JKSCI

한국컴퓨터정보학회논문지
Journal of The Korea Society of Computer and Information
Vol. 31 No. 1, pp. 109-120, January 2026
https://doi.org/10.9708/jksci.2026.31.01.109

An Implementation of TF-IDF Feature Extraction and Machine Learning Based Web Attack Detection System

Eun ji Song*, Seong-Cho Hong**, Ah Reum Kang***

*M.S. Student, Dept. of Cyber Security, Pai Chai University, Daejeon, Korea
**Research Professor, SMART ICT Convergence HRD Center, Pai Chai University, Daejeon, Korea
***Professor, Dept. of Cyber Security, Pai Chai University, Daejeon, Korea

[Abstract]

With the rapid proliferation of web applications, HTTP-based cyberattacks continue to rise, underscoring the need for effective web attack detection systems. This study proposes and evaluates a detection system that combines TF-IDF feature extraction with multiple machine-learning classifiers. Treating HTTP request data as text, we apply natural language processing techniques and assess Logistic Regression, Random Forest, and XGBoost using the CSIC 2010 HTTP dataset. Experiments show that XGBoost achieves the best performance with 98.77% accuracy, 0.994 ROC AUC, PR AUC, while Random Forest and Logistic Regression attain accuracies of 97.50% and 97.83%, respectively. All models deliver precision above 96%, demonstrating their viability for deployment in real-world environments. The results indicate that interpretable machine-learning approaches can achieve competitive performance without resorting to complex deep learning models.

▸ **Key words**: Web Attack Detection, TF-IDF, Logistic Regression, Random Forest, XGBoost, CSIC 2010

[요 약]

웹 애플리케이션의 급속한 확산과 함께 HTTP 기반 사이버 공격이 지속적으로 증가하고 있어 효과적인 웹 공격 탐지 시스템의 필요성이 대두되고 있다. 본 연구에서는 TF-IDF 특징 추출 기법과 다중 머신러닝 분류기를 결합한 웹 공격 탐지 시스템을 제안하고 성능을 평가하였다. HTTP 요청 데이터를 텍스트로 처리하여 자연어 처리 기법을 적용하고, Logistic Regression, Random Forest, XGBoost 분류기를 CSIC 2010 HTTP 데이터셋으로 평가하였다. 실험 결과, XGBoost 모델이 98.77%의 정확도와 0.994의 ROC AUC, PR AUC를 달성하여 가장 우수한 성능을 보였으며, Random Forest와 Logistic Regression은 각각 97.50%, 97.83%의 정확도를 기록하였다. 모든 모델이 96% 이상의 정밀도를 보여 실제 운영 환경에서의 적용 가능성을 입증하였다. 본 연구는 복잡한 딥러닝 모델을 사용하지 않고도, 해석 가능한 머신러닝 접근법으로 경쟁력 있는 성능을 달성할 수 있음을 보여준다.

▸ **주제어**: 웹 공격 탐지, TF-IDF, 로지스틱 회귀, 랜덤 포레스트, XGBoost, CSIC 2010

• First Author: Eun ji Song, Corresponding Author: Ah Reum Kang
*Eun ji Song (insam4457@gmail.com), Dept. of Cyber Security, Pai Chai University
**Seong-Cho Hong (scv.hong@pcu.ac.kr), SMART ICT Convergence HRD Center, Pai Chai University
***Ah Reum Kang (armk@pcu.ac.kr), Dept. of Cyber Security, Pai Chai University
• Received: 2025. 11. 10, Revised: 2025. 12. 10, Accepted: 2025. 12. 30.

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

I. Introduction

디지털 전환이 가속화되면서 웹 애플리케이션은 현대 사회의 핵심 인프라로 자리 잡고 있다. 전자상거래, 의료 서비스, 금융 거래, 교육 등 거의 모든 분야에서 웹 기반 서비스에 대한 의존도가 급격히 증가하고 있으며 이러한 추세는 코로나19 팬데믹 이후 더욱 가속화되었다[1].

하지만 웹 서비스 확산과 함께 사이버 공격의 규모와 복잡성도 기하급수적으로 증가하고 있다. OWASP(Open Web Application Security Project)가 발표한 2021년 웹 애플리케이션 보안 위협 보고서에 따르면, 웹 애플리케이션에 대한 공격이 지속적으로 증가하고 있으며, 특히 Broken Access Control, Cryptographic Failures, Injection 공격이 상위 3대 위협으로 식별되었다[2]. Akamai Technologies의 2025년 보고서 또한 2024년 한 해 동안 웹 공격이 33% 증가하여 총 3,110억 건의 공격이 기록되었고, 특히 API를 대상으로 한 공격이 급증하고 있다고 발표했다[3]. 마지막으로 Verizon의 2024년 데이터 침해 조사 보고서에서는 취약점 악용을 통한 초기 침투가 180% 증가했고, 웹 애플리케이션 공격의 88%가 도난된 자격 증명을 사용하고 있다고 보고하였으며 위에 따른 공격 방식이 전체 데이터 침해의 20%를 차지한다고 설명했다[4]. 이러한 통계는 기존의 규칙 기반 보안 시스템만으로는 진화하는 웹 공격에 효과적으로 대응하기 어려운 현실을 보여준다.

기존의 웹 보안 시스템들은 주로 시그니처 기반 탐지(Signature-based Detection) 방식에 의존해 왔으나 이러한 접근법은 알려지지 않은 새로운 공격 패턴이나 변형된 공격에 대해서는 탐지 능력이 제한적이다. 또한 AI의 활성화로 인해 공격자들이 인공지능을 활용한 자동화된 공격 도구를 사용하여 더욱 정교한 공격을 수행하고 있다는 점에서 기존 방어 체계의 한계가 명확히 드러나고 있다. 이러한 위협의 심각성은 국제 보안 기관들의 보고서에서도 확인된다. ENISA(유럽 네트워크 정보보안청)의 2024년 사이버 위협 전망 보고서는 가용성에 대한 위협을 최상위 위험으로 분류했으며, 랜섬웨어와 데이터 침해가 그 뒤를 이었다고 발표했다[5]. 이는 웹 애플리케이션을 대상으로 한 공격이 단순한 정보 탈취를 넘어 서비스 전체를 마비시키는 수준으로 진화했음을 보여준다.

이러한 문제에 대응하기 위해 머신러닝과 딥러닝 기반의 웹 공격 탐지 기법들이 활발히 연구되고 있다. 머신러닝 알고리즘은 대량의 HTTP 트래픽 데이터로부터 정상적인 패턴과 악성 패턴을 학습하여 자동화 및 변형된 공격도 탐지할 가능성을 제공한다. 관련된 연구로는 HTTP 헤더의 보안 특성을 기반으로 머신러닝 기법을 활용한 피싱 공격 탐지 실시간 시스템이 개발되었으며 97.8%의 높은 정확도와 1.57초의 빠른 응답 시간을 달성하여 16개의 HTTP 헤더 특징을 활용한 새로운 제로데이 피싱 공격 탐지가 가능함이 입증되었다[6]. 또한 최근 연구들에 따르면 딥러닝 기반 침입탐지 시스템은 기존 대비 90% 이상의 높은 탐지 정확도를 달성하고 있으며, 특히 TF-IDF(Term Frequency-Inverse Document Frequency)를 활용한 특성 추출과 앙상블 학습 기법을 결합한 접근법들이 우수한 성능을 보이는 것으로 나타났다[7][8].

