CodeBERT 기반 소스코드 멀티라벨 취약점 탐지

김건민[1], 이은성[1], 정민수[1], 오다영[2], 이서준[2], 장현지[2], 최광훈[3], 김경백[4]
[1]전남대학교 소프트웨어공학과 학부생
[2]전남대학교 인공지능학부 학부생
[3]전남대학교 정보보안융합학과 교수
[4]전남대학교 인공지능융합학과 교수
204869@jnu.ac.kr, 200750@jnu.ac.kr, 205306@jnu.ac.kr,
215430@jnu.ac.kr, tjwns1300@jnu.ac.kr, gka1225@jnu.ac.kr,
kwanghoon.choi@jnu.ac.kr, kyungbaekkim@jnu.ac.kr

Multi-Label Vulnerability Detection in Source Code Using CodeBERT

Geonmin Kim[1], Eunseong Lee[1], Minsoo Jeung[1], Dayoung Oh[2], Seojun Lee[2],
Hyeonji Jang[2], Kwanghoon Choi[3], Kyungbaek Kim[4]
[1]Dept. of Software Engineering, Chonnam National University
[2]Dept. of Artificial Intelligence, Chonnam National University
[3]Dept. of Information Security Convergence, Chonnam National University
[4]Dept. of Artificial Intelligence Convergence, Chonnam National University

요 약

현대 소프트웨어 개발 환경은 하나의 코드 스니펫이 다수의 보안 취약점에 동시에 노출되는 위협에 직면해 있다. 본 연구는 이러한 다중 취약점 환경을 반영한 멀티라벨 분류 전략을 기반으로, 사전학습 언어 모델인 CodeBERT의 취약점 인식 성능을 정량적으로 분석한다. SARD 데이터셋을 활용하여 6가지 CWE 유형에 대해 CodeBERT의 정밀도, 재현율, Threshold 변화에 따른 탐지 트레이드오프 등을 고찰한다. 또한, CWE 취약점 유형에 따른 탐지 성능 차이를 논한다.

1. 서론

소프트웨어 보안의 핵심 과제 중 하나는 코드 수준에서 발생할 수 있는 잠재적 보안 취약점을 사전에 식별하고 이를 체계적으로 완화하는 데 있다. 기존의 취약점 탐지 방식은 정적 분석 도구 또는 수작업 검토에 의존해 왔으나 이는 높은 오탐률, 낮은 재현성 그리고 새로운 유형의 취약점에 대한 일반화 성능 부족이라는 구조적 한계를 내포한다[1].

이러한 배경에서 최근 자연어 처리 분야에서 비약적인 성과를 보인 트랜스포머 기반의 사전 학습 언어모델이 코드 분석 영역으로 확장되며 주목받고 있다. 특히, CodeBERT[2], GraphCodeBERT와 같은 모델들은 코드의 구문적, 문맥적 구조를 효과적으로 학습할 수 있는 가능성을 보여주고 있다. 이들은 코드 내 변수 선언, 함수 호출, 포인터 참조 등 복합적인 구조에 대한 추론 능력을 갖추고 있다.

실제 시스템에서는 하나의 코드 스니펫에 복수의 보안 취약점이 혼재할 수 있으며, 이러한 상황을 반영한 탐지 모델의 설계는 현실적인 보안 위협 대응에 필수적이다. 본 연구는 이러한 다중 보안 위협 중첩 문제를 효과적으로 다루기 위해 CodeBERT 기반의 멀티라벨 분류 기반 보안 취약점 탐지기를 설계하고, 탐지 성능을 정량적으로 평가한다.

2. 이론적 배경 및 관련 연구

2.1 CWE(Common Weakness Enumeration)

CWE[3]는 MITRE에서 관리하는 보안 취약점 분류 체계로 소프트웨어 설계, 구현 상의 결함을 유형별로 분류한 공개된 사전이다. CWE는 취약점 유형의 식별, 보고 및 교차 비교를 가능하게 하며, 실제 소스코드 상에서는 이러한 CWE 유형이 동시에 존재하거나 인과 관계를 가질 수 있어 여러 CWE 유형을 병렬적으로 예측하는 멀티라벨 분류가 요구된다. 멀티라벨 분류란 하나의 입력에 대해 복수의 라벨을 예측하는 문제로, 일반적인 Softmax 기반의 단일 클래스 분류와 달리 Sigmoid 함수를 기반으로 각 라벨의 확률을 독립적으로 계산하며 일정 Threshold를 초과할 경우 탐지로 간주한다.

2.2 CodeBERT

CodeBERT는 코드 전용 사전학습 언어모델로, 자연어와 프로그래밍 언어의 양방향 표현 학습을 목표로 설계된 Bimodal 트랜스포머 모델이다. 이 모델은 대규모 코드 및 주석 데이터를 기반으로 학습되었으며, 코드-자연어 간 일치 판단, 코드 요약, 취약점 탐지 등에 효과적으로 활용된다.

