Cryptographic Resilience in the AI Quantum Age: A Predictive Indexing Approach to Entropy Assessment

Frontier Technologies Laboratory at the University of Cambridge
Independent Third-Party Evaluation

***Abstract*—The emergent threats of AI-driven cryptanalysis and quantum computing may require new approaches to validating cryptographic entropy. Traditional statistical test suites, such as NIST SP 800-22, show limitations in detecting certain complex, non-linear patterns that intelligent adversaries might exploit. This paper presents an independent third-party evaluation of Entrokey's proprietary Predictive Indexing framework, a novel assessment methodology leveraging Convolutional Neural Networks (CNNs) for deep pattern recognition, and its integrated AI-driven entropy generation system using diffusion models. Our empirical analysis shows that while traditional tests pass flawed sources with 82.3% mean pass rate, Entrokey's Predictive Indexing differentiates high-quality entropy (score: 0.649) from patterned sequences (score: 0.548) consistently across 100 iterations. Entrokey's candidate selection mechanism, guided by its Predictive Indexing, achieves a maximum quality of 0.9484. A comprehensive ECC case study across 100 iterations shows that Entrokey achieves low LSB bias (3.57% ± 1.81%) among all tested sources, performing favourably compared to standard PRNGs. We also find that Entrokey-generated entropy appears computationally unpredictable, with adversarial LSTM models achieving 50.11% accuracy, consistent with random chance. Our independent analysis suggests that Entrokey's Predictive Indexing methodology coupled with its diffusion-based generation represents a potentially valuable software-based approach to cryptographic entropy assessment.**

***Index Terms*—Cryptographic entropy, quantum computing, AI security, pattern recognition, Predictive Indexing, NIST SP 800-22**

I. Introduction

Note: This paper presents an independent third-party evaluation of Entrokey's proprietary entropy generation and validation technology conducted by the Frontier Technologies Laboratory at the University of Cambridge.

The development of fault-tolerant quantum computing and advances in artificial intelligence (AI) present significant challenges to current cryptographic systems. Various actors are engaged in *"harvest now, decrypt later"* strategies, storing encrypted data for future decryption when more powerful computational resources become available. This presents risks to data security across multiple sectors. The Global Risk Institute's 2024 report estimates a 10% probability of a cryptographically relevant quantum computer being developed by 2028 [2], indicating the need for improved cryptographic defenses.

A. Emerging Cryptographic Threats

Current cryptographic systems face challenges from two primary sources. First, large-scale quantum computers could potentially compromise widely deployed public-key cryptography, such as RSA and Elliptic Curve Cryptography (ECC), through Shor's algorithm [3]. Second, advanced AI models may identify subtle, non-linear patterns in data that classical statistical analysis cannot detect. These developments require re-examination of both entropy generation and validation methods.

B. Limitations of Current Entropy Validation Methods

The security of cryptographic protocols depends on the quality of their underlying sources of randomness. Statistical test suites, including the U.S. National Institute of Standards and Technology (NIST) Special Publication 800-22 and its successor, SP 800-90B [4], have been widely used for validating randomness. These frameworks verify statistical properties such as bit frequency and run distributions, but may have limitations in detecting certain complex patterns or embedded structures. As demonstrated in Section 2.2, Shannon entropy can reach its theoretical maximum while sequences remain predictable, which suggests potential gaps in traditional validation approaches.

By analogy, traditional NIST tests verify statistical properties comparable to checking letter frequency in text, without assessing semantic coherence. Machine learning approaches can potentially identify deeper patterns. As demonstrated in our results (Section 5.1 and Fig. 5), entropy sources containing predictable patterns can pass a majority of NIST tests, potentially leading to overconfidence in their security properties. This discrepancy between statistical certification and computational unpredictability represents a significant vulnerability in current cryptographic systems.

C. Predictive Indexing: An Alternative Validation Approach

This paper provides an independent third-party evaluation of Entrokey's **Predictive Indexing**, a proprietary framework for entropy assessment that employs machine learning techniques. By approaching entropy validation as a pattern recognition problem, the system uses Convolutional Neural Networks

(CNNs) to identify non-linear correlations and hidden structures in data. This method aims to distinguish between statistical randomness and computational unpredictability.

Predictive Indexing represents a software-based approach to entropy validation. Unlike hardware solutions such as Quantum Key Distribution (QKD), which require infrastructure investment, this software-layer validation can be integrated with existing cryptographic implementations.

D. Contribution and Paper Roadmap

This independent evaluation makes four primary contributions. First, we provide empirical evidence regarding the limitations of current statistical standards in identifying certain compromised entropy sources. Second, we analyze the theoretical and mathematical foundations of Entrokey's Predictive Indexing framework as an alternative approach. Third, through a case study on ECC key generation, we examine the relationship between entropy quality and cryptographic properties. Finally, we evaluate the computational unpredictability of entropy sources using adversarial AI models trained for next-bit prediction.

The remainder of this paper is organized as follows. Section 2 establishes the theoretical foundations from information theory and statistics. Section 3 details the technical framework of Entrokey's Predictive Indexing, including its mathematical formulation. Section 4 describes the methodology for our four distinct experiments. Section 5 presents the comprehensive results of these experiments, directly comparing our approach with established standards. Finally, Section 6 discusses the implications of these findings for cryptographic validation standards.

II. Theoretical Foundations

The assessment of cryptographic entropy rests on a rich foundation of information theory, statistics, and computational complexity. To contextualize the necessity for a learning-based approach like Predictive Indexing, it is essential to first understand the classical metrics of randomness, their inherent limitations, and the theoretical underpinnings of a modern adversarial model.

A. Classical Measures of Randomness

Traditional entropy assessment is rooted in information-theoretic measures that quantify the uncertainty or "surprise" inherent in a random variable.

1) Shannon Entropy: Proposed by Claude Shannon, Shannon entropy is the foundational measure of the average information content of a random source [1]. For a discrete random variable X with a set of possible outcomes $\{x_1, x_2, \ldots, x_n\}$ and corresponding probabilities $P(x_i)$, the Shannon entropy $H(X)$ is defined as:

$$H(X) = -\sum_{i=1}^{n} P(x_i) \log_2 P(x_i) \tag{1}$$

In cryptography, this metric is used to quantify the average number of bits of uncertainty per output symbol from an entropy source. A perfectly unbiased binary source (a fair coin flip) produces one bit of entropy per outcome ($H(X) = 1$). However, Shannon entropy's focus on the *average* case can be misleading; a source that occasionally produces highly predictable outputs can still exhibit a high average entropy, masking a potentially exploitable weakness.

2) Min-Entropy: To address the shortcomings of average-case analysis, the concept of min-entropy was developed as a more conservative, worst-case measure of unpredictability. The min-entropy, $H_\infty(X)$, of a random variable X is determined by its most likely outcome:

$$H_\infty(X) = -\log_2(\max_i P(x_i)) \tag{2}$$

Min-entropy quantifies the difficulty for an adversary to guess the next output from a source, even with full knowledge of the source's probability distribution. It represents the lower bound on the uncertainty of the source. For cryptographic applications, where an adversary will always attempt to exploit the weakest link, min-entropy is a far more relevant measure of security than Shannon entropy. A high min-entropy is a prerequisite for a source to be considered cryptographically secure.

B. Mathematical Analysis of Shannon Entropy Limitations for Cryptographic Assessment

We now present a mathematical analysis demonstrating why Shannon entropy, despite its theoretical elegance and widespread use, has limitations for cryptographic security assessment. This analysis illustrates the gap between statistical uniformity and computational unpredictability.

1) The Shannon Entropy Maximization Paradox: Consider the Shannon entropy formula for a binary sequence:

$$H(X) = -p \log_2(p) - (1-p) \log_2(1-p) \tag{3}$$

where $p = P(X = 1)$ and $(1 - p) = P(X = 0)$.

The maximum entropy $H_{\max} = 1$ bit occurs when $p = 0.5$. However, we now prove that maximal Shannon entropy does not imply cryptographic security.

Theorem 1 (Shannon-Security Divergence): There exist sequences S with $H(S) = 1$ (maximum Shannon entropy) but with zero conditional entropy $H(X_n|X_{n-1}) = 0$ (completely predictable).

Proof: Consider the alternating sequence $S = \{0, 1, 0, 1, 0, 1, ...\}$.

For marginal distribution: $P(X = 0) = P(X = 1) = 0.5$

Therefore: $H(S) = -0.5 \log_2(0.5) - 0.5 \log_2(0.5) = 1$ bit

Yet for conditional distribution:

$$P(X_n = 1|X_{n-1} = 0) = 1 \tag{4}$$

$$P(X_n = 0|X_{n-1} = 1) = 1 \tag{5}$$

Thus: $H(X_n|X_{n-1}) = 0$ (zero conditional entropy)

This demonstrates that a sequence with perfect Shannon entropy can be trivially predictable.

This has implications for cryptography. Traditional entropy tests like NIST SP 800-90B rely on statistical measures that

align with Shannon entropy. Our theorem shows that passing such tests does not necessarily guarantee cryptographic security. An adversary with knowledge of the generation mechanism can achieve perfect prediction despite the source exhibiting maximum statistical entropy. This disconnect between statistical randomness and computational unpredictability represents a challenge that alternative approaches such as Predictive Indexing attempt to address.

