**报告正文**

参照以下提纲撰写，要求内容翔实、清晰，层次分明，标题突出。**请勿删除或改动下述提纲标题及括号中的文字。**

**（一）立项依据与研究内容**（**建议8000字以下**）：

1．**项目的立项依据**（研究意义、国内外研究现状及发展动态分析，需结合科学研究发展趋势来论述科学意义；或结合国民经济和社会发展中迫切需要解决的关键科技问题来论述其应用前景。附主要参考文献目录）；

**1.1** **研究意义**

**1.2 国内外研究现状**

**主要参考文献目录：**

1. **项目的研究内容、研究目标，以及拟解决的关键科学问题**（此部分为重点阐述内容）**；**

**2.1 研究内容**

**2.2 研究目标**

基于研究内容的剖析，申请书的研究目标包括：

（1）

（2）

（3）

整体来说，申请书关注软件质量保障问题，具有一定的产业引用价值和研究意义。

**2.3 拟解决的关键科学问题**

为了实现上述研究目标，我们在研究过程中主要解决如下关键科学问题：

（1）

（2）

（3）

3．**拟采取的研究方案及可行性分析**（包括研究方法、技术路线、实验手段、关键技术等说明）；

**3.1 研究方案**

针对上述研究问题，申请人均有明确的研究方案。

**3.2 可行性分析**

对于部分研究方案，申请者已经对其可行性进行了初步验证。具体来说：

综上所述，项目申请书所提出的研究内容具体、目标明确、技术路线清楚，相关研究基础和预研结果较为扎实，并且具有良好的软硬件环境和实验平台。因此本项目具有较强的可行性。

4．**本项目的特色与创新之处；**

5．**年度研究计划及预期研究结果**（包括拟组织的重要学术交流活动、国际合作与交流计划等）。

**5.1 年度研究计划**

本项目的研究周期为四年，根据项目的研究内容和拟解决的科学问题，预期的研究计划如下表所示。

|  |  |
| --- | --- |
| **起止时间** | **各研究阶段的实施计划** |
| 2024.01~2024.02 |  |
| 2024.03~2024.12 |  |
| 2025.01~2025.12 |  |
| 2026.01~2026.12 |  |
| 2027.01~2027.10 |  |
| 2027.11~2027.12 |  |

**5.2 预期研究成果**

**（二）研究基础与工作条件**

1．**研究基础**（与本项目相关的研究工作积累和已取得的研究工作成绩）；

2．**工作条件**（包括已具备的实验条件，尚缺少的实验条件和拟解决的途径，包括利用国家实验室、国家重点实验室和部门重点实验室等研究基地的计划与落实情况）；

3．**正在承担的与本项目相关的科研项目情况**（申请人和主要参与者正在承担的与本项目相关的科研项目情况，包括国家自然科学基金的项目和国家其他科技计划项目，要注明项目的资助机构、项目类别、批准号、项目名称、获资助金额、起止年月、与本项目的关系及负责的内容等）；

无。

4．**完成国家自然科学基金项目情况**（对申请人负责的前一个已资助期满的科学基金项目（项目名称及批准号）完成情况、后续研究进展及与本申请项目的关系加以详细说明。另附该项目的研究工作总结摘要（限500字）和相关成果详细目录）。

**（1）前一个已结题科学基金项目的说明**

项目类型：国家自然科学基金（青年基金），项目名称：软件测试和调试过程中的测试用例演化技术研究，批准号：61202006，执行时间为2013年1月~2015年12月。

