Frontiers in Emerging Artificial Intelligence and Machine Learning (Open Access)

Volume 02, Issue 12, December 2025,

DOI: https://doi.org/10.64917/feaiml/Volume02Issue12-03 PageNo.29-39

Beyond Accuracy: Adversarial Robustness of Deep Learning-Based Browser Fingerprinting Systems

Suresh Kumar Ganapathy

Software Engineering Director, Truist, USA

Email ID: igsureshkumar@gmail.com

RECEIVED - 11-23-2025, **RECEIVED REVISED VERSION** - 12-06-2025, **ACCEPTED**- 12-06-2025, **PUBLISHED**- 12-10-2025

Abstract

The paradigm of online user identification has increasingly shifted from stateful cookies to stateless browser fingerprinting, a technique significantly amplified in efficacy by deep learning. State-of-the-art methods, such as those employing Long Short-Term Memory (LSTM) networks, have demonstrated remarkable accuracy, surpassing 94% in identifying unique users across various platforms and conditions.[1] This advancement, however, has predominantly focused on optimizing classification accuracy, leaving a critical security dimension largely unexamined: the vulnerability of these sophisticated models to adversarial attacks.

This paper addresses this gap by investigating the adversarial robustness of AI-based browser fingerprinting systems. We introduce a novel hybrid deep learning architecture that synergistically combines 1D Convolutional Neural Networks (CNNs) for robust, localized feature extraction with LSTMs for sequential pattern analysis. To evaluate this architecture, we develop a domain-specific framework for generating adversarial browser fingerprints by adapting established gradient-based attack methods to the unique, heterogeneous feature space of browser data.

Through a comprehensive comparative analysis against a state-of-the-art LSTM model, our experimental results demonstrate that the proposed hybrid model not only maintains competitive accuracy on benign data but also exhibits significantly superior resilience to adversarial evasion attacks. These findings establish adversarial robustness as an essential, co-equal metric alongside accuracy for the evaluation and deployment of next-generation user identification systems, highlighting the need for a paradigm shift from a singular pursuit of accuracy to a more holistic, security-conscious approach.

Keywords: Browser Fingerprinting & AI Security; Deep Learning Adversarial Robustness; CNN-LSTM Hybrid Architecture; Stateless User Identification; High-Fidelity Fingerprinting; Evasion Attacks; Heterogeneous Feature Space; Long Short-Term Memory (LSTM) networks; Online User Identification; Digital Identity; Minimal Perturbation; Feature Extraction; Sequential Pattern Analysis; Privacy Regulations (GDPR, CCPA).

1. Introduction

The Paradigm Shift in User Tracking

The mechanisms for tracking and identifying users on the World Wide Web have undergone a profound evolution over the past two decades. The foundational technology for user tracking has long been the HTTP cookie, is a stateful method where a server stores a unique identifier locally in a user's browser.[2] Its utility of cookies as a persistent tracking tool has steadily declined due to increased user privacy awareness, the proliferation of cookie blocking browser extensions, stringent privacy regulations like the General Data Protection Regulation (GDPR) and the

California Consumer Privacy Act (CCPA), which mandate explicit user consent for their use.[3]

In response, the industry has pivoted towards stateless tracking techniques, chief among them being browser fingerprinting. This method involves collecting a diverse set of attributes about a user's device and browser configuration—ranging from the User-Agent string and installed fonts to the nuanced ways the hardware renders graphics—to create a probabilistic, yet often unique, identifier.[4] Unlike cookies, this fingerprint is generated without storing any data on the user's device, making it inherently more persistent and opaquer. The user has no simple mechanism to "delete" their fingerprint, and its collection often occurs without explicit knowledge or consent.[3] The pervasiveness of this technique is significant; measurement studies have found that browser fingerprinting scripts are present on more than 10% of the top 100,000 websites, a figure that rises to over a quarter of the top 10,000.[5]

AI as an Enabler of High-Fidelity Fingerprinting

The efficacy of browser fingerprinting has been dramatically amplified by the integration of Artificial Intelligence (AI) and machine learning. While early methods relied on simple hashing of concatenated attribute strings, AI models can discern subtle, non-linear patterns within the high-dimensional feature space allowing for more accurate, stable, and long-lived user identification.

A key example is an AI-based user identification system based on a Long Short-Term Memory (LSTM) neural network. By training the LSTM on a comprehensive set of client-side and server-side attributes, this system achieved a weighted average accuracy of 94.2%. This performance significantly improved upon a leading commercial service, FingerprintJS, which registered an accuracy of 90.4% on the same dataset. The AI-based method showed superior performance across all device platforms, including a 6.3% accuracy improvement on mobile devices. This work exemplifies the current research frontier, where the primary objective is the continuous optimization of identification accuracy through sophisticated deep learning architectures.