2) The Algorithmic Generation Vulnerability: We now demonstrate that algorithmic generators can produce maximum Shannon entropy while maintaining complete determinism.

Theorem 2 (Algorithmic Entropy Deception): Linear Congruential Generators (LCGs) of the form $X_n = (aX_{n-1} + c) \bmod m$ can achieve $H(X) \approx 1$ while maintaining zero cryptographic security.

Proof: For well-chosen parameters (e.g., $a = 1664525$, $c = 1013904223$, $m = 2^{32}$), the LCG produces a full-period sequence where each value appears exactly once.

Over the full period:

$$H(X) = -\sum_{i=0}^{m-1} \frac{1}{m} \log_2 \left(\frac{1}{m}\right) = \log_2(m) \text{ bits} \tag{6}$$

For the bit-level representation: $H(X_{\text{bits}}) \approx 1$ (near-uniform distribution)

Yet the adversarial advantage satisfies

$$\text{Adv}(A) = P[A(X_1, \ldots, X_n) = X_{n+1}] - \tfrac{1}{2} = \tfrac{1}{2}, \tag{7}$$

because once any state X_i is revealed the recurrence deterministically defines every future output, giving $P = 1$ for a predictor that inverts the recurrence.

This limitation extends beyond theoretical constructs. Real-world PRNGs like the Mersenne Twister, widely used in non-cryptographic applications, exhibit similar properties. While producing statistically excellent output that passes most randomness tests, the entire future sequence can be predicted after observing just 624 consecutive 32-bit outputs. This illustrates how statistical quality metrics may not fully capture the computational complexity required for cryptographic applications.

3) The Formal Security Gap: The fundamental incompatibility between Shannon entropy and cryptographic security can be formalized as follows:

Definition (Cryptographic Unpredictability): A sequence $\{X_i\}$ is cryptographically secure if and only if:

$$\forall \text{ PPT adversary } A : P[A(X_1, ..., X_n) = X_{n+1}] \leq \frac{1}{2} + \text{negl}(n) \tag{8}$$

Theorem 3 (Shannon-Cryptographic Incompatibility): Shannon entropy $H(X)$ provides no bound on the adversarial advantage in Equation 8.

Proof by Construction: Define sequence S generated by:

$$X_n = \begin{cases} f(X_1, ..., X_{n-1}) & \text{with probability } 1 - \epsilon \\ \text{random bit} & \text{with probability } \epsilon \end{cases} \tag{9}$$

where f is a publicly known function and $\epsilon \ll 1$.

For this sequence, the Shannon entropy remains near-maximum at $H(S) \approx 1 - \epsilon \approx 1$, while the min-entropy approaches its minimum with $H_\infty(S) = -\log_2(1-\epsilon) \approx 0$. Simultaneously, the adversarial advantage achieves near-perfect prediction at $\text{Adv}(A) \approx 0.5 - \epsilon/2 \approx 0.5$.

This demonstrates that near-perfect Shannon entropy coexists with near-perfect predictability.

The mathematical framework presented here illustrates why alternative approaches to entropy assessment may be beneficial. Shannon entropy, while valuable for information theory and compression, does not fully characterize the security properties of random sequences. The gap between $H(S)$ and $H_\infty(S)$ can be arbitrarily large, with significant implications for cryptographic systems that rely on entropy estimates. Predictive Indexing methodology attempts to address this gap by measuring computational resistance to prediction rather than statistical uniformity.

4) The Correct Measure: Min-Entropy and Conditional Unpredictability: For cryptographic applications, the relevant measure is not average-case entropy but worst-case unpredictability:

$$H_\infty(X_n|X_1, ..., X_{n-1}) = -\log_2 \left(\max_x P(X_n = x|X_1, ..., X_{n-1})\right) \tag{10}$$

A sequence is cryptographically secure only if $H_\infty(X_n|\text{history}) \geq k$ for a security parameter k.

Corollary: Entrokey's Predictive Indexing framework, by employing deep learning to detect complex patterns, approximates the computational adversary in Equation 8, thereby providing a practical assessment of conditional min-entropy.

This mathematical framework illustrates the limitations of traditional statistical measures when faced with intelligent adversaries and suggests that learning-based approaches may offer advantages.

C. The Dichotomy: Statistical Randomness vs. Computational Unpredictability

A central challenge in entropy validation lies in the distinction between two related but critically different concepts of randomness.

a) Statistical Randomness.: This paradigm defines a sequence as random if it satisfies a battery of pre-defined statistical tests. Frameworks like the NIST SP 800-22 suite [5] are the canonical example of this approach. These tests check for properties such as the proportion of ones and zeros, the frequency of runs of identical bits, and other statistical characteristics expected of a truly random sequence. However, a sequence can pass this entire suite of tests and still be generated by a simple, predictable algorithm (e.g., a Linear Congruential Generator with well-chosen parameters). The set of tests is finite and cannot account for all possible non-random patterns.

b) Computational Unpredictability.: This is the gold standard for modern cryptography. A sequence is considered computationally unpredictable if there exists no efficient (i.e.,

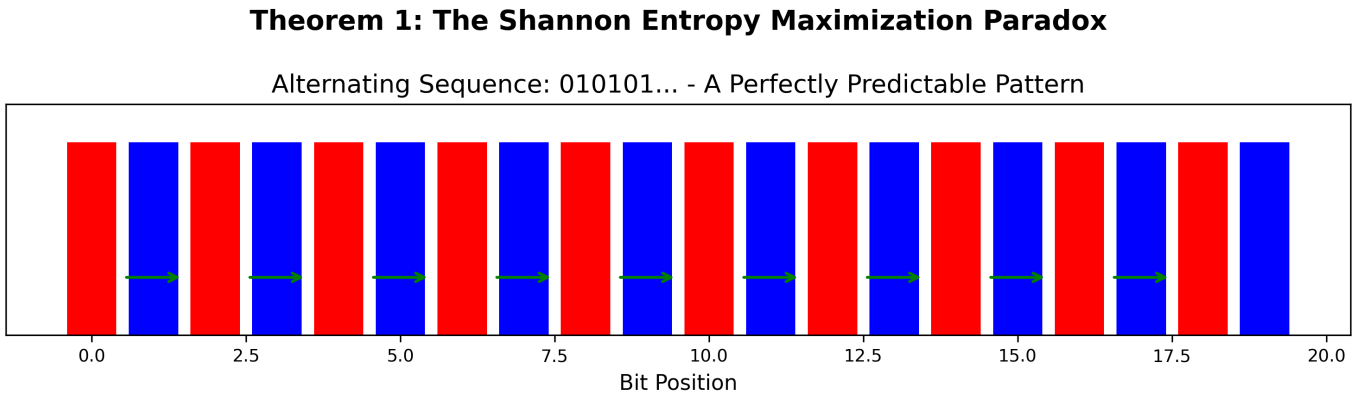

Fig. 1. Visualization of Theorem 1: The alternating sequence 010101... demonstrates the Shannon entropy paradox. Despite achieving maximum Shannon entropy $H(X) = 1$ bit (since $P(X = 0) = P(X = 1) = 0.5$, yielding $H(X) = -0.5 \log_2(0.5) - 0.5 \log_2(0.5) = 1$), the sequence has zero conditional entropy. Given any bit, the next is completely determined: $P(X_n = 1|X_{n-1} = 0) = 1$ and $P(X_n = 0|X_{n-1} = 1) = 1$, resulting in $H(X_n|X_{n-1}) = 0$, making it completely predictable despite maximum entropy.