진화하고 있는 웹 애플리케이션 공격을 효과적으로 탐지하기 위해서는 기존의 단일 탐지 기법들에서 더 나아가, 이들의 장점을 극대화하고 단점을 보완할 수 있는 앙상블 탐지 기법의 도입이 필요하다. 이에 따라서 이 연구에서는 CSIC 2010 HTTP 데이터셋을 활용하여 TF-IDF 기반 특성 추출과 머신러닝 분류 알고리즘을 결합한 효과적인 웹 공격 탐지 시스템을 제안하고자 한다. 제안하는 시스템은 HTTP 요청의 다양한 텍스트 구성요소를 통합적으로 분석하여 SQL Injection, XSS, 디렉터리 순회 등의 주요 웹 공격을 높은 정확도로 탐지할 수 있도록 설계하고자 한다. 특히 n-gram 기반 특성 추출과 다중 분류기(Classifier) 앙상블 기법을 활용하여 기존 연구 대비 향상된 성능을 달성하고자 한다. 이를 통해 실시간 웹 환경 및 실무에서 적용할 수 있는 실용적인 웹 공격 탐지 솔루션을 제공할 수 있을 것이다. 연구의 목적을 달성하기 위한 본 논문의 구성은 다음과 같다. 2장에서는 웹 애플리케이션 보안 위협의 동향과 이를 탐지하기 위한 기존의 머신러닝 기반 웹 방화벽 선행 연구들을 검토하고, 3장에서는 본 연구의 핵심인 TF-IDF를 활용한 HTTP 요청 데이터의 텍스트 특징 추출 기법과 데이터 전처리, 그리고 세 가지 머신러닝 분류기(로지스틱 회귀, 랜덤 포레스트, XGBoost)를 적용한 탐지 시스템의 설계 및 구현 과정을 설명한다. 4장에서는 제안 모델의 객관적인 성능 검증을 위해 수행한 실험 환경과 정확도, 정밀도, 재현율, 혼동행렬 등 다양한 평가지표를 바탕으로 한 분석 결과를 상세히 기술하며, 마지막으로 5장에서는 본 연구의 결론과 시스템의 기대 효과 및 향후 연구 방향을 제안한다.

II. Related Works

웹 공격의 급격한 증가와 복잡성으로 인해 머신러닝 기반 웹 공격 탐지 시스템에 대한 연구들은 앞서 서론에서 언급한 바와 같이 활발히 진행되고 있다. 기존의 선행 연구들을 크게 분류해 보자면, 1) TF-IDF 기반 접근법, 2) 딥러닝 기반 방법론, 그리고 3) 하이브리드 모델이라는 세 가지 범주로 나눌 수 있다. 각 접근법은 고유한 장점과 한계를 가지고 있으며, 실제 환경에서의 적용성과 성능 측면에서 서로 다른 특성을 보인다.

2.1. Related works

TF-IDF 기반 접근법은 HTTP 요청의 텍스트 구성요소를 벡터화하여 특성을 추출하는 방법으로써 웹 공격 탐지 분야에서 널리 활용되고 있다. 이 접근법의 주요 장점은 해석 가능성과 계산 효율성이며, 특히 실시간 처리가 요구되는 환경에서 우수한 성능을 발휘한다.

관련 연구를 살펴보면, Wu와 연구진은 허니팟(Honeypot)과 로지스틱 회귀 알고리즘을 결합한 웹 공격 탐지 방법이 제안하였다[9]. 구체적으로 웹 로그에서 수집된 텍스트 데이터를 TF-IDF를 사용하여 벡터화하고 로지스틱 회귀 모델을 통해 분류를 수행하는 방법을 사용했다. 이러한 방법을 통해, Support Vector Machine과 비교해 볼 때, 더 빠르고 정확한 웹 공격 행위 탐지를 달성한 것으로 나타났다. 반면에 TF-IDF 기반 접근법의 한계와 관련된 연구로는 Shareef와 동료들의 연구를 살펴볼 수 있다[10]. 이들은 SQL 페이로드 분류를 위한 로지스틱 회귀 알고리즘의 성능을 분석했는데, 구체적으로 로지스틱 회귀 모델의 오탐 예측이 정보보안의 CIA(기밀성, 무결성, 가용성) 원칙에 미치는 영향을 검토함과 동시에 페이로드 분류에 소요되는 시간을 단축하고자 하였다. 그러나 단순한 선형 분류기의 한계로 인해 복잡한 공격 패턴에 대한 제한된 탐지 성능만을 입증하였다. 다른 예시로는 Vajrobol 등의 연구를 들 수가 있다[11]. 이들은 상호정보(mutual information)를 활용한 로지스틱 회귀 기법으로 피싱 URL 탐지 연구를 진행하였으며, 99.97%의 높은 정확도를 달성했다고 보고했다. 그러나 이러한 결과는 피싱(phishing) URL과 같은 일반 도메인이 아닌 특정 도메인에 제한된 결과로써 다양한 웹 공격 유형에 대한 일반화 가능성은 추가 검증이 필요하다.

2.2. Deep Learning-based Approaches

딥러닝 기반 접근법은 복잡하고 거대한 패턴 학습과 자동 특성 추출 능력을 통해 웹 공격 탐지 분야에서 주목받고 있다.

관련 연구로는 Chatterjee 등은 딥러닝 기반 접근법을 바탕으로 CNN, ANN, LSTM 모델을 활용한 침입탐지 시스템을 제안했다[12]. 이들은 연구를 통해 여러 딥러닝 아키텍처를 비교 분석하여 최고 96%의 정확도를 달성했다고 보고했다. 해당 연구는 딥러닝 모델의 효율성과 신속한 대응 능력을 개선하는 데 기여했지만, 계산 복잡도가 높아 실시간 처리에는 다소 제약이 있었다. 한편 Brahmareddy와 연구진은 HEDNN(Hybrid Ensemble Deep Neural Network) ID라는 하이브리드 앙상블 딥 신경망 모델을 제안하였다[13]. 이 모델은 CNN, LSTM, 어텐션 메커니즘을 통합하여 98.68%의 정확도, 97.80%의 정밀도, 97.50%의 재현율을 달성하여 기존 단일 모델 대비 상당한 성능 향상을 보여주었다. 그러나 이 모델 또한 복잡성으로 인해 해석 가능성이 떨어진다는 단점이 존재하였다. 마지막으로 Ashiku와 동료들은 딥러닝 아키텍처를 사용한 네트워크 침입탐지시스템을 개발하였는데, 적응형이며 탄력적인 시스템은 다양한 공격 유형을 탐지하고 분류하는 데 우수한 성능을 보였다[14]. 하지만 모델 훈련에 필요한 대량의 라벨링된 데이터와 높은 계산 비용으로 인해 실용성에서 떨어진다는 한계점이 존재했다.

이처럼 딥러닝 기반 접근법은 높은 탐지 정확도를 달성할 수 있다는 장점이 있다. 하지만 앞서 살펴본 연구들이 공통적으로 지적하듯이 블랙박스 특성으로 인한 해석 가능성 부족과 높은 계산 비용이라는 근본적인 한계를 가지고 있다. 특히 앞서 소개한 선행 연구들에서 드러난 실시간 처리의 제약[12], 복잡성으로 인한 해석 불가능성[13], 그리고 대량의 라벨링 데이터 요구[14]는 모두 딥러닝 방법론의 실무적 적용을 어렵게 만드는 요인들이다. 이로 인해 과적합 위험과 대량의 훈련 데이터 요구량 등 제한된 리소스를 가진 기업이나 실시간 대응이 필요한 업무 환경에서는 도입과 운영에 상당한 제약이 따른다.

2.3. Hybrid Approaches

하이브리드 접근법은 서로 다른 머신러닝 기법들의 장점을 결합하여 단일 모델의 한계를 극복하고자 하는 방법론이다. 최근 연구들은 앙상블 학습과 다중 모델 조합을 통한 향상된 성능 달성을 보고하고 있다.

Durmuşkaya와 Bayraklı는 머신러닝 기반 웹 애플리케이션 방화벽 연구(WAF)에서 다섯 가지 분류 알고리즘

(K-NN, 로지스틱 회귀, 나이브 베이즈, SVM, 의사결정트리)을 비교했다[15]. 그 결과 의사결정트리가 93.27%의 정확도와 93.13%의 F1-score로 가장 높은 성능을 보였으며, 실시간 WAF 시스템에 성공적으로 적용되었다. 또한 Saini과 연구진은 Random Forest와 XGBoost를 결합한 하이브리드 앙상블 머신러닝 모델을 제안하였다[16]. 이를 통해 CSE-CIC-IDS2018, CIC-IDS2017, NSL-KDD, UNSW-NB15 데이터셋에서 각각 98.92%, 99.91%, 99.24%, 97.11%의 최대 예측 정확도를 달성했으며, 오탐률을 0.52%, 0.12%, 0.62%, 5.29%로 낮추는 우수한 결과를 보고했다. Doost 등은 CNN과 Random Forest를 결합한 하이브리드 침입탐지 접근법을 소개했다[17]. CNN을 통한 특성 추출과 Random Forest 기반 분류를 결합하여 KDD'99와 UNSW-NB15 데이터셋에서 97%의 정확도와 98% 이상의 정밀도를 달성했다.

