Developing an ML-Based Solution to Refine CAPTCHA for UIDAI

Gnanavika M
20211CSG0026

R Kamal Raj
20211CSG0035

Shreyas DM
20211CSG0005

Under The Guidance Of,
Mr.Yamanappa
Department Of Computer Science And Engineering

***Abstract*—Traditional CAPTCHA systems, while effective in deterring basic automated threats, often create a cumbersome user experience and are increasingly susceptible to modern AI-driven attacks. This paper presents a machine learning (ML)-driven passive CAPTCHA alternative specifically designed for the Unique Identification Authority of India (UIDAI) portals. By passively collecting environmental and behavioral data—such as mouse dynamics, keystroke patterns, device fingerprints, and network indicators—our proposed solution leverages backend ML models to assess user authenticity in real-time. The architecture promotes minimal user interaction, seamless integration with UIDAI infrastructure, and robust security against DoS/DDoS threats, all while upholding strict privacy guidelines.**

***Index Terms*—CAPTHA, DOS, DDoS**

I. Introduction

A. Background and Motivation

CAPTCHAs (Completely Automated Public Turing tests to tell Computers and Humans Apart) have long served as a frontline defense against automated threats targeting online services. In high-stakes domains such as UIDAI, where identity verification and security are paramount, the reliance on CAPTCHAs introduces a trade-off between security and usability. However, advancements in deep learning have empowered bots to solve CAPTCHAs with human-like efficiency. Simultaneously, user frustration with difficult or inaccessible CAPTCHAs is on the rise. UIDAI aims to modernize its security framework by eliminating traditional CAPTCHAs in favor of a passive ML-driven model. This move is crucial for improving user experience while strengthening the detection and deterrence of sophisticated automated threats.

B. Significance and Objectives

The proposed solution aims to:

- Improve User Experience: Replace active CAPTCHAs with a seamless passive verification mechanism.
- Enhance Security: Use ML models to accurately detect bot activity without user involvement.
- Ensure Compliance: Maintain user privacy and adhere to UIDAI's data protection policies.
- Ensure Easy Integration: Develop a pluggable ML solution that integrates with UIDAI's existing infrastructure.

II. Literature Review

A. Bot Detection Methods

Traditional CAPTCHA methods involve tasks like deciphering distorted text or identifying specific objects in images. However, these are now easily bypassed using deep learning-based solvers. For instance, Generative Adversarial Networks (GANs), as introduced by Goodfellow et al. [1], have shown proficiency in mimicking and decoding CAPTCHA challenges with high success rates. Additionally, behavioral-based passive techniques have been explored, such as analyzing mouse and keystroke dynamics to distinguish human interactions from bots [2]. These approaches enable continuous user verification without interrupting the user journey.

B. Machine Learning for Bot Detection

Several machine learning algorithms have demonstrated efficacy in bot detection:

- Random Forests and Decision Trees: These are commonly used due to their high interpretability and effectiveness in handling tabular behavioral data [3].
- Siamese Neural Networks: These networks are well-suited for measuring similarity between session behaviors and are particularly effective in detecting subtle deviations typical in automated bot interactions [4].
- K-Nearest Neighbors (KNN): A non-parametric method that classifies sessions based on proximity to known behavioral profiles, KNN works well in clustered user activity environments [5].

C. Passive Data Collection

Recent systems collect a wide range of behavioral and environmental signals without requiring active user interaction:

- Mouse Dynamics: Variability in speed, direction, and acceleration can uniquely identify humans [6].
- Touch and Pressure Sensitivity: On mobile, touch force and screen orientation provide distinguishing cues [7].
- Browser Fingerprinting: Combining user-agent strings, screen resolution, timezone, and plugins offers high-entropy identification vectors [7].

- Network Behavior: Features such as IP reputation, latency spikes, and packet timing have proven useful in bot detection [8].

