstream tasks, there remains a large potential for improvement beyond the generic alignment. This underscores the need for a model or approach, like self-specialization, that could potentially uncover specialization while maintaining cross-task generalizability with minimal supervision.

3 Self-Specialization

In this section, we describe our method called self-specialization illustrated in Figure 2.

3.1 Seed Demonstrations

Initially, we utilize a curated set of seed demonstrations S, consisting of a triplet (i, c, y), comprised of instruction i, a context c (e.g., passage), and a response y, respectively. Recognizing the difficulty of acquiring domain-specific data in real-world scenarios (Bai et al., 2021), we aim for a very mini-



(a) Self-Specialization Process (b) Abstraction of Process

Figure 2: **Self-Specialization** overview. (a) We start with a small set of human-authored domain-specific seed instructions. The base model crafts synthetic instructions and corresponding input contexts tailored to that particular domain. Subsequently, during the response generation phase, responses are curated given the generated instruction and input pairs, optionally enhanced by infusing domain-relevant knowledge obtained via a retrieval component or iterative re-generation via our previous self-specialized model. Finally, in the specialization phase, the base model is tuned for specialization (w/ QLoRA) to uncover its target domain expertise. (b) Conceptually speaking, this process can be described as uncovering latent expertise within LLMs.

mal number of seeds: only 80 for the biomedical domain and 90 for the financial domain[2]. We leverage established datasets such as Box (Parmar et al., 2022) for seed construction to fairly ensure quality (detailed in Section 4). These seeds capture essential domain concepts but are insufficient to cover the entirety of domain knowledge. We conjecture (and demonstrate) that the domain knowledge already exists within large pre-trained models in a latent state, which our approach can uncover. Seeds provide the primary scaffold upon which subsequent domain-specific instructions are built.

3.2 Domain-Specific Instruction Generation

With the seed instructions in place, we move to generating domain-specific instructions. While these new instructions are grounded in the initial seeds, they grow to cover a comprehensive scope of the domain. Specifically, a base model M_{base}, such as MPT-30B (Team, 2023) which is large enough, is prompted to produce new combinations of (i, c) given a handful of seed demonstrations which are randomly sampled from the initial seeds pool. The newly formed instructions i, coupled with their corresponding input contexts c, shape a blueprint that the model utilizes in the following stages.

[2]While manual annotation of seed data is an assumed prerequisite for this initial step in self-alignment, we consider those numbers to be reasonable to annotate.

3.3 Domain-Specific Response Generation

In this phase, it is crucial for the responses not only to be correct but also to be well-aligned with the target domain. Intuitively, as this phase is conditioned on domain-specific instructions $\{i\}$ and corresponding contexts $\{c\}$, derived from domain-specific seeds, it may be sufficient to rely on the base model itself to generate domain-specific responses. As an additional effort, we explore whether leveraging external domain-relevant knowledge would be beneficial for this case, inspired by Frisoni et al. (2022). Therefore, we optionally allow M_{base} to incorporate external knowledge via a retrieval component M_{ret}. Specifically, forming the query x as a concatenation of i and c, M_{ret} fetches top-k relevant documents $d_{1:k}$.

$$d_{1:k} = M_{ret}(x = i \oplus c)$$

Then, each document d_j is independently paired with the query x to form a prompt to M_{base}, and the final domain-specific responses y are produced from the final distribution computed by marginalizing over the probabilities of each of these k-combinations at each generation step.

$$p(y|x) = \prod_{i}^{t} \sum_{j}^{k} p_{ret}(d_j|x; M_{ret})\, p_{lm}(y_i|x, d_j, y_{1:i-1}; M_{base})$$

where p_{ret} is a relevance score (similarity) from a retriever module and p_{lm} represents the language model distribution. By integrating such external information, while domain-specific knowledge is already deemed latent within LLMs, this step further encourages the generated target responses to be more nuanced and domain-specific, leading to additional improvements (Section 5.2).

3.4 Triggering Specialization

Upon establishing a set of domain-specific instructions/responses, M_{base} undergoes tuning using the self-generated data, adjusting its internal parameters to cater specifically to the domain's nuances. This step is crucial, marking the model's transformation from being generally competent to being domain-specialized while preserving cross-task generalizability, thus resulting in the final self-aligned domain-specialized model: $M_{aligned}$.

3.5 Iterative Self-Specialization

In the spirit of continuous improvement, our approach optionally supports iterative self-specialization via re-generating instructions and responses with the better-aligned model $M_{aligned}$. This process has the potential of refining the model's domain expertise with each iteration (of considering the previous iteration $M_{aligned}$ as base each time), iteratively improving its responses.

4 Experimental Settings

Datasets. For our primary evaluation, we employ various biomedical NLP datasets, most of which are curated in BIGBIO (Fries et al., 2022). A total of 10 different datasets are adopted to encompass a wide range of NLP tasks: Question Answering (QA), Named Entity Recognition (NER), Relation Extraction (RE), Sentiment Analysis (SA), and Document Classification (DC). Following a prior work (Parmar et al., 2022), all datasets are transformed into instructional data. Additionally, we validate our method in the financial domain to showcase its generalizability. We adopt a total of 10 diverse datasets, covering numerous NLP tasks: Summarization (SUM), QA, NER, RE, SA, and Classification (CLS), detailed in Appendix A.

Models. We employ MPT-30B (Team, 2023) as a base model for main experiments. For the retriever, we use simple yet effective BM25 (Robertson et al., 1994), assuming human-labeled data is not sufficient. For benchmarking of general-purpose aligned models, we evaluate Alpaca-65B (Taori et al., 2023) and Dromedary-65B (Sun et al., 2023) that are both based on LLaMA (Touvron et al., 2023a). In addition to MPT-30B, we adopt LLaMA-2 7B (Touvron et al., 2023b) and Falcon-40B (Almazrouei et al., 2023) to further validate the general applicability of self-specialization with different scales and base models. We additionally evaluate existing domain-specific models (Wu et al., 2023): MedLLaMA and PMC-LLaMA (Details are in Section 5.2).

Metrics. In our study, all tasks are approached as a unified text generation problem, aiming to assess the capabilities of generative models. In alignment with an established convention (Parmar et al., 2022), we adopt F_1-SCORE as our main evaluation metric, given an early observation that ROUGE-L (Lin, 2004), as shown in Table 6 in Appendix, exhibits a strong correlation with F_1-SCORE.

Implementation Details. For biomedical seeds, we use data sampled from BoX (Parmar et al., 2022), encompassing 32 tasks, up to 5 instances for each dataset, resulting in a compact yet representative 80 seed samples in total, which are also used as demonstrations at inference. For optional external corpus, we leverage PubMed preprocessed in (Phan et al., 2021), which contains ≈30M abstracts. In the financial domain, based on our finding from biomedical experiments that showed surprising effectiveness of self-specialization relying on internal knowledge of LLMs without the external corpus, we opt not to employ an optional retrieval component to further validate the self-sufficiency of LLMs. We leverage a total of 90 seeds sampled from the 10 train sets in our corresponding benchmark datasets. We use a total of 5K synthetic data generated through our self-specialization for all experiments, unless otherwise specified. Being equipped with QLoRA (Dettmers et al., 2023) and 4-bit quantization, the model is trained using a simple Alpaca-style template (Taori et al., 2023) on a single A100, taking only a few hours for 3 epochs, resulting in a light-weight specialization module.

Biomedicine		k=0		k=1		k=5	
Task	**Dataset**	**Base**	**Self-Specialized**	**Base**	**Self-Specialized**	**Base**	**Self-Specialized**
QA	BioASQ-Factoid	30.90	**37.35**	47.56	**55.04**	51.96	**57.61**
	BioASQ-List	46.06	**46.99**	**47.57**	44.55	35.09	**42.17**
	BioASQ-Yesno[3]	21.20	**85.27**	10.80	**94.00**	8.80	**95.20**
	PubMedQA	11.98	**24.16**	**28.89**	24.87	**31.69**	31.31
NER	AnatEM	9.63	**11.99**	7.57	**15.76**	6.59	**21.25**
	BioNLP13CG	24.79	**24.93**	21.76	**31.80**	26.03	**41.16**
	NCBI	**18.46**	14.35	27.88	**43.11**	17.99	**46.54**
RE	DDI	**51.00**	49.40	49.20	**51.60**	49.38	**53.40**
SA	Medical Drugs	35.00	**65.80**	11.40	**54.60**	11.40	**32.80**
DC	HoC	2.44	**6.01**	**13.91**	7.61	**62.84**	62.65
Average		25.15	**36.63**	26.65	**42.29**	30.18	**48.41**

Finance		k=0		k=1		k=5	
Task	**Dataset**	**Base**	**Self-Specialized**	**Base**	**Self-Specialized**	**Base**	**Self-Specialized**
SUM	EDT-Summarization	6.40	**21.90**	13.97	**24.00**	13.87	**23.56**
QA	InsuranceQA	3.03	**19.87**	6.55	**23.79**	9.96	**24.36**
	ConvFinQA	**15.74**	5.25	**21.69**	11.84	**28.77**	20.88
NER	Fin3	9.94	**23.93**	7.53	**26.95**	6.80	**43.87**
	FiNER_139	10.24	**14.84**	**36.78**	25.81	**44.34**	35.63
RE	KPI-EDGER	11.22	**31.02**	43.28	**53.56**	49.46	**63.90**
SA	EarningsCall	46.80	**48.80**	**50.80**	48.00	**49.03**	47.74
	Financial_Phrasebank	23.60	**73.20**	9.40	**47.60**	29.20	**68.80**
	FIQA-SA	44.44	**56.84**	58.55	**61.54**	61.54	**70.09**
CLS	Gold Commodity News	21.95	**43.03**	**61.93**	55.08	38.42	**61.20**
Average		19.34	**33.87**	31.05	**37.82**	33.14	**46.00**

Table 2: Comparative results (F_1-Score) of the base LM and self-specialized one on biomedical (top) and financial (bottom) domains. The base model is MPT-30B for biomedicine and LLaMA-2 7B for finance. Self-specialized ones have the same parameters as the counterpart base ones. *k* indicates the number of demonstrations in a prompt.