앞선 연구들이 보여주듯이 하이브리드 접근법의 주요 장점은 개별 모델의 약점을 상호 보완하여 전반적인 성능과 견고성을 향상시킬 수 있다는 점이다. 반면 현장에서의 실무 적용 시 고려해야 할 한계점도 존재한다. 여러 모델을 통합할 경우, 각 모델의 파라미터를 독립적으로 관리해야 하므로 시스템 복잡성이 증가한다[16][17]. 또한 Durmuşkaya와 Bayraklı가 알고리즘들을 비교한 과정에서 드러나듯이 여러 모델을 동시에 학습하고 평가하는 과정은 단일 모델 대비 훨씬 많은 계산 자원과 시간을 요구한다[15]. 하이브리드 모델은 각 구성 모델의 하이퍼 파라미터뿐만 아니라 모델 간 가중치, 결합 방식 등 추가적인 메타 파라미터까지 조정해야 하므로 최적 구성을 찾는 데 상당한 전문 지식과 시행착오가 필요하다. 따라서 제한된 리소스를 가진 조직에서는 하이브리드 접근법의 실무적 도입에 어려움이 있다.

2.4. Summary of Previous Studies

기존 연구들을 종합적으로 분석한 결과, 각 접근법은 고유한 장단점을 가진다. TF-IDF 기반 방법은 빠른 처리 속도와 해석 가능성을 제공하지만, 복잡한 패턴 인식에는 한계가 있다. 딥러닝 접근법은 높은 성능을 달성하지만 계산 비용과 해석성 문제가 뒤따른다. 하이브리드 방법은 우수한 성능을 보이나 모델 복잡성이 증가한다.

이러한 연구적 특성은 실제 서비스 요구와도 맞물려 실산업 전반에서도 멀티 알고리즘 운영이 부상하고 있다. 글로벌 클라우드 및 보안 트렌드를 다룬 2024년 사이버 보안 보고서에서는 서비스별로 다양한 탐지 요구(정확성, 속도, 해석성 등)에 대응하기 위해 멀티 알고리즘 환경을 도입하는 사례가 크게 증가하고 있다[18]. 이러한 추세는 국내에서도 명확히 나타나고 있다. 국내 보안 전문매체의 2024 통합보안 리포트는 DPI 기반 NDR, SIEM·SOAR 연계, 다차원 위협 탐지 및 ML 상관분석 등을 한 플랫폼에서 결합해 오탐 및 과탐을 줄이고 대응 속도를 높이는 통합(멀티 알고리즘) 운영 사례가 확산 중임을 보여준다[19].

본 연구에서는 이러한 흐름을 반영하여 실제 업무 환경에서의 제약과 데이터 특성에 따라 각 모델의 장단점을 고려한 상황별 최적 알고리즘 선택 전략을 제안한다. 이를 통해 실제 웹 환경의 다양한 요구사항에 효과적으로 대응할 수 있는 실용적인 해결책을 제공할 수 있을 것이다.

III. Research Methodology

3.1. Dataset

연구의 목적인 효과적인 웹 공격 탐지 연구를 위해 CSIC 2010 HTTP 데이터셋을 사용했다[20]. 이 데이터셋은 웹 보안 연구 분야에서 널리 인정받는 벤치마크 데이터셋으로 다양한 HTTP 기반 웹 공격 패턴을 포함하고 있다. CSIC(Spanish Research National Council)의 정보보안 연구소에서 개발된 이 데이터셋은 전자상거래 웹 애플리케이션을 대상으로 자동 생성된 트래픽을 포함한다. 전체 데이터셋은 61,065개의 HTTP 요청으로 구성되어 있으며, 이 중 중복을 제거한 12,213개의 데이터로 분류하여 진행하였다. 분류된 데이터 중 7,200개(59.0%)가 정상 요청, 5,013개(41.0%)가 공격 요청으로 분류되어 있다.

CSIC 2010 HTTP 데이터셋의 구성은 실제 다양한 공격 시나리오를 충분히 반영하고 있다. 공격 유형으로는 SQL 인젝션, 버퍼 오버플로우, 정보 수집, 파일 노출, CRLF 인젝션, XSS, 서버 사이드 인클루드, 매개변수 조작 등 총 8가지 주요 공격 유형이 존재한다.

각 행은 해당 공격의 이름, 원본 HTTP 요청 예시, 그리고 공격의 특징을 설명한다. SQL Injection 유형은 URL 또는 본문(content) 필드에서 SQL 문법 키워드가 발견되는 경우를 다루며 데이터베이스 조작을 목적으로 다양한 쿼리 삽입이 이루어진다. Buffer Overflow는 매우 긴 데이터(예: 2,048바이트 이상의 연속문자)가 전송되어 서버의 버퍼를 의도적으로 넘치게 하는 경우에 발생한다. Information Gathering은 /admin, /phpmyadmin 등 관리자 경로 탐색, robots.txt·server-status 조회, 그리고 sqlmap·Nikto 같은 자동화 스캐너의 고유 User-Agent(또는 비정상/공백 User-Agent) 사용과 같이

서버 구조·자원 정보를 파악하려는 사전 정찰 요청을 포함한다. File Disclosure는 '../' 경로 순회 패턴을 이용해 민감 파일에 접근하는 사례에 해당한다. CRLF Injection은 HTTP 헤더에 개행문자 등을 삽입하여 응답 분할을 유도하는 공격을 예시로 들었으며, XSS(Cross Site Scripting)는 JavaScript 삽입으로 인증 정보 탈취나 세션 하이재킹 위험성을 보여준다. SSI(Server Side Include)는 HTML 파일 내 서버명령어 삽입, Parameter Tampering은 예측 가능한 변수명이나 값 조작을 통해 권한 상승 등을 시도하는 형태이다. 이와 같은 내용을 토대로 CSIC 2010 HTTP 데이터셋 내 주요 공격 유형 8가지를 실제 샘플과 함께 구체적으로 정리하면 아래의 Table 1과 같다.

Table 1. CSIC 2010 Dataset: Samples by Key Attack Categories

Attack Type	Attack Sample	Key Characteristics
SQL Injection	GET /tienda1/publico/vaciar.jsp?id=3' UNION SELECT 1,2,3-- HTTP/1.1	SQL statements included in the URL/content field
Buffer Overflow	POST /buffer.cgi ... A...(2048 bytes of 'A')	Reflection of abnormally long data
Information Gathering	GET /admin/ HTTP/1.1 ... User-Agent: scanner	Admin interface exposure · scanner attempt
File Disclosure	GET /../../../etc/passwd HTTP/1.1	Path traversal (../)
CRLF Injection	GET /page.jsp?param=...%0d%0a ...	Header splitting, attempt to tamper with the response
XSS	GET /search.jsp?query=<script> alert('XSS')</script> HTTP/1.1	JavaScript injection
Server Side Include(SSI)	GET /index.html?page=<!--#exec cmd="cat /etc/passwd"-->	Server-Side Include command
Parameter Tampering	POST ... username=admin&password=admin	Use of predictable/default values

3.2. Data Preprocessing

원시 요청에는 URL 인코딩(%XX), 특수문자, 다국어 문자가 혼재한다. 본 연구는 인코딩 보존과 키워드 유지를 목표로 encoding='latin1'로 CSV를 로드했으며, 14개 필드를 공백으로 연결해 request 텍스트를 만들었다. 특수문자는 제거하지 않고 그대로 남겨 두어 예컨대 %0D%0A와 같은 CRLF 토큰이나 '(quote)·--(comment) 등 SQL Injection 지표가 그대로 벡터화에 반영되도록 했다. 아래 Table 2은 SQL Injection 및 XSS 케이스의 전처리 전/후 변환 예시이다. 이 과정에서 모든 필드는 결측값, 소문자 치환, 특수문자/공백 패턴 정제와 같은 절차를 거친다.