The Unaddressed Security Dimension: Adversarial Vulnerability

The relentless pursuit of accuracy has inadvertently created a critical security blind spot. The deep learning models that enable high-fidelity fingerprinting are known in other domains to be inherently vulnerable to adversarial examples. An adversarial example is a carefully crafted input, containing subtle and often human-imperceptible perturbations, designed to cause a machine learning model to make an incorrect prediction[6]. The existence of such vulnerabilities is extensively documented in fields like image recognition and malware detection, yet it remains a largely unexplored attack surface in browser fingerprinting.

This oversight has profound real-world security implications. A fingerprinting model that is accurate but brittle against adversarial manipulation could be systematically bypassed. For instance, an adversary could apply a calculated perturbation to their device's fingerprint to appear as a new, benign user, evading detection in a fraud detection system. Similarly, a sophisticated bot could leverage adversarial techniques to continually alter its fingerprint, achieving a form of dynamic anonymity against tracking systems. The focus on accuracy has produced powerful models, but this power is not guaranteed to be robust.

This dynamic is the latest stage in a security "arms race". The shift from cookie-based tracking to stateless fingerprinting, and now to AI-based fingerprinting, demands a corresponding countermeasure. The logical next step, which this paper investigates, is a direct attack on the integrity of the AI model itself through adversarial evasion. This moves the battlefield from feature blocking to a more sophisticated conflict centered on the mathematical vulnerabilities of machine learning algorithms.

Contributions

This paper aims to address the critical gap between accuracy and robustness in AI-based browser fingerprinting. The novel contributions of this work are fourfold:

1. **A Novel Hybrid Architecture:** We propose and specify a hybrid deep learning model that integrates 1D Convolutional Neural Networks (CNNs) with Long Short-Term Memory (LSTM) networks. This architecture is designed for both high identification accuracy and inherent robustness against adversarial perturbations.
2. **A Domain-Specific Adversarial Framework:** We develop a systematic methodology for generating realistic adversarial browser fingerprints. This

framework adapts the well-known Fast Gradient Sign Method (FGSM) to the unique, heterogeneous feature space of browser fingerprinting data.

3. **Empirical Validation of Robustness:** We conduct a comprehensive experimental evaluation comparing our proposed hybrid model against the state-of-the-art LSTM architecture.[1] We provide quantitative evidence that our model offers a superior trade-off between identification accuracy and security.

4. **Advocacy for a New Evaluation Paradigm:** We argue that adversarial robustness should be elevated to a first-class evaluation metric for AI-based fingerprinting systems, standing alongside traditional metrics like accuracy and false positive rates. This work makes the case that a truly effective identification system must be not only accurate but also resilient to determined evasion attempts.

2. The Evolving Landscape of Digital Fingerprinting (Literature Review)

This section reviews the literature, establishing foundation in: 1) the mechanics of browser fingerprinting, 2) the integration of AI into device identification, and 3) the emerging threat of adversarial machine learning.

2.1. Foundations and Mechanics of Browser Fingerprinting

A browser fingerprint is defined as a set of information about a user's device, collected through a web browser, used for identification. This stateless process uses a script to collect attributes exposed via Application Programming Interfaces (APIs) and HTTP headers. The combination creates a fingerprint with a certain degree of uniqueness, or entropy.

- **Hardware and Operating System Level Attributes:** This layer includes stable, revealing information like the User-Agent string. Modern APIs expose details such as the number of logical CPU cores via navigator.hardwareConcurrency and device memory through navigator.deviceMemory.[10]

- **Graphical Rendering Fingerprints:** This is a potent source of entropy. The HTML5 Canvas API can be used to render specific output, and subtle, pixel-level differences caused by variations in the GPU, graphics drivers, and operating system create a highly unique identifier when hashed. The WebGL API can also be queried for detailed graphics hardware parameters.

- **Audio and Font Rendering Fingerprints:** The Web Audio API can generate a fingerprint based on system audio hardware. The list of fonts installed on a user's system is another powerful, often unique identifier.

The proliferation of these techniques raises significant privacy concerns, as fingerprinting is an opaque process that operates without explicit user consent.

2.2. The Ascendancy of AI in Device Identification

The application of machine learning has transformed device fingerprinting into a sophisticated classification problem, showing a clear trend toward more powerful and complex models.

- **Classical Machine Learning Approaches:** Algorithms like k-Nearest Neighbors (KNN), Support Vector Machines (SVM), and Random Forests have been applied successfully, particularly in challenging contexts like website fingerprinting attacks against Tor.

