함수 이름 추론 기법 연구 동향 분석

이상민[1], 조영필[2]
[1]한양대학교 컴퓨터소프트웨어학과(미래자동차-SW융합전공) 석박통합과정
[2]한양대학교 컴퓨터소프트웨어학과 교수
ozoesm@hanyang.ac.kr, ypcho@hanyang.ac.kr

Research Trends in Function Name Inference Techniques

Sang-min Lee[1], Yeong-pil Cho[2]
[1]Dept. of Computer Science (Automotive-Computer Convergence), HanYang University
[2]Dept. of Computer Science, HanYang University

요 약

바이너리 분석과 역공학에서 함수 이름은 코드 의미 이해와 보안 분석의 핵심 단서이다. 그러나 많은 소프트웨어 배포판은 디버그 심볼이 제거되어 함수명이 소실된다. 이를 복원하기 위한 함수 이름 추론(Function Name Inference) 기술은 초기의 통계 기반 방법에서 출발하여, 그래프·시퀀스 신경망, 멀티태스크 학습, 최근에는 대규모 언어모델(LLM) 기반 접근으로 발전하였다. 본 논문은 2020년 이후의 주요 연구들을 정리하여 기술적 배경, 기법별 특징, 한계와 향후 연구 과제를 제시한다.

1. 서론

함수 이름은 프로그램의 의미와 구조를 이해하는 데 중요한 단서로, 역공학과 보안 분석 과정에서 핵심적인 역할을 한다. 그러나 상용 소프트웨어는 디버그 심볼이 제거된 상태로 배포되는 경우가 많아 함수명이 소실되며, 이로 인해 분석자는 코드의 의도를 파악하기 어렵다. 이러한 문제를 해결하기 위해 함수 이름을 자동으로 복원하거나 새롭게 생성하는 다양한 연구가 이루어지고 있다.

함수 이름 추론 연구는 초기의 통계 기반 접근에서 출발하여, 제어 흐름과 데이터 흐름을 반영한 임베딩 기법, 서브토큰화 및 멀티태스크 학습, 그리고 최근에는 대규모 언어모델(LLM)을 적용한 방식으로 발전하고 있다. 이러한 기법들은 각각의 장단점을 지니며, 최적화 수준, 아키텍처 차이, 난독화 기법 등 다양한 변수에 따라 성능이 달라질 수 있다.

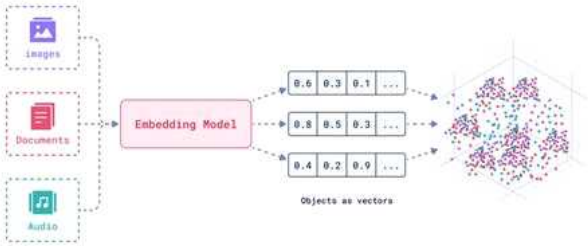
이에 본 논문은 함수 이름 추론 기법을 유형별로 분류하여 살펴보고, 최근의 연구 동향을 분석함으로써 앞으로의 관련 연구의 지속적인 필요성을 제시한다.

2. 함수 이름 추론 모델

함수 이름 추론은 디버그 심볼이 제거된 바이너리나 최적화된 코드에서 함수의 의미를 복원하기 위한 기법으로, 여러 접근 방식이 제안되어 왔다. 대표적으로 토큰·통계 기반, 그래프·임베딩 기반, 서브토큰·멀티태스크 및 요약 기반, 대규모 언어모델(LLM) 기반으로 구분할 수 있다.

2.1 토큰·통계 기반 접근

토큰·통계 기반 접근은 코드 내 토큰 분포나 호출 맥락을 분석하여 함수 이름을 예측하는 방식이다. 초기 연구인 Probabilistic Naming[1]은 단순성과 효율성에서 장점을 보였으나, 컴파일러 최적화나 난독화 환경에서는 성능이 저하되었다.

