Automated Trust-Aware Software Vulnerability Scoring via Explainable Feature Alignment

Seyedeh Leili Mirtaheri
Department of Informatics, Modeling, Electronics and System Engineering
University of Calabria
Rende, Cosenza, Italy
leili.mirtaheri@dimes.unical.it

Amirhossein Majd
Department of Informatics, Modeling, Electronics and System Engineering
University of Calabria
Rende, Cosenza, Italy
amirhossein.majd@dimes.unical.it

Reza Shahbazian
Department of Humanities
University of Palermo
Palermo, Palermo, Italy
reza.shahbazian@unipa.it

Andrea Pugliese
Department of Informatics, Modeling, Electronics and System Engineering
University of Calabria
Rende, Cosenza, Italy
andrea.pugliese@unical.it

Abstract

Transformer-based models such as BERT achieve strong accuracy in predicting vulnerability severity, but their black-box nature raises concerns about alignment with expert reasoning. Accuracy alone may therefore give a misleading view of model reliability. This paper introduces a post-hoc auditing framework that evaluates trust by measuring the semantic alignment between tokens identified via Integrated Gradients and the official CVSS definitions. The framework computes weighted similarities, applies adaptive thresholding, and integrates a dispersion penalty to derive a quantitative trust score, offering interpretable feedback for human review. Experiments on the National Vulnerability Database (NVD) and a Reduced Annotated Dataset (RAD) with BERT-based classifiers across eight Common Vulnerability Scoring System (CVSS) base metrics show that models with similar accuracy can differ in trust scores by more than 35%, revealing critical gaps in reliability. These findings highlight the need to complement accuracy with trust evaluation for interpretable and dependable automation in software vulnerability assessment.

CCS Concepts

• **Security and privacy** → **Software and application security**; **Vulnerability management**; • **Computing methodologies** → *Machine learning*; *Natural language processing*; Model verification and validation.

Keywords

Software Security, Vulnerability Scoring, Trustworthiness, Transformer, Model Explainability, CVSS

ICAAI 2025, Manchester, United Kingdom

ACM ISBN 979-8-4007-2104-5/25/11
https://doi.org/10.1145/3787279.3787294

ACM Reference Format:
Seyedeh Leili Mirtaheri, Amirhossein Majd, Reza Shahbazian, and Andrea Pugliese. 2025. Automated Trust-Aware Software Vulnerability Scoring via Explainable Feature Alignment. In *2025 9th International Conference on Advances in Artificial Intelligence (ICAAI 2025), November 14–16, 2025, Manchester, United Kingdom.* ACM, New York, NY, USA, 5 pages. https://doi.org/10.1145/3787279.3787294

1 Introduction

Software security increasingly requires rapid and accurate vulnerability assessment to mitigate threats. Manual scoring within the Common Vulnerability Scoring System (CVSS) remains a prevalent practice across many organizations; however, it is a time-consuming process that may delay patch deployment [3, 7, 10]. Since 2005, experts from the FIRST community have manually determined vulnerability severity. This task may take months [5] after a Common Vulnerabilities and Exposures (CVE) is published by MITRE[1]. Automating this process with machine learning models can improve efficiency, but high accuracy alone is insufficient: in critical domains, models must also be trustworthy, meaning that their decisions are grounded in meaningful criteria and aligned with expert reasoning [23]. The common approach to automated CVSS prediction leverages textual descriptions of vulnerabilities in the National Vulnerability Database (NVD). These data have been used across a spectrum of methods, ranging from classical approaches (SVM, Random Forest) to deep networks (CNN, LSTM) [6, 10], and more recently, transformer-based models such as BERT for predicting CVSS metrics[14, 24]. Despite success in terms of accuracy, a fundamental question remains: are these predictions genuinely based on expert-relevant signals, and can they be trusted? Transformer models, due to their black-box nature, offer limited transparency. Explainable AI (XAI) has emerged as a key tool to reveal which signals the model relies on. If the model's salient outputs (e.g., keywords in a CVE description) align with official and well-established criteria, the model's predictions may be deemed trustworthy; otherwise, high accuracy can be misleading [12, 18]. In this context,

[1]MITRE manages the CVE system, which assigns standardized identifiers (CVE IDs) to publicly known cybersecurity vulnerabilities

we introduce an innovative XAI-based framework that serves as a post-hoc auditor of trust. After predictions are generated, it evaluates whether the model's internal reasoning truly corresponds to the official CVSS definitions. In doing so, our framework bridges the gap between apparent accuracy and expert-grounded trustworthiness, providing a practical step toward the safe deployment of automated systems in software security.