该项目以测试用例集演化为切入点，通过深入挖掘软件历史仓库、配套测试用例集和项目源代码，对软件测试和调试过程中的一些关键问题（例如软件缺陷预测、软件缺陷定位和回归测试中的缺陷检测等）展开了深入研究。具体来说：（1）通过对软件缺陷预测问题的研究，有助于更为精准的预先识别出项目内的缺陷程序模块，从而指导测试用例集的高质量演化，以确保对这些程序模块的充分测试。在项目执行期间，课题组对缺陷预测数据集内的特征选择、类不平衡问题以及噪声问题等进行了一定的研究，最终提高了缺陷预测模型的预测性能。（2）已有研究表明，通过搜集测试用例的程序频谱（program spectrum）和执行结果，可以提高缺陷定位的效果（即有助于开发人员减少需要审查的代码量，以确保可以更为迅速的找到缺陷语句并进行修复）。在项目的执行期间，课题组从程序建模角度入手，基于程序切片和形式化概念分析等技术提出了多种有效的缺陷定位方法。（3）为提高回归测试中的缺陷检测效率，课题组从分析代码修改角度入手，提出了合理的测试用例演化框架，从而可以针对修改后的程序生成高质量的测试用例集。课题组认为针对上述三个问题的深入研究，有助于测试部门合理分配测试资源，并最终提高软件产品的质量，因此具有一定的理论意义和实践价值。

在前一个项目的研究过程中，借助软件缺陷预测技术，希望能够更为精准的预测出被测项目内的缺陷程序模块，从而优化测试资源分配，更好的指导测试用例集演化。这部分研究工作的积累可以有效支持该申请书中的研究内容。

**（2）已结题项目研究工作总结摘要**

以高质量测试用例集演化为切入点，通过深入挖掘和分析软件历史仓库、配套测试用例集和项目源代码，对软件测试和调试过程中的重要问题展开了一系列深入研究。具体来说，从缺陷预测数据集中的维数灾难、类不平衡问题以及抗噪声干扰能力入手，提出了多种有效的数据集预处理方法，可以有效提高随后构建的缺陷预测模型的性能。从程序频谱构造方式和缺陷数量这两个影响因素入手，提出了基于混合切片谱的缺陷定位方法和基于遗传算法的多缺陷定位方法，可以有效提高生成的缺陷定位报告质量。从分析代码修改角度入手，提出了合理的测试用例演化框架，从而可以针对修改后的程序生成高质量的测试用例集。随后对框架内的测试用例优先级排序、测试用例集缩减和测试用例集扩充等问题进行了研究。针对上述这三个问题的深入研究以及期间取得的研究成果，有助于测试部门合理分配测试资源，并最终提高软件产品的质量，因此具有一定的理论意义和实践价值。

**（3）项目研究期间的代表性成果（\*为通讯作者）**

代表性成果主要发表在国外权威期刊、权威会议和国内一级学报上，其中CCF-A类期刊会议（TSE、TOSEM、ICSE、ESEC/FSE、ASE）7篇、CCF-B类期刊会议（IST、JSS、JSEP、JCST、SPE、SCIS、IETS、ICSME、SANER、ISSRE等）31篇、国内一级学报（中国科学：信息科学、计算机学报、软件学报和计算机研究与发展）33篇。

* **CCF A类论文**

1. Xiang Chen, Chunyang Chen, Dun Zhang, Zhenchang Xing. SEthesaurus: WordNet in Software Engineering. IEEE Transactions on Software Engineering. 2021, 47(9):1960-1979.
2. Zhichao Chen, Junjie Chen, Weijing Wang, Jianyi Zhou, Meng Wang, Xiang Chen, Shan Zhou, Jianmin Wang. ACM Transactions on Software Engineering and Methodology. 2022, Accepted.
3. Yingquan Zhao, Zan Wang, Shuang Liu, Jun Sun, Junjie Chen, Xiang Chen. Achieving High MAP-Coverage through Pattern Constraint Reduction. IEEE Transactions on Software Engineering. 2023, 49(1): 99-112.
4. Chao Ni, Xin Xia, David Lo, Xiang Chen, Qing Gu. Revisiting Supervised and Unsupervised Methods for Effort-Aware Cross-Project Defect Prediction. IEEE Transactions on Software Engineering. 2022, 48(3):786-802. **(Invited to ICSE 2021 as part of the Journal First Paper Track.)**
5. 【🏆**ACM SIGSOFT Distinguished Paper Award**】Kaibo Cao, Chuyang Chen\*, Sebastian Baltes, Christoph Treude, Xiang Chen\*. Automated Query Reformulation for Efficient Search Based on Query Logs from Stack Overflow. In: Proceedings of 43rd International Conference on Software Engineering (ICSE 2021). 2021, 1273-1285.
6. Qingchao Shen, Haoyang Ma, Junjie Chen\*, Yongqiang Tian, Shing-Chi Cheung, Xiang Chen. A Comprehensive Study of Deep Learning Compiler Bugs. In Proceedings of The ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering (ESEC/FSE 2021). 2021,968-980.
7. Zan Wang, Yingquan Zhao, Shuang Liu, Jun Sun, Xiang Chen, Huarui Lin. MAP-Coverage: A Novel Coverage Criterion for Testing Thread-Safe Classes. In Proceedings of the 34th IEEE/ACM International Conference on Automated Software Engineering (ASE 2019). 2019, 722-734.

