KnowDomain: Self Knowledge Generative Prompting for Large Language Models in Zero-Shot Domain-Specific QA

Anonymous ACL submission

Abstract

This pdf explain the revisions made to the paper "KnowDomain: Self Knowledge Generative Prompting for Large Language Models in Zero-Shot Domain-Specific QA"

1 Summary

This paper is the updated version of a previous submission to the EMNLP 2025 Main Conference. Based on the reviewer's feedback, we have carefully updated the paper to address all major concerns, with the goal of improving clarity, completeness, and reproducibility.

Dataset and Methodological Clarifications: We have also included the medical domain datasets used in the paper along with previously submitted Plant Pathology dataset. The clarifications on keyword filtering is added and knowledge generation procedures and w. We have updated the appendix section of Prompts and Examples with the examples of generated data and are now clearly outlined in Table 15 and sections 3 and 4.

Justification for Model Choices: The choice of LLaMA models for knowledge generation was guided by the availability of domain-adapted variants such as BioLLaMA and MedLLaMA, which are based on LLaMA. However, our method remains model-agnostic and does not require LLaMA specifically. This has been clarified in Section 2.3.

Prompt Design and Performance Analysis: We have strengthened the explanation of our prompting strategy, including the role of different knowledge sources (e.g., keyword definitions, notes, similar questions). Table 3 and Figure 2 are now discussed with additional insight to show how concise or noisy knowledge affects performance. We acknowledge the cases where simpler prompts perform well and contextualize why complex prompts (e.g., KD-NQ) still provide value, especially in challenging domains.

On Agentic Models and Broader Frameworks: While we acknowledge the relevance of agent-based approaches and deep-research paradigms, our work focuses specifically on domain-specific QA under a strict constraint of not using any external knowledge source or retriever module. We emphasise minimal prompting as a practical and lightweight alternative, and we now clarify this positioning in the Introduction and Related Work sections.

Reproducibility and Resources: We have clarified all hyperparameters, generation steps, and inference details in the paper. The PlantPathologyQA dataset and the medicalQA dataset is submitted alongside the paper. We have also submitted the full code in this review cycle.

Presentation Improvements: We have revised all tables (e.g., Tables 1, 3, and 6) to improve clarity, fixed formatting issues (e.g., row alignment), and added captions and footnotes to aid understanding.

Conclusion: We believe our revised paper offers a more complete, transparent, and compelling case for leveraging minimal prompt-based techniques to adapt general-domain LLMs to domain-specific QA tasks in a zero-shot setting. We again thank the reviewers for their valuable input, which has helped us significantly strengthen the work.

2 Metareview

This paper address the problem of LLMs providing inaccurate or generic answers for questions in specialized fields like medicine or plant pathology. The authors propose KnowDomain (KD), a prompting technique designed to force an LLM to generate its own domain-specific knowledge before answering a question. The process works in two steps: 1) for a given user question, the LLM first generates relevant keywords, definitions, and a list of similar question-answer pairs; 2) this generated text is then bundled with the original question into

a new, much larger prompt, which is fed back into the LLM to get the final answer. The authors evaluate this on models like Llama and Qwen across four datasets, including their own new PlantQA dataset, and find it improves accuracy over simpler prompting methods. Summary Of Reasons To Publish:

1. S1. Importance: This paper directly investigates a practical and significant limitation of current LLMs whose unreliability in specialized domains. The proposed self-knowledge generation method is a clever approach to improving domain-awareness without needing external databases or model retraining.

2. S2. Evaluation: The authors tested across multiple model families (Llama, Qwen) and sizes (7B to 70B), compared against a strong suite of prompting baselines (e.g., Chain-of-Thought, EchoPrompt), and performed detailed ablations. This comprehensive evaluation shows the method consistently outperforms baselines by a solid margin (4-10

3. S3. Ablation: The analysis comparing the full KD method to its KD-NQ variant (which omits keyword definitions) provides a crucial insight: simply adding more information is not always better, highlighting the type and relevance of the generated knowledge.

1. W1. Term Misleading: The current "zero-shot" is misleading and was challenged by all reviewers. The multi-step knowledge generation stage is different from the standard single-pass zero-shot prompt.

2. W2. Reproducibility: The paper currently lacks sufficient detail to reproduce the work. Some algorithm descriptions like keyword extraction, filtering, and the similar question generation are unclear. And code is not avaliable during the review stage.

