Which Log Level Should Developers Choose for a New Logging Statement? (Journal-First Abstract)

Heng Li*, Weiyi Shang†, Ahmed E. Hassan*
Software Analysis and Intelligence Lab (SAIL), Queen's University, Canada*
Department of Computer Science and Software Engineering, Concordia University, Canada†
{hengli, ahmed}@cs.queensu.ca*, shang@encs.concordia.ca†

***Abstract*—This is an extended abstract of a paper published in the Empirical Software Engineering journal. The original paper is communicated by Mark Grechanik. The paper empirically studied how developers assign log levels to their logging statements and proposed an automated approach to help developers determine the most appropriate log level when they add a new logging statement. We analyzed the development history of four open source projects (Hadoop, Directory Server, Hama, and Qpid). We found that our automated approach can accurately suggest the levels of logging statements with an AUC of 0.75 to 0.81. We also found that the characteristics of the containing block of a newly-added logging statement, the existing logging statements in the containing source code file, and the content of the newly-added logging statement play important roles in determining the appropriate log level for that logging statement.**

***Index Terms*—software logging, log level, ordinal regression model**

I. Overview

Logging statements are used to record valuable runtime information about applications. Each logging statement is assigned a log level such that users can disable some verbose log messages (e.g., "debug" messages) while allowing the printing of other important ones (e.g., "error" messages). Log levels are beneficial for both developers and users to trade-off the rich information in logs with their associated overhead.

However, prior research finds that developers often have difficulties when determining the appropriate level for their logging statements, and that developers spend much effort on adjusting the levels of their logging statements. Prior research also explains this issue by arguing that developers rarely have complete knowledge of how the code will ultimately be used. To the best of our knowledge, there exists no prior research regarding log level guidelines.

In this paper [1], we propose an automated approach to help developers determine the appropriate log level when they add a new logging statement. We analyze the development history of four open source projects (Hadoop, Directory Server, Hama, and Qpid). Our preliminary study shows that logging statements have a different distribution of log levels across the different containing code blocks, and particularly, in different types of exception handling blocks. Based on our preliminary study and our intuition, we calculate a set of software metrics and build ordinal regression models to automatically suggest the most appropriate level for a newly-added logging statement. We also carefully analyze our models to find the important factors for determining the most appropriate log level for a newly-added logging statement.

We find that our ordinal regression model can accurately suggest the levels of logging statements with an AUC (area under the curve; the higher the better) of 0.75 to 0.81 and a Brier score (the lower the better) of 0.44 to 0.66, which is better than randomly guessing the appropriate log level (with an AUC of 0.50 and a Brier score of 0.80 to 0.83) or naively guessing the log level based on the proportional distribution of each log level (with an AUC of 0.50 and a Brier score of 0.65 to 0.76). We also find that the characteristics of the containing block of a newly-added logging statement, the existing logging statements in the containing source code file, and the content of the newly-added logging statement play important roles in determining the appropriate log level for that logging statement.

This is the first work to support developers in making informed decisions when determining the appropriate log level for a logging statement. Developers can leverage our models to receive automatic suggestions on the choices of log levels or to receive warnings on inappropriate usages of log levels. Our results also provide an insight on the factors that influence developers when determining the appropriate log level for a newly-added logging statement.

Please cite the original paper which is published in the Empirical Software Engineering (EMSE) journal:
H. Li, W. Shang, and A. E. Hassan, "Which log level should developers choose for a new logging statement?" Empirical Software Engineering, vol. 22, no. 4, pp. 1684–1716, Aug 2017.
Available online:
https://doi.org/10.1007/s10664-016-9456-2

The full version of this work was published as journal article [1].

References

[1] H. Li, W. Shang, and A. E. Hassan, "Which log level should developers choose for a new logging statement?" *Empirical Software Engineering*, vol. 22, no. 4, pp. 1684–1716, Aug 2017. [Online]. Available: https://doi.org/10.1007/s10664-016-9456-2