Large language models for data mining in software repositories

supervised by

Dr Zhiwei Lin

Enterprise software systems change from time to time to meet the requirements from end users. In modern software development practice, the requirements are recorded as issues (business stories, technical tasks or bug reports) in an issue-tracking system, such as JIRA. The developer who is assigned an issue, will make necessary changes (adding, deleting, modifying) to the source code, in order to meet the needs defined in the assigned issue.

However, software issues are usually created by different persons (users, analysts, testers), leading to incomplete or duplicate information stored in the issue tracking system. For example, two or more bug issues may be recorded by different users but the issues may all refer to the same bug. Establishing links between duplicate bug issues is key to help software developers to reduce their efforts in fixing software bugs [1,2,3,4].

This project aims to develop a novel pipelines using large language models to understand software development documents, including software issues and source code, in order to create links between issues and to predict software bugs. To achieve this, the candidate will (1) analyse software issues using large language models, (2) study new approaches for detecting the association patterns between software code changes and software issues; and (3) develop intelligent tools for software developers to predict software bugs.

Please feel free to contact Dr Zhiwei Lin (z.lin@qub.ac.uk) before you make your application.

References

[1] Sun, C., Lo, D., Khoo, S., Jiang, J.: Towards more accurate retrieval of dupli- cate bug reports. In: Alexander, P., Pasareanu, C.S., Hosking, J.G. (eds.) 26th IEEE/ACM International Conference on Automated Software Engineering (ASE 2011), Lawrence, KS, USA, November 6-10. pp. 253–262.

[2] D. Hu, et al. Bugs and features, do developers treat them differently. In ICAIBD'18, pp 250-255.

[3] M. Zhang, et al. Boosting spectrum-based fault localization using pagerank. ACM ISSTA'17, pp 261-272.

[4] F. Horváth, et al. Using contextual knowledge in interactive fault localization. (2022). https://doi.org/10.1007/s10664-022-10190-x.