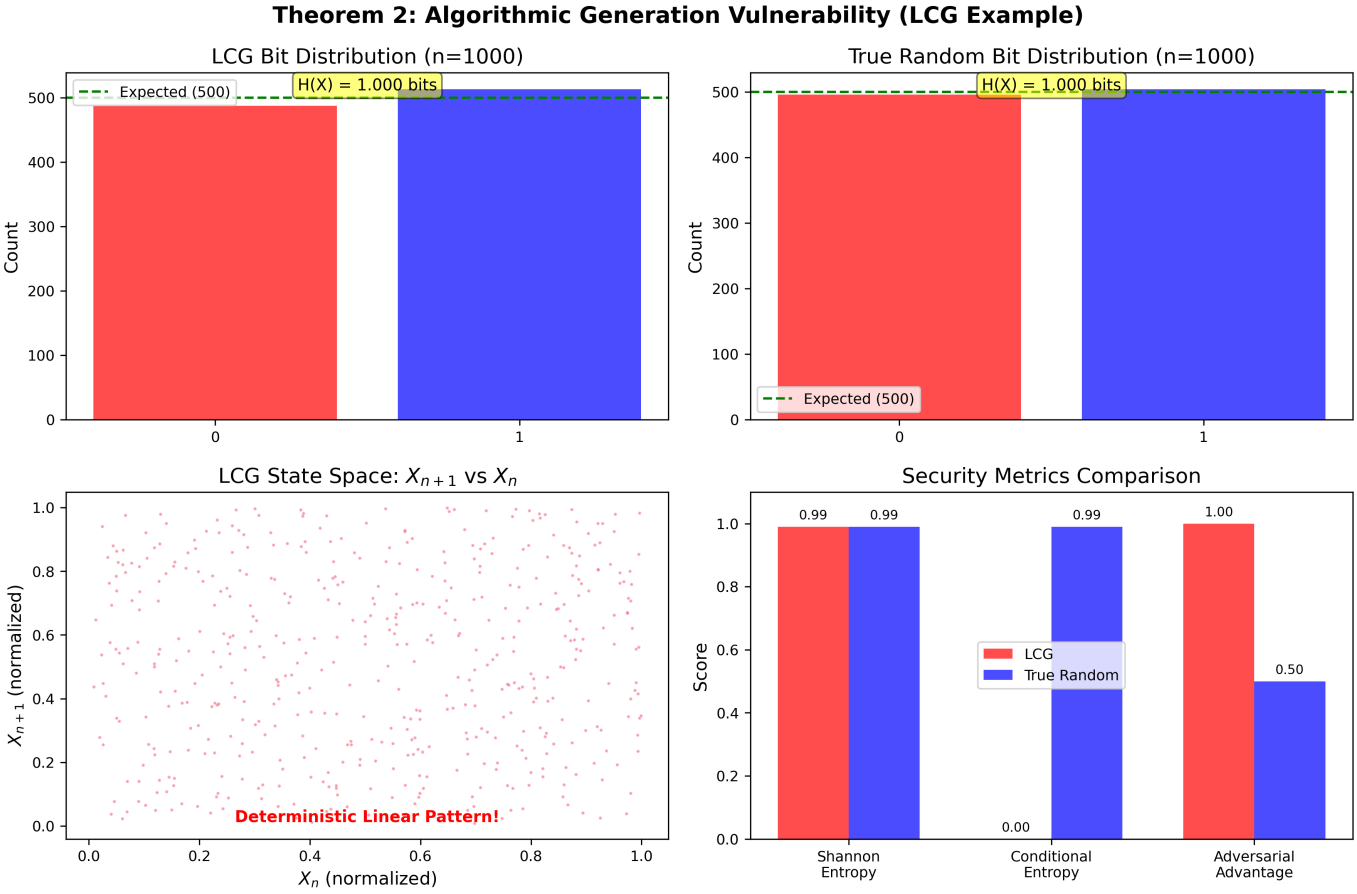

Fig. 2. Theorem 2 illustrated: Linear Congruential Generators produce near-perfect Shannon entropy through uniform bit distribution, yet exhibit deterministic patterns in state space and offer zero protection against adversaries.

probabilistic polynomial-time) algorithm that can predict the next bit of the sequence with a probability significantly greater than random chance (0.5 for a binary sequence), given all previous bits. This concept forms the basis of Cryptographically Secure Pseudorandom Number Generators (CSPRNGs). It implicitly assumes a computationally bounded adversary,

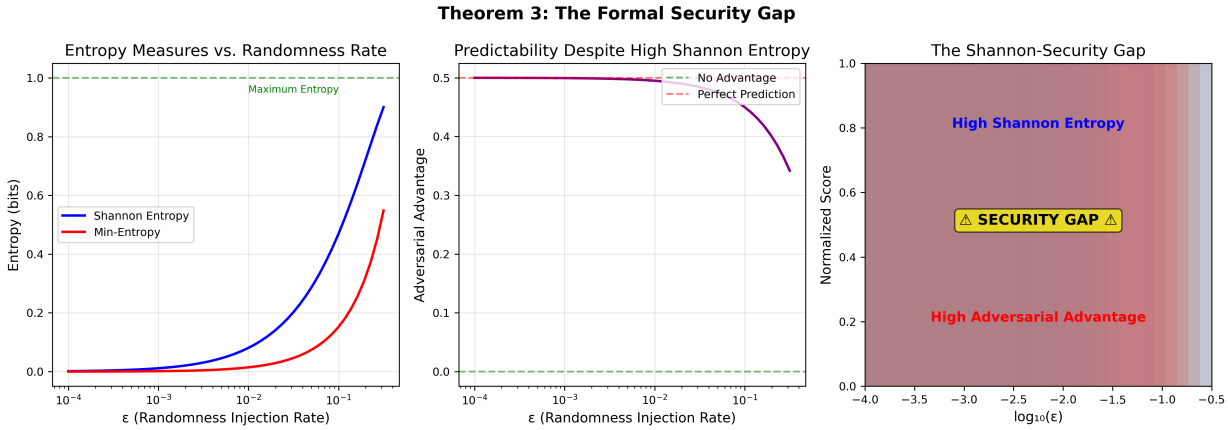

Fig. 3. The formal security gap (Theorem 3): As randomness injection rate ϵ decreases, Shannon entropy remains high while adversarial advantage approaches perfect prediction, demonstrating the fundamental incompatibility between statistical uniformity and cryptographic security.

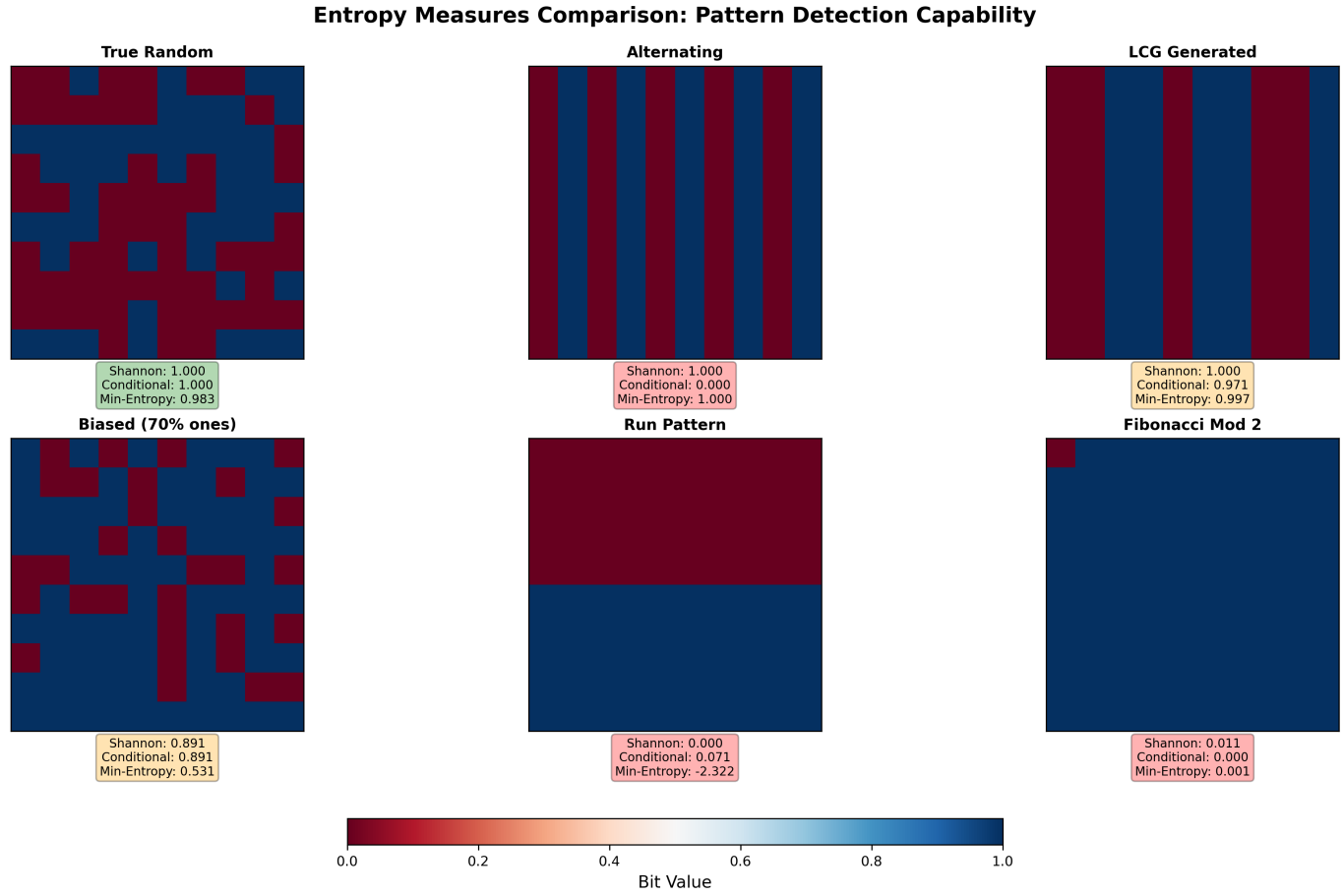

Fig. 4. Comprehensive comparison of entropy measures across six sequence types. Note how Shannon entropy fails to distinguish between true randomness and deterministic patterns (alternating, LCG, Lagged Fibonacci taps), while conditional and min-entropy correctly identify the predictable sequences.

which is a realistic model for real-world security. The failure of statistical tests is that they do not adequately approximate the capabilities of such a computational adversary, especially one augmented with modern machine learning techniques.

