JKSCI

한국컴퓨터정보학회논문지
Journal of The Korea Society of Computer and Information
Vol. 30 No. 11, pp. 63-70, November 2025
https://doi.org/10.9708/jksci.2025.30.11.063

A Study on Multi-Task CVSS Metric Prediction via Fine-Tuned SLM

Junhyuk Park*, Jaehee Lee*, Hyo-Beom Ahn**

*Undergraduate Student, Div. of Artificial Intelligence, Kongju National University, Cheonan, Korea
**Professor, Div. of Artificial Intelligence, Kongju National University, Cheonan, Korea

[Abstract]

This study proposes a lightweight model for automatically predicting CVSS v3.1 Base Metrics from vulnerability descriptions. A multi-task architecture using DistilBERT as a shared encoder with parallel classification heads was trained on about 220,000 NVD records. Experiments showed improved efficiency and consistency over single-task approaches, with input token length identified as a key factor affecting performance. The results demonstrate the feasibility of automating CVSS metric prediction, and future work will extend to unstructured data and explainable AI to enhance reliability.

▸ **Key words**: Small Language Model(SLM), Vulnerability Management, Automated Risk Scoring, CVSS, Multi-Task Learning

[요 약]

본 연구는 보안 취약점 개요를 입력으로 CVSS v3.1 Base Metrics를 자동 예측하는 경량 언어 모델 기반 방법을 제안한다. DistilBERT를 공통 인코더로 사용하고 각 항목별 분류 헤드를 병렬로 연결한 Multi-Task 구조를 설계하여 NVD에서 수집한 약 22만 건의 데이터셋을 학습하였다. 실험 결과 제안된 모델은 기존 단일 태스크 접근보다 효율성과 일관성이 향상되었으며 입력 토큰 길이가 성능에 중요한 영향을 미침을 확인하였다. 본 연구는 CVSS 지표 예측의 자동화 가능성을 입증하였으며 향후 비정형 데이터 활용과 설명 가능 인공지능 기법 접목을 통해 신뢰성을 강화할 수 있을 것으로 기대된다.

▸ **주제어**: Small Language Model(SLM), 취약점 관리, 리스크 분석 자동화, CVSS, Multi-Task Learning

• First Author: Junhyuk Park, Corresponding Author: Hyo-Beom Ahn
*Junhyuk Park (pjun0315@icloud.com), Div. of Artificial Intelligence, Kongju National University
*Jaehee Lee (jaehee7813@gmail.com), Div. of Artificial Intelligence, Kongju National University
**Hyo-Beom Ahn (hbahn@kongju.ac.kr), Div. of Artificial Intelligence, Kongju National University
• Received: 2025. 09. 10, Revised: 2025. 09. 25, Accepted: 2025. 10. 22.

http://www.ksci.re.kr pISSN:1598-849X | eISSN:2383-9945

I. Introduction

최근 BERT (Bidirectional Encoder Representations from Transformers), GPT (Generative Pre-trained Transformer), T5 (Text-to-Text Transfer Transformer)와 같은 LLMs (Large Language Models) 또는 사전학습 언어 모델 (Pretrained Language Models)이 발전함에 따라 매우 다양한 작업에 활용되고 있다[1]. 이 모델들은 대규모 데이터셋과 복잡한 모델 구조를 통해 미리 학습하여 거의 대부분의 분야에 준수한 성능을 보인다. 이러한 특성 때문에 기본적인 예측 문제부터 자연어 처리 (Natural Language Processing), 시계열 분석 등의 문제를 해결할 모델로 많이 선택받는 추세이다.

하지만 대형 언어 모델 (LLM)은 수십억에서 수천억 개의 파라미터를 가지며 학습 단계에서 매우 많은 양의 데이터셋을 필요로 한다. 또한 수십~수백 GB 이상의 메모리, 고성능 GPU 등의 대규모 연산 자원이 필요하다. 결정적으로 거의 대부분의 작업에서 대형 언어 모델은 필요 이상으로 크고 복잡하다. 이러한 단점을 극복하고 더 경량화된 AI 기술이 필요한 환경을 위하여 경량 언어 모델 (Small Language Models; SLMs)이 제안되었다.

경량 언어 모델은 대형 언어 모델에 비해 상대적으로 적은 파라미터 수 (수천만 개에서 수억 개)를 가진 모델로 빠르고 가볍게 효율적이면서도 특정 작업에서 대형 언어 모델에 버금가는 성능을 낼 수 있도록 설계되었다. 실제로 BERT의 경량 모델 중 하나인 DistilBERT는 BERT의 성능을 최대한 유지하면서도 모델의 크기와 연산 비용을 줄이기 위해 지식 증류 (Knowledge Distillation) 기법을 활용하였다[2]. 이 연구에서는 DistilBERT와 같은 Transformer 기반 소형 모델을 BERT, GPT-3와 같은 대규모 모델과 대조하여 '경량 언어 모델'로 표현한다.

