

Subject: Asking for your opinion on how to write good commit messages?

Hello xxx,

I am a researcher at Beijing Institute University, China. Our team is investigating how to help developers write high-quality Commit Messages effectively. In literature, many tools have been proposed to generate Commit Messages automatically, such as DeltaDoc, ChangeScribe, and NNGen. However, these tools are trained and evaluated with original datasets that may affect the performance because of containing bad Commit Messages.

Therefore, we want to deeply explore what kinds of Commit Messages can help developers understand the context of a code change and facilitate their collaboration. Then we intend to propose a method automatically identifying high-quality Commit Messages. We can improve the real-life performance of the automatic tools by training and evaluating them on the dataset of high-quality Commit Messages. We also expect our study could give developers diverse examples of good and bad Commit Messages to improve their writing.

From what I could gather, you've made outstanding contributions (e.g., commits ) to <OSS project>, and we deem you have a deep understanding of collaboration development. We would much appreciate it if you could please answer the following three questions that should not take more than three minutes. (Please note all the answers will be kept anonymous and only for research purpose):

Q1: Do you think the following four commit messages are well-written (noted as W) or need to be improved (noted as N).

(1) Commit Message: <content of commit message>

URL: <URL of commit message>

Your answer: ( )

(2) Commit Message: <content of commit message>

URL: <URL of commit message>

Your answer: ( )

(3) Commit Message: <content of commit message>

URL: <URL of commit message>

Your answer: ( )

(4) Commit Message: <content of commit message>

URL: <URL of commit message>

Your answer: ( )

Q2: What criteria should a good commit message meet (the provided options are mentioned by other developers, FYI)

A. Need to explain why the change is needed and summarize what happened.

B. Change is small and easy to understand.

C. Text-related changes that are self-explanatory.

D. I know these changes well, so Commit Messages are unnecessary.

E. Others: \_\_\_\_\_

Q3: What factors do you think affect the quality of a Commit Message?

## Here is an example:

Hello xxx,

I am a researcher at Beijing Institute University, China. Our team is investigating how to help developers write high-quality Commit Messages effectively. In literature, many tools have been proposed to generate Commit Messages automatically, such as DeltaDoc, ChangeScribe, and NNGen. However, these tools are trained

and evaluated with original datasets that may affect the performance because of containing bad Commit Messages.

Therefore, we want to deeply explore what kinds of Commit Messages can help developers understand the context of a

code change and facilitate their collaboration. Then we intend to propose a method automatically identifying high-quality Commit Messages. We can improve the real-life performance of the automatic tools by training and evaluating

them on the dataset of high-quality Commit Messages. We also expect our study could give developers diverse examples of

good and bad Commit Messages to improve their writing.

From what I could gather, you've made outstanding contributions (e.g., commits ) to error-prone, and we deem you

have a deep understanding of collaboration development. We would much appreciate it if you could please answer the following three questions that should not take more than three minutes. (Please note all the answers will be kept anonymous and only for research purpose):

Q1: Do you think the following four CMs are well-written (noted as W) or need to be improved (noted as N).

(1) Commit Message: "[Dubbo-2177]Fix the issue: Supplementary unit test #2177 team 3 (#2260)

\* test

\* Please enter the commit message for your changes. Lines starting

\* test “,

URL: <https://github.com/apache/dubbo/commit/f364a44fa21c90120749de4335ce1cee4215536d>

Your answer: ( )

(2) Commit Message: ` Polish `,

URL: <https://github.com/spring-projects/spring-boot/commit/c3b344fdc2d2074f3dc1f451e0f56844fa9462f1> .

Your answer: ( )

(3) Commit Message: ` constant names should be all uppercase, separated by underscores (#5356) `,

URL: <https://github.com/apache/dubbo/commit/ed7be11af9a8a9dddb4d23eea089f3890391575b> .

Your answer: ( )

(4) Commit Message: ` Backport build script updates `,

URL: <https://github.com/spring-projects/spring-boot/commit/53296cd16109dc53de867c4cb12b10cecf47bb78> .

Your answer: ( )

Q2: What criteria should a good CM meet (the provided options are mentioned by other developers, FYI)

A. Need to explain why the change is needed and summarize what happened.

B. Change is small and easy to understand.

C. Text-related changes that are self-explanatory.

D. I know these changes well, so Commit Messages are unnecessary.

E. Others: \_\_\_\_\_

Q3: What factors do you think affect the quality of a Commit Message?