* **CCF B类论文**

1. Chi Yu, Guang Yang, Xiang Chen\*, Ke Liu and Yanlin Zhou. BASHEXPLAINER: Retrieval-Augmented Bash Code Comment Generation based on Fine-tuned CodeBERT. In Proceedings of the 38th International Conference on Software Maintenance and Evolution (ICSME 2022). 2022, 82-93.
2. Ke Liu, Guang Yang, Xiang Chen\* and Chi Yu. SOTitle: A Transformer-based Post Title Generation Approach for Stack Overflow. In Proceedings of The 29th IEEE International Conference on Software Analysis, Evolution and Reengineering (SANER 2022). 2022, 577-588.
3. Guang Yang, Xiang Chen\*, Yanlin Zhou and Chi Yu. DualSC: Automatic Generation and Summarization of Shellcode via Transformer and Dual Learning. In Proceedings of The 29th IEEE International Conference on Software Analysis, Evolution and Reengineering (SANER 2022). 2022, 361-372.
4. Bin Du, Yuxiaoyang Cai, Haifeng Wang, Yong Liu and Xiang Chen. Improving the Performance of Mutation-based Fault Localization via Mutant Bias. In Proceedings of The 33rd International Symposium on Software Reliability Engineering (ISSRE 2022). 2022, 309-320.
5. Wei Zheng, Manqing Zhang, Hui Tang, Yuanfang Cai, Xiang Chen, Xiaoxue Wu and Abubakar Omari Abdallah Semasaba. Automatically Identifying Bug Reports with Tactical Vulnerabilities by Deep Feature Learning. In Proceedings of The 32nd International Symposium on Software Reliability Engineering (ISSRE 2021). 2021, 333-344.
6. Guang Yang, Yu Zhou\*, Xiang Chen\*, Xiangyu Zhang, Tingting Han, Taolue Chen. Journal of Systems and Software. 2023, 197: 111577.
7. Hao Lin, Xiang Chen\*, Xuejiao Chen, Zhanqi Cui, Yun Miao, Shan Zhou, Jianmin Wang, Zhan Su. TitleGen-FL: Quality Prediction-based Filter for Automated Issue Title Generation. Journal of Systems and Software. 2023, 195: 11513.
8. Chunyu Zhao, Yanzhou Mu, Xiang Chen\*, Jingke Zhao, Xiaolin Ju\*, Gan Wang. Can Test Input Selection Methods for Deep Neural Network Guarantee Test Diversity? A Large-scale Empirical Study. Information and Software Technology. 2022, 150: 106982.
9. Hengyuan Liu, Zheng Li, Haifeng Wang, Yong Liu, Xiang Chen. CRMF: A Fault Localization Approach based on Class Reduction and Method Call Frequency. Software: Practice and Experience. 2022. Accepted.
10. Haifeng Wang, Hengyuan Liu, Zheng Li, Yong Liu, Fuxiang Sun, Xiang Chen. A Token-based Compilation Error Categorization and Its Applications. Journal of Software: Evolution and Process. 2022. Accepted.
11. Jiaming Zhang, Zhanqi Cui, Xiang Chen, Huanhuan Wu, Liwei Zheng, Jianbin Liu. DeltaFuzz: Historical Version Information-Guided Fuzz Testing. Journal of Computer Science and Technology. 2022, 37(1):29-49.
12. Wei Zheng, Tianren Shen, Xiang Chen, Peiran Deng. Interpretability Application of the Just-in-Time Software Defect Prediction Model. Journal of Systems and Software. 2022, 188: 11245.
13. Zheng Li, Yonghao Wu, Bin Peng, Xiang Chen, Zeyu Sun, Yong Liu, Deli Yu. SeCNN: A Semantic CNN Parser for Code Comment Generation. Journal of Systems and Software. 2021, 181:111036:1-111036:17. **(Invited to SANER 2023 as part of the Journal First Paper Track)**
14. Xiang Chen, Zhidan Yuan, Zhanqi Cui, Dun Zhang, Xiaolin Ju. Empirical Studies on the Impact of Filter based Ranking Feature Selection on Security Vulnerability Prediction. IET Software. 2021, 15(1):75-89.
15. Xiaoxue Wu, Wei Zheng, Xiang Chen, Yu Zhao, Tingting Yu, Dejun Mu. Improving High-impact Bug Report Prediction with Combination of Interactive Machine Learning and Active Learning. Information and Software Technology. 2021, 133: 106530:1 - 106530:12.