Problem Statement. In just the first half of 2024, the number of reported vulnerabilities increased by 30% compared to the same period in 2023, with approximately 0.91% of them reaching the "weaponized" stage according to Qualys [1]. This trend highlights that exclusive reliance on manual scoring is unsustainable. At the same time, although modern deep learning models achieve high accuracy, they operate as black boxes, leaving it unclear which evidence underlies their decisions. Common indicators such as accuracy fail to measure true trust in the model. This study addresses the problem of automated vulnerability scoring using transformer-based models (such as BERT). The input is the textual description of a vulnerability (CVE), and the output consists of the eight base components of CVSS v3.1: AV, AC, PR, UI, C, I, A, S. These metrics are combined through the standard CVSS formula to automatically compute the numerical base score [2]. To address this, we propose a post-hoc XAI-based framework that, after model training and prediction generation, evaluates the alignment of salient tokens with the official definitions and provides a quantitative trust score. For implementation, eight fine-tuned BERT models—one for each CVSS component—were trained on historical NVD data (Figure1). Subsequently, salient tokens are extracted using Integrated Gradients, their alignment with the official definitions is assessed, and then they are filtered through an adaptive thresholding process. Finally, a quantitative trust score is computed to indicate the extent to which a model's prediction aligns with the reasoning of security experts.

Contributions. We propose a Trust Score to quantify the alignment between model explanations and official CVSS definitions. Our method combines cosine similarity, adaptive thresholding, and a dispersion penalty to select salient tokens. The framework supports multiple perspectives through configurable definitions and is evaluated on eight fine-tuned BERT classifiers over NVD and RAD datasets. Results show that, beyond accuracy, the Trust Score highlights critical differences in expert alignment and reliably identifies low-trust cases requiring human review.

2 Related Works

With the growing body of research on automated vulnerability severity prediction and explainability of deep learning models, four main research streams can be distinguished: (a) machine learning for CVSS prediction, (b) the use of transformers and large language models (LLMs) in this domain, (c) the application of XAI methods in cybersecurity, and (d) ongoing efforts to define trust and auditing metrics for AI systems. Our framework builds upon these foundations but emphasizes post-hoc auditing of model trustworthiness rather than novel predictive architectures.

Machine learning approaches for vulnerability assessment. Significant research has leveraged Common Vulnerability and Exposure (CVE) text data for predicting CVSS components and severity. Approaches range from classical machine learning models (e.g., SVM, Random Forest) to deep learning methods such as CNNs, LSTMs, and particularly transformers like BERT, which capture contextual semantics effectively. Le et al.'s DeepCVA deployed deep neural networks on vulnerability descriptions for CVSS scoring [11]. Li et al. enhanced prediction by integrating commit histories alongside textual data [13].

Transformers and large language models for CVSS prediction. BERT-based models are prominent in this area, with Shahid et al. introducing CVSS-BERT, an ensemble of classifiers predicting CVSS base metrics, establishing transformers' effectiveness on vulnerability text [21]. Recent explorations into large language models such as GPT and T5 by Mirtaheri et al. highlight that while LLMs excel in some CVSS aspects like Attack Vector and User Interaction, fine-tuned BERT models still outperform in interpretative metrics of confidentiality and availability [16]. These insights motivate hybrid models combining the reasoning capacity of LLMs and the discriminative power of BERT classifiers.

Explainable AI in cybersecurity. The complexity of models in vulnerability assessment has spurred the adoption of XAI tools like LIME and SHAP for interpreting cybersecurity tasks, including intrusion detection and malware analysis [15, 19]. Nonetheless, their application to vulnerability prediction remains limited, as these tools primarily indicate feature importance without capturing deeper semantic or causal relationships inherent in code and technical documentation [22].