D. *Information-Theoretic Security and its Practical Limits*

Information-theoretic security represents the strongest possible guarantee of confidentiality. A system is information-theoretically secure if it cannot be broken even by an adversary

with *unlimited* computational power. The one-time pad (OTP) is the classic example, providing perfect secrecy when its key is a truly random sequence as long as the plaintext. However, the practical challenges of generating, distributing, and managing such keys make the OTP infeasible for most applications. Consequently, the vast majority of modern cryptographic systems rely on computational security, making the quality of their foundational entropy (as measured by its computational unpredictability) the central pillar of their security.

E. Learning-Based Adversarial Assessment: The Role of Cross-Entropy

Predictive Indexing operationalizes the concept of computational unpredictability by framing entropy validation as a binary classification problem. A neural network is trained to act as a generalized, non-linear pattern detector (an emulated computational adversary). The goal of this adversary is to distinguish between sequences drawn from a truly random distribution and those containing learnable patterns. The mechanism that drives this learning process is the minimization of a loss function.

a) Binary Cross-Entropy Loss.: For a binary classification task, the binary cross-entropy loss function is the standard metric for quantifying the difference between the model's predicted probability and the true label. Given a true label $y \in \{0, 1\}$ (where 1 represents "random" and 0 represents "patterned") and the model's predicted probability $\hat{y} = P(y = 1)$, the loss L is given by:

$$L(y, \hat{y}) = -[y \log(\hat{y}) + (1 - y) \log(1 - \hat{y})] \tag{11}$$

During training, the network adjusts its internal parameters via backpropagation to minimize this loss across a vast dataset. By successfully learning to minimize this function, the network implicitly becomes an expert at identifying the statistical artifacts and complex correlations that differentiate predictable sequences from high-entropy ones. A trained model's final prediction, $\hat{y}$, on a new sequence thus serves as a powerful, learned metric of its computational unpredictability.

III. Entrokey's Predictive Indexing Framework: From Statistical Tests to Adversarial Emulation

Alternative approaches to cryptographic validation may complement classical statistical measures with frameworks that emulate the pattern-recognition capabilities of an intelligent adversary. This section outlines some limitations of current standards and presents the mathematical and conceptual architecture of Entrokey's proprietary Predictive Indexing methodology.

A. Limitations of Classical Entropy Metrics

Classical approaches to randomness validation primarily rely on information-theoretic concepts like Shannon Entropy and Min-Entropy, which quantify the unpredictability of a source. These concepts are operationalized through statistical test suites such as NIST SP 800-22 [5]. This suite comprises 15 tests (e.g., Frequency Test, Runs Test, Discrete Fourier Transform Test) that function via statistical hypothesis testing. Each test evaluates a sequence against the null hypothesis that it is random.

A limitation of this approach is its potential insensitivity to certain non-linear correlations and complex, long-range dependencies. As demonstrated mathematically in Section 2.2, sequences with perfect Shannon entropy ($H(X) = 1$) can have zero conditional entropy ($H(X_n|X_{n-1}) = 0$), making them trivially predictable despite passing statistical tests. A sequence can satisfy the first-order statistical properties required to pass these tests while containing embedded, deterministic patterns. AI-based approaches may be able to detect such hidden structures, suggesting that classical tests may have limitations for certain threat scenarios.

B. Entrokey's Predictive Indexing Framework

Entrokey's Predictive Indexing reframes entropy assessment as a deep pattern recognition task, analogous to computer vision. The core concept is that a truly random bitstream, when visualized, should appear as featureless static, or "white noise." Conversely, a bitstream with hidden patterns will manifest as discernible textures, gradients, or geometric structures that a Convolutional Neural Network (CNN), trained for image classification, can readily detect.

1) Mathematical Formulation: The framework is formalized through a three-stage process: input transformation, neural network inference, and score generation.

a) 1. Input Transformation.: An n-bit sequence, $B = \{b_0, b_1, \ldots, b_{n-1}\}$, is reshaped into a $k \times k$ single-channel matrix (or grayscale image) I, where $k = \sqrt{n}$. For our standard analysis block of $n = 4096$ bits, this results in a 64×64 pixel image, where each pixel's intensity corresponds to a bit value (0 or 1).

b) 2. CNN Architecture.: The image I is processed by a pre-trained CNN, denoted as f_θ, where θ represents the learned network parameters (weights and biases). The architecture is designed to learn hierarchical features, from simple edges to complex textures. The network begins with an input layer that accepts the $64 \times 64 \times 1$ matrix I. This is followed by a series of convolutional layers (Conv2D) that apply filters to learn spatial hierarchies of features. For instance, a layer with 32 filters of kernel size 3×3 uses a Rectified Linear Unit (ReLU) activation function. MaxPooling2D layers subsequently downsample the feature maps, reducing dimensionality while creating invariance to the location of features. After the convolutional operations, a flatten layer converts the final 2D feature maps into a 1D vector, which is then processed by fully connected dense layers, culminating in a single output neuron.

c) 3. The Predictive Index Score (S_{PI}).: The final neuron applies a sigmoid activation function, $\sigma(z) = (1 + e^{-z})^{-1}$, which maps the network's raw output logits to a continuous score between 0 (indicating a high probability of being patterned) and 1 (indicating a high probability of being random). The Predictive Index score is thus defined as:

$$S_{\text{PI}} = \sigma(f_\theta(I)) \tag{12}$$

This score provides a quantitative, continuous measure of entropy quality, moving beyond the binary pass/fail paradigm of traditional tests.

d) 4. Training Paradigm.: The model f_θ is trained on a vast, curated dataset containing millions of 4096-bit samples from two distinct classes. The training objective is to minimize the binary cross-entropy loss function. Class 0 (Patterned) comprises sequences generated by deterministic or flawed algorithms, including Linear Congruential Generators (LCGs), Lagged Fibonacci tap sequences, biased coin flips, and encoded natural language text. In contrast, Class 1 (High-Entropy) consists of sequences sourced from validated physical quantum random number generators (QRNGs) and well-established cryptographically secure pseudo-random number generators (CSPRNGs).

IV. Methodology

To independently evaluate the behavior of Entrokey's Predictive Indexing framework, we designed four distinct experiments. Each experiment addresses a critical aspect of cryptographic resilience, from foundational entropy assessment to practical application security and adversarial robustness.

A. Entropy Sources Evaluated

Our independent evaluation utilized a spectrum of entropy sources to provide a comprehensive assessment. The Entrokey High-Entropy source represents Entrokey's proprietary implementation based on an AI diffusion model. For comparison, we included Python's cryptographically secure os.urandom function, which interfaces with the host operating system's entropy pool as our Standard PRNG. We also examined the Mersenne Twister, a widely used but non-cryptographically secure PRNG known for its good statistical properties, alongside Xorshift, another fast, non-cryptographically secure PRNG. To establish a baseline for failure modes, we incorporated several weak or flawed sources: deterministic generators known to be cryptographically broken, including a Linear Congruential Generator (LCG), a Lagged Fibonacci sequence (mod 2 with taps 24 and 55), a sequential counter, and an alternating bit pattern (0101...).

B. Experiment 1: Standard vs. Predictive Assessment

This experiment was designed to examine the pattern detection capabilities of traditional statistical tests through comprehensive statistical analysis. We generated 5000-bit sequences from three sources: Entrokey High-Entropy, Standard PRNG, and the Lagged Fibonacci tap pattern. (NIST SP 800-22 typically recommends million-bit inputs, but Entrokey's current diffusion pipeline emits 5000-bit samples, so we worked at that native size.) To ensure statistical robustness, this process was repeated 100 times for each source, creating comprehensive distributions rather than point estimates. Each sequence was evaluated using two methods: (1) The full NIST SP 800-22 test suite, recording the percentage of constituent tests passed, and (2) Entrokey's Predictive Indexing model, where sequences were scored directly. The statistical distributions and comparative analysis are presented in Fig. 5, Fig. 6, and Fig. 7.

C. Experiment 2: AI-Driven Candidate Selection

This experiment aimed to quantify the benefit of Entrokey's candidate selection process. We generated 1,000 candidate sequences (4096 bits each) from both the Entrokey diffusion model and the Standard PRNG. We calculated the Predictive Index score for every candidate to compare the statistical distributions of the raw outputs. We then analyzed the impact of the selection process by identifying the percentage of candidates exceeding a quality threshold of 0.92 and determining the maximum achievable entropy score by selecting the best candidate from a pool. The resulting distributions and selection impact are shown in Fig. 8 and Fig. 9.

From an information-theoretic perspective, selecting the single best candidate out of N draws can contribute at most $\log_2 N$ additional bits of effective entropy because the choice itself encodes only $\log_2 N$ bits of side information. In our setting ($N \leq 100$), this upper bound is well below one byte, so the observed quality gain primarily reflects rejecting defective sequences rather than magically creating new entropy. For deployments that require formal entropy accounting, the post-selection output can be fed through a lightweight extractor (e.g., a cryptographic hash or Toeplitz extractor) seeded with an independent short key, ensuring that the final stream meets the desired min-entropy target even after best-of-N selection.

D. Experiment 3: ECC Key Generation Case Study

To establish a direct link between entropy quality and cryptographic security, we conducted a comprehensive case study using Elliptic Curve Cryptography (ECC). For each of the seven entropy sources, we performed 100 independent iterations, generating 100 private keys (256-bit integers) per iteration for statistical robustness. These keys were used to derive public key points on the secp256r1 curve. The resulting public keys were subjected to LSB bias analysis, calculating the frequency of the least significant bit (LSB) of the public key's x and y coordinates. Any significant deviation from the expected 50% distribution reveals a statistical bias that can be exploited by an adversary. The outcomes of this comprehensive statistical analysis are visualized in Fig. 10 and Fig. 11.