16. Xiang Chen, Yanzhou Mu, Ke Liu, Zhanqi Cui, Chao Ni. Revisiting Heterogeneous Defect Prediction Methods: How Far Are We? Information and Software Technology. 2021, 130: 106441:1-106441:16.
17. Zhengliang Li, Zhiwei Jiang, Xiang Chen\*, Kaibo Cao, Qing Gu\*. LaProb: A Label Propagation Based Software Bug Localization Method. Information and Software Technology. 2021, 130: 106410:1-106410:15.
18. Yonghao Wu, Zheng Li, Yong Liu, Xiang Chen. FATOC: Bug isolation-based multi-fault localiation by using optics clustering. Journal of Computer Science and Technology. 2020, 35(5):979-998.
19. Wei Zheng, Jialiang Gao, Xiaoxue Wu, Fengyu Liu, Yuxing Xun, Guoliang Liu, Xiang Chen. The Impact Factors on the Performance of Machine Learning-Based Vulnerability Detection: A Comparative Study. Journal of Systems and Software. 2020, 168: 110659:1-110659:12.
20. Xiang Chen, Yanzhou Mu, Yubin Qu, Chao Ni, Meng Liu, Tong He, Shangqing Liu. Do Different Cross-project Defect Prediction Methods Identify the Same Defective Modules? Journal of Software: Evolution and Process. 2020, 32(5):e2234:1-e2234:24.
21. Chao Ni, Xiang Chen, Xin Xia, Qing Gu, Yingquan Zhao. Multitask Defect Prediction. Journal of Software: Evolution and Process. 2019, 31(12):e2203. **(Invited to ICSME 2019 as part of the Journal First Paper Track.)**
22. Xiaoxue Wu, Wei Zheng, Xiang Chen, Fang Wang, Dejun Mu. CVE-Assisted Large-Scale Security Bug Report Dataset Construction Method. Journal of Systems and Software. 2020, 160: 110456:1-110456:14.
23. Xiang Chen, Dun Zhang, Zhanqi Cui, Qing Gu, Xiaolin Ju. DP-Share: Privacy-preserving Software Defect Prediction Model Sharing through Differential Privacy. Journal of Computer Science and Technology. 2019, 34(5): 1020-1038.
24. Tianchi Zhou, Xiaobing Sun, Xin Xia, Bin Li, Xiang Chen. Improving Defect Prediction with Deep Forest. Information and Software Technology. 2019, 114: 204-216.
25. Chao Ni, Xiang Chen\*, Fangfang Wu, Yuxiang Shen, Qing Gu\*. An Empirical Study on Pareto Based Multi-objective Feature Selection for Software Defect Prediction. Journal of Systems and Software. 2019, 152: 215-238.
26. Xiang Chen, Dun Zhang, Yingquan Zhao, Zhanqi Cui, Chao Ni. Software Defect Number Prediction: Unsupervised vs Supervised Methods. Information and Software Technology. 2019, 106: 161-181.
27. Xiaobing Sun, Wenyuan Xu, Xin Xia, Xiang Chen, Bin Li. Personalized project recommendation on GitHub. Science China: Information Sciences. 2018,61(5):050106:1-050106:14.
28. Yan Zheng, Zan Wang, Xiangyu Fan, Xiang Chen, Zijiang Yang. Localizing Multiple Faults based on Evolution Algorithm. Journal of Systems and Software, 2018,139:107-123.
29. Xiang Chen, Yingquan Zhao, Qiuping Wang, Zhidan Yuan. MULTI: Multi-Objective Effort-Aware Just-in-Time Software Defect Prediction. Information and Software Technology，2018,93:1-13. **(Invited to SANER 2019 as part of the Journal First Paper Track.)**
30. Chao Ni, Wangshu Liu, Xiang Chen, Qing Gu, Daoxu Chen, Qiguo Huang. A Cluster Based Feature Selection Method for Cross-Project Software Defect Prediction. Journal of Computer Science and Technology. 2017,32(6):1090-1107.
31. Xiaolin Ju, Shujuan Jiang, Xiang Chen, Xingya Wang, Yanmei Zhang, Heling Cao. HSFal: Effective Fault Localization using Hybrid Spectrum of Full Slices and Execution Slices. Journal of Systems and Software, 2014,90:3-17.