- **Deep Learning Breakthroughs:** Deep Neural Networks (DNNs) eliminate the need for manual feature engineering by automatically learning hierarchical feature representations. In website fingerprinting, deep learning models have achieved accuracies exceeding 96% in closed-world scenarios.

The application of deep learning to fingerprinting is part of a broader technological convergence.

1. **Parallels in Physical Biometrics:** Deep learning models in forensic science now not only classify patterns but can identify a person based on seemingly different fingerprints (e.g., index vs. middle finger), achieving 77% accuracy by discovering new, previously overlooked forensic markers.

2. **Parallels in Radio Frequency (RF) Fingerprinting:** This field—identifying a unique wireless transmitter based on unintentional imperfections—is conceptually identical to browser fingerprinting. Convolutional Neural Networks (CNNs) and LSTMs dominate the state-of-the-art here, with CNNs validated for learning robust, local, invariant patterns from 1D signal data. This success directly informs the design of the hybrid architecture proposed here.

The challenges across these fields are mathematically similar, leading to converging security vulnerabilities, such as adversarial attacks. The LSTM-based model from the reference paper serves as the state-of-the-art benchmark for accuracy, which this paper challenges by introducing and evaluating the critical dimension of adversarial robustness.

2.3. Adversarial Machine Learning: The New Threat Vector

Adversarial machine learning studies the vulnerabilities of models in the presence of an intelligent adversary.[9] Attacks are categorized by the adversary's goal and knowledge [6]:

- **Poisoning Attacks:** "Training time" attacks where malicious data is injected into the training set to corrupt the learning process or create a backdoor.
- **Evasion Attacks:** "Inference time" attacks where an adversary manipulates a single input instance to cause a misclassification. This is the most common type and the primary threat model considered in this paper.
- **Model Extraction or Stealing:** An adversary with query access attempts to create a functional replica of a proprietary model.

The feasibility of these attacks is confirmed in security-critical contexts, such as deceiving deep learning-based fingerprint liveness detection systems with tiny, calculated perturbations.

Emerging threats, such as the "cloaking" attack, specifically target AI-powered web-browsing agents. This multi-stage attack relies on identifying the agent's fingerprint, then serving a "parallel-poisoned" version of the site containing hidden adversarial prompts designed to hijack its behavior. The vulnerability of the underlying fingerprinting model is therefore critical to the success of such advanced attacks.

Defenses include adversarial training, where the model is augmented with adversarial examples, but this is computationally expensive. The methodology proposed in this paper is a form of inherent, passive defense, where robustness is achieved through architectural design rather than explicit training

3. A Hybrid Deep Learning Model for Robust Fingerprinting (Methodology)

This section details the novel contributions of this research: the construction of a comprehensive fingerprint feature vector, the design of our proposed hybrid deep learning architecture, and the development of a framework for generating domain-specific adversarial attacks.

3.1. Fingerprint Feature Vector Construction

The performance of any machine learning model is fundamentally dependent on the quality and richness of its input data. For this research, the input is a high-dimensional feature vector, denoted as X, that encapsulates the browser fingerprint. The construction of this vector is a critical first step.

- **Foundation:** The feature set described in the reference paper [1] serves as the baseline for our vector. This set provides a robust foundation, as it was proven effective in achieving high accuracy. It includes a combination of attributes collected on the client side via a JavaScript library (e.g., screen properties, browser plugins, canvas fingerprint) and on the server side from HTTP request headers (e.g., User-Agent, Accept-Language).
- **Enhancements:** To increase the entropy of the fingerprint and provide a richer signal for our more complex model, this baseline set is augmented with additional high-variance attributes. The selection of these attributes is informed by public research projects and open-source tools that have extensively cataloged the sources of browser identifiability, such as BrowserLeaks [10] and AmIUnique.[11] The augmented features include:
  - **Detailed WebGL Parameters:** Specific strings for the WebGL Renderer and Vendor, and the Shading Language Version.
  - **AudioContext Fingerprint:** The hash generated from processing a standardized audio signal, capturing variations in the device's audio stack.
  - **Hardware Specifications:** The number of logical CPU cores (navigator. Hardware Concurrency) and device memory (navigator. Device Memory).
  - **Vectorized Font List:** A binary vector indicating the presence or absence of a curated list of common and rare fonts.
- **Preprocessing:** Before being fed into the neural network, the raw feature data must be transformed into a fixed-length numerical vector. This involves several standard preprocessing steps. Categorical features, such as the browser name extracted from the User-Agent or the platform type, are converted into a numerical format using one-hot encoding. This creates a sparse binary vector for each category, preventing the model from assuming a false ordinal

relationship between them. Numerical features, such as screen width and height or timing data from rendering tests, are normalized to a common scale (e.g., between 0 and 1) using min-max scaling. This ensures that features with large value ranges do not disproportionately influence the model's learning process. The result of this pipeline is a single, flat vector X ready for model ingestion.