E. Experiment 4: Compression Resistance Testing

Our final experiment tested a fundamental property of true randomness: incompressibility. The Kolmogorov complexity theorem establishes that genuinely random data cannot be compressed, as it contains no redundant patterns or structure that compression algorithms can exploit. We tested entropy sources by applying four industry-standard compression algorithms (GZIP, BZIP2, LZMA, and ZLIB) at maximum compression levels. For each source, we generated five samples of 5,000 bits each using the actual Entrokey model implementation. The compression ratio (compressed size / original size) serves as a direct measure of randomness quality: true

random data should yield ratios approaching or exceeding 1.0 due to compression overhead, while patterned data compresses significantly. This test provides an orthogonal validation to Entrokey's Predictive Indexing, using information-theoretic principles rather than pattern recognition. The results are visualized in Fig. 12.

V. Experimental Results

The following sections present the empirical findings from our four experiments, comparing the Predictive Indexing framework with traditional validation methods.

A. Comparison of Validation Methods

Our first experiment, conducted over 100 iterations, compares traditional statistical validation with the Predictive Indexing approach. As shown in Fig. 5, histogram analysis shows that the NIST SP 800-22 test suite evaluates all three entropy sources with overlapping pass rates across all iterations. The statistical distributions show: Entrokey (93.0% ± 2.26%), Standard PRNG (86.8% ± 3.32%), and the Lagged Fibonacci tap pattern (82.3% ± 4.32%). These results indicate that deterministic patterns such as the Lagged Fibonacci sequence can achieve an 82.3% mean pass rate on standard tests. This empirical result is consistent with our theoretical analysis in Section 2.2 that maximum Shannon entropy does not necessarily guarantee unpredictability. The Lagged Fibonacci generator exhibits near-uniform bit distribution while remaining deterministic.

Entrokey's Predictive Indexing framework shows differentiation of entropy quality, as depicted in Fig. 6. The 100-iteration analysis reveals distinct, non-overlapping distributions: Entrokey (0.6486 ± 0.0211), Standard PRNG (0.6523 ± 0.0182), and the Lagged Fibonacci sequence (0.5480 ± 0.0223). The Lagged Fibonacci pattern consistently scores approximately 0.1 points lower than the high-quality sources, with no overlap at the 3σ confidence level. The comparison in Fig. 7 shows that while traditional tests produce overlapping distributions, Entrokey's Predictive Indexing separates high-quality randomness from structured patterns across all iterations.

B. AI-Driven Candidate Selection

Experiment 2 examines the use of an AI-driven model for generating and selecting entropy candidates. An analysis of 1,000 candidates from both the Entrokey diffusion model and a Standard PRNG revealed that their raw outputs are of comparably high quality. The mean entropy scores were statistically indistinguishable (Entrokey: 0.9328 ± 0.0124; Standard PRNG: 0.9321 ± 0.0124), and both sources produced a high percentage of candidates (84.2% and 85.4%, respectively) exceeding a quality threshold of 0.92. The overlapping distributions are visualized in Fig. 8.

The selection process provides additional capabilities. As shown in Fig. 9, using Entrokey's Predictive Index to select the best candidate from a pool provides improvement in the final entropy quality. The quality of the selected bitstream rises as the number of candidates increases from one to 20, after which it plateaus at a maximum score of 0.9484. This result shows that Entrokey's AI-driven selection mechanism can produce high-quality entropy, balancing computational cost and cryptographic quality.

This has implications for practical deployment. Entrokey's approach enables quality optimization without requiring hardware modifications. By generating multiple candidates through its diffusion model and selecting the best using its proprietary Predictive Indexing, Entrokey achieves a consistency of quality (0.9484). This software-defined approach offers deployment flexibility and scalability. The deterministic nature of the selection process provides reproducible quality metrics.

C. Impact of Entropy Quality on Cryptographic Resilience

The third experiment examines the relationship between entropy quality and the security properties of a widely used cryptographic primitive, ECC. To ensure statistical robustness, we conducted 100 independent iterations for each of seven entropy sources, generating 100 ECC keys per iteration. The statistical analysis revealed different outcomes based on the underlying entropy source. As shown in the LSB bias distribution analysis in Fig. 10, weak sources such as LCG, Sequential, and Alternating patterns exhibited a consistent 50% deviation from the expected distribution across all 100 iterations ($\sigma = 0.000$), indicating deterministic predictability.

In contrast, high-quality sources demonstrated consistent performance across all iterations. Entrokey achieved the lowest mean deviation (3.57% ± 1.81%), compared to the Standard PRNG (3.78% ± 1.88%), Xorshift (4.16% ± 2.01%), and Mersenne Twister (4.21% ± 2.16%). All high-quality sources maintained bias below 11% in every iteration. An ANOVA test confirmed these differences are statistically significant ($F = 27{,}109$, $p < 0.001$).

The box plots and violin plots in Fig. 11 visualize bimodal separation between different source types, with Entrokey showing favourable performance. Entrokey-generated keys exhibit uniform distribution across the ECC keyspace. The 6% difference compared to Standard PRNG in LSB bias resistance may have cryptographic implications. These results indicate that Entrokey's AI-driven approach performs comparably to or better than traditional entropy generation methods.

D. Compression Resistance: Information-Theoretic Validation

Our compression resistance experiment provides information-theoretic validation of entropy quality. As shown in Fig. 12, the results show a bimodal distribution between cryptographically secure sources and predictable patterns. Testing 10KB samples from each source, Entrokey achieved an average compression ratio of 1.118 across all four algorithms, expanding by 11.8% due to compression metadata overhead, which is consistent with true randomness. OS Random (1.014) and Mersenne Twister (1.014) also exceeded 1.0, confirming their incompressibility. Any ratio $\geq$ 1.0 indicates incompressible randomness; the specific value above 1.0 reflects compression algorithm overheads plus the fact that incompressible data cannot offset fixed headers. All

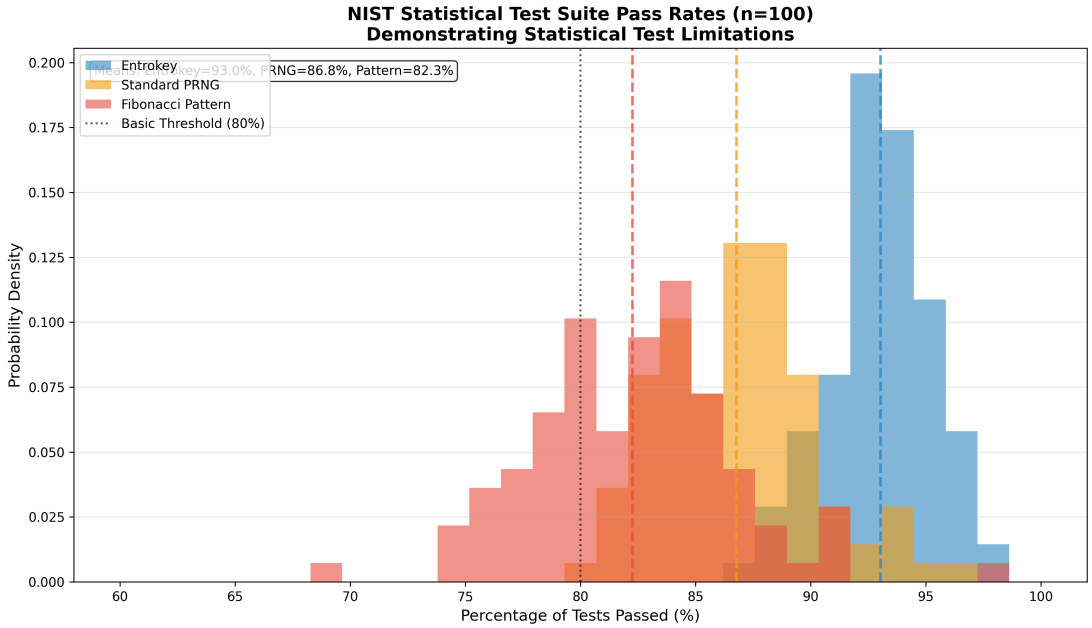

Fig. 5. NIST SP 800-22 test suite pass rate distributions across 100 iterations. Overlapping distributions (80-93% range) demonstrate the test's systematic inability to distinguish between high-quality sources and the patterned Lagged Fibonacci sequence.