Table 2. Example of Pre- and Post- Data Processing

Step	SQL Injection Sample	XSS Attack Sample
Pre-processing	GET /store1/public/empty.jsp?id=3' UNION SELECT 1,2,3--	GET /search.jsp?... <script>alert('XSS')</script> ...
Post-processing	GET tienda1 publico vaciar jsp id UNION SELECT HTTP Host ...	GET search jsp query script alert XSS script HTTP ...

본 연구에서 활용한 14개 HTTP 요청 필드는 CSIC-2010 데이터셋의 모든 HTTP 요청을 구성하는 핵심 요소로 웹 공격 탐지를 위한 특징 추출의 기본 데이터 소스가 된다. 이들 필드를 기능적 특성에 따라 6개 유형으로 분류하여 체계적으로 분석하였다.

Method는 요청 방식을 나타내며 공격자가 특정 HTTP 메서드(PUT, DELETE 등)를 악용하는 패턴을 식별하는 데 활용된다. URL은 요청 대상 자원의 경로로, SQL Injection이나 Directory Traversal 공격의 핵심 페이로드가 포함되는 가장 중요한 영역 중 하나다. 예를 들어 /admin/../../../etc/passwd 같은 경로 조작 시도나 /search.php?id=1' OR '1'='1처럼 SQL 구문이 포함된 경우 공격으로 분류될 가능성이 높다. Host는 대상 서버 정보를 담고 있어 가상 호스팅 환경에서의 공격 패턴 분석에 유용하다.

User-Agent는 브라우저나 클라이언트 애플리케이션 정보를 포함하며 자동화된 공격 도구나 취약점 스캐너 사용을 탐지하는 중요한 지표가 된다. 정상적인 브라우저 식별자와 달리 scanner, bot, sqlmap, nikto 등의 키워드를 포함하거나 비정상적으로 짧은 문자열을 보이는 경우, 혹은 존재하지 않는 브라우저/버전 정보가 기재된 경우, 의심 활동으로 분류할 수 있다.

Cookie는 세션 식별자와 사용자 상태 정보를 담고 있어 XSS(Cross Site Scripting)나 세션 하이재킹 시도에서 악성 스크립트가 삽입되는 경우가 빈번하다. 예를 들어 sessionid=<script>alert('XSS')</script> 또는 userid=admin'; DROP TABLE users;--처럼 JavaScript 코드나 SQL 구문이 포함되면 공격 신호로 간주할 수 있다.

Pragma와 Cache-Control은 캐시 동작을 제어하는 지

시어다. 일반적으로 공격 탐지에서는 상대적으로 낮은 가중치를 갖지만, 캐시 우회를 통한 공격에서는 no-cache="Set-Cookie"와 같은 비정상 지시어나 캐시 무력화 시도가 관찰될 수 있어 보조 지표로 활용된다. 이들 필드는 클라이언트가 처리 가능한 콘텐츠 형식, 인코딩, 문자셋, 언어를 나타낸다. 값이 비정상적이거나 인코딩 우회 시도가 보이면 공격 징후로 해석할 수 있다. 예를 들어 Accept-Encoding에 ../../../ 같은 경로 조작 문자열이 등장하거나, Accept-Charset에 비표준 인코딩이 포함되었을 때 인코딩 기반 우회 공격으로 분류될 수 있다.

Connection은 연결 유지 정책을 나타내는데 Content-Type은 특히 POST 요청의 데이터 형식을 나타낸다. Content는 POST 본문 데이터 자체로, 다수의 웹 공격에서 핵심 페이로드가 위치하는 영역이다. 예를 들어 username=admin' OR '1'='1&password=<script>alert(1)</script> 같은 폼 데이터에는 SQL Injection, XSS, Command Injection 등의 악성 코드가 삽입되는 경우가 많아, TF-IDF 가중치 계산에서 가장 높은 중요도를 가질 것으로 예상된다. 요약된 내용은 Table 3에 정리하여 제시하였다.

Table 3. HTTP Fields Description

Field Type	Field Name	Sample Value
Request Fundamentals	Method	GET, POST, PUT, DELETE
	URL	/index.html, /login.jsp?user=admin
	Host	www.example.com:8080
Client Identification	User-Agent	Mozilla/5.0 (Windows NT 10.0; Win64; x64)
Session Management	Cookie	sessionid=abc123; userid=user1
Cache Control	Pragma	no-cache
	Cache-Control	max-age=3600, no-store
Content Negotiation	Accept	text/html, application/json
	Accept-Encoding	gzip, deflate, br
	Accept-Charset	UTF-8, ISO-8859-1
	Language	ko-KR, en-US
Connection & Content	Content-Type	application/x-www-form-urlencoded
	Connection	keep-alive, close
	Content	username=admin&password=123456

본 연구에서는 이들 필드를 공백으로 연결해 단일 문서로 처리함으로써 필드 간 상관관계와 복합적인 공격 패턴까지 효과적으로 포착할 수 있는 특징 공간을 구성하였다. 전처리 과정에서는 1~3의 n-gram 범위를 설정했으며 본 연구에서 설정한 n-gram 범위(1~3)는 웹 공격 탐지에서 단일 단어와 연속된 단어 조합의 패턴을 모두 포착하기 위해 전략적 선정하였다. 구체적인 특징 추출은 다음과 같다.

먼저 1-gram(unigram)은 개별 토큰의 출현 빈도를 측정하여 'script', 'alert', 'UNION', 'SELECT' 등 공격에 특징적인 키워드를 식별한다. 이를 통해 2-gram(bigram)과 3-gram(trigram)은 연속된 단어 조합을 통해 공격의 문맥적 패턴을 포착할 수 있다. 예를 들어 위 표의 예시에서 1-gram 분석을 통해서는 'GET', 'search', 'jsp', 'query', 'script', 'alert', 'XSS'라는 개별 토큰들이 추출되며, 이 중 'script', 'alert', 'XSS'는 XSS 공격을 식별할 지표가 된다. 다음으로 2-gram 분석에서는 'GET search', 'search jsp', 'jsp query', 'query script', 'script alert', 'alert XSS'와 같은 연속 토큰 쌍이 생성되어, 단순히 'script'라는 단어의 존재뿐만 아니라 'query script'나 'script alert'와 같은 공격 시퀀스의 맥락을 파악하도록 하였다. 마지막으로 3-gram에서는 'GET search jsp', 'search jsp query', 'jsp query script', 'query script alert', 'script alert XSS'와 같은 삼중 토큰 조합이 생성되어 더욱 구체적인 공격 패턴의 식별이 가능하도록 하였다.

각 공격 유형별로 n-gram이 갖는 특성을 살펴보면 다음과 같다. 먼저 SQL Injection의 경우, 1-gram에서는 'UNION', 'SELECT', 'FROM', 'WHERE' 등의 SQL 키워드가 높은 TF-IDF 가중치를 가진다. 다음으로 2-gram에서는 'UNION SELECT', 'SELECT FROM' 등의 조합이 SQL Injection 공격을 식별할 강력한 공격 시그니처가 된다. 마지막으로 3-gram에서는 'UNION SELECT FROM'과 같은 완전한 SQL 구문 패턴까지 포착할 수 있어 오탐률을 크게 줄일 수 있다. XSS 공격에서는 1-gram의 'script', 'alert', 'document' 등이 기본 지표가 되고, 2-gram의 'script alert', 'alert document' 등이 공격 의도를 명확히 드러낸다. 특히 3-gram의 'script alert XSS'나 'document cookie steal' 등은 매우 구체적인 공격 패턴으로 분류 정확도를 높이는 데 기여하게 된다. 앞선 특징 추출을 예시와 함께 요약해 보면 아래의 Table 4와 같다.

Table 4. n-gram Examples

original data	1-gram	2-gram	3-gram
GET search jsp query script alert XSS	GET, search, jsp, ...	GET search, search jsp, ...	GET search jsp, ...