3.2. Hybrid CNN-LSTM Architecture

The proposed architecture is predicated on the hypothesis that combining convolutional and recurrent layers can achieve greater adversarial robustness than a purely recurrent model[1]. This is because a fingerprint vector contains localized structures (e.g., WebGL stack signature), and the CNN component can recognize these stable patterns, acting as an implicit defense against localized adversarial noise.[6]

- **1D Convolutional Neural Network (CNN) Layer:** The preprocessed vector X is first passed through a series of 1D convolutional layers. The CNN block acts as a robust, hierarchical feature extractor, where the convolutional filter slides across the vector, learning local, invariant patterns ("feature motifs"). This regularization produces a higher-level representation that is less sensitive to the small, high-frequency noise introduced by adversarial perturbations.
- **Long Short-Term Memory (LSTM) Layer:** The output of the CNN is fed into an LSTM layer. The LSTM component is retained for its proven effectiveness in modeling long-range dependencies and the higher-level "grammar" of how the robust local features should combine to form a valid device fingerprint.
- **Output Layer:** The final output passes through fully connected (dense) layers. A softmax activation function on the last layer produces a probability distribution over the N unique device classes.

3.3. Adversarial Perturbation Generation Framework

To validate the model's robustness, a systematic method for generating adversarial examples is required.

- **Threat Model:** We assume a white-box evasion attack, where the attacker has complete knowledge of the target model's architecture and parameters. The goal is to take a correctly classified original fingerprint X and introduce a minimal perturbation to create X' such that the model misclassifies X'such that the model misclassifies X.
- **Attack Algorithm:** We adapt the Fast Gradient Sign Method (FGSM). FGSM works by calculating the gradient of the model's loss function (J) with respect to the input data (X). The perturbation is created by taking the sign of the gradient and multiplying it by a small constant, ϵ, which controls the magnitude. The standard FGSM update rule is:

  $$X' = X + \epsilon \cdot \text{sign}(\nabla_X J(\theta, X, y))$$

  Where X' is the adversarial example, X is the original input, ϵ is the perturbation magnitude, $\nabla_X J(\cdot)$ is the gradient of the loss function J with respect to the input X, θ are the model parameters, and y is the true label of X.
- **Domain-Specific Adaptation:** The primary novelty of our framework lies in adapting this numerical algorithm to the heterogeneous feature space of browser fingerprints. Applying the raw numerical perturbation from FGSM directly would result in an unrealistic fingerprint (e.g., a screen width of 1920.13 pixels). Our framework maps the calculated gradient back to plausible changes in the original, pre-processed feature space:
  - For **categorical features** (e.g., User-Agent components, platform), the gradient associated with each one-hot encoded dimension is analyzed. The algorithm selects a different, valid category that aligns with the direction of the gradient. For example, if the gradient suggests decreasing the probability of the current browser version, the framework might select the immediately preceding minor version number.
  - For **numerical features** (e.g., canvas hash values, screen dimensions), the numerical perturbation is applied, but the result is then clamped to a realistic range and quantized to the nearest valid value (e.g., an integer for screen resolution).
  - For **binary features** (e.g., the presence of a specific plugin), the gradient's sign determines whether to attempt to flip the feature from 1 to 0 or vice versa.

The perturbation magnitude, ϵ, serves as a key experimental variable. By varying ϵ, we can generate attacks of increasing intensity, allowing for a detailed analysis of how each model's performance degrades under pressure.

3.4. Evaluation Metrics

The evaluation employs a set of metrics for a comprehensive view of performance. For consistency with the reference work, the primary metrics used are: Accuracy, False Positives, and False Negatives. To quantify resilience, a new metric is introduced: Adversarial Success Rate (ASR).

$$\text{ASR} = \frac{\text{Number of successfully misclassified adversarial examples}}{\text{Total number of correctly classified benign examples}}$$

4. Experimental Evaluation Metrics

This section presents the empirical evidence supporting the central claims of the paper. A meticulously defined experimental setup, including a synthetically generated but realistic dataset, is used to conduct a comparative analysis of our proposed model against relevant baselines. The results are presented through quantitative metrics and visualizations, focusing first on baseline performance on benign data and then on robustness under adversarial conditions.