5 Results and Analyses

Here, we provide a set of experimental results and analyses to address relevant research questions.

5.1 Comparison with Baselines

How effective is the self-specialization of base models? In Table 2, we present the comparative results of our self-specialized model against its base counterpart across 10 distinct biomedical NLP and 10 financial NLP datasets. The evaluation is conducted with varying numbers of in-context demonstrations, *k*.

Our findings reveal that the self-specialized model exhibits remarkable progress in the majority of tasks across all configurations in both domains, yielding a substantial (up to 18 points) improvement in average scores. Specifically, the average scores (F_1) in biomedicine rise from 30.18 to 48.41 in a 5-shot setting.[3] In finance, the improvements are 14.53 (0-shot), 6.77 (1-shot), and 12.86 (5-shot), respectively. These advancements in both domains underscore the self-specialization's generalizability in addressing a wide array of tasks across different specialized domains.

Imact on ICL capability. A potential concern on self-specialization tuning is its impact on the base LLM's in-context learning capabilities, as we did not tune the model with demonstrations. Comparing the capabilities before and after self-specialization, the improvement after adding demonstrations (from 0 to 5) of our self-specialized model on biomedicine in Table 2 is 36.63 to 48.41 (Δ=11.78), while that of the base model is 25.15 to 30.18 (Δ=5.03), indicating even better ICL capability with in-domain knowledge acquisition.

Performance drop on some tasks. Our analysis does identify a few instances where performance drops as shown in Table 2. This indicates room for further refinement, especially for tasks like ConvFinQA that require a set of specific capabilities beyond mere domain knowledge. We evidenced

[3]Even excluding BioASQ-Yesno as an outlier due to the base model's low performance, self-specialization still shows significant gain over the base model: 32.55 to 43.21 (5-shot). Appendix C.3 includes the detailed discussion.

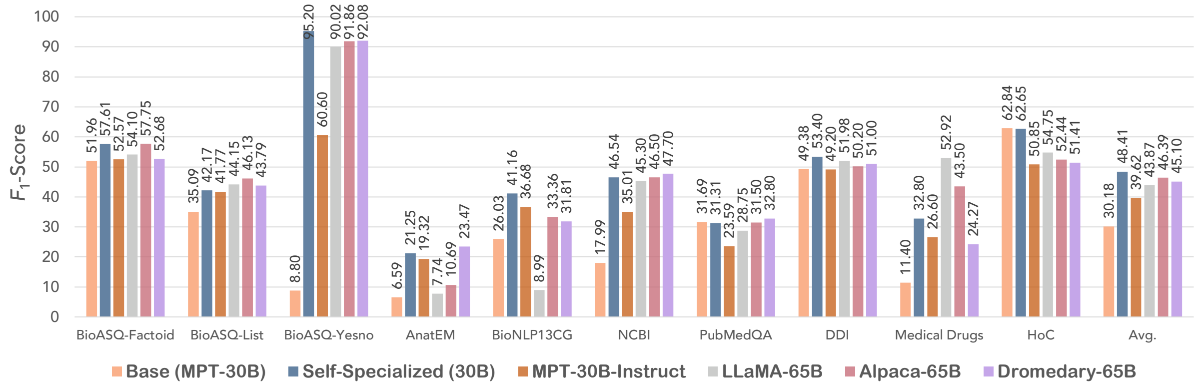


Figure 3: Comparing (with F_1-SCORE, 5-shot) our self-specialized MPT-30B model to 65B models in biomedicine.

that a minor proportion ($\leq 2\%$) of generated data partially resembles ConvFinQA, due to our generation's nature involving creative brainstorming for diversity. The specific demands of ConvFinQA, including numerical reasoning, structured tables, and conversations extend beyond basic domain knowledge and were insufficiently covered within our dataset. This gap likely contributes to the observed performance trade-offs.

However, we re-emphasize that there are significantly bigger gains in many of the cases (e.g., 45 out of 60 experiments across datasets and k), outweighing the regression overall. Acknowledging the inherent variability of in-context learning (Min et al., 2022), we present the variances with 5 different sets of demonstrations in Figure 4 based on LLaMA-2-7B in biomedicine, showing significant average improvements of 8.25 ($p = 0.003, k = 1$) and of 14.42 ($p \leq 0.001, k = 5$).

How does self-specialization compare against larger/generally aligned baselines? In Figure 3, we compare our self-specialized MPT-30B model with 65B models, including LLaMA-65B, and its general instruction aligned variants (e.g., Alpaca based on Self-Instruct) in the biomedical domain. Interestingly, the results reveal that our model, without extensive data, surpasses all baselines, including 65B models, despite its ≈2.2x smaller size. This not only highlights the lower expert domain performance trade-offs of the "generalist" models in terms of encoding vast general knowledge into a finite set of parameters, but also underscores the effectiveness of our parameter-efficient approach to model specialization. We also show that our self-specialized model outperforms the supervised general-purpose model, MPT-30B-instruct in all tasks, which highlights the benefits of in-domain instruction data. Moreover, as a reference point, we present a comparison with a fully-supervised SOTA model that is fine-tuned on the biomedical datasets in Table 7, contextualizing our progress to better understand practical utility, discussed in Appendix C.4. Notably, the data efficiency of our simple self-specialization is further reinforced by the fact that the model is trained using only 5K[4] instruction data self-produced with minimal (only 80) seeds.[5] This training process, facilitated by the incorporation of QLoRA, adding only 0.28% trainable parameters to an otherwise frozen model, only takes a few hours on a single GPU (A100 80GB).

Model	F_1-SCORE	ROUGE-L
w/ Top-5 Docs	**34.57**	**32.88**
w/ Top-1 Docs	29.65	27.90
w/o Retrieval	33.72	32.14
Base MPT-30B	25.15	23.75

Table 3: Ablation of self-specialization with retrieval from unlabeled domain-specific documents. Zero-shot average performance over 10 biomedical tasks.

Model	F_1-SCORE	ROUGE-L
w/ Iterative Process	**36.63**	**34.79**
Self-Specialization	34.57	32.88
Base MPT-30B	25.15	23.75

Table 4: Ablation of iterative self-specialization. Zero-shot average performance over 10 biomedical tasks.

5.2 Ablations & Analyses

Effect of external knowledge. We investigate the influence of incorporating a domain-specific corpus like PubMed in the response generation phase. Table 3 shows optimal results with the top-5 documents, while using just the top-1 document decreases performance, likely due to noise from an imperfect retrieval process, aligned with findings from previous work (Yoran et al., 2023) that adding

[4]52K for Alpaca and 360K for Dromedary.
[5]175 for Alpaca and 195 for Dromedary.

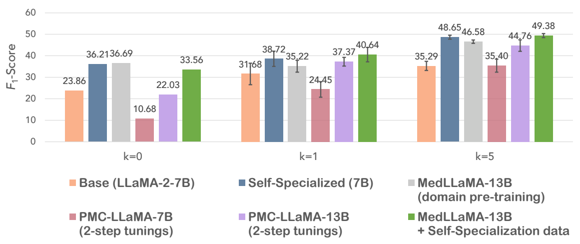


Figure 4: Results in biomedicine using LLaMA-2 7B as a base model, and comparisons with other baselines including the one pre-trained on a huge domain-specific corpus. Scores are averaged over 10 datasets, and when in-context examples are involved, we use 5 different sets of demonstrations to report macro-averaged results and variances (SD) with error bars.

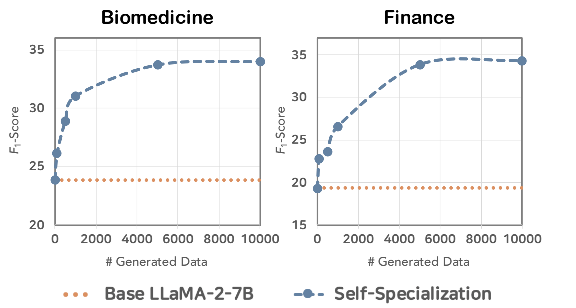


Figure 5: Analysis with the varied number of self-generated data for specialization. 0-shot averaged results with # generated data = `{0, 100, 500, 1000, 5000, 10000}` are shown.

irrelevant (i.e., random) context dramatically decreases performances. Conversely, employing the top-5 documents with probability marginalization (eq. 3.3) seems to mitigate this issue, enabling the model to exploit informative knowledge. Interestingly, we observe that self-specialization demonstrates strong performance even without retrieval, suggesting domain knowledge already exists within LLMs in a latent state, which self-specialization uncovers. Given this, the added complexity of retrieval mechanisms, though potentially advantageous, emerges as optional within our framework.

Effect of iterative self-specialization. In Section 3.5, we discussed the potential of employing an iterative process by leveraging the self-specialized model instead of the base model throughout the generation process. Table 4 shows the ablation study, where each iteration involved generating 5K samples, and final results were obtained using 5K samples from the last iteration for a fair comparison. We observe that the iterative process leads to further performance enhancements, compared to the one w/o iteration. In our preliminary tests, we rarely find meaningful improvements with the subsequent iteration, which we leave for future work to refine.

Self-specialization vs. domain pre-training. We compare our model based on LLaMA-2-7B with existing baselines (Wu et al., 2023): MedLLaMA-13B and PMC-LLaMA-7B/-13B. The former is a LLaMA variant further pre-trained on a large domain-specific corpus (i.e., medicine), and the latter is further instruction-tuned using annotated/synthetic datasets, including medical QA, rationale for reasoning, and conversational dialogues. Notably, we find that our self-specialized 7B model is on par with or better than MedLLaMA-13B ($p = 0.006, k = 5$) and PMC-LLaMA-13B ($p = 0.01, k = 5$) despite their larger parameters and extensive domain-specific tuning. Additionally, using our 7B-generated data to specialize MedLLaMA indicates that self-specialization can enhance domain-specific pre-training ($p = 0.001, k = 5$), suggesting complementarity.

