

2023 한국스마트미디어학회 종합학술대회 2023 Conference of KISM

Smart
Media

- 일시 4. 27(목)~ 4. 29(토)
- 장소 제주대학교 아라컨벤션홀
- 주최 (사)한국스마트미디어학회
- 주관 제주대학교 SW중심대학사업단
전남대학교 SW중심대학사업단
원광대학교 SW중심대학사업단



Smart Media
KOREAN INSTITUTE OF SMART MEDIA
한국스마트미디어학회

구두 발표 O3 : Smart Media, Smart Energy ICT, Smart Information

시간 : 4월 28일 (금) 09:00-12:00

발표장 : 세미나실 III

좌장 : 김경백(전남대)

O3-01	제목: Improving Human Parsing with Self-correction and Instance-aware Edge Detection 저자: Muhammad Yasir, 최창(가천대)
O3-02	제목: Word2Vec와 FastText를 이용한 한국어 워드 임베딩 비교 분석 저자: 전석원, 장준혁(한남대)
O3-03	제목: 생성적 적대 신경망을 활용한 비지도 도메인 적응 개선 저자: 카리나 바알치메그, 이재홍(대전대)
O3-04	제목: BMC 커널 부팅 시간 단축 방법 저자: 홍지만, 박민호, 황대웅(숭실대)
O3-05	제목: Bad Code 가시화 및 인공지능 학습을 위한 고품질 데이터 수집 툴 체인 구축 저자: 박찬술, 장우성, 김영철(홍익대)
O3-06	제목: 이미지기반 차량 파손 탐지 AI 모델 저자: 이대호, 박정훈, 변선오(아주대)
O3-07	제목: 별-은하 이미지의 자동 레이블링을 위한 YOLOv5 기반 객체 탐지 연구 저자: 이민규, 김준영, 박성욱, 박준, 정세훈, 심춘보(순천대)
O3-08	제목: 5G 초기인증의 순방향 비밀성 지원을 위한 5G-IPAKA 보안 프로토콜 취약점 분석 저자: 김건우(국민대), 박훈용(순천향대), 권호석, 김보남, 유일선(국민대)
O3-09	제목: 음성 및 얼굴기반 감정인식에 대한 연구 저자: 이원행, 강성현, 박정훈(아주대)
O3-10	제목: YOLOv6 및 PoseNet 기반 공원 내 불법행위 단속 시스템 설계 저자: 박정민, 김하연, 김준영, 심춘보, 정세훈(순천대)

P2-35	제목: LED광촉매 기반 유해오염 화합물질 제거에 관한 연구 저자: 홍성한(원광대)
P2-36	제목: LED광촉매 필터 종류에 따른 유해가스제거 연구 저자: 홍성한(원광대)
P2-37	제목: UV-LED 반사면 재질에 따른 반사율 성능 비교 저자: 김병철(원광대)
P2-38	제목: UV-LED 열확산계수에 따른 열전도도 비교분석 저자: 김병철(원광대)
P2-39	제목: 홀로렌즈를 이용한 전시 콘텐츠의 인터랙션 처리 및 인식 기술 연구 저자: 김동조(순천대)
P2-40	제목: 인스턴스 세그멘테이션을 위한 연속학습 저자: Heng Zhou, Sang-cheol Kim(전북대)
P2-41	제목: 사용자 골격 동적추적 기반 상호작용 미디어아트 개발 저자: 박대원((주)다원테크), 오태원(호남대)
P2-42	제목: 홀로렌즈 상에 균열 영역과 설계 도면 매핑을 위한 AR 기반 메커니즘 저자: 조재형, 박찬술, 홍영우, 장우성, 김영철(홍익대)
P2-43	제목: AutoEncoder를 활용한 섬유 드레이프 이미지 생성 모델 구현에 관한 연구 저자: 표규진, 김동현, 최윤성(다이텍연구원)
P2-44	제목: AI-MoveNet 모델 및 모션 인지 기반의 비정상 자세 검출 추론 시스템 저자: 김형진(전북대), 송현우(편웨이브)