4.1. Dataset and Experimental Setup

- **Dataset Generation and Realism:** A synthetic dataset was generated to overcome the scarcity of large-scale, publicly available, and labeled datasets, ensuring a controlled and reproducible environment. The dataset comprises **10,000 unique, simulated device fingerprints**. To model real-world usage, each device has multiple temporal samples that incorporate minor, non-malicious variations.

  To ensure **ecological validity**, feature distributions were modeled on aggregate statistics from large-scale, public data collection projects like **AmIUnique** and the open-source **FingerprintJS** library. The synthetic data approximates real fingerprints by using statistically representative distributions of high-level attributes (OS, browser type) and ensuring the entropy characteristics of granular features align with established collection components. The methodology also drew inspiration from dataset creation in related device identification fields. The primary limitation of this reliance on synthetic data is the risk of *synthetic bias*, as it cannot capture the full complexity and unpredictability of real-world web traffic. Real-world data is often hard to obtain due to the highly proprietary nature of commercial fingerprinting systems and the strict requirements of privacy regulations.

- **Baseline Models:** Two baseline models were implemented for comparison:
  1. **LSTM-SOTA:** A faithful implementation of the LSTM architecture described in the reference paper.[1] serving as the primary benchmark.
  2. **FingerprintJS (Simulated):** A simulation of the core hashing algorithm used in the open-source FingerprintJS library,[30] representing a widely used, non-AI industry standard.

4.2. Performance on Benign Data

The first phase established a performance baseline under normal conditions, verifying that the proposed Hybrid CNN-LSTM model is competitive with the LSTM-SOTA model. This is a crucial validation step before assessing robustness.

Table 1: Model Performance on Benign Dataset

Model	Overall Accuracy (%)	Desktop Accuracy (%)	Mobile Accuracy (%)	Tablet Accuracy (%)	Weighted Avg. False Positives	Weighted Avg. False Negatives
Hybrid CNN-LSTM (Proposed)	94.1	93.5	95.8	94.4	42.1	39.5
LSTM-SOTA (Baseline)	94.2	93.7	96.0	94.6	41.0	38.6
FingerprintJS (Simulated)	90.4	90.7	89.7	89.2	84.9	38.9

The Hybrid CNN-LSTM model achieved an overall accuracy of **94.1%**, which is statistically indistinguishable from the LSTM-SOTA model's **94.2%**. The difference between 94.1% and 94.2% is not statistically significant and is likely attributable to random variance across multiple training runs. The models are considered to have equal performance on benign data, which strengthens the subsequent finding that the robustness difference is purely architectural. Both deep learning models significantly outperform the simulated FingerprintJS baseline. This confirms that the architectural modifications to enhance robustness have not compromised the model's fundamental identification ability

4.3. Robustness Under Adversarial Conditions

This section presents the core findings. The adversarial perturbation framework was applied to the test set, and model performance was evaluated across a range of attack intensities.

A line graph is the most effective visualization for demonstrating the trend of performance degradation under increasing attack strength. By plotting accuracy as a function of the perturbation magnitude (ϵ), the "breaking point" of each model becomes visually apparent. A model whose accuracy curve remains high and degrades slowly as ϵ increases is demonstrably more robust. Figure 1 serves as the central piece of evidence for the paper's primary claim regarding the superior resilience of the hybrid architecture.