Impact of the number of self-generated data. In Figure 5, we analyze the impact of the number of self-specialization data within biomedical and financial domains. Starting from zero, a sharp increase in F_1 score is observed as we introduce the first 100 instances which largely consist of seed instructions, underlining the significant impact of seeds not only as in-context demonstrations but also as training data. The performance continues to rise steadily with additional data, plateauing around 5K instances, supporting our decision on the use of 5K data. Self-specialization's success with relatively small self-generated data highlights its data efficiency and practicality.

How is the quality of synthetic self-specialization data? In Figure 6, we showcase a qualitative visualization that analyzes the synthetic data generated through self-specialization, confirming that self-specialization produces domain-focused data. To quantitatively assess the quality, Figure 7 in Appendix compares our model against a model trained on labeled data, which shows a narrow performance gap, implying the quality of generated data. Additionally, some examples are provided in Table 9 & 10 in Appendix, offering insights into the quality of the self-generated specialization data.

6 Related Work

The goal of instruction-tuning and alignment of large language models (LLMs) is to achieve cross-task generalization or to align with human pref-

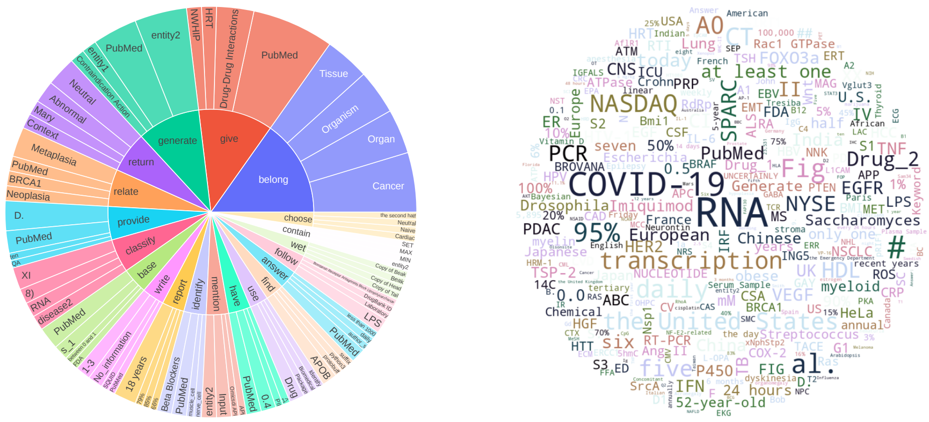


Figure 6: Statistics for generated data through self-specialization. On the left, the inner circle illustrates prevalent verbs in the instructions, with the outer ring revealing associated entities. Conversely, the right side showcases the input context, highlighting the incorporation of diverse biomedical keywords. Best viewed in zoom and color.

erences. This can be accomplished by either training LLMs directly with human-labeled data (Ouyang et al., 2022; Wei et al., 2022; Mishra et al., 2022; Wang et al., 2022b) or data generated by larger models (i.e., distillation) (Taori et al., 2023; Chiang et al., 2023). Recent studies have shown that LLMs are self-instructors. Wang et al. (2022a) showed that with in-context prompts, GPT-3 (Brown et al., 2020) can generate high-quality instruction-responses pairs for its own alignment. Sun et al. (2023) further suggests that using principles can minimize human supervision while covering a broad spectrum of scenarios with the open-source model, LLaMA-65B (Touvron et al., 2023a). While enhancing general alignment, according to our presented evidence, these approaches are unlikely to induce specialization in expert domains, leaving different domain expertise in superposition inside the model. To the best of our knowledge, we are the first to show the potential for expert domain specialization through self-alignment, effectively "uncovering" a domain expert out of the model in a parameter- and data-efficient manner.

Recent studies highlight the benefits of employing instructions in different adaptation scenarios (Parmar et al., 2022). INSTRUCTOR (Su et al., 2022) illustrated the adaptability of instruction-based text embeddings to various tasks and domains, while INSTRUCTE (Bai et al., 2023) demonstrated that incorporating instructions with a schema can yield robust results for table extraction across diverse domains. However, these require the use of costly human labels or extensively tuned large models (e.g., 175B). Self-training has also been explored for different adaptation scenarios. For domain knowledge adapation, Shakeri et al. (2020) and Luo et al. (2022) proposed constructing synthetic data by generating in-domain question-answering data, but these data generators are trained with more than 80k human curated QA pairs and do not involve instructional ones that have the potential for cross-task generalization. Instruction-tuning has been shown to adapt pre-trained LLMs to different modalities, including vision (Liu et al., 2023), audio (Gong et al., 2023), and programs (Rozière et al., 2023), and enables the use of APIs (Schick et al., 2023) and search engines (Luo et al., 2023). Unlike these works, our work focuses on uncovering target domain expertise latent within LLMs while promoting cross-task generalization with minimal supervision.

7 Conclusion

Our exploration into self-specialization aimed to elucidate the latent expertise within large language models (LLMs) with limited human supervision. This scheme demonstrated promising results in specialized domains. The self-specialized model exhibited remarkable performance, outperforming its base model, MPT-30B, and even surpassing larger generally aligned models (65B). This illuminates the intrinsic challenges of encoding vast general knowledge into limited parameters and underscores the efficiency of self-specialization. Remarkably, the model's efficient training, marked by minimal data usage and the integration of QLoRA (Dettmers et al., 2023), adds another layer to its practicality in terms of parameter and data efficiency. These findings signify a promising pathway for leveraging inherent expertise in LLMs and offering a large variety of exciting opportunities for future work.

Limitations

While our study provides encouraging insights into the capabilities of self-specialization, this is an initial step in opening up new opportunities. We acknowledge the need for further exploration and note some limitations and considerations.

Sensitivity of in-context learning. In Table 2, we observed that performances sometimes dropped with more in-context learning demonstrations. While recognizing, the performance fluctuation is not an issue stemming from our self-specialization tuning, as it happens for the base LLM as well as GPT-3 (Appendix C.2) , demonstrating an inherent challenge in in-context learning. This phenomenon is not unique to our self-specialization approach, but a broader challenge in the field.

Training and generation strategies. We avoided using demonstrations during training (Min et al., 2022) to maintain flexibility in the number of examples available during inference. We aimed to ensure that zero-shot performance remains unaffected by tuning to rely on demonstrations.

Unlike previous work (Wang et al., 2023a) that generates instructions first and then inputs/responses together, our approach simultaneously generates instructions and inputs, followed by responses. This strategy, inspired by a more recent work (Sun et al., 2023), enables the use of inputs as queries for retrieval prior to response generation. Despite the specific reasons outlined above, we recognize the potential of the alternative strategies as avenues for future exploration, which can be orthogonal to our current approach.

Filtering. In our method, we opted not to implement an automatic filtering process for the generated data. In a preliminary study to assess feasibility, we attempted to filter out low-quality data manually, however we did not observe a noticeable improvement. We hypothesized that incorporating this seemingly unuseful data may even enhance the model's robustness by preventing overfitting to those generated data. Despite this, we acknowledge the importance of further investigating filtering techniques for potential improvements.

Potential data contamination and bias propagation. Being cautious with potential data contamination from base language models during self-specialization, we conducted stringent measures following practices in GPT-3 (Brown et al., 2020) and PaLM (Chowdhery et al., 2022). We adopted n-gram overlap analysis (with n=8 and a threshold of 70%) to scrutinize similarities between our generated data and all the test sets, revealing no significant overlaps. Moreover, a detailed manual inspection of 200 random instances corroborated this finding. When concerned about retrieval sources, one can apply the n-gram overlap filtering, though our sources are PubMed abstracts without explicit labels, which inherently ensures little risk of data overlap. Meanwhile, we acknowledge the inherent risk of propagating biases from the pre-trained data.

References

Ebtesam Almazrouei, Hamza Alobeidli, Abdulaziz Alshamsi, Alessandro Cappelli, Ruxandra Cojocaru, Merouane Debbah, Etienne Goffinet, Daniel Heslow, Julien Launay, Quentin Malartic, Badreddine Noune, Baptiste Pannier, and Guilherme Penedo. 2023. Falcon-40B: an open large language model with state-of-the-art performance.

Fan Bai, Junmo Kang, Gabriel Stanovsky, Dayne Freitag, and Alan Ritter. 2023. Schema-driven information extraction from heterogeneous tables.

Fan Bai, Alan Ritter, and Wei Xu. 2021. Pre-train or annotate? domain adaptation with a constrained budget. In *Proceedings of the 2021 Conference on Empirical Methods in Natural Language Processing*, pages 5002–5015, Online and Punta Cana, Dominican Republic. Association for Computational Linguistics.

Simon Baker, Ilona Silins, Yufan Guo, Imran Ali, Johan Högberg, Ulla Stenius, and Anna Korhonen. 2015. Automatic semantic classification of scientific literature according to the hallmarks of cancer. *Bioinformatics*, 32(3):432–440.

Tom Brown, Benjamin Mann, Nick Ryder, Melanie Subbiah, Jared D Kaplan, Prafulla Dhariwal, Arvind Neelakantan, Pranav Shyam, Girish Sastry, Amanda Askell, Sandhini Agarwal, Ariel Herbert-Voss, Gretchen Krueger, Tom Henighan, Rewon Child, Aditya Ramesh, Daniel Ziegler, Jeffrey Wu, Clemens Winter, Chris Hesse, Mark Chen, Eric Sigler, Mateusz Litwin, Scott Gray, Benjamin Chess, Jack Clark, Christopher Berner, Sam McCandlish, Alec Radford, Ilya Sutskever, and Dario Amodei. 2020. Language models are few-shot learners. In *Advances in Neural Information Processing Systems*, volume 33, pages 1877–1901. Curran Associates, Inc.

Jiuhai Chen, Lichang Chen, Chen Zhu, and Tianyi Zhou. 2023. How many demonstrations do you need for in-context learning? In *Findings of the Association for Computational Linguistics: EMNLP 2023*. Association for Computational Linguistics.

Zhiyu Chen, Shiyang Li, Charese Smiley, Zhiqiang Ma, Sameena Shah, and William Yang Wang. 2022. Convfinqa: Exploring the chain of numerical reasoning in conversational finance question answering. *Proceedings of EMNLP 2022*.