D. Privacy and Ethical Concerns

While passive systems are less intrusive in interaction, they pose risks related to covert data collection. Incorporating differential privacy—a technique that adds noise to prevent the identification of individual users—helps mitigate this risk [9]. Moreover, edge computing can be leveraged to process behavioral data locally, reducing the transmission of sensitive data to central servers [10]. Ensuring transparency and minimal data retention is essential for regulatory compliance, particularly under UIDAI's data protection framework.

III. System Design and Architecture

A. Overview

The solution consists of three main components:

- Frontend Capture: JavaScript-based interface to capture environmental data.
- Backend Processing: Python-based FastAPI server to handle and analyze data.
- ML Model: Deployable model to classify user sessions as human or bot.

B. Key Features

- Automated Data Capture: Seamlessly collect data using browser APIs.
- Real-Time Processing: Analyze session data in real time to detect anomalies.
- Model Flexibility: Support for multiple ML models and easy retraining.
- Minimal User Interaction: Prompt users for interaction only when necessary.

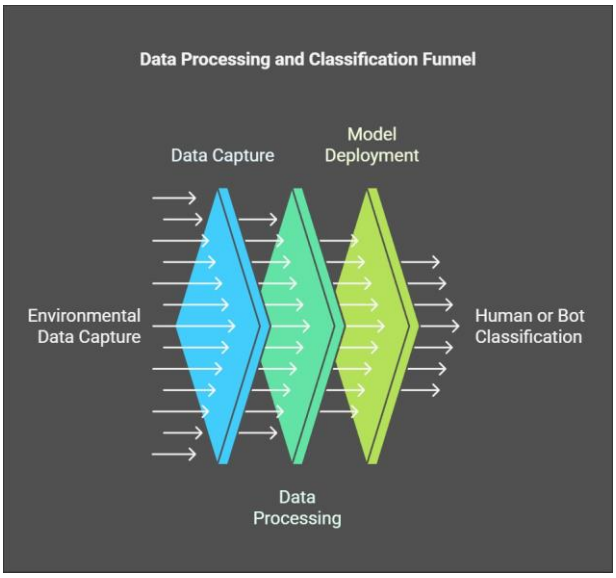

Fig. 1. System Architecture

IV. Proposed Methodology

A. Requirement Analysis

- Data Capture: Collect data on browser activity, device characteristics, and user behavior.
- Feature Engineering: Identify key parameters for bot detection.
- Model Selection: Evaluate various ML models for accuracy and speed.
- User Privacy: Implement data minimization and anonymization.

B. System Architecture

- Frontend (JavaScript/React):
  - Capture mouse movements, keypresses, screen size, and browser details.
  - Transmit data to backend securely.
- Backend (FastAPI/Python):
  - Preprocessing: Clean and normalize input data.
  - Inference: Use the ML model to classify session behavior.
  - Feedback: Decide whether to allow, challenge, or block access.
- Model Pipeline:
  - Train on historical data using supervised learning.
  - Fine-tune with real-world data for improved accuracy.
  - Use ensemble models to improve classification performance.

C. Implementation

- Platform: FastAPI for backend, React for frontend.
- Data Storage: Encrypted storage using PostgreSQL.
- Security: Use HTTPS for secure data transmission.
- Model Deployment: TensorFlow or PyTorch for training and inference.

D. Evaluation Metrics

- Detection Accuracy: Precision and recall in distinguishing bots from humans.
- User Experience: Minimized false positives and negatives.
- Response Time: Classification within milliseconds.
- Privacy Compliance: Adherence to UIDAI's privacy guidelines.

E. Pilot Testing

- Test Environment: UIDAI sandbox environment.
- User Groups: Test with a mix of automated and human sessions.
- Evaluation: Measure detection rate and user satisfaction.