3.3. TF-IDF Feature Representation and Model Evaluation

TF-IDF는 텍스트 마이닝에서 중요한 특징 추출 방법으로 Term Frequency(TF)와 Inverse Document Frequency(IDF)를 결합한 기법이다. 이 연구에서는 HTTP 요청을 문서로 간주하고, TF-IDF를 적용하여 각 요청의 특징 벡터를 생성했다.

TF는 특정 문서에서 단어의 빈도를 나타내고, IDF는 단어가 전체 문서 집합에서 얼마나 희귀한지를 나타낼 때, d는 HTTP 요청 문서, t는 단어, D는 전체 문서 집합으로 표기하고, 문서 d에서 단어 t의 TF-IDF 값을 구하고자 하면 아래의 식(1)과 같이 계산된다. 이러한 방식을 통해 공격에 특징적으로 나타나는 단어들이 높은 가중치를 갖도록 한다면 공격 패턴을 효과적으로 식별할 수 있다.

$$TF-IDF(t,d,D) = TF(t,d) \times IDF(t,D) \quad (1)$$

이에 따라서 이 연구는 TF-IDF 기반 표현을 바탕으로 세 가지 기계학습 알고리즘을 비교 평가했다. 구체적으로 sklearn의 로지스틱 회귀, 랜덤 포레스트 분류기, XGBoost 분류기를 사용했으며, 모든 실험에서 동일한 조건으로 80%의 훈련 데이터와 20%의 테스트 데이터로 분할하여 평가했다. 로지스틱 회귀는 선형 분류기로서 빠른 학습과 예측이 가능하며 특징의 중요도를 계수를 통해 직관적으로 해석할 수 있다. 랜덤 포레스트 분류기는 다수의 의사결정 트리를 결합한 앙상블 방법으로 과적합을 방지하고 특징 중요도를 제공한다. XGBoost 분류기는 그래디언트 부스팅 기법을 사용하여 순차적으로 약한 학습기들을 결합하여 강한 학습기를 만드는 방법으로 높은 예측 성능을 기대할 수 있다.

3.4. Experimental Environment

연구 환경은 Python 3.12를 기반으로 하며, 주요 라이브러리로 scikit-learn, pandas, numpy를 활용했다. 실험 과정에서 TF-IDF 특징 추출, 데이터 전처리, 모델 학습 및 평가를 위해 로지스틱 회귀, 랜덤 포레스트 분류기, XGBoost 분류기를 적용했으며, 성능 평가는 sklearn.metrics 라이브러리를 통해 수행했다. 연구 목적을 달성하기 위해 HTTP 요청 데이터를 텍스트로 처리하여 자연어 처리 기법을 적용하였고, 각 모델의 성능을 정확도, F1-점수, ROC-AUC, PR-AUC 등 다양한 지표를 종합적으로 평가하여 웹 공격 탐지에 가장 적합한 모델을 선정하고자 하였다.

특히 객관적인 성능 평가를 보장하기 위해 중복을 제거한 전체 데이터셋(12,213개)을 층화 추출(stratified sampling)하여 훈련 데이터셋(80%, 9,770개)과 테스트 데이터셋(20%, 2,443개)으로 분할하였다. 각 모델에 적용된 주요 하이퍼파라미터 설정은 Table 5와 같다.

Table 5. Hyperparameter Settings

Model	Key Hyperparameters
Logistic Regression	solver='lbfgs', max_iter=500, class_weight='balanced'
Random Forest	n_estimators=300, max_features='sqrt', class_weight='balanced_subsample'
XGBoost	n_estimators=400, max_depth=6, learning_rate=0.1, subsample=0.9, tree_method='hist'

IV. Results

본 연구에서는 CSIC 2010 HTTP 데이터셋을 사용하여 세 가지 머신러닝 모델의 성능을 종합적으로 평가했다. 12,213개의 HTTP 요청 데이터를 8:2 비율로 분할하여 9,770개는 훈련용으로, 2,443개는 테스트용으로 활용했다. 각 모델의 성능을 다각도로 분석하기 위해 전체 성능 지표, 클래스별 상세 분석, 오탐률 검토, 개별 알고리즘의 특성으로 나누어 평가했다.

4.1. Overall Performance Metrics

테스트 데이터셋을 대상으로 세 가지 머신러닝 모델의 성능을 평가한 최종 결과는 Table 6과 같다.

먼저 로지스틱 회귀 모델은 97.83%의 정확도로 두 번째로 높은 성능을 보였으며, 특히 공격 클래스에 대한 정밀도(Precision_Attack)가 사실상 100%로 세 모델에 비해 상대적으로 높은 수치를 기록했다. 이는 테스트 데이터 내에서 정상 트래픽을 공격으로 오인하는 오탐(False Positive)이 전혀 발생하지 않았음을 의미한다. 그러나 재현율(Recall_Attack)은 0.9472로 세 모델 중 가장 낮은 수치를 기록하여 일부 공격을 놓치는 명확한 한계가 있었다. 랜덤 포레스트 모델의 경우 97.50%로 다른 모델에 비해 상대적으로 낮은 정확도를 보였으나, 모든 지표에서 96% 이상의 안정적인 성능을 유지하였다. 반면에 XGBoost 모델은 98.77%의 정확도를 달성하였다. 공격 클래스에 대한 정밀도는 0.9949로 로지스틱 회귀에 비해 소폭 낮았으나 여전히 높은 수준을 유지하여 실제 웹 서비스 운영 시 오탐으로 인한 가용성 저하 우려가 적음을 의미한다. 특히 재현율 측면에서 XGBoost는 0.9751을 기록하며 세 모델 중 가장 우

수한 성능을 보여 실제 유입되는 공격을 놓치지 않고 탐지하는 능력이 가장 탁월한 것으로 나타났다. F1-Score 또한 0.9849로 가장 높은 수치를 보여 정밀도와 재현율 간의 균형이 가장 잘 맞는 모델임이 입증되었다.

종합해 보았을 때, XGBoost 모델은 정확도, 재현율, F1-Score 등 주요 지표에서 가장 높은 성능을 보였다. 세 모델 모두 97.5% 이상의 높은 정확도를 달성했으나 XGBoost의 경우 미탐(False Negative)을 최소화하면서도 99% 이상의 높은 정밀도를 유지해 실제 보안 시스템 구현에 가장 적합한 모델임을 입증했다. 이러한 결과는 복잡한 딥러닝 모델이 없이도, 전통적인 머신러닝 접근법과 TF-IDF 기반 특성 추출을 통해 웹 공격 탐지에서 실무에서도 활용 가능한 경쟁력 있는 성능을 달성할 수 있음을 보여준다. 구체적인 지표는 아래 Table 6와 Fig. 1을 통해 제시하였으며 이를 통해 모델별 특성과 성능 차이를 명확히 확인할 수 있다.

Table 6. Overall Performance Metrics (Evaluated on Test Set, n=2,443)

Model	Precision Attack	Recall Attack	F1-Score Attack	Accuracy
Logistic Regression	1.00	0.9472	0.9729	97.83%
Random Forest	0.9777	0.9611	0.9693	97.50%
XGBoost	0.9949	0.9751	0.9849	98.77%

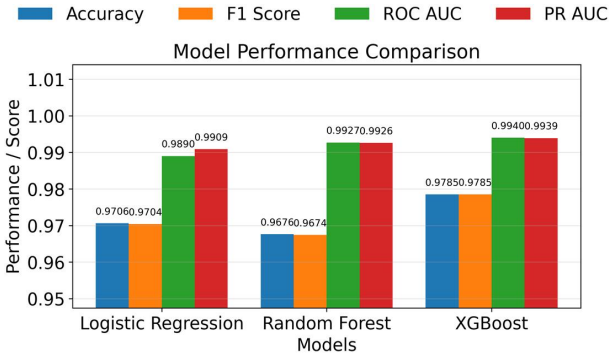

Fig. 1. Performance comparison of three machine learning models for web attack detection

4.2. Confusion Matrix Analysis

각 모델의 분류 성능을 면밀히 분석하기 위해 테스트 데이터셋(2,443건)에 대한 혼동행렬을 도출하였다. 혼동행렬은 모델이 범하는 오류의 유형(오탐 또는 미탐)을 구체적으로 식별할 수 있어 보안 시스템의 운영 정책을 수립하는 데 중요한 지표가 된다.