한편 4차 산업 혁명이 본격화되고 다양한 IT 기술들이 다양화, 고도화되면서 다차원적인 사이버보안 문제 또한 이슈가 되고 있다. 기술이 복잡해지면서 취약점이 발생할 가능성이 높아지고, 실제로 이를 악용한 보안 이슈 또한 증가하는 추세이다. CVE (Common Vulnerabilities and Exposures) 데이터는 보안 취약점의 발생 양상이 시간에 따라 명확하게 증가하고 있다는 것을 보여준다[3]. 이러한 배경에서는 취약점 관리 및 스코어링 시스템의 소요 자원 또한 증가할 것으로 예상할 수 있다.

CVSS (Common Vulnerabilities Scoring System)는 식별된 취약점의 위험도를 수치화하는 표준화 체계로써 CVE와 보안 취약점 관리 분야에서 서로 긴밀하게 연결되어 있다. 새로운 취약점이 발견되면 관리 기관이 해당 문제에 대해 CVE ID를 부여하고, 해당 취약점의 CVE 개요 (Description)를 기반으로 CVSS 점수를 산정하여 위험도를 수치로 제공한다. 보안 전문가, 기업은 이 점수를 근거로 대응 우선순위 결정에 활용한다.

CVSS 점수는 보안 취약점의 특성을 분류할 수 있는 여러 기준들을 합산해서 계산된다. 이때 이 특성에 따른 분류는 기준에 따라 자동화된 것이 아니라 보안 전문가 그룹이 CVE 개요를 참조하여 수작업으로 일일이 분류한다. 취약점 발생 건수가 일정 수준 이상으로 증가하면 원활한 시스템 유지가 매우 우려되는 부분이다. 이미 NVD (National Vulnerability Database)의 CVE 데이터베이스에 CVE 지표(Metric) 분류가 할당되지 않은 취약점이 다수 존재한다. 따라서 CVSS 지표 분류 작업의 자동화가 요구되는 단계이다.

본 연구는 비정형 자연어 데이터인 CVE 개요를 정형 지표인 CVSS 지표로의 변환 자동화 방법론을 제시한다. 이는 경량 언어 모델인 DistilBERT를 Fine-tuning하여 CVSS 지표에 특화된 분류 모델로써 구현되었다. 이를 활용하여 CVE 개요의 문맥 정보를 정교하게 반영함으로써 분류 성능을 높이는 것을 목표로 하며 다중 Metric 동시 예측이 가능한 Multi-Task 구조를 설계한다. 이는 단일 Metric 예측 기반의 기존 연구와 차별성을 가진다. 본 모델은 높은 정확성을 겸비하여 실제 보안 실무에서의 자동화 시스템 설계와 위험도 분석 정량화에 기여할 수 있는 기반 기술로 작용할 것으로 기대한다.

본 논문의 구성은 다음과 같다. II장에서는 국내외 동향과 함께 CVSS 지표 분류 모델 구현과 관련된 이론적 배경을 다룬다. III장에서는 본 연구에 활용된 데이터셋과 모델 설계, 학습 전략 등에 대해 자세히 기술한다. 마지막으로 IV장에서는 연구 결과를 요약하고 주요 시사점과 향후 연구 방향을 제시한다.

II. Preliminaries

1. Related works

최근 취약점 텍스트 기반 모델링은 경량 LLM/SLM(지식 증류·프루닝·양자화·어댑터)로 추론 비용을 낮추고, 멀티태스크 학습(공유 인코더+태스크별 헤드, 보조 손실)로 데이터 효율과 일반화를 높이며, XAI 보조 기법(attribution/saliency, attention-rollout, 오류 유형화 등)으로 판단 근거의 투명성을 강화하는 방향으로 수렴하

고 있다. 본 연구는 DistilBERT(경량 백본)에 멀티헤드 구조(멀티태스크)를 결합해 CVSS Base Metrics를 동시에 예측하는 경량 멀티태스크 설계를 채택하며 추론 오버헤드를 크게 늘리지 않는 경량 XAI 신호와의 호환성을 전제로 실무 적용성을 논의한다. 즉, 본 접근은 경량화-멀티태스크-설명성의 최근 추세와 정합적으로 위치한다.

1.1 A Software Vulnerability Risk Scoring System Using Public Vulnerability Information

국내에서는 CVSS 기반의 위험도 산정을 자동화하고자 한 시도로 위험도 스코어링 시스템을 제안한 연구가 있다[4]. 이 연구는 CVE 데이터와 CVSS v3.1의 Base Metrics 항목을 활용하여 각 항목 간 영향을 수치화하고 사용자 선택에 따라 위험도를 계산할 수 있는 점수화 시스템을 설계하였다. 특히 항목별로 사전에 정의된 가중치 테이블을 기반으로 점수를 산정함으로써 기존의 수작업 중심 위험도 평가에서 벗어나 보다 일관된 기준에 따른 기계적 판단을 가능케 하였다는 점에서 의의가 있다. 이는 국내 보안 분야에서 CVSS 산정 방식을 정형화하려는 초기의 실용적 시도였으며 이후 다양한 도메인에 확장 가능한 기반을 마련하였다고 할 수 있다.