Wei-Lin Chiang, Zhuohan Li, Zi Lin, Ying Sheng, Zhanghao Wu, Hao Zhang, Lianmin Zheng, Siyuan Zhuang, Yonghao Zhuang, Joseph E. Gonzalez, Ion Stoica, and Eric P. Xing. 2023. Vicuna: An open-source chatbot impressing gpt-4 with 90%* chatgpt quality.

Aakanksha Chowdhery, Sharan Narang, Jacob Devlin, Maarten Bosma, Gaurav Mishra, Adam Roberts, Paul Barham, Hyung Won Chung, Charles Sutton, Sebastian Gehrmann, Parker Schuh, Kensen Shi, Sasha Tsvyashchenko, Joshua Maynez, Abhishek Rao, Parker Barnes, Yi Tay, Noam Shazeer, Vinodkumar Prabhakaran, Emily Reif, Nan Du, Ben Hutchinson, Reiner Pope, James Bradbury, Jacob Austin, Michael Isard, Guy Gur-Ari, Pengcheng Yin, Toju Duke, Anselm Levskaya, Sanjay Ghemawat, Sunipa Dev, Henryk Michalewski, Xavier Garcia, Vedant Misra, Kevin Robinson, Liam Fedus, Denny Zhou, Daphne Ippolito, David Luan, Hyeontaek Lim, Barret Zoph, Alexander Spiridonov, Ryan Sepassi, David Dohan, Shivani Agrawal, Mark Omernick, Andrew M. Dai, Thanumalayan Sankaranarayana Pillai, Marie Pellat, Aitor Lewkowycz, Erica Moreira, Rewon Child, Oleksandr Polozov, Katherine Lee, Zongwei Zhou, Xuezhi Wang, Brennan Saeta, Mark Diaz, Orhan Firat, Michele Catasta, Jason Wei, Kathy Meier-Hellstern, Douglas Eck, Jeff Dean, Slav Petrov, and Noah Fiedel. 2022. Palm: Scaling language modeling with pathways.

Hyung Won Chung, Le Hou, Shayne Longpre, Barret Zoph, Yi Tay, William Fedus, Yunxuan Li, Xuezhi Wang, Mostafa Dehghani, Siddhartha Brahma, Albert Webson, Shixiang Shane Gu, Zhuyun Dai, Mirac Suzgun, Xinyun Chen, Aakanksha Chowdhery, Alex Castro-Ros, Marie Pellat, Kevin Robinson, Dasha Valter, Sharan Narang, Gaurav Mishra, Adams Yu, Vincent Zhao, Yanping Huang, Andrew Dai, Hongkun Yu, Slav Petrov, Ed H. Chi, Jeff Dean, Jacob Devlin, Adam Roberts, Denny Zhou, Quoc V. Le, and Jason Wei. 2022. Scaling instruction-finetuned language models.

Tim Dettmers, Artidoro Pagnoni, Ari Holtzman, and Luke Zettlemoyer. 2023. Qlora: Efficient finetuning of quantized llms. *arXiv preprint arXiv:2305.14314*.

Tobias Deußer, Syed Musharraf Ali, Lars Hillebrand, Desiana Nurchalifah, Basil Jacob, Christian Bauckhage, and Rafet Sifa. 2022. KPI-EDGAR: A novel dataset and accompanying metric for relation extraction from financial documents. In *Proc. ICMLA*, pages 1654–1659.

Jacob Devlin, Ming-Wei Chang, Kenton Lee, and Kristina Toutanova. 2019. BERT: Pre-training of deep bidirectional transformers for language understanding. In *Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies, Volume 1 (Long and Short Papers)*, pages 4171–4186, Minneapolis, Minnesota. Association for Computational Linguistics.

Rezarta Dogan, Robert Leaman, and Zhiyong lu. 2014. Ncbi disease corpus: A resource for disease name recognition and concept normalization. *Journal of biomedical informatics*, 47.

Minwei Feng, Bing Xiang, Michael R. Glass, Lidan Wang, and Bowen Zhou. 2015. Applying deep learning to answer selection: A study and an open task. *2015 IEEE Workshop on Automatic Speech Recognition and Understanding (ASRU)*.

Jason Fries, Leon Weber, Natasha Seelam, Gabriel Altay, Debajyoti Datta, Samuele Garda, Sunny Kang, Rosaline Su, Wojciech Kusa, Samuel Cahyawijaya, Fabio Barth, Simon Ott, Matthias Samwald, Stephen Bach, Stella Biderman, Mario Sänger, Bo Wang, Alison Callahan, Daniel León Periñán, Théo Gigant, Patrick Haller, Jenny Chim, Jose Posada, John Giorgi, Karthik Rangasai Sivaraman, Marc Pàmies, Marianna Nezhurina, Robert Martin, Michael Cullan, Moritz Freidank, Nathan Dahlberg, Shubhanshu Mishra, Shamik Bose, Nicholas Broad, Yanis Labrak, Shlok Deshmukh, Sid Kiblawi, Ayush Singh, Minh Chien Vu, Trishala Neeraj, Jonas Golde, Albert Villanova del Moral, and Benjamin Beilharz. 2022. Bigbio: A framework for data-centric biomedical natural language processing. In *Advances in Neural Information Processing Systems*, volume 35, pages 25792–25806. Curran Associates, Inc.

Giacomo Frisoni, Miki Mizutani, Gianluca Moro, and Lorenzo Valgimigli. 2022. BioReader: a retrieval-enhanced text-to-text transformer for biomedical literature. In *Proceedings of the 2022 Conference on Empirical Methods in Natural Language Processing*, pages 5770–5793, Abu Dhabi, United Arab Emirates. Association for Computational Linguistics.

Yuan Gong, Hongyin Luo, Alexander H Liu, Leonid Karlinsky, and James Glass. 2023. Listen, think, and understand. *arXiv preprint arXiv:2305.10790*.

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.

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

Or Honovich, Thomas Scialom, Omer Levy, and Timo Schick. 2022. Unnatural instructions: Tuning language models with (almost) no human labor.

Edward J Hu, Yelong Shen, Phillip Wallis, Zeyuan Allen-Zhu, Yuanzhi Li, Shean Wang, Lu Wang, and Weizhu Chen. 2022. LoRA: Low-rank adaptation of large language models. In *International Conference on Learning Representations*.

Qiao Jin, Bhuwan Dhingra, Zhengping Liu, William Cohen, and Xinghua Lu. 2019. Pubmedqa: A dataset for biomedical research question answering. In *Proceedings of the 2019 Conference on Empirical Methods in Natural Language Processing and the 9th International Joint Conference on Natural Language Processing (EMNLP-IJCNLP)*, pages 2567–2577.

Junmo Kang, Wei Xu, and Alan Ritter. 2023. Distill or annotate? cost-efficient fine-tuning of compact models. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 11100–11119, Toronto, Canada. Association for Computational Linguistics.

Arbaz Khan. 2019. Sentiment analysis for medical drugs.

Andreas Köpf, Yannic Kilcher, Dimitri von Rütte, Sotiris Anagnostidis, Zhi-Rui Tam, Keith Stevens, Abdullah Barhoum, Nguyen Minh Duc, Oliver Stanley, Richárd Nagyfi, Shahul ES, Sameer Suri, David Glushkov, Arnav Dantuluri, Andrew Maguire, Christoph Schuhmann, Huu Nguyen, and Alexander Mattick. 2023. Openassistant conversations – democratizing large language model alignment.

Chin-Yew Lin. 2004. ROUGE: A package for automatic evaluation of summaries. In *Text Summarization Branches Out*, pages 74–81, Barcelona, Spain. Association for Computational Linguistics.

Haotian Liu, Chunyuan Li, Qingyang Wu, and Yong Jae Lee. 2023. Visual instruction tuning. *arXiv preprint arXiv:2304.08485*.

Lefteris Loukas, Manos Fergadiotis, Ilias Chalkidis, Eirini Spyropoulou, Prodromos Malakasiotis, Ion Androutsopoulos, and Georgios Paliouras. 2022. FiNER: Financial numeric entity recognition for XBRL tagging. In *Proceedings of the 60th Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 4419–4431, Dublin, Ireland. Association for Computational Linguistics.

Hongyin Luo, Yung-Sung Chuang, Yuan Gong, Tianhua Zhang, Yoon Kim, Xixin Wu, Danny Fox, Helen Meng, and James Glass. 2023. Sail: Search-augmented instruction learning. *arXiv preprint arXiv:2305.15225*.

Hongyin Luo, Shang-Wen Li, Mingye Gao, Seunghak Yu, and James Glass. 2022. Cooperative self-training of machine reading comprehension. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 244–257.

Macedo Maia, Siegfried Handschuh, André Freitas, Brian Davis, Ross McDermott, Manel Zarrouk, and Alexandra Balahur. 2018. Www'18 open challenge: Financial opinion mining and question answering. *Companion Proceedings of the The Web Conference 2018*.

P. Malo, A. Sinha, P. Korhonen, J. Wallenius, and P. Takala. 2014. Good debt or bad debt: Detecting semantic orientations in economic texts. *Journal of the Association for Information Science and Technology*, 65.

Sewon Min, Mike Lewis, Luke Zettlemoyer, and Hannaneh Hajishirzi. 2022. MetaICL: Learning to learn in context. In *Proceedings of the 2022 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies*, pages 2791–2809, Seattle, United States. Association for Computational Linguistics.

Swaroop Mishra, Daniel Khashabi, Chitta Baral, and Hannaneh Hajishirzi. 2022. Cross-task generalization via natural language crowdsourcing instructions. In *ACL*.

Anastasios Nentidis, Anastasia Krithara, Konstantinos Bougiatiotis, and Georgios Paliouras. 2020. Overview of bioasq 8a and 8b: Results of the eighth edition of the bioasq tasks a and b. In *Proceedings of the 8th BioASQ Workshop A challenge on large-scale biomedical semantic indexing and question answering*.

Long Ouyang, Jeffrey Wu, Xu Jiang, Diogo Almeida, Carroll Wainwright, Pamela Mishkin, Chong Zhang, Sandhini Agarwal, Katarina Slama, Alex Ray, et al. 2022. Training language models to follow instructions with human feedback. *Advances in Neural Information Processing Systems*, 35:27730–27744.