* **国内一级学报**

1. 吴欢欢，谢瑞麟，乔塬心，陈翔，崔展齐。基于可解释性分析的深度神经网络自动修复方法。计算机研究与发展。2023。已录用。
2. 陈翔，于池，杨光，濮雪莲，崔展齐。基于双重信息检索的Bash代码注释自动生成方法。软件学报。2022。已录用。
3. 李硕川，王赞，马明旭，陈翔，赵英全，王海弛，王昊宇。GC-MCR:基于有向图构建约束的并发缺陷检测方法。软件学报。2022。已录用。
4. 谢瑞麟，崔展齐，陈翔，郑丽伟。IATG：基于可解释性分析的自动驾驶软件测试数据生成方法。软件学报。2022。已录用。
5. 孙家泽，温苏雷，郑炜，陈翔。基于可攻击空间假设的陷阱式集成对抗防御网络。软件学报。2022。已录用。
6. 郑炜，刘程远，吴潇雪，陈翔，成婧源，孙小兵，孙瑞阳。基于知识图谱的跨项目安全缺陷报告预测方法。软件学报。2022。已录用。
7. 潘建文，崔展齐，林高毅，陈翔，郑丽伟。Android恶意应用静态检测方法研究综述。计算机研究与发展。2022。已录用。
8. 李伟湋，陈翔，张恒伟，黄志球，贾修一。一种基于同步语义对齐的异构缺陷预测方法。软件学报。2022。已录用。
9. 崔展齐\*，谢瑞麟，陈翔\*，刘秀磊，郑丽伟。DeepRager：覆盖制导的深度森林测试方法。软件学报。2021。已录用。
10. 李铮，崔展齐，陈翔，王荣存，刘建宾，郑丽伟。Deep-SBFL: 基于频谱的深度神经网络缺陷定位方法。软件学报。2021。已录用。
11. 沐燕舟，王赞，陈翔，陈俊洁，赵静珂，王建敏。采用多目标优化的深度学习测试优化方法。软件学报。2022，33（7）：2499-2524。
12. 郑炜，王晓龙，陈翔\*，夏鑫，廖慧玲，刘程远，孙瑞阳。重复软件缺陷报告检测方法综述。软件学报。2022，33（6）：2288-2311。
13. 郑炜，唐辉，陈翔\*，张满青，夏鑫。安卓移动应用兼容性测试研究综述。计算机研究与发展。2022，59（6）：1370-1387。
14. 杨慧文，崔展齐，陈翔，贾明华，郑丽伟，刘建宾。基于软件度量的Solidity智能合约缺陷预测方法。软件学报。2022，33（4）：1587-1611。
15. 李征，吴永豪，王海峰，陈翔\*，刘勇\*。软件多缺陷定位方法研究综述。计算机学报。2022，45（2）：256-288。
16. 陈翔，杨光，崔展齐，孟国柱，王赞。代码注释自动生成方法综述。软件学报。2021，32（7）：2118-2141。
17. 钟文康，葛季栋，陈翔，李传艺，唐泽，骆斌。面向神经机器翻译系统的多粒度蜕变测试。软件学报。软件学报。2021，32（4）：1051-1066。
18. 李政亮，陈翔\*，蒋智威，顾庆\*。基于信息检索的软件缺陷定位方法综述。软件学报。2021，32（2）：247-276。
19. 王赞，闫明，刘爽，陈俊洁，张栋迪，吴卓，陈翔。深度神经网络测试研究综述。软件学报。2020，31（5）：1255-1275。
20. 郑炜，陈军正，吴潇雪，陈翔，夏鑫。基于深度学习的安全缺陷报告预测方法实证研究。软件学报。2020，31（5）：1294-1313。