로지스틱 회귀(Logistic Regression) 모델의 결과는 테스트셋 데이터에서 오탐이 관측되지 않아 정밀도가 사실상 100%에 가까웠다. 정상 트래픽 1,440건을 100% 정확하게 분류하여 오탐(False Positive)이 단 한 건도 발생하지 않았다(0건). 공격 트래픽 1,003건 중에서는 950건을 탐지하고 53건을 놓쳐 약 5.28%의 미탐률(FNR)을 보였다. 정밀도 측에서는 매우 높은 수치를 보이지만, 세 모델 중 가장 많이 공격을 놓친다는 한계점을 보여준다. 하지만 공격을 일부 놓치더라도 정상 사용자의 차단을 원천적으로 배제해야 하는 보수적인 보안 환경에 적합함을 시사한다.

랜덤 포레스트(Random Forest) 모델은 정상 트래픽에서 1,418건을 맞추고 22건을 오분류 하였으며, 공격 트래픽에서는 964건을 탐지하고 39건을 놓쳤다. 로지스틱 회귀에 비해 공격 탐지 능력은 소폭 우수했으나, 1.53%의 오탐률(FPR)을 보여 세 모델 중에서는 가장 높은 오분류율을 기록했다.

반면 XGBoost 모델은 가장 균형 잡힌 성능을 입증했다. 정상 트래픽 1,440건 중 1,435건을 정확히 분류해 오탐을 단 5건(0.35%)으로 억제했으며, 공격 트래픽에서는 1,003건 중 978건을 탐지해 가장 낮은 미탐 건수(25건)를 기록했다. 미탐률(FNR) 또한 2.49%로 세 모델 중 가장 낮아 지능화된 웹 공격을 가장 효과적으로 방어할 수 있는 모델임이 확인되었다.

결론적으로 세 모델 모두 0.35%~1.53% 범위의 매우 낮은 오탐률(FPR)을 기록하여 실제 운영 환경에 적용 시 가용성 저하 우려가 낮음을 확인하였다. 각 모델의 구체적인 혼동행렬 수치는 Table 7과 같다.

Table 7. Confusion Matrix Results for Each Model

Model	TN	FP	FN	TP	FPR (%)	FNR (%)
Logistic Regression	1,440	0	53	950	0	5.28
Random Forest	1,418	22	39	964	1.53	3.89
XGBoost	1,435	5	25	978	0.35	2.49

4.3. ROC and PR Curve Analysis

ROC 곡선 분석은 모델의 곡선이 왼쪽 상단에 가깝게 위치할수록 높은 분류 성능을 나타내는 것으로 여겨진다. ROC-AUC가 1에 가까울수록 임계값 전 범위에서 양성과 음성을 올바르게 구분하는 능력이 높은 것으로 볼 수 있다. Fig. 2는 앞선 세 모델의 ROC 곡선을 비교한 그래프이다. XGBoost 모델의 ROC-AUC 값은 0.994로 가장 높은 수치를 기록했으며, 이는 모델이 정상 트래픽과 공격 트래픽을 매우 정확하게 구분할 수 있음을 의미한다. 로지스틱 회귀 모델의 ROC-AUC 값은 0.990이었으며, 랜덤 포레스트 모델는 0.993이라는 ROC-AUC 값을 달성했다.

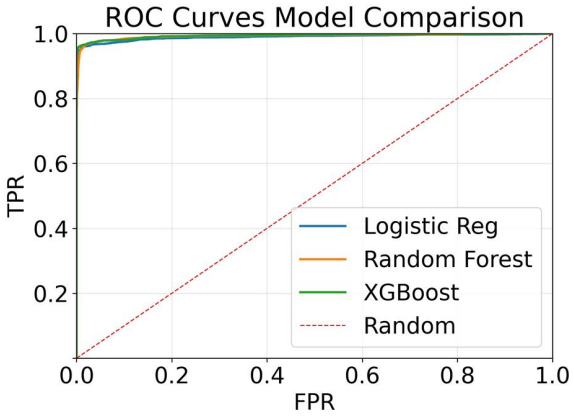

Fig. 2. ROC curves comparison for the three machine learning models

PR(Precision-Recall) 곡선은 유형 불균형 상황에서 모델의 성능을 평가하는 데 특히 유용하다. Fig. 3은 세 모델의 PR 곡선을 보여주는데, 곡선이 오른쪽 상단에 위치할수록 우수한 성능을 보이는 것으로 여겨진다. 모든 모델은 1에 가까운 AUC 값을 달성해 임계값 설정과 무관하게 우수한 분류 성능을 보장하는 것으로 나타났다. 실적용 관점에서 볼 때 세 모델 모두 높은 정밀도와 재현율을 보이며 우수한 성능을 보이는 것으로 나타났으나, 특히 XGBoost 모델의 PR-AUC 값은 0.994로 로지스틱 회귀 모델의 0.991보다 다소 높게 나타나 ROC-AUC와 PR-AUC 모두에서 가장 높은 성능을 보였다.

4.4. Discussion

각 모델에 따른 개별적 특성을 구체적으로 분석해 보면 다음과 같다. 먼저 로지스틱 회귀(Logistic Regression) 모델은 97.83%의 전체 정확도를 달성하며, 랜덤 포레스트보다 소폭 높은 성능을 보였다. 특히 주목할 점은 공격 유형에 대해 1.0000(100%)의 정밀도를 기록했다는 것이다.

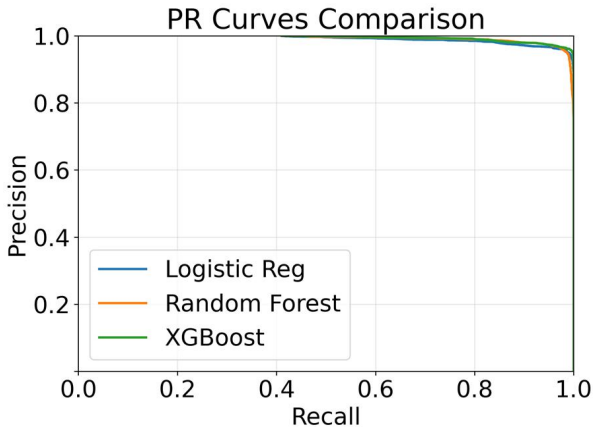

Fig. 3. Precision-Recall curves for the three machine learning models

이는 테스트셋 내에서 정상 트래픽을 공격으로 오인한 사례가 단 한 건도 없었음을 의미한다. 다만 재현율은 94.72%로 XGBoost에 비해 다소 낮아 일부 정교한 공격 패턴을 놓치는 경향이 있었다. 계산 효율성 측면에서는 평균 0.0010 ms의 예측 시간으로 세 모델 중 가장 빠른 처리 속도를 보였다. 따라서 로지스틱 회귀의 주요 장점은 모델의 투명성과 오탐 없는 신뢰성에 있다고 할 수 있다. 각 특성의 가중치를 명확히 파악할 수 있어 공격 탐지의 근거를 직관적으로 이해할 수 있으며, 빠른 속도와 완벽한 정밀도로 인해 오탐이 허용되지 않는 보안 환경에 유리하다.

다음으로 랜덤 포레스트(Random Forest) 모델은 97.50%의 전체 정확도를 기록했다. 공격 유형에서는 96.11%의 재현율과 97.77%의 정밀도를 달성했으며 F1-Score는 96.93%로 나타났다. 비록 정확도는 세 모델 중 가장 낮았으나, 모든 지표에서 96% 이상의 고른 분포를 보이며 과적합 없이 안정적인 성능을 유지했다. 이 모델의 특징적인 장점은 특성 중요도(Feature Importance) 분석을 통해 HTTP 요청의 어떤 필드가 공격 탐지에 가장 중요한지 파악할 수 있다는 점이다. 본 연구에서는 URL과 Content 필드가 가장 높은 중요도를 보였으며, 이는 실제 웹 공격에서 이들 필드에 정상 페이로드와 섞여 악성 페이로드가 함께 포함되는 경우가 많다는 사실과 일치한다.

