Exploring the Limits of AI Cryptanalysis with Semantic Discrimination Evaluation: From Enigma to RSA

Kazuya Yusa† and Akihito Nakamura† *
† Computer Science Division, University of Aizu
Aizu-Wakamatsu, Fukushima, Japan
* Email: nakamura@u-aizu.ac.jp

***Abstract*—Recent advancements in artificial intelligence (AI), including machine learning and deep learning, have shown high performance in pattern recognition. This raises a serious question: Can AI break modern cryptography? In this study, we empirically establish the boundary of AI-based cryptanalysis by evaluating four neural network models, Seq2Seq, Attention-based Seq2Seq, CNN, and Transformer, against four cipher types spanning classical ciphers (Simple Substitution, Vigenère), mechanical ciphers (Enigma), and modern ciphers (Simplified RSA). All models were trained on datasets generated from English literature. A central methodological contribution is the introduction of a "discrimination task." Unlike traditional character-level accuracy, which measures only spelling correctness, this task evaluates whether a model can identify semantically valid plaintext based on its output confidence, even when character reconstruction is imperfect. Formally, it tests whether the mutual information between plaintext and ciphertext remains exploitable. A score above the 50% random-guessing baseline indicates that the cipher leaks semantically exploitable structure. The results reveal a precise empirical boundary. Attention-based models exploited the statistical residue left by Enigma's deterministic rotor mechanics, achieving over 97% discrimination accuracy despite character-level accuracy below 27%—demonstrating that semantic leakage persists even without full decryption. In contrast, RSA's combination of probabilistic padding and modular exponentiation drives mutual information toward zero, causing all models, including the Transformer, to collapse to approximately 50% discrimination accuracy. These findings demonstrate that algorithmic complexity alone does not confer AI resistance. True resistance requires mathematical randomness that structurally eliminates all statistical dependencies between plaintext and ciphertext, not merely an increase in rule complexity.**

***Index Terms*—Cryptanalysis, Vigenère, Enigma, RSA, Seq2Seq, CNN, Transformer**

I. INTRODUCTION

Cryptography [1], [2] is the foundation of information security, and its safety and reliability are indispensable to today's information-driven society. The primary uses of cryptography are confidentiality, authentication, and digital signatures. The main cryptographic systems include symmetric-key cryptography and public-key cryptography. Cryptographic algorithms have evolved over time and become increasingly secure. Conversely, older cryptographic algorithms may become vulnerable over time.

Cryptanalysis is the science and study of breaking ciphers. A cipher is breakable if it is possible to determine the plaintext or key from the ciphertext, or to determine the key from plaintext-ciphertext pairs. Several methods are known, including the ciphertext-only attack, the known-plaintext attack, and the chosen-plaintext attack [1], [2]. For older ciphers, such as simple substitution, character frequency analysis is an effective method [3].

Deep learning has transformed pattern recognition, challenging the security assumptions of cryptographic systems that rely on obscuring linguistic statistics. Classical ciphers were designed under the assumption of limited adversarial computing power, but neural networks can now detect subtle statistical regularities that were previously exploitable only through manual cryptanalysis.

Recent studies [4], [5] have demonstrated neural network attacks on classical ciphers, yet systematic evaluation of when and why these approaches fail remains limited. We address this by testing four architectures, Seq2Seq [6] with Long Short-Term Memory (LSTM) [7], Attention-based Seq2Seq [8], 1D-CNN [9], [10], and Transformer [11], against four cipher types: Simple Substitution and Vigenère (classical) [2], [3], Enigma (rotor-based) [12], and RSA (public-key) [13]. Through training on English corpora, we establish empirical bounds for AI-based cryptanalysis and identify structural properties of ciphers that confer resistance to Such attacks.

The remainder of this paper is organized as follows. Section II reviews the related work. In section III, we explain the experimental method of cryptanalysis. In Section IV, we present the evaluation results. Section V discusses the characteristics and effectiveness of our method and results. Section VI concludes the paper.

II. RELATED WORK

A. Target Classical Ciphers

Recent advances have demonstrated that neural networks are highly effective against classical polyalphabetic substitution ciphers. Specifically, Ahmadzadeh et al. [4] applied LSTM-based Seq2Seq models to break the Vigenère Cipher. Their study demonstrated that deep learning models could detect periodic statistical patterns, achieving high decryption accuracy (over 90%) without any prior knowledge of the encryption key.

B. Target Mechanical Ciphers

Research has also extended to mechanical rotor machines, which are significantly more complex than hand-calculated ciphers. Greydanus [5] utilized Recurrent Neural Networks (RNNs) [14] to attack the Enigma Machine. This study demonstrated that the model was not merely memorizing text but was able to learn the dynamic rules and function of the Enigma mechanism to some extent. This suggested that even state-dependent encryption logic could be approximated by neural networks.