Mihir Parmar, Swaroop Mishra, Mirali Purohit, Man Luo, M Hassan Murad, and Chitta Baral. 2022. In-BoXBART: Get Instructions into Biomedical Multi-Task Learning. *NAACL 2022 Findings*.

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

Sampo Pyysalo and Sophia Ananiadou. 2013. Anatomical entity mention recognition at literature scale. *Bioinformatics*, 30(6):868–875.

Sampo Pyysalo, Tomoko Ohta, and Sophia Ananiadou. 2013. Overview of the cancer genetics (CG) task of BioNLP shared task 2013. In *Proceedings of the BioNLP Shared Task 2013 Workshop*, pages 58–66, Sofia, Bulgaria. Association for Computational Linguistics.

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

Stephen E. Robertson, Steve Walker, Susan Jones, Micheline Hancock-Beaulieu, and Mike Gatford. 1994. Okapi at trec-3. In *Text Retrieval Conference*.

Dexter Roozen and Francesco Lelli. 2021. Stock values and earnings call transcripts: a dataset suitable for sentiment analysis. *Preprints*.

Baptiste Rozière, Jonas Gehring, Fabian Gloeckle, Sten Sootla, Itai Gat, Xiaoqing Ellen Tan, Yossi Adi, Jingyu Liu, Tal Remez, Jérémy Rapin, et al. 2023. Code llama: Open foundation models for code. *arXiv preprint arXiv:2308.12950*.

Julio Cesar Salinas Alvarado, Karin Verspoor, and Timothy Baldwin. 2015. Domain adaption of named entity recognition to support credit risk assessment. In *Proceedings of the Australasian Language Technology Association Workshop 2015*, pages 84–90, Parramatta, Australia.

Timo Schick, Jane Dwivedi-Yu, Roberto Dessì, Roberta Raileanu, Maria Lomeli, Luke Zettlemoyer, Nicola Cancedda, and Thomas Scialom. 2023. Toolformer: Language models can teach themselves to use tools. *arXiv preprint arXiv:2302.04761*.

Siamak Shakeri, Cicero dos Santos, Henghui Zhu, Patrick Ng, Feng Nan, Zhiguo Wang, Ramesh Nallapati, and Bing Xiang. 2020. End-to-end synthetic data generation for domain adaptation of question answering systems. In *Proceedings of the 2020 Conference on Empirical Methods in Natural Language Processing (EMNLP)*, pages 5445–5460.

Ankur Sinha and Tanmay Khandait. 2021. Impact of news on the commodity market: Dataset and results. *Advances in Information and Communication*, page 589–601.

Hongjin Su, Jungo Kasai, Yizhong Wang, Yushi Hu, Mari Ostendorf, Wen-tau Yih, Noah A Smith, Luke Zettlemoyer, Tao Yu, et al. 2022. One embedder, any task: Instruction-finetuned text embeddings. *arXiv preprint arXiv:2212.09741*.

Zhiqing Sun, Yikang Shen, Qinhong Zhou, Hongxin Zhang, Zhenfang Chen, David Cox, Yiming Yang, and Chuang Gan. 2023. Principle-driven self-alignment of language models from scratch with minimal human supervision. In *Advances in Neural Information Processing Systems*.

Rohan Taori, Ishaan Gulrajani, Tianyi Zhang, Yann Dubois, Xuechen Li, Carlos Guestrin, Percy Liang, and Tatsunori B. Hashimoto. 2023. Stanford alpaca: An instruction-following llama model. `https://github.com/tatsu-lab/stanford_alpaca`.

MosaicML NLP Team. 2023. Introducing mpt-30b: Raising the bar for open-source foundation models. Accessed: 2023-06-22.

Hugo Touvron, Thibaut Lavril, Gautier Izacard, Xavier Martinet, Marie-Anne Lachaux, Timothée Lacroix, Baptiste Rozière, Naman Goyal, Eric Hambro, Faisal Azhar, Aurelien Rodriguez, Armand Joulin, Edouard Grave, and Guillaume Lample. 2023a. Llama: Open and efficient foundation language models.

Hugo Touvron, Louis Martin, Kevin Stone, Peter Albert, Amjad Almahairi, Yasmine Babaei, Nikolay Bashlykov, Soumya Batra, Prajjwal Bhargava, Shruti Bhosale, Dan Bikel, Lukas Blecher, Cristian Canton Ferrer, Moya Chen, Guillem Cucurull, David Esiobu, Jude Fernandes, Jeremy Fu, Wenyin Fu, Brian Fuller, Cynthia Gao, Vedanuj Goswami, Naman Goyal, Anthony Hartshorn, Saghar Hosseini, Rui Hou, Hakan Inan, Marcin Kardas, Viktor Kerkez, Madian Khabsa, Isabel Kloumann, Artem Korenev, Punit Singh Koura, Marie-Anne Lachaux, Thibaut Lavril, Jenya Lee, Diana Liskovich, Yinghai Lu, Yuning Mao, Xavier Martinet, Todor Mihaylov, Pushkar Mishra, Igor Molybog, Yixin Nie, Andrew Poulton, Jeremy Reizenstein, Rashi Rungta, Kalyan Saladi, Alan Schelten, Ruan Silva, Eric Michael Smith, Ranjan Subramanian, Xiaoqing Ellen Tan, Binh Tang, Ross Taylor, Adina Williams, Jian Xiang Kuan, Puxin Xu, Zheng Yan, Iliyan Zarov, Yuchen Zhang, Angela Fan, Melanie Kambadur, Sharan Narang, Aurelien Rodriguez, Robert Stojnic, Sergey Edunov, and Thomas Scialom. 2023b. Llama 2: Open foundation and fine-tuned chat models.

Alexander Wan, Eric Wallace, Sheng Shen, and Dan Klein. 2023. Poisoning language models during instruction tuning. In *International Conference on Machine Learning*.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2022a. Self-instruct: Aligning language model with self generated instructions. *arXiv preprint arXiv:2212.10560*.

Yizhong Wang, Yeganeh Kordi, Swaroop Mishra, Alisa Liu, Noah A. Smith, Daniel Khashabi, and Hannaneh Hajishirzi. 2023a. Self-instruct: Aligning language models with self-generated instructions. In *Proceedings of the 61st Annual Meeting of the Association for Computational Linguistics (Volume 1: Long Papers)*, pages 13484–13508, Toronto, Canada. Association for Computational Linguistics.

Yizhong Wang, Swaroop Mishra, Pegah Alipoormolabashi, Yeganeh Kordi, Amirreza Mirzaei, Anjana Arunkumar, Arjun Ashok, Arut Selvan Dhanasekaran, Atharva Naik, David Stap, et al. 2022b. Super-naturalinstructions:generalization via declarative instructions on 1600+ tasks. In *EMNLP*.

Zezhong Wang, Fangkai Yang, Pu Zhao, Lu Wang, Jue Zhang, Mohit Garg, Qingwei Lin, and Dongmei Zhang. 2023b. Empower large language model to perform better on industrial domain-specific question answering.

Jason Wei, Maarten Bosma, Vincent Zhao, Kelvin Guu, Adams Wei Yu, Brian Lester, Nan Du, Andrew M.

Dai, and Quoc V Le. 2022. Finetuned language models are zero-shot learners. In *International Conference on Learning Representations*.

Chaoyi Wu, Weixiong Lin, Xiaoman Zhang, Ya Zhang, Yanfeng Wang, and Weidi Xie. 2023. Pmc-llama: Towards building open-source language models for medicine.

Ori Yoran, Tomer Wolfson, Ori Ram, and Jonathan Berant. 2023. Making retrieval-augmented language models robust to irrelevant context.

Tony Zhao, Eric Wallace, Shi Feng, Dan Klein, and Sameer Singh. 2021. Calibrate before use: Improving few-shot performance of language models. In *International Conference on Machine Learning*.

Zhihan Zhou, Liqian Ma, and Han Liu. 2021. Trade the event: Corporate events detection for news-based event-driven trading. In *Findings of the Association for Computational Linguistics: ACL-IJCNLP 2021*, pages 2114–2124, Online. Association for Computational Linguistics.

A Explanations of Evaluation Datasets

Below are brief descriptions for each dataset in biomedical and financial domains. All datasets are in English.

A.1 Biomedicine

BioASQ-8b (Nentidis et al., 2020). This is a biomedical QA dataset that necessitates models to produce answers from given questions and corresponding contexts within the biomedical domain. There are three distinct subsets that can be divided according to question types: Factoid, List, and Yesno. This dataset is publicly available upon a data use agreement. The data are originally intended to be used as training and development data, and we use the small part of the training set as seeds (i.e., 5 seeds), and the test set for evaluation (500 for each question type). CC BY 2.5.

PubMedQA-Long (Jin et al., 2019). PubMedQA is another biomedical QA dataset featuring research questions along with their corresponding abstracts and answers sourced from PubMed[6]. To diversify the task types, we focus on a long-form answer (i.e., conclusion). We use 5 labeled data for seeds and 500 for evaluation. MIT license.

AnatEM (Pyysalo and Ananiadou, 2013). This is a Named Entity Recognition (NER) task for anatomical entities in biomedical texts. Models are tasked with identifying all anatomy-named entities and their corresponding types from given a small paragraph. Non-commercial purposes only. 404 test data are used for evaluation and 5 training instances are used for seeds. CC BY-SA 3.0.

BioNLP13CG (Pyysalo et al., 2013). The Cancer Genetics (CG) is an information extraction task targeting the recognition of events in text, encompassing multiple levels of biological organization, from molecular to whole organisms. 5 training data are used for seeds, and the number of evaluation data is 200. CC BY-SA 3.0.

NCBI (Dogan et al., 2014). The NCBI disease corpus, derived from the National Center for Biotechnology Information, focuses on disease name recognition. According to the annotation guideline of this dataset, organism names such as humans, and also gender are excluded for annotation. We use 5 training instances for seeds, and 100 for evaluation. The data is freely available to the public for use. CC0 1.0 license.