1.2 Attack Path Analysis for Vulnerability Assessment Based on Attack Graph and Reinforcement Learning

공격 그래프 기반의 보안 분석에 강화학습을 적용한 연구도 국내에서 수행된 바 있다[5]. 해당 연구는 네트워크 내 다중 취약점들이 상호작용하는 과정을 공격 그래프 형태로 모델링하고, 이에 대해 Q-learning을 적용하여 최적의 공격 경로를 도출하는 방식의 분석 기법을 제안하였다. 특히 학습의 보상 함수에 CVSS v3.1 점수를 반영함으로써, 취약점의 심각도를 정량적 수치로 해석하고 모델 학습에 통합하였다. 이는 CVSS 점수를 단순 참조 지표가 아닌 자동화된 보안 의사결정 도구로 활용한 사례로, 본 연구가 다루는 CVSS 지표 자동 예측 결과의 실제 응용 가능성과 연계될 수 있다. 또한, 취약점의 특성과 맥락을 고려한 고위험 경로를 실질적으로 식별해내는 접근은 CVSS 분류 기반 평가 모델의 확장성과 활용성을 논의하는 데 있어 시사점을 제공한다.

1.3 CVSS-BERT: Explainable Natural Language Processing to Determine the Severity of a Computer Security Vulnerability from its Description

해외에서는 CVE 설명을 기반으로 개별 CVSS Base Metric을 자동 예측하고자 하는 다양한 시도가 이루어졌다. 대표적으로 CVSS-BERT 모델을 제안한 연구가 있다[6]. 이 연구는 취약점 개요 텍스트로부터 각 CVSS 항목(예: AV, AC, UI 등)을 개별 분류기로 예측하는 구조를 통해, 예측 결과의 설명 가능성을 확보하는 데 중점을 두었다. 각 항목에 대해 독립적인 BERT 기반 단일 태스크 분류기를 구성하였으며 Gradient 기반 saliency 기법을 활용해 입력 토큰의 영향도를 시각화함으로써 해석 가능성을 제시하였다. 다만, 해당 연구는 CVSS 항목 간의 상호 연관성을 고려하지 않은 단일 태스크 기반 구조에 머물렀다는 한계가 있으며 그로 인해 학습의 일관성과 효율성 측면에서 제약이 존재한다.

1.4 Can LLMs Classify CVEs? Investigating LLMs Capabilities in Computing CVSS Vectors

한편, 최근에는 대형 언어 모델을 활용한 CVSS 벡터 예측 가능성에 대한 연구도 수행된 바 있다[7]. 해당 연구는 GPT 계열의 LLM과 CVE 설명 텍스트만을 입력으로 하여 전체 CVSS v3.1 벡터를 자동 생성할 수 있는지 검토하였으며, 다양한 프롬프트 전략(Few-shot, CWE 정보 포함 등)을 활용해 실험을 수행하였다. 실험 결과 LLM은 Attack Vector, User Interaction과 같이 비교적 명확한 항목에서 높은 정확도를 보였으나 Confidentiality, Integrity, Availability와 같은 주관적 항목에 대해서는 정확도가 낮았으며 학습 제어의 어려움과 연산 자원의 소모가 큰 단점으로 지적되었다. 특히, 이 연구는 대형 언어 모델 기반 접근에만 집중하고 경량 모델(SLM)의 효율성은 고려하지 않았다는 점에서의 한계가 존재한다.

2. Core Concepts

2.1 CVE 및 CVSS

CVE는 미국 MITRE 재단에서 관리하는 공개 보안 취약점 데이터베이스로 각 보안 이슈에 대해 고유 식별자(CVE ID)와 함께 간략한 개요를 제공한다. 이 개요는 일반적으로 취약점의 발생 조건, 영향 범위, 관련된 시스템 또는 소프트웨어 등에 대한 자연어 기반 텍스트로 구성된다. CVE는 전 세계적으로 통용되는 취약점 식별체계로 다양한 보안 시스템 및 스코어링 도구에서 참조 기준으로 사용된다[8].

CVSS는 이러한 CVE에 포함된 취약점에 대해 정량적 위험도를 산정하기 위한 표준화된 평가 체계이다. CVSS v3.1은 Base, Temporal, Environmental 세 범주로 구성되며 이 중 Base Metrics는 취약점 그 자체의 고정적 속성을 평가하는 항목들로 구성된다. Base Metrics는 CVE의 개요 텍스트를 기반으로 보안 전문가가 직접 수작

업으로 분류하고 판단하는 과정에서 결정되며 여기에는 공격 벡터(Attack Vector; AV) 권한 요구(Privileges Required; PR), 사용자 상호작용(User Interaction; UI) 등의 요소가 포함된다[9].

CVE와 CVSS는 상호 보완적인 체계로 CVE가 취약점의 정성적 개요를 제공한다면 CVSS는 이를 기반으로 정량적 위험도를 부여한다는 점에서 밀접한 상관성을 가진다. 특히 CVSS의 Base Metrics는 CVE 설명으로부터 유추 가능한 내용에 기반하므로 이 둘 간에는 높은 개념적·내용적 연결성이 존재한다.