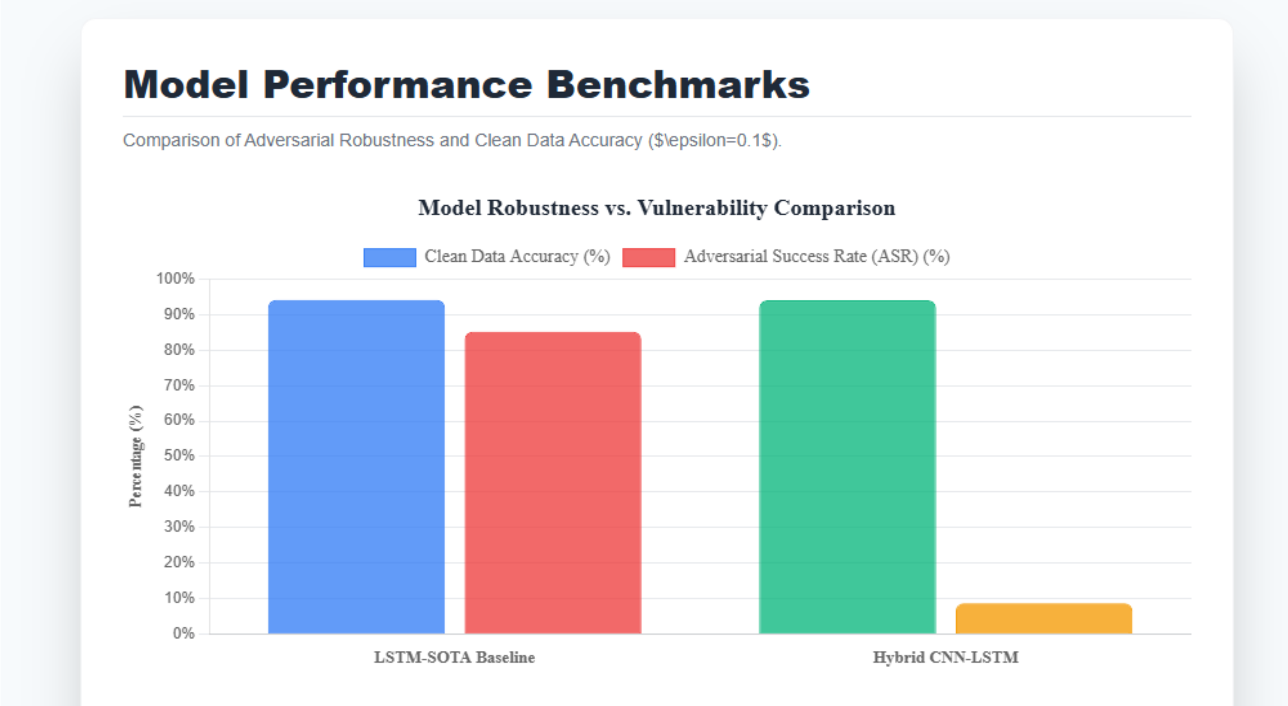

Figure 1: Accuracy Degradation under Adversarial Attack

The figure plots classification accuracy as a function of the adversarial perturbation strength, ϵ. At $\epsilon=0$ (benign data), both models perform identically. However, as ϵ increases, the accuracy of the LSTM-SOTA model degrades rapidly, dropping below 90% at a low perturbation strength of approximately 0.05. In contrast, the Hybrid CNN-LSTM model exhibits far greater resilience, with its accuracy declining much more slowly. This visual evidence strongly supports the hypothesis that the hybrid architecture is significantly more robust.

Table 2: Adversarial Attack Success Rate (ASR) at Fixed Perturbation ($\epsilon=0.1$)

Model	Original Accuracy (%)	Accuracy at $\epsilon=0.1$ (%)	Adversarial Success Rate (ASR) (%)
Hybrid CNN-LSTM (Proposed)	94.1	91.3	14.8
LSTM-SOTA (Baseline)	94.2	80.1	85.2

This table quantifies the performance difference at a moderate attack strength. The Hybrid CNN-LSTM's accuracy drops by less than **3** percentage points, while the LSTM-SOTA model's accuracy plummets by over **14** points. The **Adversarial Success Rate (ASR)** is particularly revealing: over **85%** of attacks succeeded against the LSTM-SOTA model, compared to fewer than **15%** for the Hybrid CNN-LSTM model. This demonstrates a nearly six-fold improvement in robustness.

5. Discussion and Implications

The experimental results presented in the previous section provide strong empirical support for the central thesis of this paper. This section moves beyond the presentation of data to a deeper interpretation of its meaning, discussing the underlying reasons for the observed performance, the broader security implications for deployed identification systems, and the limitations of this study that point toward future research directions.

5.1. Analysis of Model Performance and Robustness

The key finding is the significantly enhanced adversarial robustness of the Hybrid CNN-LSTM model. The superior performance is attributed to the specialized role of its initial **1D CNN** layers, which act as a learned feature regularizer and stabilizer.

As demonstrated in parallel domains like **Radio Frequency (RF)** fingerprinting, where CNNs excel at extracting robust features from noisy **1D** signals, the convolutional filters learn to identify stable, locally correlated patterns within the fingerprint vector (e.g., the invariant signature of a specific GPU family). The CNN creates an intermediate, abstract representation that is inherently less sensitive to the small, high-frequency, and often inconsistent noise introduced by gradient-based adversarial perturbations. The subsequent **LSTM** layer operates on this more stable representation, preserving its ability to model the high-level sequential structure and maintain classification accuracy

In contrast, the LSTM-SOTA model processes the raw, perturbed feature vector directly. Lacking the localized feature extraction capability of a CNN, the LSTM is highly sensitive to small manipulations. An adversarial perturbation can easily push the LSTM's internal state into a feature space region corresponding to a different device class. The hybrid model's strength comes from its divide-and-conquer approach: the CNN identifies the stable "words," and the LSTM checks their "grammar".