코드 가시화 기반의 Bad Code 학습을 위한 고품질 데이터 식별 및 수집 튜체인 화

박찬술^{1*}, 장우성², 김영철³

홍익대학교 소프트웨어융합학과 소프트웨어공학 연구실

e-mail : c2193102@g.hongik.ac.kr¹, {uriel200², bob³}@hongik.ac.kr

Tool Chain Mechanism with Identifying and Collecting High Quality Data for Learning Bad Code based on Code Visualization

Chansol Park, Woo Sung Jang, R. Young Chul Kim

SE Lab, Dept. of Software and Communications Engineering, Hongik
University

요 약

현재 전 세계에서 인공지능과 소프트웨어 공학을 접목하는 연구를 진행하고 있다. 인공지능이 Bad Code의 패턴을 학습하고 코드로부터 Bad Code 영역(복잡 모듈, 취약 모듈, Bad Smell 모듈 등)을 식별하는 것이 필요하다. 이를 위해 CWE 취약점에 적용했다. 문제는 인공지능에 Bad Code 패턴 학습을 위해 필요한 고품질의 많은 코드 데이터가 매우 부족하다. 기존 방법은 오픈 소스 취약점 검출 도구를 이용해 Bad Code 데이터 수집했지만, 데이터의 정확성이 보장할 수 없다. 이를 해결하기 위해 코드 가시화를 통해 복잡한 모듈 식별 및 해당 모듈의 취약점 데이터를 수집하는 튜체인을 제안한다. 이를 통해 기존 Bad Code 검출 도구만을 이용해 데이터를 수집하는 경우보다 높은 정확도의 Bad Code 검출 데이터 수집할 수 있다. 이렇게 수집된 데이터셋을 통해 인공지능의 정확한 Bad Code 학습이 기대된다.

키워드: 소프트웨어 공학, 코드 가시화, 코드 복잡도, 코드 취약점, 인공지능

1. 서 론

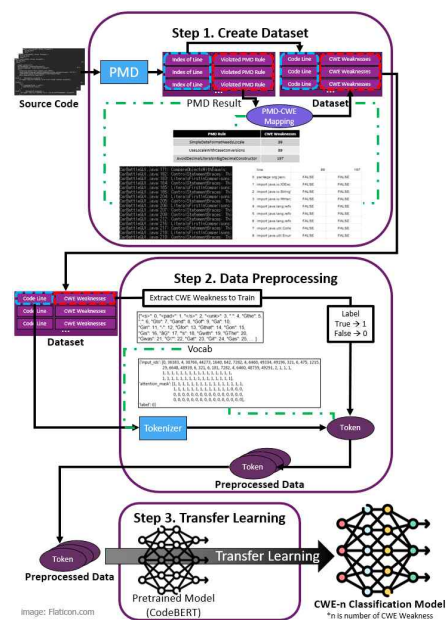
현재 인공지능과 소프트웨어 공학을 접목하려는 연구가 활발하다[1]. 세부적으로 인공지능을 위한 소프트웨어 공학 분야와 소프트웨어 공학을 위한 인공지능 분야로 나눌 수 있다. 그중 소프트웨어 공학에 인공지능을 접목해 패턴을 스스로 학습하여 Bad Code를 식별하는 연구를 진행 중이다. 인공지능 모델의 식별 정확도를 향상하기 위해서는 다량의 고품질 데이터가 필요하다. 하지만 인공지능을 학습하기 위한 Bad Code 데이터는 턱없이 부족한 상황이다. 따라서 고품질의 데이터를 수집하기 위해 기존 코드 가시화 도구를 이용한다. 코드 가시화 도구를 통해 식별한 Bad Code 요소를 종합하여 정확도가 높은 Bad Code 데이터를 수집한다. 수집된 데이터를 통해 성능이 좋은 인공지능 모델을 학습시킬 수 있을 것을 기대한다.