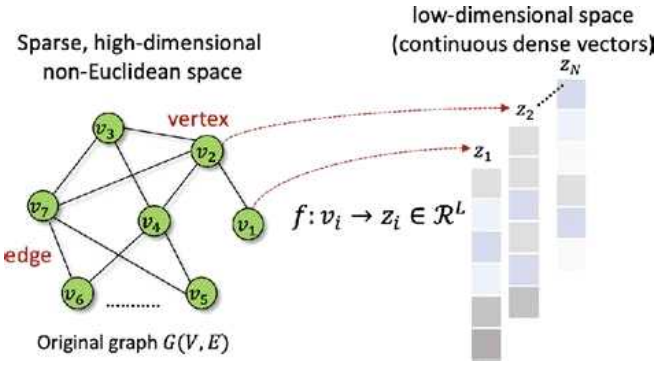


(그림 1) 토큰 기반 접근 개념도

2.2 그래프·임베딩 기반 접근

그래프·임베딩 기반 접근은 제어 흐름 그래프

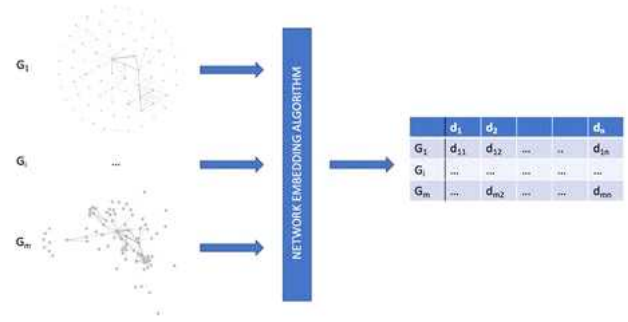
(CFG)나 데이터 흐름 그래프(DFG)를 벡터 공간에 임베딩하여 함수 의미를 학습하는 방식이다. SymLM[2]은 호출 관계와 실행 맥락을 결합해 높은 정확도를 달성하였고, Function Embedding Alignment[3] 역시 CFG·DFG를 정렬해 함수 의미를 효과적으로 반영하였다.



(그림 2) 그래프 임베딩 기반 함수 의미 표현

2.3 서브토큰·멀티태스크 및 요약 기반

Epitome[4]는 함수명을 서브토큰 단위로 분해하여 예측 정확도를 높이고, 멀티태스크 학습을 통해 품질을 향상시켰다. Bin2Summary[5]는 함수 요약을 먼저 생성한 뒤 이를 기반으로 이름을 도출하는 접근을 제안해, 단순한 이름 예측을 넘어 가독성까지 개선하였다.



(그림3) 함수 요약 기반 이름 생성 예시

2.4 대규모 언어모델(LLM) 기반 적응

가장 최신 연구들은 사전학습된 LLM을 바이너리 분석 도메인에 적응시킨다. Jiang et al.[6]는 다양한 컴파일러와 최적화 조건에서 높은 일반화 성능을 보였으며, BLens[7]는 함수 캡션 생성과 대조학습을 결합해 의미 기반 이름 복원을 가능하게 하였다.

3. 함수 이름 추론 연구 동향

함수 이름 추론 연구는 지난 수년간 다양한 방향으로 발전해왔다. 초기 연구인 Probabilistic Naming[1]은 코드 내 토큰 빈도와 호출 맥락을 이용한 확률적 접근으로, 단순성과 경량성 측면에서 유용했으나 최적화나 난독화 환경에서는 성능이 크게 저하되는 한계가 있었다. 이러한 초기 연구는 이후 복잡한 구조적 특징을 반영하는 연구의 출발점이 되었다.

그래프 임베딩을 활용한 연구들은 구조적 의미를 반영하기 위해 제어 흐름 그래프(CFG), 데이터 흐름 그래프(DFG) 등을 사용하였다. SymLM[2]은 호출 관계와 실행 맥락을 동시에 반영하여 높은 Top-k 정확도를 달성하였으며, Function Embedding Alignment[3]는 함수 간 의미적 유사도를 벡터 공간에 정렬함으로써 대규모 코드베이스 분석에도 강점을 보였다. 하지만 이들 연구는 대규모 학습 데이터와 높은 연산 비용이 요구된다는 점에서 현실적 한계가 존재하였다.

최근 연구들은 이러한 문제를 해결하기 위해 새로운 접근을 시도하였다. Epitome[4]는 함수명을 서브토큰 단위로 분해하고 멀티태스크 학습을 적용하여 OOV(Out-Of-Vocabulary) 문제를 완화하였으며, Bin2Summary[5]는 함수 요약을 먼저 생성한 뒤 이를 바탕으로 이름을 도출하는 방식을 도입하여 단순한 이름 복원에서 나아가 분석자의 이해를 지원하는 방향으로 발전하였다. 이 과정에서 함수 이름 추론 연구는 점차 보안 분석 및 역공학 실무에 직접적인 도움을 줄 수 있는 방향으로 확장되었다.