2.2 DistilBERT

DistilBERT는 BERT 모델을 지식 증류(knowledge distillation) 기법으로 경량화한 모델로, 전체 파라미터 수를 약 40% 줄이면서도 원래 BERT 성능의 약 97%를 유지한다고 보고된 바 있다. 이를 통해 추론 속도가 빨라지고 메모리 사용량이 감소하여, 대규모 데이터셋 처리나 자원 제약 환경에서 효과적으로 활용될 수 있다.

자연어 처리 분야에서는 문서 분류, 감정 분석, 질의응답 등 다양한 태스크에서 DistilBERT가 BERT에 준하는 성능을 보이면서도 훨씬 가벼운 연산량을 제공한다는 점이 입증되었다. 이러한 특성 덕분에 보안 분야에서도 취약점 데이터의 대량 처리에 DistilBERT가 적합한 대안으로 주목받고 있다. 최근 연구에서는 보안 공지 및 취약점 리포트와 같은 긴 텍스트를 처리할 때 DistilBERT를 활용하여 취약점 유형 분류, 공격 벡터 예측, 보안 경고 자동화 등에 적용하려는 시도가 이루어지고 있다. 예를 들어, NVD(National Vulnerability Database) 기반 취약점 설명을 입력으로 하여 CVSS 메트릭을 예측하거나, 보안 경고 문서의 위험도 수준을 자동 산출하는 연구들이 보고되었다.

2.3 Fine-Tuning

Fine-Tuning은 사전 학습된 모델을 특정 과제에 맞게 재학습하는 전이학습 기법으로, 대규모 비지도 데이터로 학습된 일반 표현을 소량의 레이블 데이터로 과제 특화 예측에 활용한다[10]. BERT는 출력층만 태스크에 맞게 바꾸고 전체 파라미터를 end-to-end로 미세 조정하며 이 단순한 구조 변경만으로 다양한 자연어 처리에서 높은 성능을 보여왔다[11]. 이를 통해 모델 재사용으로 학습 효율을 높이고 정제된 표현을 활용해 일반화 성능을 향상시킬 수 있다. 본 연구도 DistilBERT를 CVSS 예측에 맞게 Fine-Tuning하여 텍스트 기반 취약점 설명에서 보안 지표를 효과적으로 분류하였다[10].

III. The Proposed Scheme

1. Dataset

본 연구의 모델 학습에 사용된 여러 데이터는 자체 제작한 자동화 도구를 통해 미국 국립표준기술연구소(National Institute of Standards and Technology; NIST)에서 운영하는 NVD의 CVE, CVSS 정보를 수집하여 구성하였다.

Table 2. Model Dataset

Category	Description
Source	National Vulnerability Database, NIST
CVE ID Range	From CVE-2015-0001 to CVE-2025-58760
Collection Period	Jan 2015 - Jun 2025
Total Samples	220,219 CVEs
Main Features	cve, AV, AC, PR, UI, S, C, I, A, vector, description
Labels	Text Data
Missing Value Handling	on Metric Columns, 'vector', and 'description'

수집된 샘플은 총 220,219건의 취약점으로 구성되어 있으며 이는 대략 2015년부터 2025년까지의 범위이다. 주요 속성으로는 샘플을 식별하기 위한 CVE 번호와 CVSS 주요 지표의 각 항목, 이를 하나의 문자열로 표현한 CVSS 벡터, NVD에 작성된 해당 CVE에 대한 개요 텍스트가 있다. 이 값들은 모두 문자열 데이터로 이루어져 있으며 특히 'description' 속성에는 자연어 텍스트에서 쉽게 발견되는 특수문자 또한 포함되어 있다. 수집 과정의 특성 상 'cve'속성을 제외한 나머지 컬럼에는 결측치가 존재하는 샘플이 있으며 이는 추가적인 전처리를 수행하였다. 본 연구에서는 이를 'CVE-DescSet(Ver.3.1)'이라는 이름으로 명명하였다.

2. Training Configuration

공통된 Backbone을 공유하는 Multi-Task CVSS 지표 분류 모델의 파이프라인은 데이터 전처리와 토큰화(Tokenization), 모델 객체 정의, 학습 하이퍼파라미터 설정 및 학습 과정으로 구성된다.

2.1 Data Preprocessing

모델 학습을 위한 데이터셋은 크게 개요 텍스트와 CVSS Vector로 이루어져 있으며 이는 모델에서 요구하

는 입력과 최종 출력으로 사용된다. 개요 텍스트는 자연어 데이터로, 모델이 학습하기 위해서 수치형 데이터로 변환하는 토큰화 작업이 필요하다. 아래의 Fig. 1.에서 이 과정을 구현하였다.