2. 관련 연구

2.1. Bad Code 패턴의 지도학습을 통한 Bad Code 식별 적용 사례

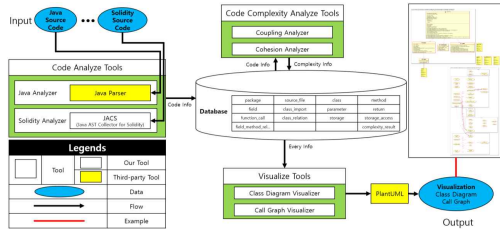
그림 1은 CodeBERT 모델에 대한 Bad Code 패턴 전이 학습 과정의 구조도이다[2]. 첫 번째 단계에서는 PMD를 통해 코드 라인의 취약점을 검출한다. 검출한 PMD 취약점을 PMD 취약점과 Common Weakness Enumeration(CWE) 취약점 간 매핑 테이블을 이용해 CWE 취약점으로 변환한다. 각 코드 라인에 대해 CWE 취약점 목록 검출 여부를 True/False로 표기하고,

Comma-Separated Values(CSV) 파일로 저장한다. 두 번째 단계에서는 코드 라인을 CodeBERT 모델에 입력할 수 있도록 토큰화한다. 토큰화된 라인에 CodeBERT 모델에 학습할 CWE 취약점 항목의 검출 여부를 1과 0으로 라벨링 한다. 세 번째 단계에서는 두 번째 단계에서 생성된 토큰들을 통해 CodeBERT 모델을 전이 학습한다.



(그림 1) CodeBERT 모델에 대한 Bad Code
패턴 전이 학습 과정

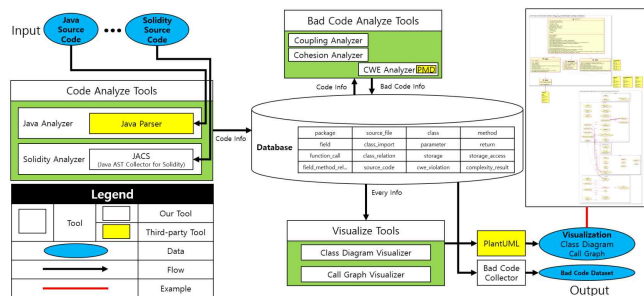
2.2. 코드 가시화 연구



(그림 2) 기존 코드 가시화 도구 구조도

그림 2는 연구 중인 코드 가시화 도구의 구조도이다[3]. 도구는 세부적으로 세 종류의 도구와 데이터베이스로 나눌 수 있다. 데이터베이스는 코드 가시화 공정 중 도구 간 공유되어야 하는 데이터를 수집 및 배포한다. 코드 분석 도구는 주어진 객체 지향 소스 코드를 분석하여 데이터베이스에 코드 내부 정보를 저장한다. 코드 복잡도 분석 도구는 데이터베이스로부터 코드 내부 정보를 불러와 복잡도를 계산한다. 가시화 도구는 데이터베이스의 정보를 통해 다이어그램을 도식한다.

3. Bad Code 가시화 및 인공지능 학습을 위한 고품질 데이터 수집 틀체인 구축



(그림 3) 코드 가시화 및 데이터 수집 도구 구조도

그림 3은 CWE 위반 요소 검출 도구와 Bad Code 데이터 수집 도구가 추가된 코드 가시화 도구의 구조도이다. CWE 위반 요소 검출 도구는 PMD 도구를 이용하여 코드 라인의 PMD 취약점을 검출한다. 검출된 PMD 취약점 항목을 PMD 취약점 요소와 CWE 취약점 요소 간 매핑 테이블을 이용해 CWE 취약점으로 변환한다. 변환된 CWE 취약점 ID를 CWE 취약점 테이블에 취약점이 발견된 코드 라인 ID와 함께 저장한다.