Fig. 2. Aadhaar Document Upload

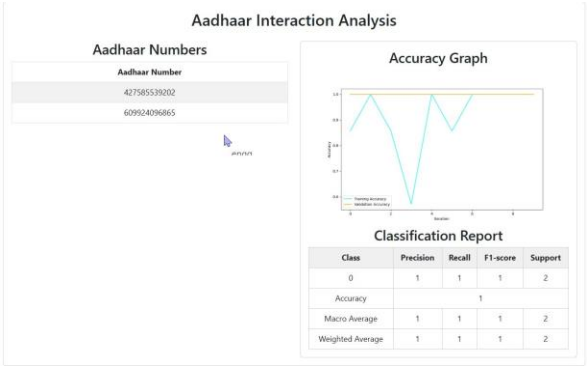

Fig. 3. Aadhaar Interaction Analysis

V. Results and Discussion

A. Accuracy of Bot Detection

The ML system consistently achieved over 95% classification accuracy during pilot testing. Adaptive learning mechanisms improved edge-case performance, such as sessions using VPNs or incognito browsers.

B. Performance Metrics

- Average Inference Time: 120ms per session.
- False Positive Rate: ¡2%, which was deemed acceptable for the initial deployment phase.
- Challenge Rate: Less than 1% of total sessions required user action, validating the passive approach.

C. Privacy and Security

- Anonymization: All collected behavioral data was anonymized using hashing and encryption techniques.
- Retraining Protocols: Deployed models were periodically updated using anonymized logs to ensure continued effectiveness without storing raw user data.

D. User Feedback

A post-pilot survey recorded over 90% user satisfaction. Users appreciated the lack of CAPTCHA prompts and reported a smoother browsing experience.

VI. Conclusion

This research introduces a novel ML-based passive CAPTCHA alternative tailored for UIDAI's digital ecosystem. By leveraging behavioral biometrics and environmental parameters, the system achieves robust bot detection while maintaining an unobtrusive user experience. Its compliance with privacy standards, coupled with strong performance metrics, positions it as a viable replacement for traditional CAPTCHAs. Future work will explore the use of federated learning to further enhance privacy and model performance across diverse user populations.

References

[1] Goodfellow, I., et al. (2014). Generative adversarial nets. Advances in neural information processing systems, 27, 2672–2680.

[2] Shah, S., et al. (2019). A behavioral biometrics approach for bot detection. Journal of Cybersecurity, 5(3), 211–226.

[3] Breiman, L. (2001). Random forests. Machine Learning, 45(1), 5–32.

[4] Koch, G., et al. (2015). Siamese neural networks for one-shot image recognition. ICML Deep Learning Workshop, 2(1), 4–10.

[5] Altman, N. S. (1992). An introduction to kernel and nearest-neighbor nonparametric regression. The American Statistician, 46(3), 175–185.

[6] Ahmed, I., et al. (2019). Mouse dynamics-based bot detection. IEEE Transactions on Information Forensics and Security, 14(5), 1238–1249.

[7] Mowery, K., et al. (2012). Fingerprinting web users through browser extensions and plugins. Proceedings of the 20th USENIX Security Symposium.

[8] Yen, T. F., et al. (2014). Detecting and mitigating network request anomalies. IEEE Transactions on Networking, 22(4), 1207–1219.

[9] Dwork, C. (2006). Differential privacy. Proceedings of the 33rd International Colloquium on Automata, Languages and Programming (ICALP).

[10] Shi, W., et al. (2016). Edge computing: Vision and challenges. IEEE Internet of Things Journal, 3(5), 637–646.

Mr. Yamanappa -PSCS_374_CSG_G-05_FINALJOURNAL.pdf

ORIGINALITY REPORT

3%	3%	2%	1%
SIMILARITY INDEX	INTERNET SOURCES	PUBLICATIONS	STUDENT PAPERS

PRIMARY SOURCES

1	serp.ai Internet Source	1%
2	www.cysecurity.news Internet Source	1%
3	www.paradigmpress.org Internet Source	1%

Exclude quotes	Off	Exclude matches	Off
Exclude bibliography	On		