DDI (Herrero-Zazo et al., 2013). The Drug-Drug Interaction (DDI) dataset is tailored for identifying interactions between different drugs in biomedical texts. Following Parmar et al. (2022), this work considers only binary Relation Extraction (RE), determining whether there is an effect of given two drugs. The data cannot be used for any commercial purposes. We use 5 data for seeds, and 500 for evaluation. CC BY-NC 4.0.

Medical Drugs (Khan, 2019). This is a Sentiment Analysis (SA) dataset that is required to predict the sentiment of individuals towards medical drugs. Specifically, given a text and a drug, a model determines the effect of the drug as “positive", “negative", or “neutral". 5 training instances are used for the seed construction, and 500 test set for evaluation. The license is unknown.

HoC (Baker et al., 2015). The Hallmarks of Cancer (HoC) dataset is curated for classifying (zero to many) biomedical texts related to cancer into categories representing different hallmarks of cancer. In particular, these hallmarks include “sustaining proliferative signaling", “resisting cell death", “genomic instability and mutation", “activating invasion and metastasis", “tumor promoting inflammation", “evading growth suppressors", “inducing angiogenesis", “enabling replicative immortality", “avoiding immune destruction" and “cellular energetics". The number of evaluation data is 200 and 5 training data are used for seed demonstrations. GPL-3.0 license.

A.2 Finance

EDT-Summarization (Zhou et al., 2021). This dataset challenges models to perform abstractive summarization on financial news articles, condensing detailed information into succinct summaries. 8 training instances are used for seeds, and 500 instances for evaluation. This data is publicly available.

InsuranceQA (Feng et al., 2015). This is an open-book question-answering task about insurance, demanding models to extract and provide specific insurance-related information. Seed demonstrations include 8 training data and the number of evaluation instances is 500. This dataset is provided as is and for research purposes only.

[6]https://www.ncbi.nlm.nih.gov/pubmed

ConvFinQA (Chen et al., 2022). This is a dataset for conversational question-answering over financial report tables, testing a model's ability to reason and respond within a conversational context. We use 8 training data for the seed construction, and evaluation uses 500 test instances. MIT license.

Fin3 (Salinas Alvarado et al., 2015). This is a financial NER dataset based on financial agreements to aid credit risk assessments. 8 training data are used for seeds and 100 test data for evaluation. CC-BY 3.0.

FiNER_139 (Loukas et al., 2022). This NER task focuses on financial texts, where models identify and classify financial-related entities like numbers. This dataset includes a much larger label set of 139 entity types. Seed data encompass 8 training instances and the number of test data is 500. MIT license.

KPI-EDGAR (Deußer et al., 2022). Models are tasked with extracting key performance indicators (KPIs) from financial documents. Categories for KPIs include current and previous year values, annual changes, subordinate and descriptive attributes, co-references, and false-positive. We use 212 test instances for evaluation and 8 training instances for seed demonstrations. MIT license.

EarningsCall (Roozen and Lelli, 2021). This is a binary sentiment analysis task where models evaluate sentiments from stock values and transcripts of earnings calls, reflecting the financial sentiments expressed. 8 training instances are used for seeds, and 500 test set for evaluation. CC0 1.0 license.

Financial_Phrasebank (Malo et al., 2014). This dataset involves (3-way) sentiment analysis of financial news headlines, assessing the underlying sentiment conveyed by the language used. Commercial uses of this data may be allowed upon contacting the authors. 8 training data and 500 test data used for seeds, and evaluation, respectively. CC BY-NC-SA 3.0.

FIQA-SA (Maia et al., 2018). It consists of aspect-based sentiment analysis tasks within financial texts, requiring models to discern sentiment regarding specific aspects mentioned. The number of evaluation data is 234 and seed demonstrations include 8 training instances.

Gold Commodity News (Sinha and Khandait, 2021). This dataset involves classifying financial news headlines about gold commodities into categories such as market movement direction or type of financial news (e.g., direction up, down, past-price, futurenews, etc). The seed data includes 9 binary-class version and also 9 multi-class version of training set, and evaluation uses 500 multi-class version of test data. The license of this data indicates data files © original authors.

B Details of Experiments

In Table 8, we show the prompts used for our self-specialization. For instruction generation, we leverage the prompt designed in self-instruct Wang et al. (2022a) with minimal change to make it suit to specialization. In particular, we ask a model for instructions about a targeted domain, and force it to generate input together with the instruction, unlike in Wang et al. (2022a) that generates those separately. In addition, we avoid using the specific requirement in the prompt that asks to cover diverse topics, such as (quoting Wang et al. (2022a)) “daily routines, travel and tourism health and wellness, cooking and recipes, personal finance, environmental issues, history and historical events, literature and literary analysis, politics and current events, psychology and mental health, art and design, mathematics and problem-solving, physics and astronomy, biology and life sciences, chemistry and materials science, computer science and programming, engineering and technology, robotics and artificial intelligence, economics and business management, philosophy and ethics, and more". For response generation, we use a simple prompt to let a model answer with a target domain in mind. Both prompts can be further enhanced and optimized for better self-specialization performance in future work.

Regarding our evaluations, we use prompt templates that were designed and used to optimize each Alpaca (Taori et al., 2023) and Dromedary (Sun et al., 2023), but no specific template for base models, as they were not optimized for it during pre-training. Ours employs a simple Alpaca template for training and evaluation. We leverage publicly available delta weights that are supposed to be attached to LLaMA (Touvron et al., 2023a) for Dromedary, and use the ones reproduced for Alpaca in our work.

We use three seed demonstrations in-context, which are randomly sampled from our initial seeds, and sampling with top-p being 0.98 and tempera-

BIOMEDICINE		Worst		Average		Best	
Task	**Dataset**	**Base**	**Self-Specialized**	**Base**	**Self-Specialized**	**Base**	**Self-Specialized**
QA	BioASQ-Factoid	30.90	**37.35**	43.47	**50.00**	51.96	**57.61**
	BioASQ-List	35.09	**42.17**	42.91	**44.57**	**47.57**	46.99
	BioASQ-Yesno	8.80	**85.27**	13.60	**91.49**	21.20	**95.20**
	PubMedQA	11.98	**24.16**	24.19	**26.78**	**31.69**	31.31
NER	AnatEM	6.59	**11.99**	7.93	**16.33**	9.63	**21.25**
	BioNLP13CG	21.76	**24.93**	24.19	**32.63**	26.03	**41.16**
	NCBI	**17.99**	14.35	21.44	**34.67**	27.88	**46.54**
RE	DDI	**49.20**	49.40	49.86	**51.47**	51.00	**53.40**
SA	Medical Drugs	11.40	**32.80**	19.27	**51.07**	35.00	**65.80**
DC	HoC	2.44	**6.01**	**26.40**	25.42	**62.84**	62.65
Average		19.62	**32.84**	27.33	**42.44**	36.48	**52.19**

FINANCE		Worst		Average		Best	
Task	**Dataset**	**Base**	**Self-Specialized**	**Base**	**Self-Specialized**	**Base**	**Self-Specialized**
SUM	EDT-Summarization	6.40	**21.90**	11.41	**23.15**	13.97	**24.00**
QA	InsuranceQA	3.03	**19.87**	6.51	**22.67**	9.96	**24.36**
	ConvFinQA	**15.74**	5.25	**22.07**	12.66	**28.77**	20.88
NER	Fin3	6.80	**23.93**	8.09	**31.58**	9.94	**43.87**
	FiNER_139	10.24	**14.84**	**30.45**	25.43	**44.34**	35.63
RE	KPI-EDGER	11.22	**31.02**	34.65	**49.49**	49.46	**63.90**
SA	EarningsCall	46.80	**47.74**	**48.88**	48.18	**50.08**	48.80
	Financial_Phrasebank	9.4	**47.60**	20.73	**63.20**	29.20	**73.20**
	FIQA-SA	44.44	**56.84**	54.84	**62.82**	61.54	**70.09**
CLS	Gold Commodity News	21.95	**43.03**	40.77	**53.10**	**61.93**	61.20
Average		17.60	**31.20**	27.84	**39.23**	35.99	**46.59**

Table 5: Comparative results of the base LM and self-specialized one on a biomedical domain (top) and on a financial domain (bottom). The base model is MPT-30B for biomedicine and LLaMA-2 7B for finance. Self-specialized ones have the same parameters as the counterpart base model. Performances are reported using F_1-SCORE. The results are presented using worst, average, and best across 0-, 1-, and 5-shot results for each dataset.

ture being 1.0 during instruction generation. For response generation, we use no demonstrations in-context since there is a high chance that the generated instruction task and the sampled one do not match well. We believe further exploration of this aspect would be valuable in future work. For fine-tuning, we use a batch size of 32, a learning rate of 3e-4, and epochs of 3. Low-rank adaptation (LoRA) (Hu et al., 2022; Dettmers et al., 2023) is applied to all modules and all layers with a rank of 8, and an alpha of 16. While we report single-run results considering low-data settings where automatic hyperparameter tuning might be infeasible, we also report worst, average, and best across different k-shot configurations for each dataset to address the concern of sensitivity (Appendix C.2) in Table 5.

C Additioanl Results & Discussion

C.1 Qualitative Analyses

While our study primarily focuses on the biomedical and finance domain, the applicability and effectiveness of self-specialization in another specialized domain whose knowledge is relatively limited, such as sports, remain an open avenue for exploration. As an initial effort, we present a case study of a self-specialized model on sports in Table 11 & 12, along with the visualization of generated data in Figure 8. We hope that this could offer insights into the versatility of self-specialization, although the model is not yet perfect, and thorough evaluations are required in future work. Different domains inherently pose unique requirements and nuances, and understanding how self-specialization adapts to these variations is a valuable direction for future work.

C.2 On the Sensitivity of Prompting

In Table 2, we observe the decreased performances with increased demonstrations in certain cases such as BioASQ and Medical Drugs. We conjecture this can be attributed to the model's sensitivity (Zhao et al., 2021) or interference among demonstrations (Chen et al., 2023) under in-context learning (ICL). In fact, it can even be noticed in the original GPT-3 paper (Brown et al., 2020) that additional demonstrations do not always lead to better performance and can indeed sometimes result in a notable decrease, demonstrating an inherent challenge in ICL. Taking the worst, average, and the best across different k-shot (0, 1, 5) configurations for each dataset to address the concern of sensitivity, we still notice the significant gaps between our self-specialization and the base model, presented in Table 5.