Trustworthiness and auditing in AI for cybersecurity. Trustworthiness in AI extends beyond accuracy to include robustness, transparency, and alignment with domain expert reasoning. In cybersecurity, this entails resistance to adversarial manipulation, explainability tailored to expert understanding, and prioritization in line with security practices [9, 20]. Recent works propose quantifiable trust metrics, including stability under attacks and concordance with expert consensus [4, 8, 9, 17]. Our framework extends this discourse by introducing a Trust Score to quantify the alignment between model-extracted salient tokens and official CVSS definitions, enabling rigorous post-hoc auditing and identification of low-trust cases needing human review.

3 Proposed Framework

The proposed framework consists of six sequential stages: 1-AI Model Loading, 2-Explainability Method Selection, 3-High-Attention Feature Extraction, 4-Automated Trustability Assessment, 5-Selection of High-Impact Candidates, and 6-Automated Trustability Calculation. In the following, we provide a detailed description of each stage.

3.1 AI Model Loading

This stage involves receiving a pre-trained deep learning model (e.g., fine-tuned BERT for each metric in Figure 1) as input. Our design employs a dedicated classifier for each of the eight CVSS base metrics, ensuring that predictions are made independently and inter-metric dependencies are reduced (a design choice revisited later). The system takes as input a textual vulnerability description

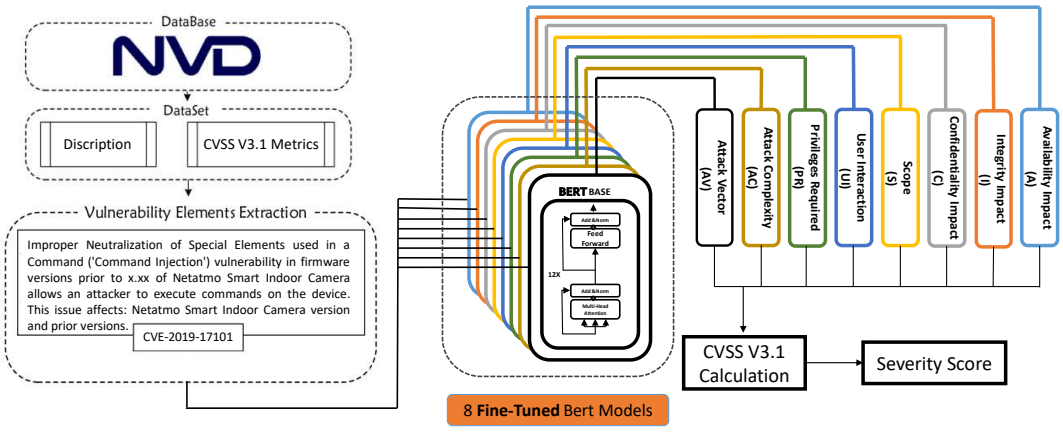

Figure 1: The general architecture of a Transformer (BERT)-based vulnerability scoring system.

(e.g., a CVE entry), and the expected output is the predicted values for the eight CVSS metrics, accompanied by a trust score for those predictions.

3.2 Explainability Method Selection

In this stage, an XAI (Explainable AI) technique is chosen and applied to identify the features most influential in the model's decision. Our approach is flexible and can accommodate any explainability method compatible with the model. In our implementation, we adopted the Captum library and specifically employed Integrated Gradients (IG). This method was chosen due to advantages such as fidelity to the model (leveraging its internal gradients), computational stability, and adherence to the principle of completeness (assigning input contributions such that their sum equals the model's output difference from a baseline input). Nevertheless, our framework is not restricted to IG and can be integrated with other explanation techniques as well. In this step, the words or input features with the highest influence on the model's prediction are extracted. For each CVE description input, IG computes the contribution of each token to the model's output. The mathematical formulation of this attribution is expressed in Eq. 1:

$$IG_i(x) = (x_i - x_i') \times \int_{\alpha=0}^{1} \frac{\partial f(x' + \alpha \cdot (x - x'))}{\partial x_i} \, d\alpha \tag{1}$$

where x is the input vector (the CVE description), and x' is the baseline (e.g., a neutral or empty input). The term IG_i represents the attribution of feature x_i (such as a specific word in the CVE text) in the final decision of the model f. Using multiple samples between x' and x, we approximate the integral numerically via the trapezoidal rule. The output of this step for each input is thus a set of salient tokens, each accompanied by a positive or negative importance score. For clarity, Algorithm 1 presents the implementation procedure, while Table I summarizes the notations and functions used. At the end of this stage, the original input is effectively distilled: noisy and low-importance tokens are filtered out, focusing attention on the critical keywords that drive the model's behavior. This not only improves interpretability but also simplifies subsequent steps and reduces computational overhead.