C. The Gaps and Our Contributions

Despite these successes, two major gaps remain in existing literature. First, previous studies have largely focused on specific "pattern-based" ciphers (such as Vigenère and Enigma), in which linguistic traces or algorithmic patterns remain. Second, there is a lack of direct comparison with modern "math-based" ciphers (such as RSA) within a unified experimental setting. This study bridges these gaps by applying the same neural network models to both cipher types. By doing so, we aim to empirically define the boundary where deep learning attacks fail.

III. CRYPTANALYSIS METHOD

This section discusses how to experiment and evaluate the performance of AI cryptanalysis. This study evaluates the cryptanalytic resistance of four cipher types using synthetic datasets and four neural network architectures.

A. Cryptanalysis Framework

Fig. 1 shows the AI cryptanalysis framework. In the training stage, plaintext is processed through three categories of cryptosystems—classical (Simple Substitution, Vigenère), mechanical (Enigma with rotors 1 to 3), and modern (Simplified RSA). Each cipher produces encrypted text, which is paired with the original plaintext to train our AI models.

During testing, we reverse the process. The trained models receive ciphertext as input and attempt to output the decrypted plaintext. We evaluate all four AI model architectures—Seq2Seq, Attention-based Seq2Seq, CNN, and Transformer—to determine which ciphers they can successfully break and where they fail.

B. Data Generation and Preprocessing

We generated synthetic training data from English literature in the NLTK Gutenberg corpus [15], which includes classic works such as "Moby Dick" and "Emma." The corpus was loaded sequentially from multiple books until reaching approximately 1.5 times the target dataset size. Text preprocessing consisted of converting all characters to lowercase and retaining only alphabetic characters and spaces, resulting in a 27-character vocabulary ('a' through 'z' and space).

To prevent data leakage, we adopted a strict hold-out strategy, partitioning the corpus into training (90%) and test (10%) sets before encryption. This pre-encryption split ensures minimal n-gram overlap between the training and evaluation

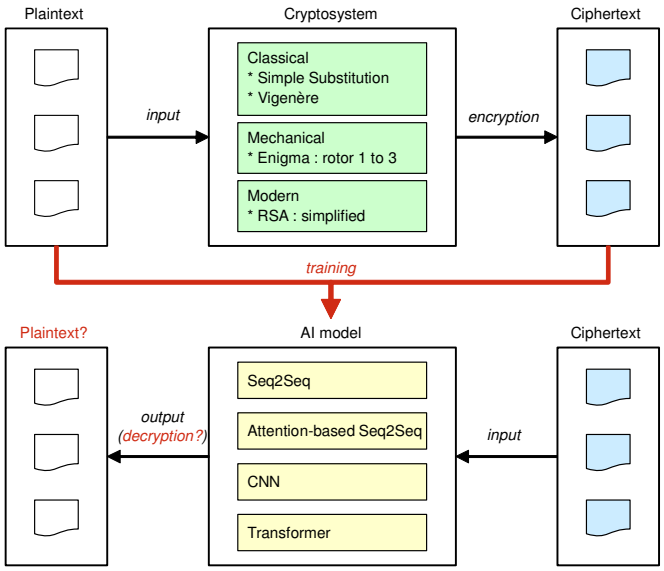


Fig. 1. AI Cryptanalysis Framework

data, preventing the model from memorizing specific plaintext sequences and encouraging it to learn generalizable decryption patterns.

From each partition, we extracted overlapping windows with a stride of two characters. For three ciphers, Simple Substitution, Vigenère, and Enigma, we used 100-character sequences directly. For RSA, due to block expansion, we extracted 12-character plaintext windows (seq_len/8), which expanded to approximately 100 characters after encryption.

C. Target Cryptosystems

We implemented four encryption algorithms representing distinct cipher classes [Fig. 1].

1) Monoalphabetic Cipher: Simple Substitution applies a monoalphabetic substitution in which each character is replaced by a fixed, random character throughout the text, serving as a baseline that preserves character frequency patterns.

2) Polyalphabetic Cipher: The polyalphabetic ciphers include Vigenère and Enigma. Vigenère uses the repeating 5-character key "lemon," shifting each plaintext character by the corresponding key position to create a periodic statistical structure with period length 5. Our Enigma simulation replicates the three-rotor Enigma I configuration (rotors I, II, III; reflector UKW-B) with randomized initial rotor positions and authentic stepping mechanics, in which the rightmost rotor advances with each keystroke and the middle rotor is at a notch position. Spaces were preserved without encryption to maintain sequence alignment.