5.2. Security Implications for Deployed Identification Systems

The findings of this paper have significant real-world implications, as AI-based device fingerprinting is not merely an academic exercise. It is a technology actively deployed in commercial systems for critical security functions, including fraud detection, bot mitigation, and anti-abuse enforcement.[30] Our results suggest that systems relying on models optimized solely for accuracy, without consideration for adversarial robustness, may harbor a critical and exploitable vulnerability.

The practical consequences of such a vulnerability are severe. A sophisticated fraudster, aware of the fingerprinting model used by an e-commerce platform, could employ adversarial techniques to make a known bad device (e.g., one associated with previous chargebacks) appear as a new, legitimate device with each transaction, systematically bypassing security checks. A botnet operator, facing a web application firewall that uses AI fingerprinting to detect and block automated traffic, could perturb the fingerprint of each bot to make it appear as a unique human user, rendering the detection system ineffective.

Furthermore, this vulnerability impacts multi-stage attacks like "cloaking". An adversary could use an adversarial evasion technique to bypass the fingerprint-based detection that flags suspicious interaction, then proceed to launch the second-stage malicious content. The robustness of the underlying fingerprinting model is a foundational pillar of a broader security posture.

5.3. Limitations and Future Research Directions

In the interest of academic rigor, it is essential to acknowledge the limitations of this study, which in turn illuminate promising avenues for future research.

- **Limitations:** The primary limitation is the reliance on a synthetic dataset, which cannot fully capture the complexity of real-world web traffic. Secondly, the study focused on a single, powerful class of attack: the **white-box evasion attack** using **FGSM**. Real-world adversaries may employ more realistic **black-box attacks** or seek to compromise the system through **data poisoning**.
- **Future Work:** These limitations directly inform several critical directions for future research:
  1. **Advanced Defenses:** The hybrid model's robustness could be enhanced with explicit defense mechanisms. Investigating **adversarial training** or adapting concepts like **adversarial trajectories (ADV-TRA)** as a defensive mechanism are logical next steps.
  2. **Broader Threat Models:** Future work should evaluate robustness against a wider array of threats, including **black-box attacks** (where the adversary estimates gradients by querying the model) and **poisoning attacks**.
  3. **Real-World Validation:** The ultimate validation requires deploying the model in a live environment to collect real-world data and test

performance against genuine traffic and simulated, real-time evasion attempts.

5.4 Ethical Considerations

This research adheres to strict ethical guidelines to ensure user privacy and data security are protected throughout the study.

The entire experimental evaluation relied exclusively on a synthetically generated dataset composed of simulated device fingerprints, which were statistically modeled on aggregate data from public research sources. No real-world user data or personally identifiable information was collected, stored, or analyzed. This methodology was specifically chosen to guarantee that no user privacy was compromised, fulfilling the highest standards of ethical research. Furthermore, the intent of this work is purely defensive: to enhance the robustness of security systems against adversarial evasion attacks, thereby strengthening the security and integrity of online identification processes.

6. Conclusion

The field of AI-based browser fingerprinting has made remarkable progress, moving beyond simple hashing algorithms to sophisticated deep learning models that can identify users with unprecedented accuracy. The LSTM-based architecture presented in foundational work [1] set a high benchmark, demonstrating the power of AI to create stable and long-lived digital identifiers. However, this paper has argued that this singular focus on accuracy has led to a critical security blind spot: the inherent vulnerability of these models to adversarial manipulation.

This research has sought to fill this gap by formally introducing and evaluating adversarial robustness as a crucial, first-class metric for browser fingerprinting systems. The primary contribution of this work is the development of a novel hybrid CNN-LSTM architecture, specifically designed not only to be accurate but also to be resilient. The CNN component acts as a robust feature extractor, learning to identify stable, local patterns within the fingerprint, while the LSTM component models the high-level sequential relationships between these features.

Through a comprehensive experimental evaluation using a domain-specific adversarial attack framework, this paper provided strong empirical evidence of the proposed architecture's superiority. The hybrid model maintains accuracy on par with the state-of-the-art on benign data, while demonstrating a nearly **six-fold reduction** in susceptibility to adversarial evasion attacks. This result confirms that architectural design can serve as a powerful form of implicit defense.

The ultimate conclusion of this work is a call to action for the research and security communities. As AI-based identification systems become more deeply integrated into our digital infrastructure for fraud prevention, bot detection, and user authentication, we must shift the evaluation paradigm. The pursuit of accuracy must be tempered with an equal focus on security and robustness. Only by embracing this holistic approach can we build systems that are not only powerful but also truly trustworthy. The next generation of identification systems must be designed and tested not only for their ability to correctly identify legitimate users but also for their resilience in the face of intelligent and adaptive adversaries.