3.3 Semantic Similarity Assessment

After identifying the most important features, the next step is to evaluate how well these features align with the official CVSS criteria. The key idea is that if the model truly makes decisions based on expert reasoning, the high-attention words it selects should be semantically close to the formal definitions of the corresponding CVSS metrics. To this end, it is necessary to establish appropriate reference definitions for each metric.

For similarity assessment, each extracted feature from a CVE description and each reference definition are mapped into a semantic vector space. Specifically, we use *text embeddings*, which transform words or sentences into vector representations (e.g., output embeddings from BERT or GloVe). Suppose vector $A = [A_1, A_2, \ldots, A_n]$ represents a feature (or a set of features), and vector $B = [B_1, B_2, \ldots, B_n]$ represents a reference definition. To compare A and B, we employ *Weighted Cosine Similarity (WCS)*:

$$\text{WCS}(A, B) = \frac{\sum_{i=1}^{n} w_{A_i} A_i B_i}{\sqrt{\sum_{i=1}^{n} (w_{A_i} A_i)^2} \sqrt{\sum_{i=1}^{n} B_i^2}} \tag{2}$$

In this formulation, w_{A_i} is the weight assigned to the i-th component of vector A. These weights emphasize repeated or highly important features. The WCS score ranges between $[-1, +1]$, where $+1$ indicates perfect alignment (maximum semantic similarity) and -1 indicates complete opposition (absolute dissimilarity).

3.4 High-Impact Candidates

To identify keywords and final candidates for trustworthiness assessment, it is essential to establish a threshold that determines which attributes qualify as primary candidates for subsequent actions. The threshold should be adaptive, according to the data characteristics and the individual challenge, to ensure optimal outcomes. Let the computed cosine similarity values be arranged in a descending order, denoted as $\{x_1, x_2, x_3, \ldots, x_n\}$. As shown in Figure **??**, we compute the first quartile (Q_1) and the third quartile (Q_3) employing Eq. 3 to enhance our comprehension of the data distribution:

$$Q_1 = x_{\left(\frac{n+1}{4}\right)}, \; Q_3 = x_{\left(\frac{3(n+1)}{4}\right)}, \; IQR = Q_3 - Q_1. \tag{3}$$

The first quartile (Q_1) and the third quartile (Q_3) denote the thresholds below which 25% and 75% of the data are situated, respectively. The interquartile range IQR quantifies the dispersion of data between the first and third quartiles, facilitating comprehension of data variability. The adaptive multiplier is calculated using the ratio of the standard deviation (σ) for all cosine similarity values to the interquartile range (IQR), as presented in Eq. 4:

$$\text{Adaptive Multiplier} = \frac{\sigma}{IQR} = \frac{\sqrt{\frac{1}{n}\sum_{i=1}^{n}(x_i - \mu)^2}}{IQR}. \tag{4}$$

where μ represents the mean value.

The adaptive threshold is ultimately computed using Eq. 5:

$$\text{Threshold}_{IQR} = Q_3 + \text{Adaptive Multiplier} \times IQR. \tag{5}$$

3.5 Automated Trustability

To compute the overall trustworthiness, we propose a formulation that is based on the cosine similarities between the selected candidates and the adaptive threshold T. Let (d_i, d_j) be the set of feature pairs whose cosine similarity is equal to or greater than the threshold T as follows:

$$D_t = \{(d_i, d_j) \in D \mid \text{cosineSimilarity}(d_i, d_j) \geq T\},$$

where D_t contains the set of cosine similarities that remain after applying the threshold, D_{all} includes all computed cosine similarities for the extracted words, d_i and d_j are the vectors generated for the words and definitions, respectively. The model's trustworthiness is then calculated using Eq. 6:

$$\text{Trust}_{\text{model}} = (\alpha \cdot S_t) + (\beta \cdot S_{\text{all}}) \left(1 - \frac{\delta_t}{\delta_{\text{all}}}\right), \tag{6}$$