Algorithm 1 Data Preprocessing & Tokenization

Input : CVE database file (CVE-DescSet(Ver.3).csv)
CVSS metrics $M = \{AV, AC, PR, UI, S, C, I, A\}$
Output : Tokenized text sequences and encoded labels

$df \leftarrow$ Read Data(**Input**)
Remove rows with missing values in $vector \cup M$
for each metric $m \in M$ **do**
  $df[m\_label] \leftarrow$ EncodeLabels($df[m]$)
end for
Extract texts: $texts \leftarrow df['description']$
Build label dictionary: $labels\_dict \leftarrow \{m: df[m\_label]\ for\ m \in M\}$
Initialize tokenizer with pretrained *DistilBERT*
Define Dataset class:
  for each text t:
    Tokenize text: $tokenized \leftarrow$ Tokenize$(t, \max_length = 256)$
    return $tokenized$, $\{m\_label: label[m]\ for\ m \cup M\}$
  end for

Fig. 1. A Pseudo Code of Data Preprocessing and Tokenization

토크나이저(Tokenizer)는 HuggingFace에서 제공하는 'distilbert-base-uncased'라는 사전학습된 버전을 사용하며 max_length 하이퍼파라미터를 통해 입력 시퀀스에서의 토큰 길이를 제한한다. CVE-DescSet(Ver.3.1)의 'description' 데이터 토큰 수는 대부분 500개 이하 ($Q_3 = 91.00$)에 분포하는 것으로 확인되었다. 본 연구 데이터셋의 경우 384개의 토큰 수 제한이 가장 최적임을 3.4절의 실험을 통해 확인하였다.

2.2 Model Architecture

본 연구의 모델 핵심 구조는 Fig. 2.에서와 같이 객체로 정의된다. 기존 연구와의 차이점 중 하나는 8개의 CVSS Base Metrics를 예측하기 위한 인코더를 공유하여 사용한다는 것이다. 이는 아래 Fig. 2.의 2번째 열에서 확인할 수 있다.

Algorithm 2 Model Architecture Definition

Define class MultiTaskDistilBERT:
  Shared Encoder: $H \leftarrow$ DistilBERTModel($input\_ids$)
  for each metric $m \in M$:
    Classification Head:
      $head\_m$ = Linear($hidden\_dim \rightarrow |C_m|$)
  end for
  Forward Pass:
    for each metric $m \in M$ **do**
      $logits[m] \leftarrow head\_m(H[CLS])$
    end for
    return $m: logits[m]$ **for** $m \in M$

Fig. 2. A Pseudo Code of Model Architecture Definition

모든 Metrics에 대응하는 단일 인코더를 사용하면 모델의 성능 측면에서 적은 공간을 차지한다는 이점을 가진다. 이는 학습된 모델이 다양한 임베디드 환경에서도 활용할 수 있음을 시사한다. Naïve한 형태의 사전학습된 DistilBERT 모델을 통해 각 Metric 별 헤드를 추가하고, 최종 출력을 도출하는 Multi-Task CVSS 분류기를 학습하는 과정은 위의 Fig. 3.과 같다.

3. Evaluating Model

3.1 Train & Evaluation Setup

DistilBERT와 같은 상대적으로 파라미터의 수가 많은 대규모 모델은 학습 단계에서의 수치 조정에 따라 성능에

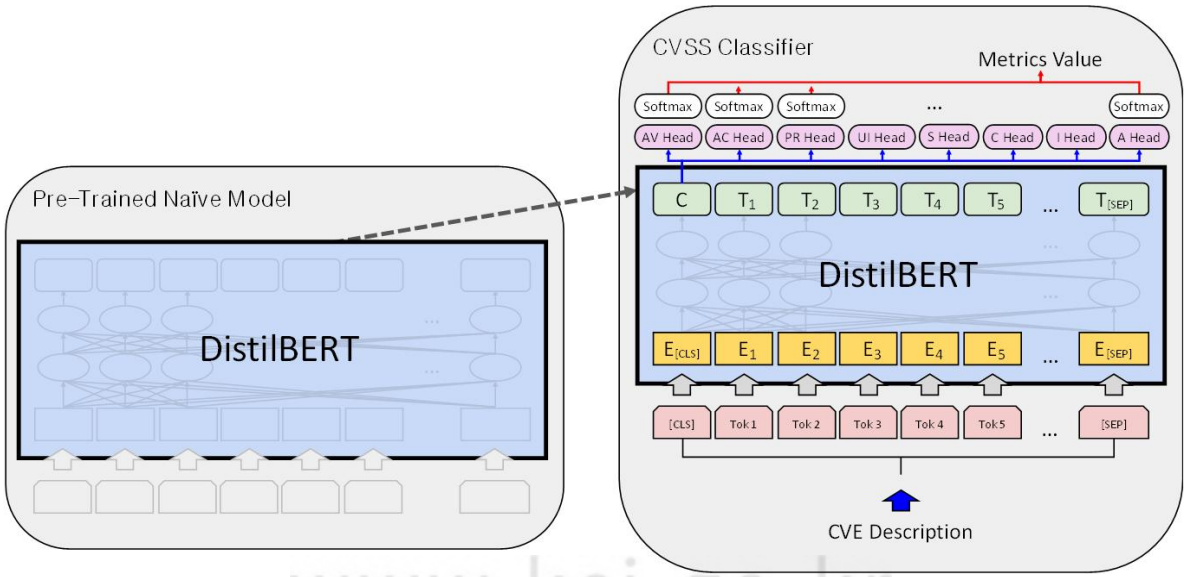

Fig. 3. Overview of the Fine-tuned Multi-task CVSS Metric Classifier Based on Pre-trained DistilBERT

큰 영향을 받는다. 본 연구에서의 구조화된 모델을 학습시키기 위한 정의는 아래의 Fig. 4와 같다.