21. 陈翔，赵英全，顾庆，倪超，王赞。基于文件粒度的多目标软件缺陷预测方法实证研究。软件学报。2019，30（12）：3694-3713。
22. 谢肖飞，李晓红，陈翔，孟国柱，刘杨。基于符号执行与模糊测试的混合测试方法。软件学报。2019，30（10）：3071-3089。
23. 倪超，陈翔\*，刘望舒，顾庆\*，黄启国，李娜。基于特征迁移和实例迁移的跨项目缺陷预测方法。软件学报。2019，30（5）：1308-1329。
24. 王赞，郜健，陈翔\*，傅浩杰，樊向宇。自动程序修复方法研究述评。计算机学报，2018,41（3）：588-610。
25. 刘望舒，陈翔，顾庆，刘树龙，陈道蓄。一种面向软件缺陷预测的可容忍噪声的特征选择框架。计算机学报，2018,41（3）：506-520。
26. 陈翔，王莉萍，顾庆，王赞，倪超，刘望舒，王秋萍。跨项目软件缺陷预测方法研究综述。计算机学报，2018，41（1）：254-274。
27. 何吉元，孟昭鹏，陈翔，王赞，樊向宇。一种半监督集成跨项目软件缺陷预测方法。软件学报，2017,28（6）：1455-1473。
28. 刘望舒，陈翔，顾庆，刘树龙，陈道蓄。软件缺陷预测中基于聚类分析的特征选择方法。中国科学：信息科学，2016，46（9）：1298–1320。
29. 王赞，樊向宇，邹雨果，陈翔。一种基于遗传算法的多缺陷定位方法。软件学报，2016，27（4）： 879-900。
30. 【🏆《软件学报》2018年高影响力论文】陈翔，顾庆，刘望舒，刘树龙，倪超。静态软件缺陷预测方法研究。软件学报，2016，27（1）：1-25。
31. 陈翔，鞠小林，文万志，顾庆。基于程序频谱的动态缺陷定位方法研究。软件学报，2015，26（2）：390-412。
32. 陈翔，陈继红，鞠小林，顾庆。 回归测试中的测试用例优先排序技术述评。软件学报，2013，24(8)：1695-1712。
33. 鞠小林，姜淑娟，陈翔，张艳梅，邵浩然。基于切片谱的错误定位框架影响因素分析。计算机研究与发展，2014，51(12)：2772-2787。

**（三）其他需要说明的情况**

1. 申请人同年申请不同类型的国家自然科学基金项目情况（列明同年申请的其他项目的项目类型、项目名称信息，并说明与本项目之间的区别与联系）。

无。

2. 具有高级专业技术职务（职称）的申请人或者主要参与者是否存在同年申请或者参与申请国家自然科学基金项目的单位不一致的情况；如存在上述情况，列明所涉及人员的姓名，申请或参与申请的其他项目的项目类型、项目名称、单位名称、上述人员在该项目中是申请人还是参与者，并说明单位不一致原因。

无。

3. 具有高级专业技术职务（职称）的申请人或者主要参与者是否存在与正在承担的国家自然科学基金项目的单位不一致的情况；如存在上述情况，列明所涉及人员的姓名，正在承担项目的批准号、项目类型、项目名称、单位名称、起止年月，并说明单位不一致原因。

无。

4. 其他。

无。