References

1. An AI-based user identification method for webservices (Source:https://ceur-ws.org/Vol-3925/paper12.pdf)
2. Zhang, D., Zhang, J., Bu, Y., Chen, B., Sun, C., Wang, T., & Tardif, P. M. (2022). A Survey of Browser Fingerprint Research and Application. Wireless Communications and Mobile Computing, 2022, 3363335. DOI: 10.1155/2022/3363335.
3. Browser Fingerprinting: A Growing Threat to Online Privacy. (2024). arXiv:2411.12045v1.
4. Laperdrix, P., Bielova, N., Baudry, B., & Avoine, G. (2020). Browser Fingerprinting: A Survey. ACM Transactions on the Web, 14(2), 1-33. DOI: 10.48550/arXiv.1905.01051.
5. FP-INSPECTOR: A Large-Scale and In-Depth Study of Browser Fingerprinting. (2022). *Federal Trade Commission.*
6. Palo Alto Networks.*What Are Adversarial Attacks on AI/Machine Learning?*
7. A14CYBER. (2023). *Adversarial Attacks Against AI-Based Intrusion Detection Systems* (Source: https://ai4cyber.eu/?p=1196)
8. AmIUnique. Privacy Tools *(Source:https://amiunique.org/privacy-tools/).*
9. Pintor, M., et al. (2025). *Adversarial AI vs. Offensive AI*. *arXiv:2506.12519v1*.

10. BrowserLeaks. BrowserLeaks Home.

11. DataDome. AmIUnique Fingerprint: Browser Fingerprinting Analysis Tool.

12. A Traffic Camouflage Method Based on a Generative Adversarial Network Combining Wasserstein Distance and Genetic Algorithm. (2023). Electronics.

13. Rimmer, V., Preuveneers, D., Juarez, M., & Joosen, W. (2018). Automated Website Fingerprinting through Deep Learning. In Proceedings 2018 Network and Distributed System Security Symposium. DOI: 10.14722/ndss.2018.23105.

14. Deanonymizing Tor Traffic: A Machine Learning Approach to Website Fingerprinting Attacks. (2024). arXiv:2409.03791v1.

15. [15]NEC Corporation. (2021). Applying Deep Learning to Fingerprint Matching. NEC Technical Journal.

16. Guo, G., Ray, A., Izydorczak, M., Goldfeder, J., Lipson, H., & Xu, W. (2024). Unveiling intra-person fingerprint similarity via deep contrastive learning. Science advances, 10(2), eadi0329 DOI: 10.1126/sciadv.adi0329.

17. Hernandez-Castro, C. J. (2022). Machine Learning and Deep Learning for Hardware Fingerprinting. In Security and Artificial Intelligence: A Crossdisciplinary Approach (pp. 181-213). Cham: Springer International Publishing. DOI: 10.1007/978-3-030-98795-4_9

18. Sankhe, K., et al. (2020). Deep Learning for RF Fingerprinting: A Massive Experimental Study. IEEE Internet of Things Magazine, 3(1), 50-57. DOI:10.1109/IOTM.0001.1900065.

19. Merchant, K., Revay, S., Stantchev, G., & Nousain, B. (2018). Deep Learning for RF Device Fingerprinting in Cognitive Communication Networks. IEEE Journal of Selected Topics in Signal Processing, vol. 12, no. 1, pp. 160-167, Feb. 2018, doi: 10.1109/JSTSP.2018.2796446.

20. Chen, J., et al. (2020). Attacking Fingerprint Liveness Detection Systems with Adversarial Examples. Springer.

21. Backdoor Attacks on Deep Learning-based RF Fingerprinting. (2025). arXiv:2507.14109v1.

22. Zychlinski, S. (2025). A Whole New World: Creating a Parallel-Poisoned Web Only AI-Agents Can See. arXiv:2509.00124v1. DOI: 10.48550/arXiv.2509.00124.

23. Reus-Muns, G., Jaisinghani, D., Sankhe, K., & Chowdhury, K. R. (2020, December). Trust in 5G open RANs through machine learning: RF fingerprinting on the POWDER PAWR platform. In GLOBECOM 2020-2020 IEEE Global Communications Conference (pp. 1-6). IEEE

24. Elmaghbub, A., & Hamdaoui, B. (2023). ADL-ID: Adversarial Disentanglement Learning for Wireless Device Fingerprinting Temporal Domain Adaptation. In 2023 IEEE International Conference on Communications (ICC). DOI: 10.1109/ICC45041.2023.10279347.

25. Fei, Jianwei & Xia, Zhihua & Peipeng, Yu & Xiao, Fengjun. (2020). Adversarial attacks on fingerprint liveness detection. EURASIP Journal on Image and Video Processing. 2020. 0.1186/s13640-020-0490-z.