where S_t is the mean of the similarities in the set D_t, S_{all} is the mean of the similarities in the set D_{all}, δ_t is the standard deviation of similarities in D_t, and δ_{all} is the standard deviation of similarities in D_{all} defined as follows:

$$S_t = \frac{1}{|D_t|} \sum_{(d_i,d_j)\in D_t} \text{cosineSimilarity}(d_i, d_j),$$

$$S_{\text{all}} = \frac{1}{|D_{\text{all}}|} \sum_{(d_i,d_j)\in D_{\text{all}}} \text{cosineSimilarity}(d_i, d_j),$$

$$\alpha = \frac{\delta_{\text{all}} \cdot S_t}{\delta_t \cdot S_{\text{all}} + \delta_{\text{all}} \cdot S_t}, \quad \beta = \frac{\delta_t \cdot S_{\text{all}}}{\delta_t \cdot S_{\text{all}} + \delta_{\text{all}} \cdot S_t}.$$

4 Experimental Results

We evaluated the performance of the proposed framework on (1) the original NVD dataset, containing complete vulnerability descriptions, and (2) the reduced RAD dataset, in which high-importance keywords were removed to assess model robustness.

The performance of fine-tuned BERT classifiers across the eight CVSS base metrics (AV, AC, PR, UI, S, C, I, A) is summarized in Table 1. As shown in Table 1, the models trained on NVD achieve strong accuracy and F1-scores (up to 0.96 and 0.92, respectively), confirming their predictive power. Interestingly, performance on RAD remains comparable, even though this dataset was created by deliberately eliminating high-importance keywords. This suggests that the models can still reach good accuracy by relying on alternative patterns, which raises concerns: accuracy alone may mask a lack of alignment with expert-relevant features, highlighting the need for explicit trust evaluation.

Table 1: Performance Comparison of Models on NVD Dataset and RAD Dataset. The Accuracy, Recall, Precision, F1-Score, and Cohen's Kappa are reported for 8 fine-tuned BERT models over the CVSS vectors.

	Accuracy		Recall		Precision		F1-Score		Cohen's Kappa	
Model	NVD	RAD	NVD	RAD	NVD	RAD	NVD	RAD	NVD	RAD
AV	0.904	0.89	0.654	0.66	0.793	0.76	0.698	0.70	0.738	0.73
AC	0.943	0.95	0.722	0.74	0.864	0.81	0.773	0.77	0.548	0.54
PR	0.814	0.81	0.684	0.72	0.721	0.77	0.701	0.74	0.616	0.62
UI	0.901	0.93	0.894	0.92	0.893	0.93	0.893	0.92	0.787	0.85
S	0.963	0.96	0.904	0.92	0.946	0.94	0.923	0.93	0.847	0.86
C	0.840	0.83	0.788	0.80	0.837	0.81	0.809	0.80	0.711	0.70
I	0.870	0.80	0.848	0.80	0.875	0.81	0.859	0.81	0.787	0.69
A	0.902	0.87	0.621	0.67	0.714	0.83	0.632	0.70	0.798	0.75

4.1 Robustness Analysis with RAD Dataset (Key Feature Removal)

Table 2 reports the trustability comparison of models trained on NVD versus RAD across all CVSS metrics and levels. Since the RAD dataset was created by deliberately removing high-attention keywords, one would expect degraded performance in terms of semantic alignment. Indeed, while conventional accuracy metrics (see Table 1) suggested only minor differences between NVD and RAD models, Table 2 highlights a large drop in trust scores. For example, for AV:N, the NVD model achieves 70.21 compared to only 25.07 for RAD; for Scope at Changed (S:C), NVD reaches 85 while RAD drops to 37. These results confirm that although RAD models may maintain acceptable accuracy, they do so by relying on alternative or superficial patterns rather than definitions aligned with CVSS semantics.