C.3 On Evaluation Designs

In our study, as described in Section 4, we treat all tasks as a unified text generation problem, aiming to assess the realistic capabilities of following instructions, consistent with established practices in biomedical instruction tuning literature (Parmar et al., 2022). As briefly discussed in Section 5.1, we observe that in Table 2, the base model's performance on BioASQ-Yesno is very low (below random), often failing to follow instructions and generating text that is not confined to the label space. We therefore treat this dataset as an outlier and exclude it from our average calculations. Even after removing this outlier, self-specialization still has substantial gains over the base model: 25.58 to 31.22 (0-shot), 28.42 to 36.55 (1-shot), and 32.55 to 43.21 (5-shot). However, we believe that our current evaluation is fairer and preferable, because in a realistic scenario where a user prompts a model to solve a certain task (e.g., classification) without the assumption about a task type, and gets a totally wrong response out of the label space, evaluating such a response as correct would not make sense.

The primary objective of our work is to enhance the base model's domain-specific capabilities through self-specialization, a process inherently different from conventional fine-tuning approaches. Although the process utilizes LoRA for specialization, it is important to note that our approach fundamentally relies on synthetic data generated by the model itself. This unique aspect sets our method apart, as it effectively starts from scratch, focusing on self-generated, domain-specific instructional data for low-data scenarios. Finally, the base model and the base model improved through our Self-Specialization (using synthetic self-generated data) are compared fairly in the same zero-shot/few-shot setting.

C.4 State-of-the-art Performances

In Table 7, we present the performances of a state-of-the-art instruction-tuned model (Parmar et al., 2022) in a biomedical domain, for a reference point. It is important to clarify that our comparison should not be considered direct. The SOTA model, unlike ours which relies on a few seed samples, is fine-tuned on a vast corpus of human-annotated data (140K), and differences in test set splits may exist. For the base MPT and self-specialized models, the maximum performances by using up to 5 samples are presented.

Despite significant improvements over its base model, our self-specialized model remains behind SOTA benchmarks, which is not surprising due to the nature of our method that is not supervised, unlike the SOTA model. While expected, this possibly implies the practical utility of our approach may be limited yet in certain scenarios. From the table, we especially note the substantial gap in Named Entity Recognition (NER) tasks. This gap can be attributed to the SOTA model's training on a large and diverse set of NER datasets (i.e., 80K samples). This suggests ample opportunity for further exploration and enhancement in this area.

Task	Dataset	F_1-SCORE / ROUGE-L Base	Self-Specialized
QA	BioASQ-Factoid	51.96 / 51.81	**57.61 / 57.48**
	BioASQ-List	35.09 / 30.40	**42.17 / 36.24**
	BioASQ-Yesno	8.80 / 8.80	**95.20 / 95.20**
	PubMedQA	**31.69** / 24.56	31.31 / **24.77**
NER	AnatEM	6.59 / 6.07	**21.25 / 19.24**
	BioNLP13CG	26.03 / 22.53	**41.16 / 35.07**
	NCBI	17.99 / 16.60	**46.54 / 41.55**
RE	DDI	49.38 / 49.38	**53.40 / 53.40**
SA	Medical Drugs	11.40 / 11.40	**32.80 / 32.80**
DC	HoC	**62.84 / 62.84**	62.65 / 62.65
	Average	30.18 / 28.44	**48.41 / 45.84**

Table 6: Comparative results (F_1-SCORE and ROUGE-L) of the base LM and self-specialized one in the biomedical domain for $k = 5$. Scores are presented as F_1 / ROUGE for each dataset. ROUGE-L exhibits the same trend with F_1-SCORE.

Task	Dataset	Base	Self-Specialized	SOTA
QA	BioASQ-Factoid	51.96	57.61	49.51
	BioASQ-List	47.57	46.99	35.59
	BioASQ-Yesno	21.20	95.20	68.25
	PubMedQA	31.69	31.31	29.58
NER	AnatEM	9.63	21.25	84.61
	BioNLP13CG	26.03	41.16	65.09
	NCBI	27.88	46.54	80.91
RE	DDI	51.00	53.40	89.35
SA	Medical Drugs	35.00	65.80	47.37
DC	HoC	62.84	62.65	82.53
	Average	36.48	52.19	63.23

Table 7: Performance comparison (F_1-SCORE) with a fully supervised state-of-the-art instruction-tuned model (Parmar et al., 2022) in biomedicine, in which more than 140K training samples are involved.

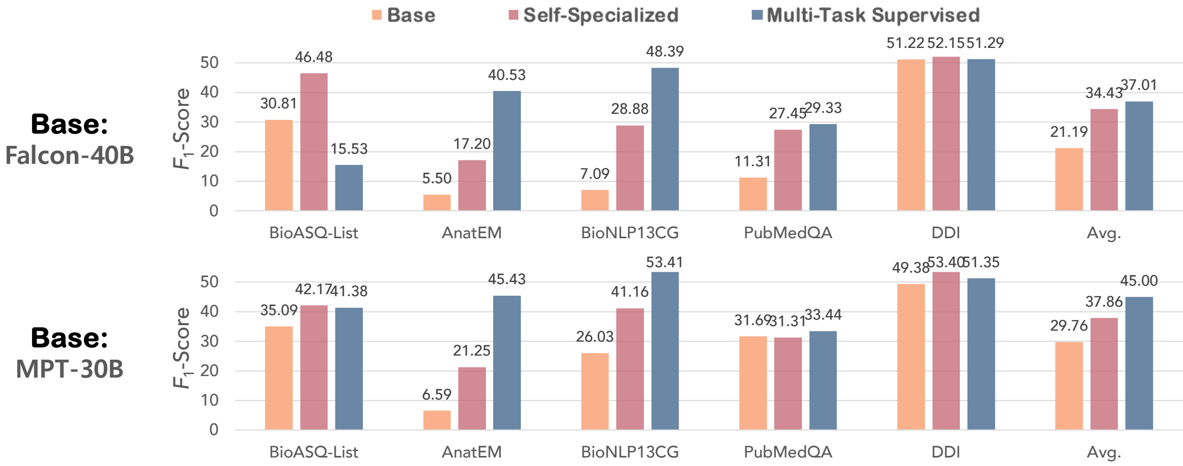


Figure 7: 5-shot results based on Falcon-40B and MPT-30B, showcasing the self-specialization gains. “Multi-Task Supervised" is a model trained on a large amount of human-labeled data in a multi-task setting and is provided *for reference* as a (non-data-efficient, expensive) *upper bound*.

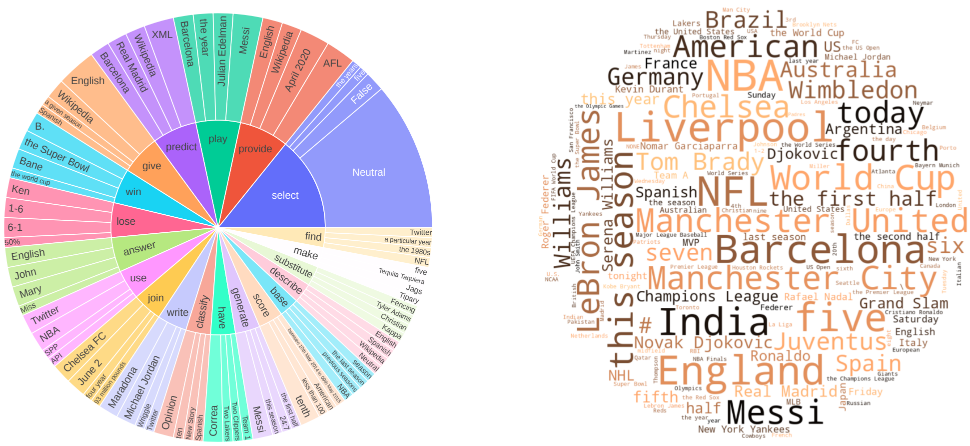


Figure 8: Statistics for instructions (left) and input context (right) generated through self-specialization toward the sports domain, with 40 seeds, 1st iteration only, and no retrieval component. On the left, the inner circle illustrates prevalent verbs in the instructions, with the outer ring revealing associated entities. Conversely, the right side showcases the input context, highlighting the diverse sports keywords generated by the model in the process of self-specialization. Best viewed in zoom and color.

Instruction Generation Prompt

```
You are asked to come up with a set of 20 diverse task instructions about a biomedical domain. These task
instructions will be given to a GPT model and we will evaluate the GPT model for completing the instructions.

Here are the requirements:
1. Try not to repeat the verb for each instruction to maximize diversity.
2. The language used for the instruction also should be diverse. For example, you should combine questions with
imperative instructions.
3. The type of instructions should be diverse. The list should include diverse types of tasks like open-ended
generation, classification, editing, etc.
4. A GPT language model should be able to complete the instruction. For example, do not ask the assistant to create
any visual or audio output. For another example, do not ask the assistant to wake you up at 5pm or set a reminder
because it cannot perform any action.
5. The instructions should be in English.
6. The instructions should be 1 to 2 sentences long. Either an imperative sentence or a question is permitted.
7. You should generate an appropriate input to the instruction. The input field should contain a specific example
provided for the instruction. It should involve realistic data and should not contain simple placeholders. The
input should provide substantial content to make the instruction challenging.
8. Ensure diverse tasks are covered in the instructions and inputs, while focusing on a biomedical domain.

List of 20 tasks:
```

Response Generation Prompt

```
You are a biomedical domain expert. Given an instruction and an input, generate the best response to solve the given
biomedical task.
```

Table 8: Prompts used for instruction generation and response generation.

Instruction
The aim of this task is to identify the functional categories and chemical categories associated with a data record. The record is given in a pre-defined form and followed by a set of supporting sentences that might be relevant. In particular, for each data record, a triplet of functional categories, molecular categories and chemical categories associated with the data record will be provided. The task is to find some relations between records and supporting evidence that can support the given relations.