3. W3. Ablation and Analysis: Some qualitative examples of the intermediate outputs are not included in paper, such as generated keywords, notes, and QA pairs. So it is hard to tell whether the improvement come from the intermediate outputs. In addition, the KD-NQ variant sometimes outperforms the full KD method, which is interesting but not fully explored. Since the compute time for knowledge generation cannot be ignored. This trade-off between accuracy and efficiency should be discussed and analyzed in the main paper by setting different budgets.

Summary Of Suggested Revisions:

1. We respectfully clarify that our use of the term "zero-shot" aligned with its broader definition in the literature—as a setting where the model is applied to new tasks or domains without task-specific training or fine-tuning. As our method includes a multi-step prompt-based knowledge generation process, it does not rely on any external corpora, labeled data, or in-context exemplars and even though all intermediate knowledge is generated by the LLM itself using only the input question. We have updated the paper such that it now emphasises the multistep prompting in order to avoid any confusion.

2. Along with previous details we have added the example of generated knowledge and further clarified the steps for better understanding. We have released all the datasets and code in current submission.

3. We include additional discussion in Section 5 to analyze why KD-NQ can outperform KD in some cases. One reason is that KD includes both definitions and notes, and definitions—being general—may sometimes conflict with question-specific context, introducing noise. We discuss the impact of knowledge length: KD averages over 2500 tokens, while KD-NQ is more concise (394 tokens), which can reduce confusion and improve model performance in some cases. We emphasize this trade-off between accuracy and efficiency, particularly in scenarios with compute budget constraints. We agree this is a valuable direction and have flagged it for future ablation-focused work.

3 Response to Reviewer LZvN

Thank you for reviewing our work. We have provided detailed responses to your comments below.

1. Misclassification of Method : Even though our method introduces an additional step of generating a knowledge base, we would like to clarify that the approach remains within

the zero-shot framework because we do not use any external knowledge sources, annotated data, or task-specific training. All the knowledge used is generated by the model itself, and it is not handcrafted or extracted from existing databases. The generated questions are not shown as explicit exemplars to the model (neither in answer format nor in reasoning steps). Instead, they provide context that helps the model answer the original question. Therefore, although there is an overhead in knowledge creation, we believe the method aligns with the zero-shot QA paradigm, since it requires no labelled data, no fine-tuning, and no external supervision. The difference of our method from RAG is that in this work, we do not retrieve information from any external knowledge base. Instead, all the contextual knowledge is generated by the model itself, using only the input question without access to external corpora. : *The total generation time (750hours) covers creating entities, definitions, notes, similar questions, and hints(for HintQA) across datasets using Llama8B and Llama70B. Average per-datapoint times(in sec):- generation took 1.67(Llama8B), 3.57(Llama70B) and hint generation 0.14; For QA (Llama8B): Base-2.53, COT-18.65, Echo-23.51, ARR-21.05, QAP-15.42, HintQA-6.42, KD-K -4.72, KD-NQ -6.50, KD-8.67; For QA (Llama70B): Base-3.49, COT-53.51, ARR-53.51, KD-K -4.79, KD-NQ -1.86, KD-2.13;. Adding generation time to QA time shows our approach is slower than Base and HintQA but still faster than larger prompts like ARR, COT, QAP and Echo.*

2. Figures and Method The purpose of Figure 1 is to illustrate how a general-domain LLM struggles to understand domain-specific knowledge and context in the questions. This helps motivate the need for our approach, which aims to enhance the model's understanding of domain-specific QA. Even though we have developed the model for a zero-shot setting, it is possible to fine-tune the second model if training data becomes available. We mention this as additional information that could be explored in future work if training data is also available. : *Figure 3 is a stacked bar plot where we combine the accuracy scores across five datasets for each approach and model. The y-axis shows the cumulative accuracy obtained by stacking the results, which explains why the values exceed the usual range (for example, reaching around 400). We chose this stacked format to provide an overall visual comparison of total performance across datasets, although individual dataset scores are not labelled in the figure due to space and readability constraints.*

3. Note: Our goal is to show general-domain LLMs can handle domain-specific QA with minimal change (just updating domain names). Even though the improvement over other prompts is sometimes small, our approaches still performed best or among the best across all datasets and models, which shows the reliability of our approach. Also, in a zero-shot setting, trying out many different prompt designs for each new domain would create extra work and complexity. Our methods aim to avoid that by offering a prompt that works well in most cases without extra tuning. Finally, even for the smaller MNOTA dataset (with only 500 QA pairs), our method did achieve the best result in at least one setting, which supports its usefulness even when data is limited. We will add detailed run time tables in the appendix as suggested.