4.2 Model Trust Score Evaluation

The proposed framework assigns a trust score to each prediction, quantifying the alignment between high-importance model features and official CVSS definitions. Multiple sources were considered: Def.1 and Def.2 (two official CVSS references), their combination, and a GPT-generated definition (Def.GPT). The evaluation was conducted in both Full (metric + level) and Label (level only) modes to test robustness across definition richness. As shown in Table 2, NVD consistently outperforms RAD across all metrics and definition sources. Confidentiality at High (C:H) illustrates this clearly, with NVD scoring nearly 90 while RAD remains close to 13. Even when RAD maintains similar predictive accuracy, its semantic basis is much weaker. Averaging across optimal references, the NVD models achieve roughly 37% higher trust scores than RAD, underscoring that accuracy alone is insufficient for evaluating decision quality.

5 Conclusion

We proposed an automated framework to evaluate the trustworthiness of AI models in vulnerability scoring by aligning their reasoning with expert-defined CVSS metrics. Beyond accuracy, our approach combines explainability, adaptive thresholding, and

Table 2: Average Trustability Comparison Across NVD and RAD Datasets.

Metric	Label	NVD	RAD
Attack Vector (AV)	Network (N)	70.21	25.07
	Physical (P)	65.07	25.69
	Local (L)	69.58	25.65
	Adjacent (A)	68.22	14.24
Attack Complexity (AC)	High (H)	59.21	27.27
	Low (L)	51.42	24.70
Privileges Required (PR)	None (N)	69.88	40.79
	Low (L)	70.13	38.42
	High (H)	62.09	42.46
User Interaction (UI)	None (N)	78.55	41.18
	Required (R)	75.36	53.80
Scope (S)	Changed (C)	85.36	37.84
	Unchanged (U)	86.25	36.61
Confidentiality (C)	None (N)	68.92	13.82
	Low (L)	72.59	14.74
	High (H)	89.57	13.86
Integrity (I)	None (N)	69.55	29.32
	Low (L)	69.01	26.67
	High (H)	67.26	31.49
Availability (A)	None (N)	51.60	31.92
	Low (L)	46.83	31.83
	High (H)	43.70	31.89

similarity-based scoring to expose when models rely on irrelevant cues. Experiments on NVD and RAD show that trust scores can diverge by over 37% despite similar accuracy, underscoring the risk of accuracy-only evaluation. This highlights the value of trust scores as a complementary measure for identifying reliable models. Future work will focus on real-time trust monitoring, integration with larger LLMs, and extending the framework to other high-stakes domains.

References

[1] S. Abbasi. 2024. 2024 Midyear Threat Landscape Review. Qualys Security Blog. https://blog.qualys.com/vulnerabilities-threat-research/2024/08/06/2024-midyear-threat-landscape-review Accessed: 2025-01-14.

[2] Manuj Aggarwal. 2023. A Study of CVSS v4.0: A CVE Scoring System. In *2023 6th International Conference on Contemporary Computing and Informatics (IC3I)*, Vol. 6. IEEE, 1180–1186.

[3] Mohammad Ali, Ahsan Ullah, Md Rashedul Islam, and Rifat Hossain. 2025. Assessing of Software Security Reliability: Dimensional Security Assurance Techniques. *Computers & Security* 150 (2025), 104230.

[4] Aseel Alshuaibi, Mohammed Almaayah, and Aitizaz Ali. 2025. Machine Learning for Cybersecurity Issues: A Systematic Review. *Journal of Cyber Security and Risk Auditing* 2025, 1 (2025), 36–46. doi:10.63180/jcsra.thestap.2025.1.4

[5] Haipeng Chen, Jing Liu, Rui Liu, Noseong Park, and V. S. Subrahmanian. 2019. VEST: A System for Vulnerability Exploit Scoring & Timing. In *Proceedings of the 28th International Joint Conference on Artificial Intelligence (IJCAI)*. 6503–6505.

[6] Neophytos Christou, Di Jin, Vaggelis Atlidakis, Baishakhi Ray, and Vasileios P. Kemerlis. 2023. IvySyn: Automated Vulnerability Discovery in Deep Learning Frameworks. In *32nd USENIX Security Symposium (USENIX Security 23)*. 2383–2400.

[7] Sarah Elder, Md Rayhanur Rahman, Gage Fringer, Kunal Kapoor, and Laurie Williams. 2024. A Survey on Software Vulnerability Exploitability Assessment. *Comput. Surveys* 56, 8 (2024), 1–41.