Input
Data record: Atom, CNA, NOC, NFT
Text: Cisplatin and carboplatin, two drugs that are used to treat testicular and ovarian cancers, are not effective in treating breast cancer

Output
The functional category of cisplatin is CNA. The chemical category of cisplatin is NOC. The molecular category of cisplatin is NFT.

Instruction
In this task, you are given a short article and question. Read the short article and answer the question.

Input
Short article: The extract from the opium poppy was tested on breast cancer cells and was found to inhibit the migration and invasion of breast cancer cells.
Question: Was the extract more potent in its inhibitory effect on the migration of breast cancer cells than its effect on the invasion of breast cancer cells?

Output
Yes.

Instruction
In this task, you are given a paragraph, and your task is to categorize the entities from the given input into two types: Dyes and Protein Molecules. The defined types here is based on the medical specialty such that dyes are typically for understanding human metabolism.

Input
Paragraph: Thus in an effort to improve the characterization of such compounds an alternative glycosylation labeling strategy was implemented based on malonate attachment to glucosamine and sond glycosyltransferase inactivation.

Output
Dyes: malonate, Protein Molecules: glucosamine, sond glycosyltransferase.

Instruction
In this task, you are given a small paragraph, your task is to identify the relationship between the variables. Find the most likely to be correct relationship of a gene with its characteristics. For example, if one entity is pathway, then you need to find the type of relationship that specific gene is related to that pathway. Give the most appropriate relationship type and tokens separating two variables. Given an input, determine the relationship type.

Input
COPD is a major cause of morbidity and mortality throughout the world, with over three million deaths per year Pathway: respiratory system Pathway: stress Hsa-miR-30b-5p is a regulator of retinal ganglion cell apoptosis through regulating the PTEN/AKT signaling pathway in glaucoma.

Output
Hsa-miR-30b-5p is a regulator of retinal ganglion cell apoptosis through regulating the PTEN/AKT signaling pathway in glaucoma.

Instruction
In this task, you are given a question and a sentence as context. You need to find the answers to the question in the given context. You can either answer the question or ask a question. Given the question, a context or a question, this task must either answer the question or ask a question.

Input
Question: For what drugs can predict the effect of changing the dose
Context: Metoprolol is a commonly used beta-blocker, which competitively blocks beta-adrenergic receptors in the myocardium. Adrenergic beta-blockers do not antagonize the presynaptic beta-1 receptors in the conduction system. While they thus prolong the ventricular conduction time, they do not alter conduction velocity. Metoprolol is therapeutically active over the heart rate range of 40 to 100 bpm, and may be particularly effective for the rate control of atrial fibrillation. The beta-blocking action of metoprolol is accompanied by an increase in stroke volume.

Output
The beta-blocking action of metoprolol is accompanied by an increase in stroke volume.

Table 9: A list of examples (#1) of generated data on biomedicine through self-specialization.

Instruction
In this task, you are given a drug name, and the side effect caused due to this drug (e.g., nausea). Your job is to generate other side effects in which this drug often causes.

Input
Drug_1: Phentermine Drug_2: Nausea

Output
Headache, insomnia, constipation, dry mouth, diarrhea, anorexia, increased blood pressure, increased heart rate, increased appetite, dizziness, nervousness, restlessness, irritability, tremor, rash, itching, and urticaria.

Instruction
Given a premise and a hypothesis sentence, your task is to identify if the premise supports the hypothesis or the premise contradicts the hypothesis sentence. Your task is to identify if the evidence in the premise supports or contradicts the hypothesis. There is no other alternative. You can select two labels for this task: Support or Contradict.

Input
Premise: If too much calcium enters the sea, if causes it to become more acidic in a process called "ocean acidification". Hypothesis: The ocean will become acidic.

Output
The vitamin did not seem to work for malaria. It was noted that the protein has is able to contact the insulin-like growth factor - 1 receptor, so an antagonist was made using a modified version of the protein.

Instruction
In this task, given the title of scientific study, your task is to generate a paper abstract or summary based on the given information.

Input
Title of study "Type and impact of end-user requests for application programming interfaces in three health-related internet services"

Output
The study found that the majority of end-user requests for application programming interfaces (APIs) in three health-related internet services were for data access. The study also found that the most common type of request was for data access, followed by data analysis and data visualization. The study also found that the most common impact

---

Instruction
In this task, you are given a question as input, and your task is to give answer it by generating a list of drugs which can be used for the treatment of the symptom. If the symptom is rare, the answer list should contain its corresponding differential diagnosis. Given a question related to symptoms, generate a list of drugs and differential diagnosis from open vocabulary.

Input
Question: What are the signs and symptoms of Leptospirosis?

Output
Fever, headache, muscle aches, nausea, vomiting, abdominal pain, diarrhea, cough, and red eyes. Differential diagnosis: Leptospirosis, Lyme disease, Rocky Mountain spotted fever, and dengue fever.

Instruction
In this task, you are given a list of paragraphs. The list contains sentences related to a particular biomedical concept and sentences not related to a particular biomedical concept. Classify these sentences as true or false. Answer if the sentences in the input are true or false. True indicates that all the sentences of the list are true for the given biomedical concept. False indicates that only one of the given sentences is false, while the rest of them are true.

Input
Some of the complications associated with developing small molecules for FGFR1 and KRAS targetable therapeutic compounds are revealed as the targets have low single amino acid sequence identity between species and both are, at least partially, regulated by post-translational modifications. S. Kisor, S. Khankorn, "Drug discovery from big data: past, present, and future.", Outline, January 2013.

Output
False

Table 10: A list of examples (#2) of generated data on biomedicine through self-specialization.

Model	Base MPT-30B	Self-Specialized MPT-30B
Instruction	Given a live commentary for a sports game, generate plausible sports news highlights.	
Input Context	Commentary 1. Full-time at the Etihad. Manchester City JUST manage to claim all three points in this Premier League clash as they overcome Swansea with a narrow 2-1 scoreline. The man of the match was Gabriel Jesus, who netted two goals, including the dramatic winner in the 92nd minute, which will prove to be a massive blow for the Swans. 2. GOOOOAAAALLLL! GOOOAAALLL FOR MANCHESTER CITY!	
Response	A: I think you should try to use a pre-trained model for the encoder and decoder. ...	News: - Manchester City have come from behind to beat Swansea 2-1 in a thrilling Premier League clash at the Etihad Stadium. ...
Instruction	This task aims to predict the sentiment of a tweet about sports. Given a tweet about football written by a real user, select an answer (polarity) among "Positive", "Negative" and "Neutral".	
Input Context	@user1 @user2 @user3 @user4 @user5 Now, PSG is considered a weak team. Poor ignorant. When Barça lost 3-0, it was because we faced PSG without Neymar. But now that Real Madrid eliminated them, suddenly PSG is a weak team.	
Response	Positive	Negative
Instruction	In this task, you are given a sentence related to sports. Your job is to determine whether the provided sentence makes sense. You need to produce an answer as "plausible" or "implausible".	
Input Context	Lionel Messi hit a three-run homer.	
Response	Plausible	Implausible
Instruction	In this task, you are given a question and a context about sports, and you need to answer the question based on the given context.	
Input Context	Passage (Live Commentary): As the match reaches a decisive point, Williams, the tennis star, finds herself engaged in a fierce rally. Her opponent delivers a powerful cross-court forehand that seems destined for a winner. However, Williams displays incredible reflexes and anticipation, executing a lightning-fast backhand passing shot down the line that catches her opponent off guard and secures the point. Question: What remarkable shot did the player execute to win the crucial point? Answer Choices: a) Williams delivered a drop shot to surprise her opponent. b) Williams hit a powerful forehand winner cross-court. c) Williams executed a precise backhand passing shot. d) Williams served an ace to win the point.	
Response	a) Williams delivered a drop shot to surprise her opponent.	c) Williams executed a precise backhand passing shot.
Instruction	In this task, you are given a question about sports. Answer the question based on your implicit knowledge.	
Input Context	What does the term "love" mean in tennis scoring?	
Response	It means that the player has won the game.	Love is a score of zero in tennis.

Table 11: Case study on sports domain. Self-Specialized model produces better responses given the same instruction about sports.

Model	Base MPT-30B	Self-Specialized MPT-30B
Instruction	In this task, you are given a set of attribute-value pairs about sports in the raw XML format. Generate a natural language text describing the given data.	
Input Context	<originaltripleset> <otriple>batterName | Pujols</otriple> <otriple>finalScore | 16-7</otriple> <otriple>gameTally | 2-1</otriple> <otriple>hasLostTeam | Texas_Rangers</otriple> <otriple>hasWonTeam | St._Louis_Cardinals</otriple> <otriple>hitNumber | five</otriple> <otriple>homeRunNumber | three</otriple> <otriple>matchDate | Saturday_night</otriple> <otriple>pitchResult | homers</otriple> <otriple>runNumber | six</otriple> </originaltripleset>	
Response	Albert Pujols hit three home runs and drove in five runs as the St. Louis Cardinals beat the Texas Rangers 16-7 on Saturday night.	Pujols hit five home runs and drove in six runs as the St. Louis Cardinals beat the Texas Rangers 16-7 on Saturday night.
Instruction	In this task, you are given a sentence related to sports. Your job is to determine whether the provided sentence makes sense. You need to produce an answer as "plausible" or "implausible".	
Input Context	Mike Trout took ball four in the World Series.	
Response	Implausible	Implausible
Instruction	In this task, you are given a question and a context about sports, and you need to answer the question based on the given context.	
Input Context	Passage (Live Commentary): With two runners on base and a full count, Johnson, the opposing team's batter, faced an intense battle against the pitcher. The tension reached its peak as the pitcher delivered a devastating curveball, catching Johnson off guard. He swung and missed, resulting in a resounding strikeout that ended the inning and stranded the runners. Question: When did the pitcher deliver a crucial strikeout to end the inning? Answer Choices: a) At the start of the inning, Johnson struck out. b) After a series of foul balls, Johnson hit a double. c) At the end of the inning, Johnson grounded out. d) With a full count, Johnson struck out to end the inning.	
Response	a) At the start of the inning, Johnson struck out.	c) At the end of the inning, Johnson grounded out.

Table 12: Case study on sports domain. Negative cases where both models produce wrong responses are presented.