마지막으로 XGBoost 모델은 98.77%의 정확도를 달성하며 모든 평가지표에서 가장 높은 성능을 보였다. 공격 유형에서 97.51%의 재현율과 99.49%의 정밀도를 기록했으며, F1-Score는 98.49%로 가장 균형 잡힌 성능을 보였다. 이러한 XGBoost 모델의 우수한 성능은 그래디언트 부스팅 알고리즘의 특성에 기인한다. 순차적으로 약한 학습기들을 결합하면서 이전 단계의 오류를 보정해 나가는 방식으로써 복잡한 패턴을 타 알고리즘 대비 효과적으로 학습할 수 있다. 이러한 방식은 특히 웹 공격의 다양하고 복잡한 공격 패턴을 포착하는 데 효과적이다.

Table 8은 CSIC 2010 데이터셋을 활용한 기존 연구들과 본 연구에서 제안하는 XGBoost 모델의 성능을 비교한 결과이다. 최신 연구인 Durmuşkaya[15]은 머신러닝 기반 웹 방화벽 연구에서 의사결정트리(Decision Tree)를 사용하여 93.27%의 정확도를 달성하였다. 반면 본 연구의 TF-IDF 및 XGBoost 기반 접근법은 98.77%의 정확도를 기록하여 기존 단일 모델 기반의 연구 대비 약 5.5% 향상된 탐지 성능을 입증하였다.

전체 파이프라인 지연 시간(Pipeline Latency)과 관련하여, Intel Core i7 환경에서 배치 처리(n=1,000)를 기준으로 텍스트 전처리 및 TF-IDF 벡터화 과정을 모두 포함한 전체 파이프라인(End-to-End) 처리 시간을 측정하였

다. 측정 결과, XGBoost는 요청당 평균 0.0100 ms, 로지스틱 회귀는 0.0060 ms가 소요되었다. 이는 일반적인 웹 애플리케이션의 네트워크 허용 지연 시간(수십 ms) 대비 미미한 수준으로, 대용량 트래픽 환경에서도 병목 없이 실시간 필터링을 수행할 수 있음을 실질적으로 보여준다.

Table 8. Performance Comparison with Existing Studies (CSIC 2010)

Study	Method	Accuracy
Durmuşkaya & Bayraklı [15]	Decision Tree	93.27%
Proposed Method	TF-IDF + XGBoost	98.77%

종합해 볼 때, 본 연구는 CSIC 2010 데이터셋을 사용한 단일 TF-IDF 특성 추출 기반 접근법으로 정확도 98.77%를 달성하면서 모델의 해석 가능성을 유지하는 데 성공하였다. 각 모델의 특징들을 요약하자면, XGBoost 모델은 가장 높은 정확도와 재현율을 제공하여 실제 공격 탐지율을 극대화해야 하는 환경에 적합하며, 로지스틱 회귀 모델은 100%의 정밀도와 신속한 속도를 제공하여 오탐(False Positive) 방지와 실시간 처리가 최우선인 환경에 적합하다. 마지막으로 랜덤 포레스트 모델은 안정적인 성능을 바탕으로 일반적인 상황에서 무난하게 활용될 수 있다.

이러한 결과는 복잡한 딥러닝이나 다중 알고리즘 조합 없이도 적절한 특성 추출과 알고리즘 선택만으로 실무에 적용할 수 있는 효과적인 웹 공격 탐지 시스템을 구축할 수 있음을 보여준다. 또한 사용자가 운영 환경의 요구사항(최고의 탐지율 vs 최저의 오탐률 vs 속도)에 따라 최적의 모델을 선택할 수 있는 유연한 선택지를 제공한다는 점에서 그 의의가 크다.

4.5. Validation on Recent Dataset

본 연구에서는 웹 페이로드의 텍스트 특성 분석에 용이한 CSIC-2010을 주 데이터셋으로 활용하였으나 데이터셋의 시의성 한계를 보완하기 위해 HTTP Params Dataset[21]을 활용하여 추가 검증을 수행하였다. 해당 데이터셋은 2016년에 최초 생성되었으며 해당 데이터셋의 마지막 주 업데이트는 2020년으로, 본 연구에서 주 데이터셋으로 활용한 CSIC-2010보다 최신화된 다양한 공격패턴을 포함하고 있다. 해당 데이터셋은 FuzzDB를 포함한 다수의 웹 공격 패턴 데이터베이스를 기반으로 구축되었다. 이는 학술적 환경뿐만 아니라 실제 모의해킹 및 WAF 우회 시도에서 사용되는 다양한 최신 페이로드를 포함하고 있어 제안 모델의 탐지 성능을 검증하기에 적합하다.

검증 실험은 HTTP Params Dataset(총 31,067건)을 대상으로 본 연구와 동일한 TF-IDF 벡터화 및 XGBoost 학습 파라미터를 적용하여 수행되었으며 데이터 분할은 동일하게 훈련 80%, 테스트 20% 비율로 진행하였다. 실험 결과는 Table 9와 같다.

Table 9. Validation Results on HTTP Params Dataset

Model	Precision Attack	Recall Attack	F1-Score Attack	Accuracy
XGBoost	0.9992	0.9868	0.9930	99.42%

실험 결과, 제안하는 XGBoost 모델은 검증용 데이터셋에서도 99.42%의 높은 정확도와 0.9992의 정밀도를 기록하였다. 이는 본 연구에서 제안한 TF-IDF 기반 특징 추출 방식이 CSIC-2010 데이터셋에 국한되지 않고 다양한 형태의 최신 웹 공격 문자열 패턴을 탐지하는 데에도 효과적으로 적용될 수 있음을 보여준다.

V. Conclusions

본 연구에서는 CSIC-2010 HTTP 데이터셋을 활용하여 TF-IDF 기반 특징 추출과 세 가지 기계학습 알고리즘을 결합한 웹 공격 탐지 시스템을 제안하고 비교 평가하였다.

연구 결과, XGBoost 모델의 정확도가 98.77%, ROC-AUC 값이 0.994 그리고 PR-AUC 값이 0.994로 가장 우수한 성능을 보였다. 로지스틱 회귀 모델은 97.83%의 정확도를, 랜덤 포레스트는 97.50%의 정확도를 보이는 것으로 나타났다. 모든 모델에서의 거짓 양성률은 매우 낮은 것으로 나타나 실제 운영 환경에서의 적용 가능성 또한 확인할 수 있었다. 특히 XGBoost 모델은 공격 트래픽에 대한 정밀도가 99.49%로 매우 높은 값을 보여 실무 적용에 적합했으며, 로지스틱 회귀 또한 미탐률은 세 모델 중 가장 높은 수치를 기록했지만, 정밀도 측에서는 사실상 거의 100%의 수치를 보여 계산 효율성을 고려할 때 유력한 대안이 될 수 있는 것으로 나타났다.

본 연구는 복잡한 딥러닝 모델이 아니라도 해석이 가능한 기계학습 접근법을 통해 경쟁력 있는 성능을 달성할 수 있음을 입증하였다. 제안된 방법은 계산 효율성과 모델 해석성 측면에서 뚜렷한 장점을 제공하여 하드웨어 자원이 제한된 IoT 엣지 디바이스 등의 환경이나 중소 규모 조직에서 활용하기에 적합하다고 볼 수 있다. 또한 각 모델의 특성을 고려해 볼 때 금융 서비스나 전자상거래 플랫폼과 같이 미탐 최소화와 최고 성능이 요구되는 환경에서는 XGBoost 모델을,

실시간 미디어 스트리밍 서비스(OTT)나 대용량 트래픽 처리를 위해 빠른 응답 시간과 해석성이 중요한 환경에서는 로지스틱 회귀 모델을, 다양한 접속 패턴이 혼재되어 과적합 방지와 안정성이 중시되는 일반 기업망과 같은 환경에서는 랜덤 포레스트 모델을 선택하는 것이 적절하다. 이처럼 운영 환경의 요구사항에 따라 최적의 모델을 선택할 수 있는 유연성을 제공한다는 점에서 그 의의가 크다.