[8] Tanikonda et al. 2025. AI-Based Continuous Compliance Monitoring Framework with High Detection Accuracy and Reduced Response Times in Healthcare. *World Journal of Advanced Research and Reviews* 26, 3 (2025), 2249–2255.

[9] E. Giunchiglia et al. 2023. Trustworthy AI Metrics: Quantitative Assessment for Security Applications. *ACM Transactions on Information and System Security* (2023).

[10] Philipp Kuehn, David N. Relke, and Christian Reuter. 2023. Common Vulnerability Scoring System Prediction Based on Open Source Intelligence Information Sources. *Computers & Security* 131 (2023), 103286.

[11] T. H. M. Le, D. Hin, R. Croft, and M. A. Babar. 2021. DeepCVA: Automated Commit-level Vulnerability Assessment with Deep Multi-task Learning. In *Proceedings of the 36th IEEE/ACM International Conference on Automated Software Engineering (ASE)*. IEEE, 717–729.

[12] Yong Li et al. 2021. Trustworthy AI: A Survey on Ensuring Trust in Artificial Intelligence Systems. *Journal of AI Research* 70 (2021), 1–35.

[13] Y. Li, A. Yadavally, J. Zhang, S. Wang, and T. N. Nguyen. 2023. Commit-level, Neural Vulnerability Detection and Assessment. In *Proceedings of the 31st ACM Joint European Software Engineering Conference and Symposium on the Foundations of Software Engineering (ESEC/FSE)*. ACM, 1024–1036.

[14] Peiyu Liu, Junming Liu, Lirong Fu, Kangjie Lu, Yifan Xia, Xuhong Zhang, Wenzhi Chen, Haiqin Weng, Shouling Ji, and Wenhai Wang. 2024. Exploring ChatGPT's Capabilities on Vulnerability Management. In *33rd USENIX Security Symposium (USENIX Security 24)*. 811–828.

[15] Scott Lundberg and Su-In Lee. 2017. A Unified Approach to Interpreting Model Predictions. *arXiv preprint arXiv:1705.07874* (2017). arXiv:1705.07874

[16] S. L. Mirtaheri, A. Pugliese, N. Movahedkor, and A. Majd. 2024. Advanced Automated Vulnerability Scoring: Improving Performance with a Fine-tuned BERT-CNN Model. In *Proceedings of the 2024 11th International Symposium on Telecommunications (IST)*. IEEE, 109–113.

[17] VZ Mohale. 2025. A Systematic Review on the Integration of Explainable AI to Enhance Intrusion Detection Systems. *Frontiers in Artificial Intelligence* (2025), 1526221. doi:10.3389/frai.2025.1526221

[18] Alex Radiuk et al. 2024. Toward Explainable Deep Learning in Healthcare Through Transition Matrix Approaches. *Frontiers in Artificial Intelligence* 8 (2024), 1482141.

[19] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2016. "Why Should I Trust You?": Explaining the Predictions of Any Classifier. In *Proceedings of the 22nd ACM SIGKDD International Conference on Knowledge Discovery and Data Mining*.

[20] Marco Tulio Ribeiro, Sameer Singh, and Carlos Guestrin. 2020. Beyond Accuracy: Behavioral Testing of NLP Models with CheckList. In *Proceedings of the 58th Annual Meeting of the Association for Computational Linguistics (ACL)*.

[21] Mustafizur R. Shahid and Hervé Debar. 2021. 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)*. IEEE, 1600–1607.

[22] A. Vizel, N. Feldman, and E. Yahav. 2023. Explainable AI for Vulnerability Assessment: Challenges and Opportunities. *IEEE Security & Privacy* 21, 2 (2023).

[23] Zijing Zhang, Vimal Kumar, Bernhard Pfahringer, and Albert Bifet. 2025. AI-Enabled Automated Common Vulnerability Scoring from Common Vulnerabilities and Exposures Descriptions. *International Journal of Information Security* 24, 1 (2025), 1–20.

[24] Xin Zhou, Ting Zhang, and David Lo. 2024. Large Language Model for Vulnerability Detection: Emerging Results and Future Directions. In *Proceedings of the 2024 ACM/IEEE 44th International Conference on Software Engineering: New Ideas and Emerging Results*. 47–51.