Review of Code Search Is All You Need? Improving Code Suggestions with Code Search

Junkai Chen, Xing Hu, Zhenhao Li, Cuiyun Gao, Xin Xia, David Lo

Summary

The paper titled Code Search Is All You Need? Improving Code Suggestions with Code Search proposes a method for enhancing the effectiveness of code suggestions using code retrieval. This method employs different code retrieval approaches(IR-based and DL-based) and search strategies(Header2Code, NL2Code, and NL2NL) to find similar code snippets, and then combine these snippets into prompts to enhance the performance of code suggestions through large language models. This paper implements a retrieval-augmented framework to employ many code retrieval methods, and compares the results through metrics like BLEU and CodeBLEU with the original models.

Reasons to accept the paper

1. The main strength of this paper is that it proposes a novel and flexible approach combining many common code search methods for code suggestion tasks. This approach bridges the gap between code suggestions and code retrieval by a combination of both technologies.
2. The authors clearly illustrate the process and framework of their approach in both Figure 1 and Figure 2, combined with the introduction of terminology like IR-based and DL-based code search enables readers to understand the conception of this approach.
3. Tables with multiple dimensions demonstrate the experience in detail, providing convincing and detailed results to prove the efficiency of the method. The results show a notable improvement in terms of BLUE-4 for both code completions and code generations.

Reasons to reject the paper

1. Although the authors introduce all components of the approach with high-level figures, they still fail to provide a detailed implementation of their approach. For example, in section 3.2, CodeBERT is used for the NL2Code search strategy, but how they fine-tune this model such as the parameters and the resources requirement for the fine-tuning, is not mentioned in this paper, leaving a question regarding the reproducibility of this approach.
2. As the authors mentioned in this paper three code search strategies are employed as retrievers in both code retrieval approaches, it could be clear for readers to understand this retrieval-augmented framework if they provide examples of input and output for these strategies respectively, allowing readers to execute different components of the approach by themselves even if some conditions like the fine-tuned model are not available.
3. The adaptability of this approach in code generation is not investigated, the authors use both GPT-3.5-turbo and text-davinci-003 as LLMs for code generation tasks, but these two models are under the sub-version of ChatGPT 3.5, therefore, it could be beneficial for the author to use other LLMs such as Llama, Deepseak to prove the adaptability of their approach.

Recommendation

My recommendation is that the paper should be accepted(Strong accept), this is because not only a novel and efficient approach is proposed in the paper, but the results shown in the tables are also convincing.

Major Comments

Section 1 Introduction : In the introduction part, the authors mention that one of the characteristics of their framework is 'plug-and-play', which emphasizes that this framework supports different language models. However, we can only see two language models in the same version from OpenAI employed in the experience part, and these models share similar request and response body, so the framework may not be compatible with other language models with different input and output.

Section 2 Background : Because of the use of General DL models as code completion tasks, it should be beneficial to detail the implementation or categories of General DL models employed in this approach, the details like parameters and training hours will enable readers to reproduce.

Section 3 Methodology-3.3 Formulator : The function of the Formulator is to produce the prompt combined with templates and code obtained from the Retriever, and then feed it to the Generator, but the prompt strategies and templates should be mentioned in this part to improve readability.

Section 4 Experimental Setup : Since the authors choose different retrieval approaches and search strategies, including both IR-based and DL-based which are implemented in two diverse technological paths, the applicability will be proved if the extra metric to examine this framework's time efficiency with different search methods(like Header2Code with IR-based strategy or NL2NL with DL-based strategy) is considered.

Section 5 Results-5.1 Results for general DL models on line-level code completion (RQ1). : As mentioned above, the time efficiency should be analyzed in this sector; this is because the code suggestions, especially for the code completions, are extremely time-sensitive, analyzing this metric will clarify the applicability of this framework. In addition, the adaptability of this framework will be proved if more language models are included in this experiment.

Section 5 Results-5.2 Table 5: Results for LLMs on code generation (RQ2). : The critical parameters, like temperature and top_p, to invoke LLM(GPT 3.5) should be introduced in this section for clarity and reproducibility.

Section 6 Discussion : It would be beneficial if the paper discusses the existing or potential drawbacks of this retrieval-augmented framework to warn readers. Furthermore, the experimental conditions or resources should be indicated in this paper for other researchers.

Section 7 Related Work : In this section, it might be useful to introduce and discuss other code suggestion methodologies, this may help readers to open their minds or build a concrete conception of why the retrieval-augmented framework works better than other implementations.

Section 8 Conclusion and Future Work : The authors should mention some potential directions for readers interested in this approach to refine and enhance this framework in the future.

Minor comments

Section 4.2 Experimental Setup : The table of 'Table 2: Statistics of the datasets.' should be put to the section of Methodology for readability

Section 6.1 Discussion : The 'sturctured' should be "structured."