**2.5 related works**

**2.5.1 Introduction**

SQL Injection remains one of the most persistent and potential threats to web application security. Over the last decade, researchers have exploited distinct and varying machine-learning and deep-learning approaches to automate the detection and mitigation of SQL injection attacks. This chapter is a synthesis of important works in the area, with special emphasis on advanced techniques that use artificial intelligence to enhance accuracy in detection, reduce false alarms, and, in some cases, help with prioritization and prevention of attacks in real-time.

**2.5.2 Detection Approaches Using Machine Learning**

Early works on SQLi detection focused on the classical machine learning paradigm, such as Support Vector Machines (SVM), Decision Trees, Naïve Bayes, and Logistic Regression. These methods generally required engaged features obtained from HTTP request content, URL, or even SQL query structure. They showed decent performances but were constrained by the quality and view of the features extracted. Sometimes, a minor change in the attack pattern would deter these methods from detecting them.

With the appearance of a few hybrid methods merging rules with machine learning classifiers to improve detection, more flexibility and accuracy were introduced but confronted again with false positive and false negative alarms, especially concerning large-scale data and obfuscated attacks.

**2.5.3 Deep Learning for SQL Injection Detection**

Recent research has increasingly turned to deep learning to overcome the limitations of traditional models. Three notable contributions illustrate the evolution and diversification of deep learning-based detection approaches.

**2.5.3.1 NLP and BERT-Based Detection (Sagar Lakhani et al.)**

Lakhani et al. proposed an NLP approach to SQLi detection. Their model uses BERT or Bidirectional Encoder Representations from Transformers for contextual feature extraction. BERT was fine-tuned on a labeled dataset of SQL queries and indicated good performance with 97% accuracy and 0.8% false positives, and 5.8% false negatives.

This study emphasizes the effectiveness of contextual embeddings over traditional TF-IDF or bag-of-words models. By capturing the semantics and syntactic structure of input queries, BERT offers a robust solution for identifying both standard and obfuscated SQLi attempts.

**2.5.3.2 MLP vs. LSTM Comparative Study (Peng Tang et al.)**

Tang et al. did a successful study by comparing Multi-Layer Perceptron and Long Short-Term Memory networks for exploiting SQL injections in real ISP traffic data. The study involved feature extraction of eight handcrafted features from URL payloads: number of keywords, number of special characters, length of the payload, and so on.

The MLP with three hidden layers gave excellent results, with an accuracy of 99.67%, precision of 100%, and recall of 99.41%. On the other hand, LSTM, on its promise in sequential learning, gave accuracy results of 97.68% with heavy training time, thus being inefficient in this task. It was concluded that the feature-rich MLPs were much more efficient in this task, but the LSTM had potential in more complex scenarios.

**2.5.3.3 CNN-LSTM Hybrid with Risk Prioritization (Alan Paul et al.)**

Alan Paul et al. proposed an all-encompassing framework, i.e., SQLR34P3R, which puts the problem of SQL injection detection into a multi-class classification setting. The system detects the SQLi, prioritizes the attacks, and aids the prevention strategies. The system consists of a CNN-LSTM hybrid model, which is trained on a massive dataset of over 520,000 samples collected from the web and network traffic, attaining an average f-score of 97%.

Unlike previous approaches, SQLR34P3R performs contextual risk assessment based on known CVE vulnerabilities and operates its detection engine in real time, catching traffic from platforms such as DVWA and Vulnerado. The work is unique in that it combines threat intelligence with deep learning for a comprehensive and usable solution.

**2.5.4 Comparative Discussion**

These three contributions reflect the maturation of SQL injection detection research:

* **Lakhani et al.** demonstrated how pre-trained language models can improve detection with minimal feature engineering.
* **Tang et al.** emphasized the efficiency of classical MLPs combined with statistical features from real traffic.
* **Alan Paul et al.** introduced a real-time system combining deep learning with vulnerability assessment.

While MLPs are computationally light and achieve high accuracy, models like BERT or CNN-LSTM provide deeper semantic understanding and adaptability to novel threats. The choice between these methods depends on the deployment context: lightweight detection at the edge (MLP), semantic understanding (BERT), or full-stack risk management (CNN-LSTM).

**2.5.5 Summary**

In sum, these works provide promising evidence of how deep learning has been effectively used to detect SQL injection attacks with high precision and recall. In addition, they show that the combination of data-driven approaches and domain knowledge (CVE, traffic patterns) can help maximize performance and applicability. The following chapter details the implementation and experimental evaluation of our own SQLi detection models, including both traditional and transformer-based models.