가장 최근의 동향은 대규모 언어모델(LLM)을 활용하는 연구로 요약된다. Jiang et al.[6]은 사전학습된 LLM을 바이너리 도메인에 적응시켜 다양한 컴파일러와 최적화 조건에서도 높은 일반화 성능을 달성하였다. 또한 BLens[7]는 함수 캡션 생성과 대조학습을 결합하여 단순히 이름을 복원하는 것을 넘어 코드 의미와 연계된 이름 생성이 가능함을 보여주었다. 이러한 연구들은 LLM의 표현력을 활용하여 함수 의미를 보다 정밀하게 추론할 수 있는 가능성을 제시했지만, 동시에 환각(hallucination) 문제와 높은 학습 비용, 학습 데이터 편향성이라는 과제를 남기고 있다.

결과적으로 함수 이름 추론 연구는 단순한 이름 복원에서 시작하여 구조적 의미 반영, 가독성 및 분석

지원, 의미 보존과 일반화로 발전해왔다. 이는 곧 함수 이름 추론 기술이 점점 더 실제 소프트웨어 보안 분석과 역공학 과정에서 실질적 가치를 제공할 수 있는 단계로 나아가고 있음을 의미한다.

4. 결론

본 논문에서는 함수 이름 추론 기법의 발전 과정과 최근 연구 동향을 분석하였다. 초기의 통계 기반 연구[1]는 단순성과 효율성에서 의의를 가지나 환경 변화에 취약했으며, 그래프 임베딩 기반 연구[2][3]는 구조적 의미 반영을 통해 성능을 향상시켰으나 데이터와 자원 요구가 크다는 한계를 보였다. 최근의 연구들은 서브토큰화 및 멀티태스크 학습[4], 함수 요약 기반 접근[5], 대규모 언어모델을 활용한 적응형 학습[6][7]을 통해 이러한 한계를 극복하고자 시도하였으며, 함수 이름 추론 기술은 점차 실질적인 분석 지원 도구로 발전하고 있다.
연구 동향을 종합하면, 함수 이름 추론은 코드 의미 보존, 다양한 환경에서의 강건성, 분석자의 이해 지원이라는 세 가지 핵심 방향으로 발전하고 있다. 향후 연구에서는 (1) 최적화 및 난독화된 코드에 대응할 수 있는 강건한 모델 설계, (2) LLM 환각 문제를 줄이기 위한 제약 기반 검증 기법, (3) 함수 이름 추론과 요약, 보안 분석을 통합하는 다목적 파이프라인, (4) 대규모 공개 벤치마크 데이터셋 구축이 중요한 과제가 될 것이다. 이러한 연구들이 지속적으로 이루어진다면, 함수 이름 추론 기술은 역공학, 악성코드 분석, 소프트웨어 보안 등 다양한 분야에서 실질적이고 신뢰할 수 있는 도구로 자리매김할 수 있을 것이다.

이 논문은 과학기술정보통신부의 재원으로 정보통신기술기획평가원(IITP)의 (연구과제번호 RS-2024-00337414, SW공급망 운영환경에서 역공학 한계를 넘어서는 자동화된 마이크로 보안 패치 기술 개발) 지원을 받아 수행된 연구임.

참고문헌

[1] Patrick-Evans, A., et al., “Probabilistic Naming of Functions in Stripped Binaries,” Annual Computer Security Applications Conference (ACSAC), Austin, USA, 2020, pp. 321–332.

[2] Jin, X., et al., “SymLM: Predicting Function Names in Stripped Binaries via Context-Sensitive Execution-Aware Code Embeddings,” Proceedings of the ACM SIGSAC Conference on Computer and Communications Security (CCS), Los Angeles, USA, 2022, pp. 612–626.

[3] Zhang, X., et al., “Function Embedding Alignment for Binary Code Understanding,” USENIX Security Symposium, Vancouver, Canada, 2021, pp. 1201–1217.

[4] Wang, Y., et al., “Epitome: Enhancing Function Name Prediction using Votes-Based Name Tokenization and Multi-Task Learning,” Proceedings of the ACM Conference, San Francisco, USA, 2024, pp. 130–142.

[5] Li, Z., et al., “Bin2Summary: Beyond Function Name Prediction in Stripped Binaries with Natural-Language Summaries,” ACM Transactions on Software Engineering and Methodology (TOSEM), vol. 33, no. 2, 2024, pp. 1–27.

[6] Jiang, H., et al., “Inferring Function Names in Stripped Binaries via Domain Adaptation Leveraging Pre-Trained Generative LLMs,” Network and Distributed System Security Symposium (NDSS), San Diego, USA, 2025, pp. 1–15.

[7] Benoit, C., et al., “BLens: Contrastive Captioning of Binary Functions using LLMs,” USENIX Security Symposium, San Francisco, USA, 2025, pp. 801–815.