이러한 연구의 의의에도 불구하고 본 연구는 몇 가지 한계점을 가지고 있다. 먼저, 이 연구는 2010년에 구축된 CSIC-2010 데이터셋을 사용함으로써 최신 웹 공격 기법에 대한 탐지 성능을 충분히 검증하지 못했다. 따라서 후속 연구는 보다 최신 데이터셋 등 다양한 벤치마크 데이터셋에서의 추가 검증이 필요하다. 또한 분석에 있어 단일 데이터셋에 의존하여 모델의 일반화 성능에 대한 검증이 부족하다고 볼 수 있다. 향후 서로 다른 데이터셋을 사용함과 동시에, 해당 데이터셋들을 사용한 연구들과 직접 비교한다면 더 좋은 결과를 얻을 수 있을 것이다. 한편, 본 연구는 실시간 차단 효율성을 극대화하기 위해 8가지 공격 유형을 통합하여 이진 분류(정상/공격)로만 평가를 수행하였다. 이는 신속한 탐지에 집중한 결과이나, 특정 공격 유형(예: Buffer Overflow)에 대한 개별적 탐지 취약점이나 유형 간 오분류 패턴을 상세히 분석하지 못했다는 한계점이 있다. 마지막으로 실시간 처리 성능과 대용량 트래픽 환경에서의 확장성에 대한 평가가 이루어지지 않았으며, 적대적 공격이나 회피 기법에 대한 견고성 검증이 부족하다는 한계점도 존재한다. 이러한 한계점들을 해결하기 위해 향후 연구로는 다양한 공격 유형에 대한 세부적인 성능 평가와 실시간 탐지 성능 최적화, CSIC-2010 외 최신 데이터셋 기반의 검증이 필요할 것으로 보인다. 후속 연구에서는 이와 더불어 앙상블 기법을 통한 세 모델의 메타 모델화, 온라인 학습 알고리즘의 적용, 그리고 각 모델의 특징 중요도 분석을 활용한 공격 패턴의 심층 분석과 새로운 공격 유형에 대한 적응성 평가를 진행한다면 연구에서 제안한 접근법의 실용성과 신뢰성을 더욱 향상시킬 수 있을 것으로 기대된다.

ACKNOWLEDGEMENT

This work was supported by the Institute of Information & Communications Technology Planning & Evaluation(IITP)-Innovative Human Resource Development for Local Intellectualization program grant funded by the Korea government(MSIT)(IITP-2026-RS-2022-00156334)

REFERENCES

[1] C. O'Toole, J. Schneider, K. Smaje, and L. LaBerge, "How COVID-19 has pushed companies over the technology tipping point—and transformed business forever," McKinsey & Company, www.mckinsey.com

[2] Open Web Application Security Project (OWASP), "OWASP Top 10 Vulnerabilities 2021," OWASP, 2021.

[3] Akamai, "State of Apps and API Security 2025 : How AI Is Shifting the Digital Terrain," Akamai, vol. 11, no. 2, 2025.

[4] Verizon, "2024 Data Breach Investigations Report (DBIR)," Verizon, 2024.

[5] European Union Agency for Cybersecurity (ENISA), "ENISA Threat Landscape 2024," ENISA, 2024.

[6] S. Shukla, M. Misra, and G. Varshney, "HTTP header based phishing attack detection using machine learning," Transactions on Emerging Telecommunications Technologies, Vol. 34, No. 11, 2023. DOI: https://doi.org/10.1002/ett.4872

[7] Z. Xu, Y. Wu, S. Wang, J. Gao, T. Qiu, Z. Wang, H. Wan, and X. Zhao, "Deep Learning-based Intrusion Detection Systems: A Survey," arXiv:2504.07839, 2025.

[8] S. V. N. S. Kumar, M. Selvi, and A. Kannan, "A Comprehensive Survey on Machine Learning-Based Intrusion Detection Systems for Secure Communication in Internet of Things," Computational Intelligence and Neuroscience, Vol. 2023, 2023.

[9] Q. Wu, S. Wen, F. Li, B. Liu, and W. Zhong, "Web Attack Detection Based on Honeypots and Logistic Regression Algorithm," Journal of Electrical Systems, Vol. 20, No. 3, pp. 814-825, 2024. DOI: https://doi.org/10.52783/jes.3013

[10] O. S. F. Shareef, R. F. Hasan, and A. H. Farhan, "Analysis of the False Prediction of the Logistic Regression Algorithm in SQL Payload Classification and its Impact on the Principles of Information Security (CIA)," Iraqi Journal for Computer Science and Mathematics, Vol. 4, No. 4, pp. 191-205, 2023.

[11] V. Vajrobol, B. B. Gupta, and A. Gaurav, "Mutual information based logistic regression for phishing URL detection," Cyber Security and Applications, Vol. 2, 2024. DOI: https://doi.org/10.1016/j.csa.2024.100044

[12] S. Chatterjee, S. Chaudhary, and A. K. Cherukuri, "Intrusion Detection System Using Deep Learning for Network Security," arXiv:2505.05810, 2025.

[13] A. Brahmareddy, S. Meghana, S. V. Kiran, K. S. Bharathi, and B. V. Kumar, "Hybrid Ensemble Deep Neural Network for Intrusion Detection (HEDNN ID)," SSRG International Journal of Electronics and Communication Engineering, Vol. 12, No. 7, pp. 184-200, 2025.

[14] L. Ashiku and C. H. Dagli, "Network Intrusion Detection System using Deep Learning," Procedia Computer Science, Vol. 185, pp. 239-247, 2021.

[15] M. E. Durmuşkaya and S. Bayraklı, “Web application firewall based on machine learning models,” PeerJ Computer Science, Vol. 11, 2025. DOI: https://doi.org/10.7717/peerj-cs.2975

[16] N. S.aini, V. B. Kasaragod, K. Prakasha, and A. K. Das, “A hybrid ensemble machine learning model for detecting advanced persistent threat attacks based on network behavior anomaly detection,” Concurrency and Computation: Practice and Experience, Vol. 35, No. 28, 2023.

[17] P. A. Doost, S. S. Moghadam, E. Khezri, A. Basem, and M. Trik, “A new intrusion detection method using ensemble classification and feature selection,” Scientific Reports, Vol. 15, No. 1, 2025. DOI: https://doi.org/10.1038/s41598-025-98604-w

[18] Check Point Software Technologies, “2024 Cyber Security Report,” Check Point Software Technologies, 2024.

[19] B. Won, “2024 Integrated Security Report: From Solution Consolidation to Unifying User Needs,” BoanNews, 2024. https://www.boannews.com/media/view.asp?idx=132426

[20] C. T. Gimenez, A. Perez-Villegas, and G. A. Maranon, “HTTP dataset CSIC 2010,” Impact Cyber Trust Dataset Repository, 2010.

[21] Morzeux, "HTTP Params Dataset," Kaggle/GitHub, 2020. Based on FuzzDB and WAF evasion payloads. https://github.com/Morzeux/HttpParamsDataset

Authors

Eun ji Song received the B.S. degree in Computer Engineering in 2022. She is currently pursuing an M.S. degree in Information Security at Pai Chai University and works at Igloo Corporation in the area of security operations monitoring. Her research interests include cybersecurity, information security, and cloud security.

Seong-Cho Hong received the B.A. degree in Psychology in 2014, M.A. in Criminology in 2018 and Ph.D. in Legal Psychology. He joined the Smart ICT Convergence Human Resource Development Center at Pai Chai University, Daejeon, South Korea, as a Research Professor. His current research interests include security, artificial intelligence, criminal behavior, theoretical framework and convergence research.

Ah Reum Kang received the M.S. and Ph.D. degrees in information security from Korea University, South Korea, in 2012 and 2016. She is a professor in the Department of Information Security at Pai Chai University in Daejeon, South Korea. Her current research interests include security, artificial intelligence, malware, medical data analysis, and online game security.