Algorithm 3 Training Pipeline

Split Dataset into Train/Validation sets
Create mini-batches with size B
Initialize optimizer () and learning rate scheduler
Define loss function: $CrossEntropy$ for each metric
for $epoch$ = 1 **to** num_epochs **do**
  Set model to training mode
  for each $batch$ **do**
    Encode input: $H \leftarrow Encoder(batch.inputs)$
    $total\_loss = 0$
    for each metric $m \in M$ **do**
      Predict logits: $logits[m] \leftarrow Head\_m(H)$
      Compute loss: $loss[m] \leftarrow CrossEntropy(logits[m], batch.labels[m])$
    end for
    Backpropagate $total\_loss$
    Update parameters and adjust learning rate
    Reset gradients
  end for
end for
return traind mode $\theta *$

Fig. 4. A Pseudo Code of Training Pipeline

여러 번의 실험 결과, 모델 학습 과정에서의 주요 파라미터들의 최적 값은 에포크 수(num_epochs) 10과 조기 종료(Early Stopping), AdamW 옵티마이저의 학습률(lr) 5e-5, 배치 사이즈(batch_size) 16임을 확인할 수 있었다. 이외에도 손실 함수(Loss Function)의 변화나 Dropout, Layer Normalization(레이어 정규화) 등의 학습 기법 적용에 따른 성능 변화 또한 확인할 수 있었다. 베이스라인 모델의 하이퍼파라미터는 배치 사이즈 16, 시퀀스 토큰 길이(max_len) 256, 에포크 수 3, 학습률 5e-5로 설정되었다. 여러 번의 실험 결과 성능에 유의미한 영향을 미치는 하이퍼파라미터는 시퀀스 토큰 길이인 것을 알 수 있었다. 또한 클래스 불균형 문제에 대응한 Focal Loss를 Cross Entropy 대신 손실 함수로 사용한 결과 오히려 성능이 더 악화되었음을 확인하였다. 이는 모델의 소수 클래스에 집중하려는 성향이 다수 클래스를 잘 분류하는 것에 악영향을 미친 것으로 판단된다. DistilBERT 인코더 통과 후의 Multi-head 구조에서 적용된 Dropout, Layer Normalization 기법 또한 특별한 성능 향상으로 이어지지 않았음을 확인하였다.

3.2 Model Evaluation Metric

최종적으로 학습된 모델의 성능 지표는 아래의 Table 4. 와 같다. 전체 정확도 0.9153, Weighted F1 점수 0.9138, Macro F1 점수(클래스별 단순 평균) 0.8582를 기록하여 전반적으로 준수한 성능을 보이는 것으로 확인되었다. 이는 이전의 연구에서 학습한 모델[6]에 비교해 최대 2.17p 향상된 것이다. 또한 Gemma 3, all-MiniLM과 같은 LLM 모델이 일부 지표(AV, UI)에서 소폭 우세하지만 다른 지표(PR, Impact Metrics 3종)에서는 제안된 모델이 월등한 성능을 보이는 것으로 확인되었다. 다만 또 다른 연구[7]에서의 Gemma 3은 CVSS 분류에 맞게 학습한 모델이 아닌 일반 생성형 모델을 분류에 단순 사용하였고, all-MiniLM 모델은 end-to-end 구조를 가지지 않는 특성을 지녀 fine-tuning을 통한 성능 개선이 어렵다는 방법론적 특성에서의 차이점이 있다. 따라서 본 연구의 제안 모델과의 비교 기준군으로 구분하였다.

Table 4. Major Evaluation Score per Metric, (*weighted)

Model	Performance Metric	Exploitability Metrics					Impact Metrics		
		AV	AC	PR	UI	S	C	I	A
Ours	Accuracy	**0.9332**	0.9602	0.8518	0.9463	0.9615	0.8855	0.8908	0.8929
	Precision*	0.9320	0.9561	0.8501	0.9462	0.9610	0.8855	0.8909	0.8903
	Recall*	0.9332	0.9602	0.8518	0.9463	0.9615	0.8855	0.8908	0.8929
	F1-Score*	0.9324	0.9560	0.8496	0.9462	0.9609	0.8848	0.8907	0.8898
CVSS-BERT[6]	Accuracy	**0.9115**	0.9607	0.8379	0.9321	0.9545	0.8704	0.8735	0.8894
	Precision*	0.9090	0.9570	0.8392	0.9318	0.9553	0.8714	0.8736	0.8868
	Recall*	0.9115	0.9607	0.8379	0.9321	0.9545	0.8704	0.8735	0.8894
	F1-Score*	0.9089	0.9574	0.8378	0.9319	0.9548	0.8681	0.8731	0.8863
Gemma 3[7]	Accuracy	0.95	0.92	0.48	0.95	0.82	0.62	0.70	0.47
all-MiniLM[7]		0.89	0.87	0.73	0.87	0.86	0.77	0.77	0.77
Max Improv. (Acc)		**2.17%**	-0.05%	1.39%	1.42%	0.70%	1.51%	1.73%	0.35%

