LLM OF BABEL

EVALUATION OF LLMS ON CODE FOR NON-ENGLISH USE-CASES

Author: Yongcheng Huang

Supervisors: Prof. Dr. Arie van Deursen, Assistant Prof. Dr. Maliheh Izadi, ir. Jonathan Katzy

Committee: Prof. Dr. Arie van Deursen, Assistant Prof. Dr. Maliheh Izadi, Assistant Prof. Dr. Gosia Migut

ABSTRACT

Large Language Models (LLMs) have shown impressive multilingual abilities and are widely used, especially in programming. However, evaluations of LLMs on code are mostly centered on English. This study assesses LLM performance in generating Chinese Java code comments via open coding. Additionally, we explore quantitatively analyzing semantic errors, particularly hallucinations, by examining cosine similarity of word embeddings.

RESEARCH QUESTIONS

Question 1: What mistakes do LLMs make when generating Chinese Java code comments?

Question 2: Can semantic errors be analyzed through the examination of cosine similarities of word embeddings?

METHODOLOGY

1 DATA PREPARATION

2 INFERENCE PIPELINE

3 OPEN CODING

4 COSINE SIMILARITY & PCA

Use ERNIE 1.0[1] to obtain word embeddings from the model-generated comments, then compare the cosine similarity[2] and perform PCA.

5 VISUALIZATION

DISCUSSION

This study identified common errors made by LLMs in generating Chinese Java code comments, developing a detailed error taxonomy that highlights prevalent semantic errors and validates the use of cosine similarity for analyzing these errors. Our findings demonstrate that lower cosine similarity correlates with significant semantic deviations, indicating potential for automated performance analysis. This research contributes valuable insights for improving LLMs in multilingual code annotation tasks and sets the groundwork for future advancements.

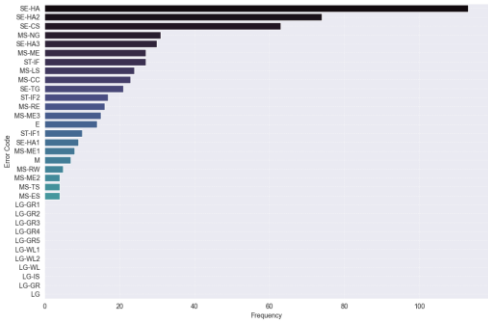
CONCLUSION

This study examined LLMs' performance in generating Chinese code comments, identifying common semantic errors like hallucinations and validating the use of cosine similarity to detect these errors. Our findings provide an error taxonomy and demonstrate potential for automated performance analysis. Future research should expand to other languages and metrics to build on these insights.

RESULTS

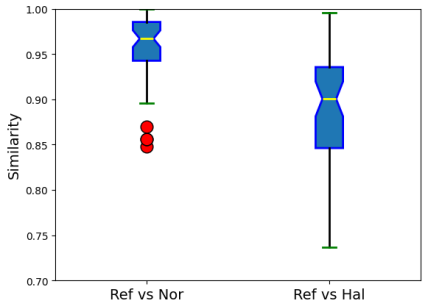
1 ERROR FREQUENCY

The error frequency of open coding result.

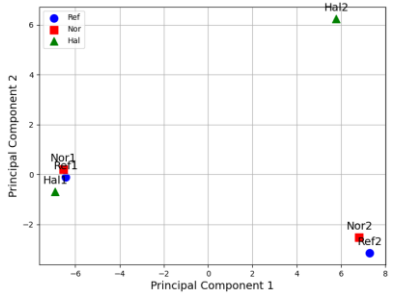


2 COSINE SIMILARITY (HALLUCINATION)

Cosine similarity between Group1 (Ref vs Normal comments) and Group2 (Ref vs Hallucinatory comments).

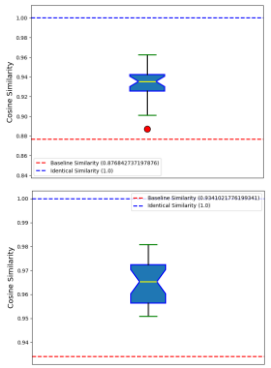


3 PCA RESULT



4 COSINE SIMILARITY (OVERLY BROAD)

Cosine similarity analysis for overly broad comments, with result in both line and block comments.



TU Delft

REFERENCE

1. Yu Sun, Shuohuan Wang, Yukun Li, Shikun Feng, Xuyi Chen, Han Zhang, Xin Tian, Hao Tian, and Hua Wu. Ernie: Enhanced representation through knowledge integration. arXiv preprint arXiv:1904.09223, 2019.

2. Ansar Aynetdinov and Alan Akbik. Semscore: Automated evaluation of instruction-tuned llms based on semantic textual similarity, 2024