Feedback on "Understanding Copilot X and it's potential for improved DevOps practices"

David Cederström
Arasp Keighobad

April 26, 2023

DD2482

We certify that generative AI, incl. ChatGPT, has not been used to write this feedback. Using generative AI without permission is considered academic misconduct.

[Some general and overviewing comments about the essay...]

High-level Strengths

- The essay chooses the very interesting topic of copilot/LLMs (and admittedly very hot right now) and ties it to the course content as well as more traditional practices like pair programming in a balanced and intriguing way.
- The essay makes some strong valid points about the strengths and weaknesses of both pair programming as well as using new AI tools to aid for this.
- The essay is written in nice and easy to read language, keeping the reader engaged as well as making the material easier to digest. Anything mentioned is explained and the text is fit for the intended audience.

High-level Weaknesses

- The challenges section really only provided one single challenge, whereas for example in the section where you listed alternative models there were some problems and challenges that the other models aimed to solve, e.g. not relying on licensed code etc.
- Weaknesses and potential problems with licensing, ethical issues etc should perhaps also be explored. Have there been any cases where problems were caused by using the AI tool? i.e. over-reliance on that the output is correct and it turning out to break in some unforeseen way.

Specific Feedback

Title and introduction

The title of the essay is good. You understand the core concept of the essay which also is reflected in the essay itself. The quite heavy focus on pair programming is not really that clear from just reading the title of the essay, although you rather quickly understand that the possible, practical implementations with Co-pilot X is done by pair programming so it is not really an issue.

The introduction does a good job of introducing the reader to the topic pair programming, AI's potential in improving this methodology and why this is important and relevant to DevOps. It sparks the readers interest and the authors does a good job in describing what the potential problems are with pair programming and what AI can potentially do in solving these issues, therefore also introducing a potential solution to the problem which makes the reader want to keep on reading.

"Another problem is that the Navigator, who gives direction to the developer, would need to be knowledgeable of the operations and infrastructure of the software environment, in order to best make suggestions." → This does not feel like a drawback of pair programming itself, perhaps the drawback would then be that you need more people to do pair programming for the same amount of value compared to if they just worked separately (if one could show that the person would be more valuable not being merely a navigator)

"In this thesis we will give a general description (...)" → Perhaps thesis is not the right word

Background

It is appreciated by the readers that the authors have included a short introduction and overview of the section. This makes it easier for the reader to understand and be prepared to what it can expect in the form of content and context.

Since pair programming is an important concept within this topic, the authors did a good choice

by providing the reader with some background information regarding the concept. A nice image was also added for additional, visual representation of what pair programming looks like and how it can be implemented. The authors also motivates the relevance of pair programming in DevOps in a good way, making the reader more convinced that the presented technologies and methodologies can be very relevant as both a problem and solution within the field. It is also nice to get a background to what GitHub Copilot X is and why it is connected to the concept of pair programming.

Although interesting, we are not sure that it is relevant or provides the reader with anything useful for this essay to list other AI code tools than Github Copilot X in a separate section.

"However Copilot is likely to remain the most widely used one, since it is integrated with the most advanced large language model." What models do the others use?

Combining pair programming with AI

Interesting section where the more traditional practice of pair programming is compared to new more high-tech tools that could potentially replace, or augment pair programming.

Suggestions and comments

"and Copilot X would allow the used to have someone to collaborate with" used → user

"an experienced user of tools like Terraform or other DevOps toolkits can bring along Copilot for auto-completion and code suggestions" → why specifically Terraform? Is it extra suited for using alongside Copilot? You bring it up as an example suddenly and seemingly out of context.

"One of the most important tools of a developer is searching. Searching for code, libraries, tools or any other part of the development process. After something suitable has been found, it then has to be shaped to fit the problem (or code) at hand, which can be time consuming and tedious" → Good point, that is a very compelling strength of using Copilot as a pair programmer

Conclusion

Nice conclusion describing and weighing and rounding up the different strengths and challenges of the topic discussed. On the other hand the same feedback about not discussing the challenges enough applies here too. Perhaps the conclusion would have been more balanced if more challenges were explored.

Corrections

"Because of it's ability to understand" it's → its

"As a result, if Copilot does shows promising results" either remove "does" or shows → show

General comments

Tiny nitpick but usually, the word "however" is idiomatically followed by a comma, you guys missed that at a couple of places.

Pointers to additional material and related work

Found this article that might be interesting for pondering about other challenges with this approach: **Why You Should Apply Caution When Using AI in Code Development** https://www.spiceworks.com/tech/artific intelligence/guest-article/why-you-should-apply-caution-when-using-ai-in-code-development/

This post on Github's own blog might be interesting: **Responsible AI pair programming with GitHub Copilot** https://github.blog/2023-02-22-responsible-ai-pair-programming-with-github-copilot/