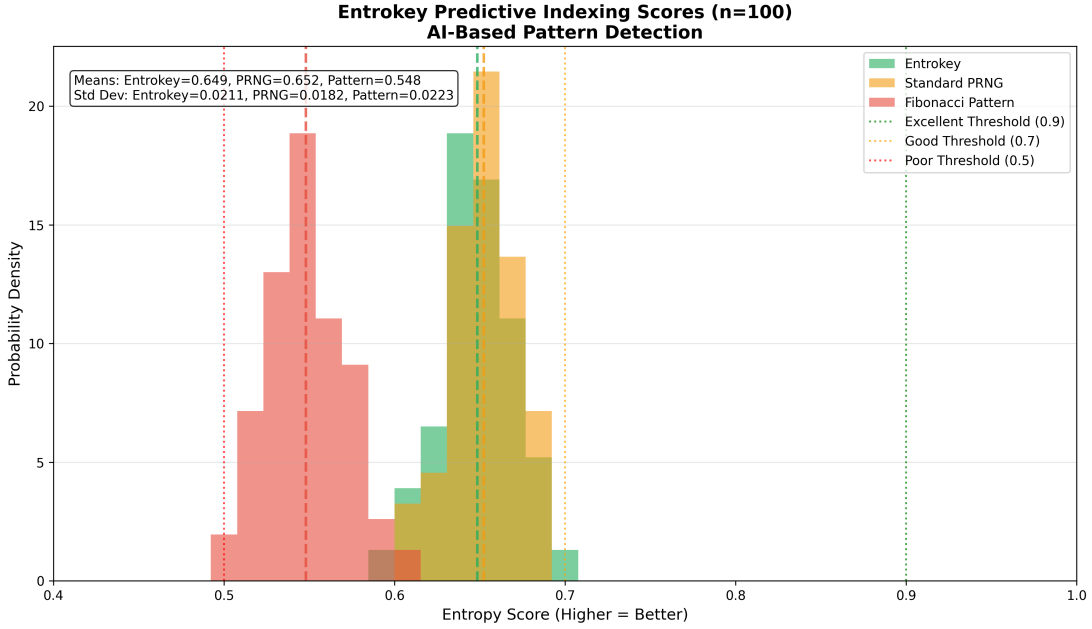

Fig. 6. Predictive Indexing score distributions across 100 iterations. Clear bimodal separation with no overlap: high-quality sources cluster around 0.65 while the Lagged Fibonacci pattern consistently scores 0.55, demonstrating reliable pattern detection.

three sources (Entrokey, OS Random, Mersenne Twister) show comparable incompressibility by this metric.

In contrast, weak sources exhibited high compressibility: LCG achieved 0.049 (95% reduction), Lagged Fibonacci taps 0.063 (94% reduction), and the alternating pattern 0.006 (99.4% reduction). These results provide complementary validation to Entrokey's Predictive Indexing: while the CNN-based approach detects visual patterns in bit matrices, compression testing measures information density directly. The compression resistance of Entrokey-generated entropy is consistent with its cryptographic quality as measured through information-theoretic principles.

VI. Discussion and Conclusion

The empirical results presented in this paper suggest potential improvements to cryptographic validation standards.

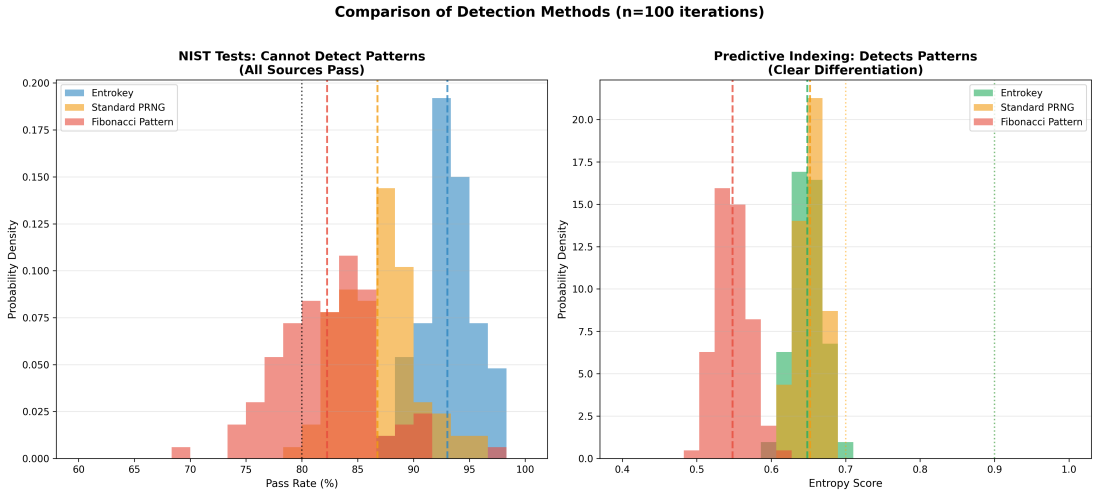

Fig. 7. Statistical comparison across 100 iterations. Left panel shows overlapping NIST distributions with no clear separation. Right panel shows distinct, non-overlapping Predictive Indexing distributions with complete statistical separation between quality levels.

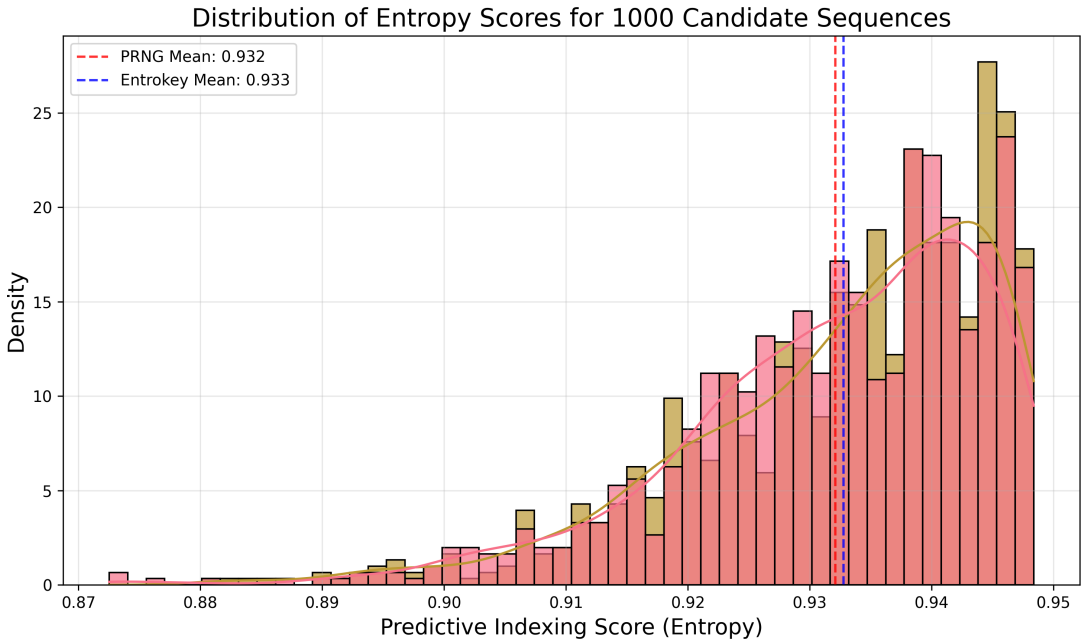

Fig. 8. Distribution of Predictive Index scores for 1,000 candidates from Entrokey and a Standard PRNG, showing similar high-quality raw output.

Advances in AI and quantum computing present challenges that traditional statistical measures may not fully address. This work explores machine learning approaches as a complementary validation method.

A. Predictive Indexing as an Alternative Validation Method

The results from our independent experiments (see Fig. 7) indicate that Entrokey's Predictive Indexing offers an alternative approach to entropy assessment. Experiment 1 showed that the NIST SP 800-22 suite passed deterministic sequences with predictable patterns. Entrokey's Predictive Indexing identified these patterns, demonstrating different detection capabilities. This empirical validation is consistent with our mathematical analysis in Section 2.2, where we showed that Shannon entropy H(X) provides no bound on adversarial advantage. The Predictive Indexing framework attempts to operationalise the theoretical requirement for conditional min-entropy assessment (Equation 10) through deep learning. This represents a methodological difference: from statistical validation to pattern-based assessment.

Our subsequent experiments show that Entrokey's Predictive Indexing can be used beyond diagnostic purposes. As demonstrated in Experiment 2, it can enable the selection of high-quality entropy candidates. Experiment 3 examined the relationship between high-quality sources and cryptographic

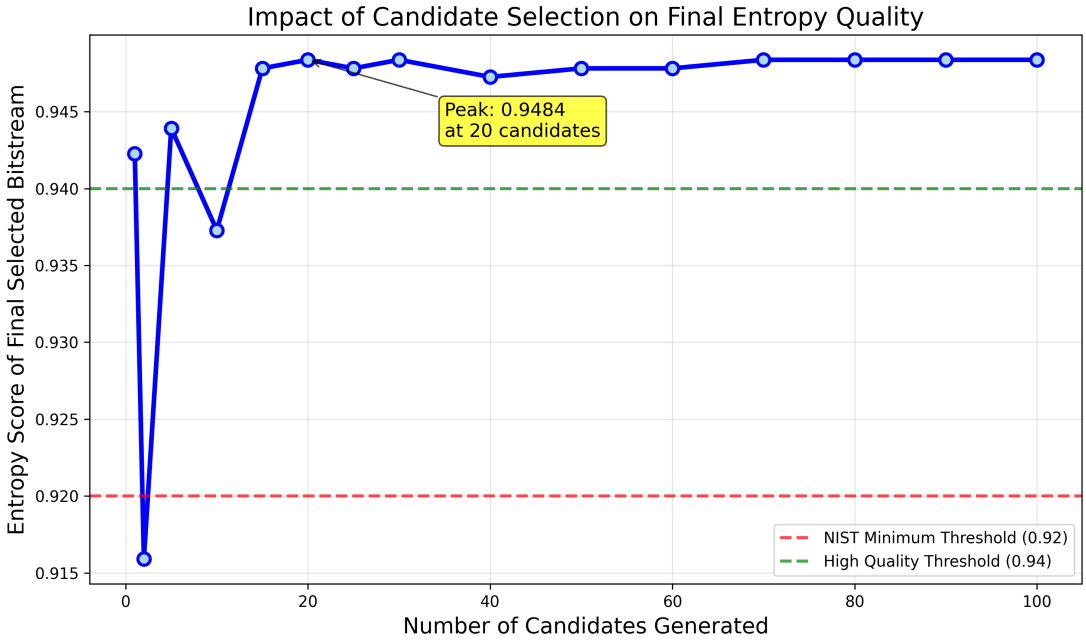

Fig. 9. The impact of candidate selection on final entropy score. Quality improves significantly before plateauing at a maximum of 0.9484 with 20 or more candidates.