3.3 Model Size Comparison

성능 향상에도 불구하고 예측 모델의 규모를 결정하는 파라미터 수를 줄여 기존 연구 대비 경량화 또한 확인하였다. 아래의 Table. 5는 제안 모델이 약 66.02백만 개의 파라미터를 사용한 데 비해 기존 모델[6]은 약 230.12백만 개의 파라미터를 사용하여 약 3.49배 규모 차이가 있음을 보여준다. 이는 FP32 가중치 기준 메모리 사용량을 약 878 MiB에서 252 MiB 수준으로 저감할 수 있다. 이러한 경량화는 기존 연구가 메트릭별로 독립 인코더를 학습하는 것에 대비해 제안된 모델이 인코더를 공유하고 8개 선형 헤드만 분기하는 구조를 가지는 것에서 구현될 수 있었다. 또 다른 연구[7]의 Gemma 3는 어떤 세부 모델을 사용하였는지 명시되지 않아 가장 작은 모델을 기준으로 추정하였고, 그 결과 약 4.09배 규모 차이가 있음을 확인하였다. 이로써 LLM이 아닌 SLM을 통해 모델 경량화를 달성하면서도 성능 향상이 가능함이 입증되었다. all-MiniLM은 제안된 모델보다 적은 파라미터를 사용하나 절대 성능은 제안 모델이 우세하다.

Table 5. Parameter comparison between the CVSS-BERT and the proposed model, (*minimum)

Model	Encoder	Params	Δ vs Other
Proposed (Multi-Task DistilBERT)	1 Shared encoder + 8 linear heads	66.02M	-
CVSS-BERT (8xBERT-small)[6]	Independent encoder per metric (8 models)	230.12M	+249% (≈3.49x bigger)
Gemma 3[7]	(Generative)	270M-27B	+309%* (≈4.09x* bigger)
all-MiniLM[7]	1 Shared encoder + 8 nonlinear heads(XGBoost)	≈22-22.7M	-66.7% (≈0.33x smaller)

IV. Conclusions

본 연구는 대규모 보안 취약점 데이터셋을 기반으로 경량 언어 모델인 DistilBERT를 활용한 Multi-Task 학습 구조를 설계하여 CVSS v3.1 Base Metric을 자동 예측하는 방법을 제안하였다. 제안된 모델은 개별 Metric 간 상관성을 고려하여 단일 백본 인코더를 공유하고 각 항목별 분류 헤드를 통해 동시에 값을 산출하는 구조를 채택함으로써 기존 단일 태스크 기반 접근보다 높은 효율성과 일관성을 확보하였다. 실험 결과, 다수의 Metric에서 우수한 정확도와 F1-Score를 보였으며 이는 실제 보안 실무에서 CVE 신규 등록 후 위험도 산정 단계에서 분석가가 수행하던 CVSS 지표 분류 과정을 자동화함으로써, 보안 담당자가 각 취약점의 특징을 일일이 판단하던 절차를 대체할 수 있음을 보여준다. 이를 통해 취약점 평가 과정에서의 지표 판정 시간을 단축하고 다수의 신규 취약점이 동시 보고되는 상황에서도 신속하게 위험도 우선순위를 산출할 수 있다. 또한 제안 모델은 기존 보안 관리 시스템(NVD, 사내 취약점 관리 플랫폼 등)에 API 형태로 연동이 가능하므로 실제 운영 환경에서 자동 분류 및 예비 스코어 산정 기능으로 활용될 수 있다. 결과적으로 이는 보안 운영자의 분석 부담과 의사결정 지연을 줄여 전체 취약점 대응 프로세스의 효율성과 신뢰성을 높이는 기반 기술로 작용할 수 있을 것이다.

그러나 본 연구에는 몇 가지 한계가 존재한다. 우선, 데이터셋은 주로 NVD에서 제공하는 정형화된 Description을 중심으로 수집되었기 때문에 실제 현장에서 보고되는 비정형 취약점 서술이나 다양한 출처의 텍스트를 충분히 반영하지 못하였다. 또한 모델이 처리하는 입력은 단일 텍스트 Description에 한정되어 있어 소스코드 패치 로그, 보안 권고문, 취약점 공개 전 블로그 포스팅 등 잠재적으로 유용한 맥락 정보를 포함하지 못하였다.