Bad Code 데이터 수집 도구는 소스 코드 분석 결과를 학습할 수 있도록 수집한다. 기존 연구에서 모델을 학습하기 위해 사용한 Bad Code 데이터셋은 PMD 도구만을 이용하여 수집했다. 이러한 방법은 PMD 도구가 오탐지한 취약점을 그대로 학습한다는 문제가 있다. 학습 데이터의 정확도를 높이기 위해 복잡도를 반영하여 복잡도가 높은 소스 코드의 코드 라인과 CWE 취약점 데이터를 수집한다. 수집한 데이터는 JavaScript Object Notation(JSON) 구조로 저장된다.

4. 적용 사례

그림 4는 데이터베이스에 수집된 소스 코드와 CWE 취약점 검출 라인이다. 그림 4의 A는 소스 코드 테이블에

저장된 정보 중 일부이다. 그림 4의 B는 검출된 CWE 취약점 테이블에 저장된 정보 중 일부이다. 그림 4에 따르면 301번째 코드 라인에서 수 가지의 CWE 취약점이 검출되었다.

그림 5는 수집된 CWE 취약점 검출 데이터 중 일부를 추출한 것이다. 그림 5의 A는 수집된 소스 코드 데이터에 관한 정보이다. 복잡도, 패키지 이름, 클래스 이름이 기록된다. 그림 5의 B는 취약점이 검출된 코드 라인 정보의 예시이다. 코드 라인, 코드의 라인 번호, 검출된 취약점 정보가 기록된다.

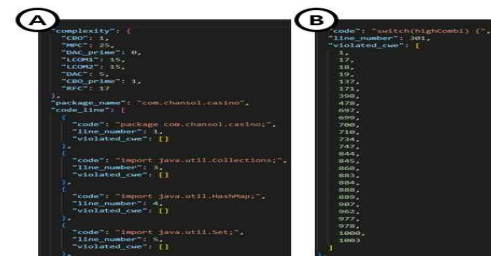
id	source_id	time_number	code_line
1	1	1	1
2	2	2	2
3	3	3	3
4	4	4	4
5	5	5	5
6	6	6	6
7	7	7	7
8	8	8	8
9	9	9	9
10	10	10	10
11	11	11	11
12	12	12	12
13	13	13	13
14	14	14	14
15	15	15	15
16	16	16	16
17	17	17	17
18	18	18	18
19	19	19	19
20	20	20	20
21	21	21	21
22	22	22	22
23	23	23	23
24	24	24	24
25	25	25	25
26	26	26	26
27	27	27	27
28	28	28	28
29	29	29	29
30	30	30	30
31	31	31	31
32	32	32	32
33	33	33	33
34	34	34	34
35	35	35	35
36	36	36	36
37	37	37	37
38	38	38	38
39	39	39	39
40	40	40	40
41	41	41	41
42	42	42	42
43	43	43	43
44	44	44	44
45	45	45	45
46	46	46	46
47	47	47	47
48	48	48	48
49	49	49	49
50	50	50	50
51	51	51	51
52	52	52	52
53	53	53	53
54	54	54	54
55	55	55	55
56	56	56	56
57	57	57	57
58	58	58	58
59	59	59	59
60	60	60	60
61	61	61	61
62	62	62	62
63	63	63	63
64	64	64	64
65	65	65	65
66	66	66	66
67	67	67	67
68	68	68	68
69	69	69	69
70	70	70	70
71	71	71	71
72	72	72	72
73	73	73	73
74	74	74	74
75	75	75	75
76	76	76	76
77	77	77	77
78	78	78	78
79	79	79	79
80	80	80	80
81	81	81	81
82	82	82	82
83	83	83	83
84	84	84	84
85	85	85	85
86	86	86	86
87	87	87	87
88	88	88	88
89	89	89	89
90	90	90	90
91	91	91	91
92	92	92	92
93	93	93	93
94	94	94	94
95	95	95	95
96	96	96	96
97	97	97	97
98	98	98	98
99	99	99	99
100	100	100	100