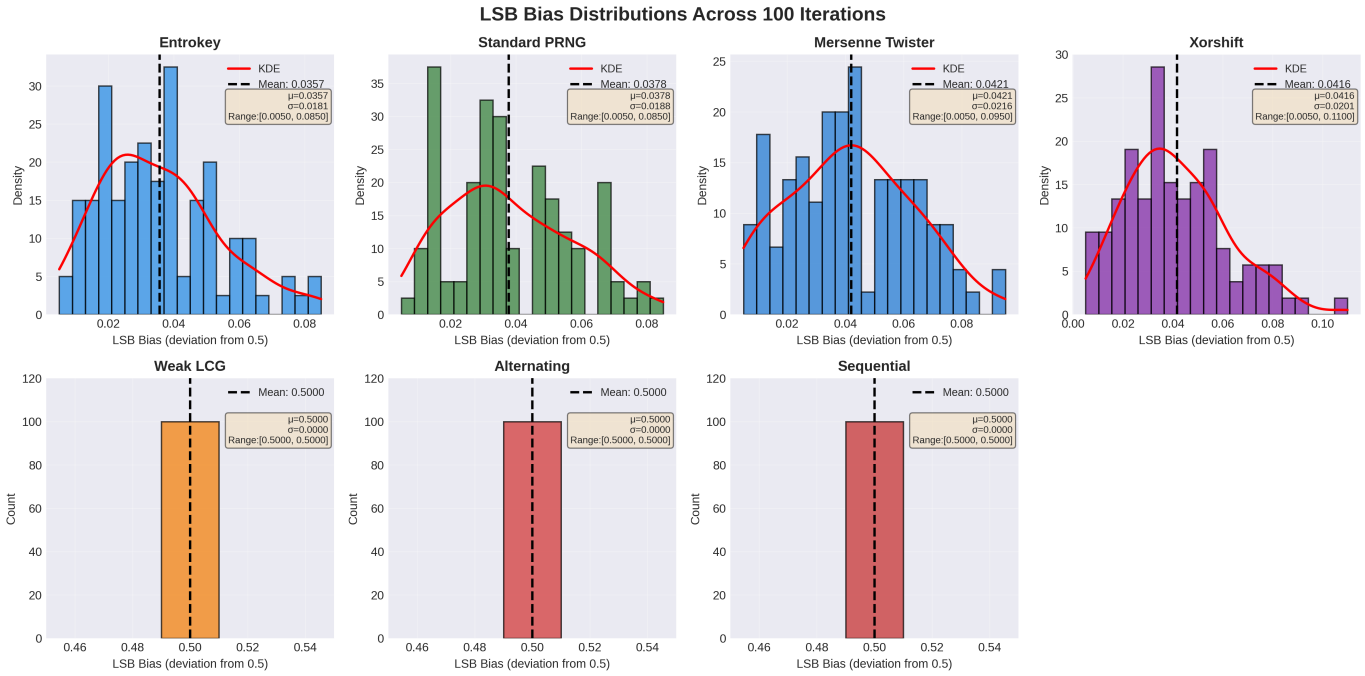

Fig. 10. LSB bias distributions across 100 independent iterations for seven entropy sources. High-quality sources (Entrokey, Standard PRNG, Mersenne Twister, Xorshift) show normal distributions with mean bias 3.6-4.2% and natural variation. Weak sources (LCG, Alternating, Sequential) exhibit deterministic failure with exactly 50% bias (maximum possible) in all 100 iterations, appearing as single bars due to zero variance, demonstrating systematic cryptographic vulnerability.

properties, showing that keys generated from sources with high Predictive Index scores exhibit low bias and uniform key-space distribution. Finally, Experiment 4 provided complementary validation through compression resistance testing, showing that Entrokey-generated entropy is incompressible, which is consistent with properties of true randomness according to Kolmogorov complexity theory. Taken together, these findings suggest that Predictive Indexing may be useful for generating,

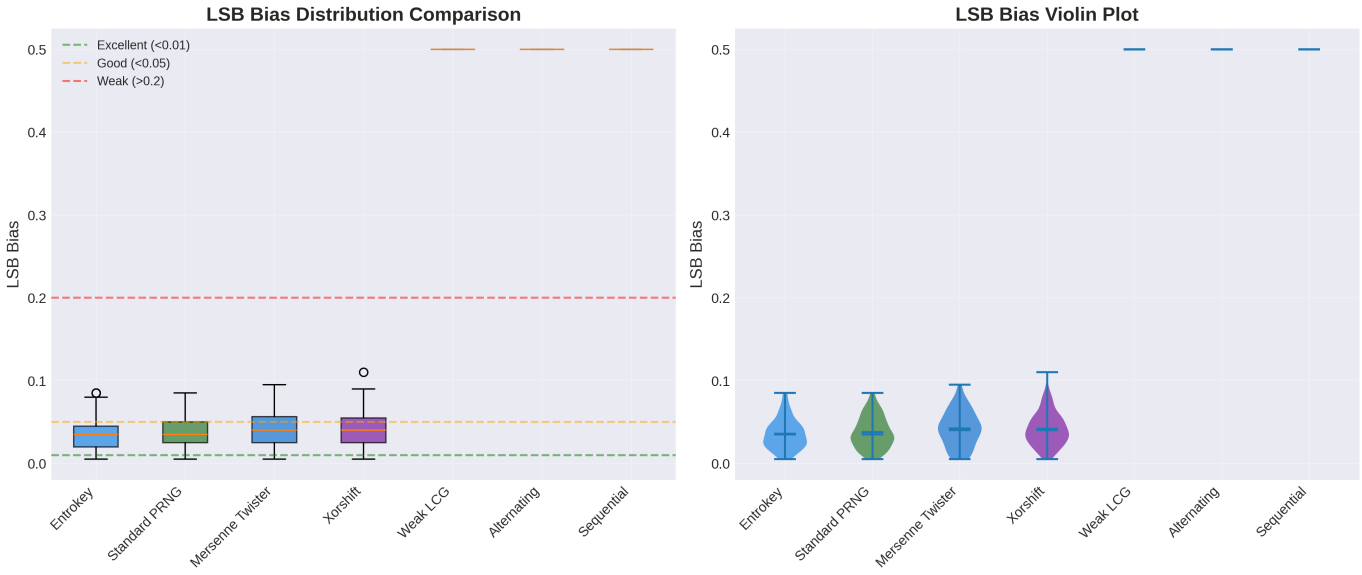

Fig. 11. Statistical comparison of LSB bias across entropy sources. Left: Box plots showing median, quartiles, and outliers. Right: Violin plots revealing distribution density. Clear bimodal separation exists between secure sources (bias < 0.1) and severely biased sources (bias = 0.5). ANOVA analysis confirms highly significant differences ($F = 27{,}109$, $p < 0.001$). Entrokey demonstrates the lowest mean bias among the tested generators.

selecting, and validating cryptographically secure entropy.

B. Implementation Considerations

Organizations considering entropy validation improvements have multiple implementation options, including hardware and software-based approaches. Hardware solutions such as Quantum Key Distribution (QKD) require significant infrastructure investment and have operational constraints. Software-based validation methods like Predictive Indexing offer an alternative approach.

Improving entropy source validation represents one approach to enhancing cryptographic security in response to emerging computational capabilities. As a software-based solution, Predictive Indexing can be deployed across existing infrastructure without requiring hardware modifications.

C. Applications and Future Work

The implications of this research extend beyond the generation of primary cryptographic keys. The principles of Predictive Indexing could be applied to cryptographic contexts requiring high-quality randomness, including the generation of protocol nonces, initialisation vectors (IVs), and padding schemes, where subtle biases can lead to vulnerabilities.

Future research could proceed along several avenues. First, the development of more advanced deep learning architectures, such as Transformer-based models, may enable the detection of longer-range and more abstract correlations within data streams. Second, the creation of a standardised, large-scale public benchmark dataset of patterned and random sequences would facilitate research and allow for the comparison of different validation models. Finally, the integration of real-time Predictive Indexing modules into hardware security modules (HSMs) and system-on-a-chip (SoC) designs could provide continuous validation of entropy sources at the hardware level.