향후 연구에서는 웹 크롤링 기술과 데이터 정제 파이프라인을 결합하여 다양한 비정형 취약점 설명 데이터를 자동 수집·통합하는 방안을 모색할 필요가 있다. 이를 통해 데이터의 다양성과 표현력을 확장하고 모델이 보다 일반화된 맥락에서 안정적인 예측 성능을 발휘할 수 있을 것으로 기대한다. 아울러, 설명 가능 인공지능(XAI) 기법을 접목하여 모델의 판단 근거를 시각화하고 보안 전문가가 신뢰할 수 있는 자동화 도구로 발전시키는 것도 중요한 연구 과제이다.

종합적으로, 본 연구는 CVSS Metric 자동 예측의 효율성과 실용성을 입증한 시도로서 사이버 보안 분야에서 경량 언어 모델의 적용 가능성을 보여주었다. 향후 확장된 데이터 수집과 모델 해석력 강화를 통해 보다 신뢰도 높은 취약점 관리 및 자동화된 위험도 평가 체계로 발전할 수 있을 것으로 기대한다.

REFERENCES

[1] J. D'Souza, "A Review of Transformer Models," Preprint, Sept. 2023. DOI: 10.48366/r640001

[2] V. Sanh, L. Debut, J. Chaumond, and T. Wolf, "DistilBERT, a distilled version of BERT: smaller, faster, cheaper and lighter,"Proc. of the EMC² Workshop at NeurIPS 2019, Vancouver, Canada, Dec. 2019. DOI: 10.48550/arXiv.1910.01108

[3] A.K. Threatt, J. Glyder, R. Franks, and L. Adams, "Some Analysis of Common Vulnerabilities and Exposures (CVE) Data from the National Vulnerability Database (NVD)," Proc. of the UNCW Honors Research Symposium, pp. 1-11, Wilmington, USA, May 2021.

[4] H. Kim, S. Park, and J. Lee, "A Software Vulnerability Risk Scoring System Using Public Vulnerability Information," Journal of the Korea Institute of Information and Communication Engineering, Vol. 25, No. 5, pp. 606-615, May 2021. DOI: 10.6109/jkiice.2021.25.5.606

[5] J. Kim and M. Han, "Attack Path Analysis for Vulnerability Assessment Based on Attack Graph and Reinforcement Learning," Journal of Korean Institute of Intelligent Systems, Vol. 35, No. 1, pp. 16-24, 2025. DOI: 10.5391/JKIIS.2025.35.1.16

[6] M.R. Shahid and H. Debar, "CVSS-BERT: Explainable Natural Language Processing to Determine the Severity of a Computer Security Vulnerability from its Description," in 2021 20th IEEE International Conference on Machine Learning and Applications (ICMLA), pp. 1600-1607, December 2021. DOI: 10.1109/ICMLA52953.2021.00256

[7] Francesco Marchiori, Denis Donadel, and Mauro Conti, "Can LLMs Classify CVEs? Investigating LLMs Capabilities in Computing CVSS Vectors," arXiv preprint arXiv:2504.10713, April 2025. DOI: 10.48550/arXiv.2504.10713

[8] MITRE Corporation, "Common Vulnerabilities and Exposures (CVE) Program Overview," cve.org, https://www.cve.org/about/overview.

[9] FIRST.Org, Inc., "Common Vulnerability Scoring System v3.1: Specification Document," first.org, https://www.first.org/cvss/v3-1/specification-document.

[10] IBM, "Fine-tuning AI models: what it is and how it works," ibm.com, *https://www.ibm.com/kr-ko/think/topics/fine-tuning.*

[11] J. Devlin, M. Chang, K. Lee, and K. Toutanova, "BERT: Pre-training of Deep Bidirectional Transformers for Language Understanding," Proceedings of the 2019 Conference of the North American Chapter of the Association for Computational Linguistics: Human Language Technologies (NAACL-HLT), pp. 4171-4186, Minneapolis, USA, June 2019. DOI: 10.18653/v1/N19-1423

Authors

Junhyuk Park is currently an undergraduate student in the Division of Artificial Intelligence at Kongju National University, Korea, and is in his third year of the B.S. program.

Junhyuk Park has been working as an undergraduate researcher at the AI Security Lab in the Division of Artificial Intelligence at Kongju National University, Korea, since 2024. His research interests include Large Language Models (LLMs) and Domain Adaptation.

Jaehee Lee is currently an undergraduate student in the Division of Artificial Intelligence at Kongju National University, Korea, and is in his second year of the B.S. program.

Jaehee Lee has been working as an undergraduate researcher at the AI Security Lab in the Division of Artificial Intelligence at Kongju National University, Korea, since 2025. His research interests include Artificial Intelligence.

Hyo-Beom Ahn received the B.S. in Computer Science and M.S., and Ph.D. in Computer Science and Statistics from Dankook University, Korea in 1992, 1994 and 2002 respectively.

Dr. Ahn has been with the Department of Information and Telecommunication at Kongju National University since then, and since 2021, he has been affiliated with the Division of Artificial Intelligence at the same university. He is interested in Computer Networks, Network Security.