4 Response to Reviewer FcqZ

1. The pseudo-code in Algorithm 1 is a little bit different from the description in section 3. For example, what are the fundamental criteria to extract domain-specific keywords? Is it the combination of step 2 and 3 in Algorithm 1? How to do the filter? Could you interpret what are the procedure to conduct knowledge generation in detail? How you generate similar and abstracting questions? What are the insight behind it? :- *We have only used prompting to tackle these, except for the filtering, where the filtering is used to only remove the basic English words. The fundamental criteria for extracting domain-specific keywords and generating similar questions are mentioned in the instruction set steps, and the complete prompt for knowledge generation is mentioned in Table 13. The idea behind showing LLM with similar questions is similar to showing LLM with in-context examples, but not exactly, as*

the sample questions' format is different and the questions don't explicitly tell the model how to answer the original question.

2. Is there any qualitative analysis of those questions in terms of quality? Why it helps? Whether the generated QA are correct? :- *In the current work, we have omitted this step, and we have mentioned this in the Limitations section*

3. From the table 3, seems like K-NQ perform even better than K on Qwn, BioLlama and MedLlama model. Is there any insight? : *While keyword definitions are short and general (within the domain), and these might not always match the context of the question, the notes and similar questions (NQ) are generated specifically with respect to the original question, and tend to be comparatively more useful. Hence, when compared to KD-K, KD-NQ performs better in most cases. In KD, we use the keyword definitions along with NQ, and conflicting keyword definitions can introduce ambiguity into the model's understanding, sometimes resulting in incorrect answers. Furthermore, BioLlama and MedLlama are medical-specific LLMs fine-tuned on medical data, which reduces their reliance on the external definition of keywords as a source of extra knowledge. In the case of the Qwen model, the performance for KD-NQ and KD is almost the same, except on MFCT and MNOTA datasets, which are comparatively smaller datasets with 96 and 500 data points, respectively. From our analysis of these datasets, no significant pattern was observed to explain this behaviour.*

5 Response to Reviewer p6LC

The paper is missing many details/justifications that hurt its reproducibility. For example:

4. (1) No details are provided regarding the new dataset, PlantQA. : *Details of the PlantQA dataset are provided in Appendix A. We have also submitted the dataset file in this submission.*

5. (2) It is not clear how the filtering of keywords was done. : *Filtering is used to remove only the basic English words.*

6. (3) It is not fully justified why using a Llama model is essential as depicted in Figure 2. :*In Figure 2, the Llama model is mentioned since we have used Llama models to generate the knowledge, either Llama8B or Llama70B. In the paper, we do not mention the explicit requirement of the only Llama model to generate the knowledge as any LLM can be used. However, we chose Llama since the BioLlama and MedLlama have the Llama model as a base hence it seemed appropriate to use Llama models for better comparison.*

7. (4) Table 6 talks about the number of questions per prompt but they are never mentioned in it. :-*Since we mentioned the column names in the same sequence as the number of questions mentioned in the caption, we omitted separately describing them. However, we can rephrase the caption for a clearer understanding.*

8. The results in Table 3 don't fully justify the proposed complex prompting technique. In particular, it looks like using only the question-answer pairs is sufficient in many cases.: *While keyword definitions are short and general (within the domain), and these might not always match the context of the question, notes and similar questions (NQ) are generated specifically with respect to the original question, and tend to be comparatively more useful. In KD, we use the keyword definitions along with NQ, and conflicting keyword definitions can introduce ambiguity into the model's understanding, sometimes resulting in incorrect answers. In the case of the Llama we can see that KD performed better in most cases and in case of Qwen model, the performance for KD-NQ and KD is almost the same, except on MFCT and MNOTA datasets, which are comparatively smaller datasets with 96 and 500 data points, respectively. Furthermore, BioLlama and MedLlama are medical-specific LLMs fine-tuned on medical data, which reduces their reliance on the external definition of keywords as a source of extra knowledge; these were only taken as baselines. Another possibility is the effect of knowledge length, as the knowledge present in KD-NQ is more concise, with an average of 394 tokens/question, than in the case of KD, where on average*

(2178+394) tokens/question are given as additional information which might have introduced noise in some cases. However, stating only the importance of notes and question-answer pairs will not be correct at this stage, since keyword definitions are general (within the domain) and further enhancement of these can significantly help the model for all cases.

9. The paper does not mention the work on agents/deep-research which will be a more general paradigm to address this problem instead of using tailored prompts.: *Since our work tackles scenarios only in the case of a zero-shot framework, where we don't have any external knowledge source which can be utilised in learning/training, we have only considered an approach which works in such scenarios.*