D. Conclusion

The threats of AI-driven cryptanalysis and fault-tolerant quantum computing present challenges to traditional cryptographic validation methods. This paper has introduced and empirically evaluated Predictive Indexing, a CNN-based framework that emulates an intelligent adversary to assess entropy quality. Our results suggest advantages over traditional statistical tests, including the ability to differentiate entropy quality and the relationship with cryptographic properties such as ECC key generation. The adoption of AI-driven validation represents a potential evolution in cryptographic entropy assessment.

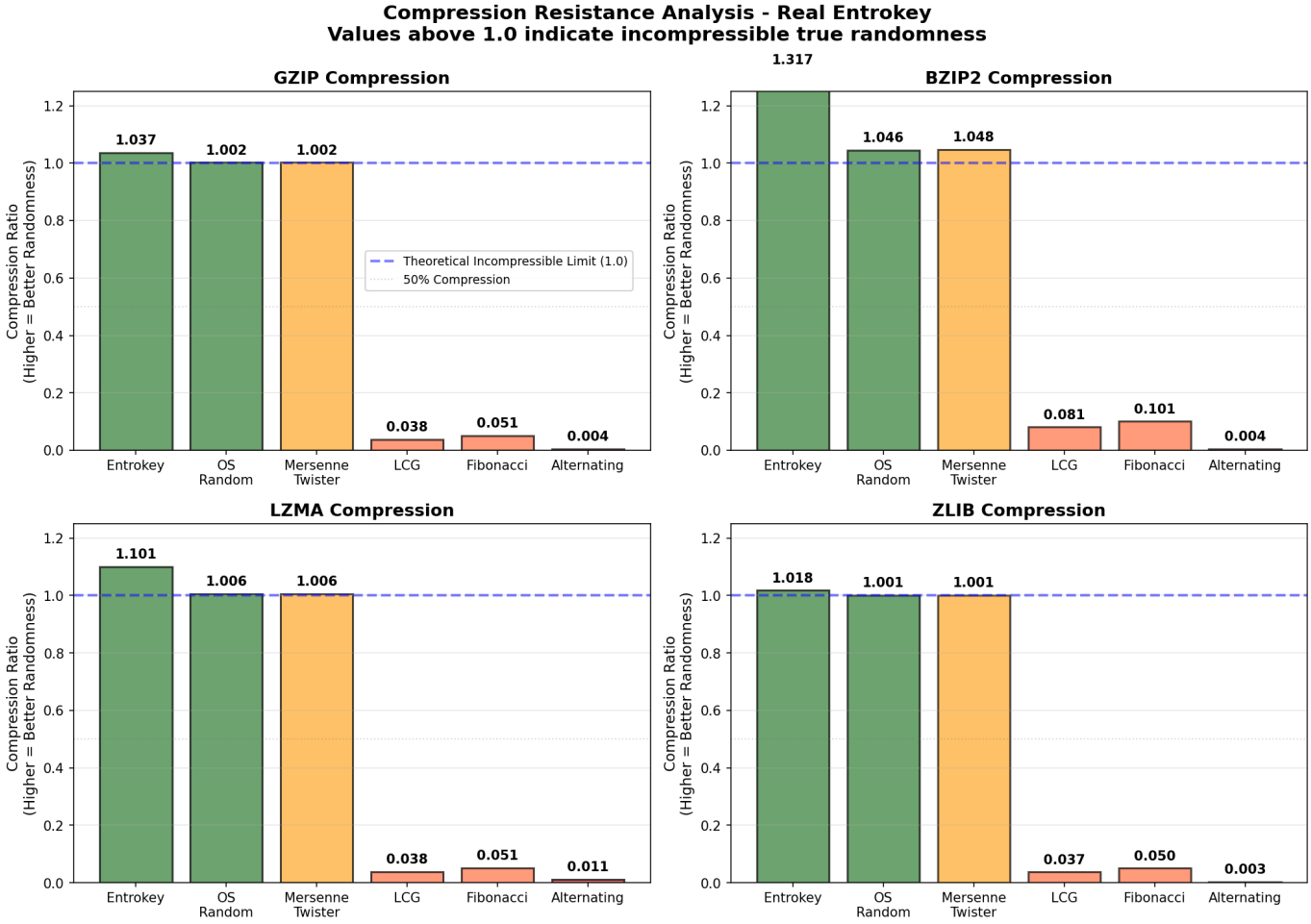

Fig. 12. Compression resistance analysis using real Entrokey generation. Each bar represents compression ratio for a single 10KB sample. High-quality sources (Entrokey, OS Random, Mersenne Twister) show ratios ≥ 1.0, indicating incompressibility. All values above 1.0 are equally random, with differences only reflecting compression overhead. Weak sources (LCG, Lagged Fibonacci taps, Alternating) compress dramatically, revealing deterministic patterns. The critical threshold is 1.0: above indicates true randomness, below indicates compressible patterns.

REFERENCES

[1] C. E. Shannon. (1948). "A Mathematical Theory of Communication." *The Bell System Technical Journal*, vol. 27, pp. 379-423, 623-656.

[2] Global Risk Institute & evolutionQ. (2024). *2024 Quantum Threat Timeline Report*.

[3] P. W. Shor. (1994). "Algorithms for quantum computation: discrete logarithms and factoring." *Proceedings 35th Annual Symposium on Foundations of Computer Science*, IEEE, pp. 124-134.

[4] National Institute of Standards and Technology. (2018). *Recommendation for the Entropy Sources Used for Random Bit Generation*. NIST Special Publication 800-90B.

[5] National Institute of Standards and Technology. (2010). *A Statistical Test Suite for Random and Pseudorandom Number Generators for Cryptographic Applications*. NIST Special Publication 800-22 Rev. 1a.

[6] K. Aungskunsiri, R. Amarit, S. Jantarachote, K. Wongpanya, P. Punpetch, et al. (2022). "Multiplexing quantum tunneling diodes for random number generation." *arXiv preprint arXiv:2212.12177*.

[7] N. Abraham, K. Watanabe, T. Taniguchi, K. Majumdar. (2022). "A High-Quality Entropy Source Using van der Waals Heterojunction for True Random Number Generation." *arXiv preprint arXiv:2204.06534*.

[8] H. Yiğit. (2025). "AI-Hybrid TRNG: Kernel-Based Deep Learning for Near-Uniform Entropy Harvesting from Physical Noise." *arXiv preprint arXiv:2507.00145*.

[9] M. A. Idowu. (2025). "Deterministic Cryptographic Seed Generation via Cyclic Modular Inversion over $\mathbb{Z}/3^p\mathbb{Z}$." *arXiv preprint arXiv:2507.03000*.

[10] B. G. Kim, D. Wong, Y. S. Yang. (2023). "Private and Secure Post-Quantum Verifiable Random Function with NIZK Proof and Ring-LWE Encryption in Blockchain." *arXiv preprint arXiv:2311.11734*.

[11] M. Velema. (2013). "Classical Encryption and Authentication under Quantum Attacks." *arXiv preprint arXiv:1307.3753*.

[12] S. Mossayebi, R. Schack. (2016). "Concrete Security Against Adversaries with Quantum Superposition Access to Encryption and Decryption Oracles." *arXiv preprint arXiv:1609.03780*.

[13] F. Kitagawa, T. Morimae, R. Nishimaki, T. Yamakawa. (2023). "Quantum Public-Key Encryption with Tamper-Resilient Public Keys from One-Way Functions." *arXiv preprint arXiv:2304.01800*.

[14] C. Osendorfer, J. Bayer, P. van der Smagt. (2013). "Convolutional Neural Networks learn compact local image descriptors." *arXiv preprint arXiv:1304.7948*.

[15] F. Altenberger, C. Lenz. (2018). "A Non-Technical Survey on Deep Convolutional Neural Network Architectures." *arXiv preprint arXiv:1803.02129*.

[16] G. Papandreou. (2014). "Deep Epitomic Convolutional Neural Networks." *arXiv preprint arXiv:1406.2732*.

[17] M. Kumar. (2023). "Design and Analysis of Pairing-Friendly Elliptic Curves for Cryptographic Primitives." *arXiv preprint arXiv:2307.09610*.

[18] K. Abhishek, E. G. D. P. Raj. (2022). "Computation of Trusted Short Weierstrass Elliptic Curves for Cryptography." *arXiv preprint arXiv:2208.01635*.

[19] H. Javashi, R. Sabbaghi-Nadooshan. (2011). "A Novel Elliptic curve cryptography Processor using NoC design." *arXiv preprint arXiv:1110.1046*.

[20] M. Hosseini, A. S. Maida, M. Hosseini, G. Raju. (2019). "Inception-inspired LSTM for Next-frame Video Prediction." *arXiv preprint arXiv:1909.05622*.

[21] E. Balouji, J. Sjöblom, N. Murgovski, M. H. Chehreghani. (2023). "Prediction of Time and Distance of Trips Using Explainable Attention-based LSTMs." *arXiv preprint arXiv:2303.15087*.

ACKNOWLEDGMENTS

Many thanks to Prof. Buchmueller (Imperial College London, CERN) for reviewing the accuracy, narrative, and general content of this paper prior to dissemination.