3. CWE 별 멀티라벨 분류 실험 설정 및 방법

본 연구는 실험의 신뢰성과 반복 가능성을 확보하기 위하여 SARD(Software Assurance Reference Dataset)의 C언어 기반 취약 코드 스니펫을 사용하였다. CWE-78(OS Command Injection), CWE-134(Uncontrolled Format String), CWE-190(Integer Overflow), CWE-400(Resource Exhaustion), CWE-416(Use After Free), CWE-476(Null Pointer Dereference) 총 6개 유형의 CWE 유형을 선별하였으며, 각 코드 스니펫은 최대 512토큰 단위로 슬라이딩 윈도우(Stride=256) 방식으로 분할되었다. 모델은 neulab/codebert-c 기반으로 설정되었으며, 샘플 수가 적은 클래스를 보강하기 위하여 클래스별로 다른 가중치를 적용하여 데이터 불균형을 보정하였다. Threshold는 각각 0.3, 0.6 두 가지 실험을 진행하였다.

4. 실험 결과

<표 1> Threshold 별 CWE 멀티라벨 분류 실험 결과

Threhold	Accuracy	Precision	Recall	F1
0.3	0.8115	0.8340	1.000	0.9069
0.6	0.8857	0.9221	0.9084	0.9046

<표 2> CWE 별 분류 실험 결과(W: 가중치 적용)

	Accuracy	Precision	Recall	F1
CWE-78	1.000	1.000	1.000	1.000
CWE-78(W)	1.000	1.000	1.000	1.000
CWE-134	0.5271	0.9885	0.5192	0.6765
CWE-134(W)	0.9537	0.7126	0.9917	0.8633
CWE-190	0.8514	0.9983	0.8503	0.9184
CWE-190(W)	0.9735	0.8685	1.000	0.9296
CWE-400	0.7054	0.9468	0.7071	0.8083
CWE-400(W)	0.9910	0.7595	1.000	0.8633
CWE-416	0.5385	0.6786	0.5298	0.5846
CWE-416(W)	0.9976	0.8000	1.000	0.8889
CWE-476	0.9531	0.9688	0.9536	0.9606
CWE-476(W)	0.9990	0.8824	1.000	0.9375

실험 결과, Threshold가 낮을수록 재현율은 극대화되나 정밀도는 상대적으로 낮아지는 현상이 발생하였다. 또한, Threshold가 높아지면 탐지의 보수성이 증가하여 정밀도는 증가하고 재현율이 하락하였다. 또한, CWE 유형에 따라 탐지 정확도가 상이하게 나타났다. 이는 CWE-78, CWE-190과 같은 명시적 패턴 기반 취약점에 비해, CWE-134, 400, 416 등은 제어 흐름과 자원 해제 시점의 정밀한 이해를 요구함에 따라 낮은 탐지율을 보인 것으로 추정된다. 다만, 이들에 대하여 가중치를 통하여 탐지 성능을 끌어올릴 수 있었다.

5. 결론 및 향후 연구

본 논문은 코드 전용 사전 학습 언어모델인 CodeBERT를 기반으로 다중 CWE 유형에 대한 멀티라벨 분류 기반 보안 취약점 탐지기를 설계하고, 이를 SARD 데이터셋 내 주요 CWE 유형에 대하여 실험적으로 분석하였다. 실험 결과 본 모델은 다중 CWE 유형이 동시에 존재하는 복합적인 취약 코드 시나리오에 효과적인 탐지 성능을 보였다. 또한, Threshold 조정과 클래스 가중치 적용이 탐지 성능 최적화에 유효함을 확인하였다. 특히, CWE 유형에 따라 탐지 정확도가 상이하게 나타나는 현상을 분석함으로써, 탐지 성능이 단순한 문법적 특징보다 시맨틱 복잡도 및 컨텍스트 종속성에 크게 영향을 받는다는 점을 실증적으로 확인하였다. 향후 연구에서는 코드 구조 정보를 학습할 수 있는 GNN 기반 모델이나 AST 인코딩 기법을 CodeBERT와 결합하는 하이브리드아키텍처로의 확장 및 시맨틱 기반 데이터 증강 기법 도입을 통한 모델의 일반화 능력 향상 등에 대한 연구가 요구된다.

Acknowledgement

본 연구는 한국인터넷진흥원(KISA)-정보보안 특성화대학 지원사업의 지원을 받아 수행된 연구임 (34%). 이 논문은 정부(과학기술정보통신부)의 재원으로 정보통신기획평가원-지역지능화혁신인재양성사업의 지원을 받아 수행된 연구임(IITP-2025-RS-2022-00156287, 33%). 본 연구는 과학기술정보통신부 및 정보통신기획평가원의 인공지능융합혁신인재양성사업 연구 결과로 수행되었음 (IITP-2023-RS-2023-00256629, 33%)

참고문헌

[1] Seshagiri, Prabhu, Anu Vazhayil, and Padmamala Sriram. "AMA: static code analysis of web page for the detection of malicious scripts."*Procedia Computer Science*93 (2016): 768-773.

[2] K. Zhao, S. Duan, G. Qiu, J. Zhai, M. Li and L. Liu, "Python Source Code Vulnerability Detection Based on CodeBERT Language Model," 2024 7th International Conference on Algorithms, Computing and Artificial Intelligence (ACAI), Guangzhou, China, 2024, pp. 1-6

[3] M. Howard, "Improving Software Security by Eliminating the CWE Top 25 Vulnerabilities," in IEEE Security & Privacy, vol. 7, no. 3, pp. 68-71, May-June 2009