(A) source_code 테이블

id	source_id	time_number	cwe_id
1	1	1	1
2	2	2	2
3	3	3	3
4	4	4	4
5	5	5	5
6	6	6	6
7	7	7	7
8	8	8	8
9	9	9	9
10	10	10	10
11	11	11	11
12	12	12	12
13	13	13	13
14	14	14	14
15	15	15	15
16	16	16	16
17	17	17	17
18	18	18	18
19	19	19	19
20	20	20	20
21	21	21	21
22	22	22	22
23	23	23	23
24	24	24	24
25	25	25	25
26	26	26	26
27	27	27	27
28	28	28	28
29	29	29	29
30	30	30	30
31	31	31	31
32	32	32	32
33	33	33	33
34	34	34	34
35	35	35	35
36	36	36	36
37	37	37	37
38	38	38	38
39	39	39	39
40	40	40	40
41	41	41	41
42	42	42	42
43	43	43	43
44	44	44	44
45	45	45	45
46	46	46	46
47	47	47	47
48	48	48	48
49	49	49	49
50	50	50	50
51	51	51	51
52	52	52	52
53	53	53	53
54	54	54	54
55	55	55	55
56	56	56	56
57	57	57	57
58	58	58	58
59	59	59	59
60	60	60	60
61	61	61	61
62	62	62	62
63	63	63	63
64	64	64	64
65	65	65	65
66	66	66	66
67	67	67	67
68	68	68	68
69	69	69	69
70	70	70	70
71	71	71	71
72	72	72	72
73	73	73	73
74	74	74	74
75	75	75	75
76	76	76	76
77	77	77	77
78	78	78	78
79	79	79	79
80	80	80	80
81	81	81	81
82	82	82	82
83	83	83	83
84	84	84	84
85	85	85	85
86	86	86	86
87	87	87	87
88	88	88	88
89	89	89	89
90	90	90	90
91	91	91	91
92	92	92	92
93	93	93	93
94	94	94	94
95	95	95	95
96	96	96	96
97	97	97	97
98	98	98	98
99	99	99	99
100	100	100	100

(B) cwe_violation 테이블

(그림 4) 데이터셋에 저장된 소스 코드와 CWE 취약점 예시



(그림 5) 수집된 CWE 취약점 검출 데이터 예시

5. 결론 및 추후 연구

본 논문에서는 복잡도를 기준으로 CWE 위반 요소 데이터를 수집한다. 취약점이 존재할 가능성이 큰 소스 코드에 대해서만 데이터를 수집함으로써 오탐지된 취약점의 데이터셋 반영을 줄일 수 있다. 정확도가 향상된 데이터셋을 통해 인공지능 모델의 성능 개선을 기대한다.

ACKNOWLEDGMENT

이 논문은 교육부 및 한국연구재단의 4단계 두뇌한국21사업의 지원(F21YY8102068)과 2023년도 정부(교육부)의 재원으로 한국연구재단 기초연구사업의 지원(No.2021R111A3050407)을 받아 수행된 연구임

참고문헌

- [1] A. D. Carleton, E. Harper, T. Menzies, T. Xie, S. Eldh and M. R. Lyu, "The AI Effect: Working at the Intersection of AI and SE", IEEE Software. Vol. 37 No. 4, July-Aug 2020. pp. 26-35.
- [2] 박찬술, 김장환, 문소영, 김영철, "Bad Code 패턴의 지도 학습을 통한 Bad Code 식별 적용 사례.", 한국 소프트웨어공학 학술대회 논문집. Vol. 25, No. 1, 2023. pp. 119-120.
- [3] 박찬술, 전병국, 김영철, "코드 내부 정보의 정규화 기반 효율적인 코드 정적 분석 및 가시화.", 한국정보처리학회 학술대회논문집. Vol. 29 No. 4, 2022. pp. 85-87.

2023 한국스마트미디어학회 종합학술대회