3) Public-Key Encryption: For RSA, we implemented a character-block variant using approximately 31-bit modulus (p = 49,999, q = 49,993, n = 2,499,650,007). To resist frequency analysis, each character was padded with a random value (from 1 to 999) multiplied by a shift constant (100) before modular exponentiation (e = 65,537), and the resulting

integer was converted back to character representation. This process produces approximately eight ciphertext characters per plaintext character.

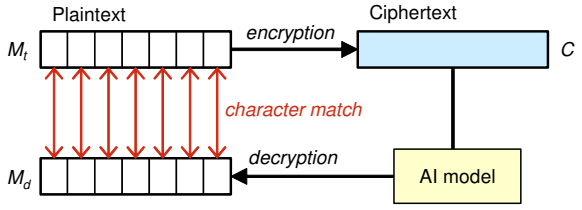
D. AI Models

We compared four AI models to identify structural requirements for successful cryptanalysis. The Seq2Seq baseline uses 256-unit LSTM layers for both encoding and decoding to establish basic sequence transduction capability. Bahdanau attention [8] in this architecture enables the decoder to dynamically weight the encoder hidden states, thereby addressing the limited context retention of standard RNNs by concatenating weighted context vectors with the decoder outputs.

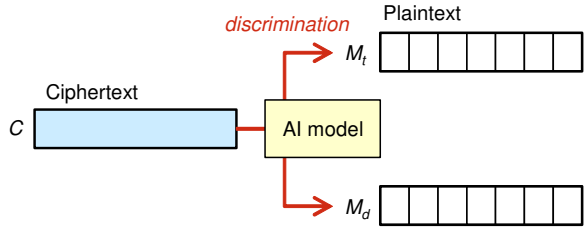
The 1D-CNN architecture tests whether local n-gram features suffice for cryptanalysis, using parallel convolutional layers (kernel sizes 3, 4, 5; 128 filters each) with global max pooling. The Transformer model uses a single encoder block with multi-head self-attention (2 heads, key dimension 64) and position-wise feed-forward networks, capturing global sequence dependencies through parallel attention operations rather than sequential recurrence. All models use 64-dimensional character embeddings and output softmax distributions over the 28-token vocabulary.

E. Evaluation Methods

To quantitatively assess the AI model's cryptanalytic capability, we employed two evaluation methods [Fig. 2].



(a) Character Match



(b) Discrimination

Fig. 2. Evaluation Methods

First, we measure character-level decryption accuracy [Fig. 2 (a)]. This metric calculates the percentage of characters in the decrypted output M_d that exactly match the ground-truth plaintext M_t. It indicates how closely the model approximates the correct spelling and serves as a measure of partial decryption success.

Second, we introduce a discrimination task to evaluate whether the models understand semantic context beyond simple pattern matching [Fig. 2 (b)]. For each test sample, we generate a pair consisting of the model's decrypted output, M_d, and the ground-truth plaintext, M_t. The model is asked to predict which text is more likely to be valid English based on its output confidence scores (softmax probability). Accuracy on this task is measured as the percentage of cases where the model correctly identifies the ground truth. Since random guessing yields 50% accuracy, this serves as our baseline. This metric tests whether the model has sufficiently learned linguistic structure to distinguish meaningful text from incorrect outputs.

The discrimination task offers a methodological advantage over character-level accuracy in the context of cryptographic security. Character-level accuracy measures whether a model can reproduce the exact plaintext spelling, but a low score does not necessarily imply that no meaningful information has been extracted from the ciphertext. From an information-theoretic perspective, a cipher should reveal no exploitable information about the plaintext. The discrimination task operationalizes this requirement. If a model can identify the true plaintext among the candidates with a probability significantly above 50%, the cipher leaks a semantically exploitable structure, regardless of its character-level performance. This property makes the discrimination task a more direct proxy for evaluating a cipher's cryptographic resistance against AI-based adversaries.

F. Experimental Setup

Models were trained to decrypt 100-character ciphertext sequences (except RSA inputs, which varied due to block expansion). For each cipher type, we generated up to 50,000 candidate samples from the training partition, then randomly sampled 10,000 for training and 4,000 from the test partition for evaluation.

All experiments were conducted using TensorFlow/Keras on Google Colab [16].

IV. Evaluation Results

In this section, we show the evaluation results of the cryptoanalysis.

A. Decryption Accuracy

Fig. 3 shows the results of standard decryption by the AI models.

For Simple Substitution, Attention-based Seq2Seq and Transformer achieved nearly perfect accuracy (99.2% and 99.8%), successfully breaking monoalphabetic ciphers. In contrast, the basic Seq2Seq achieved only 15.3%, and 1D-CNN struggled to 6.7%, indicating that local feature extraction alone is insufficient.

Vigenère introduced more complexity. Attention-based Seq2Seq and Transformer maintained moderate performance (62.1% and 60.4%), while basic Seq2Seq and CNN dropped

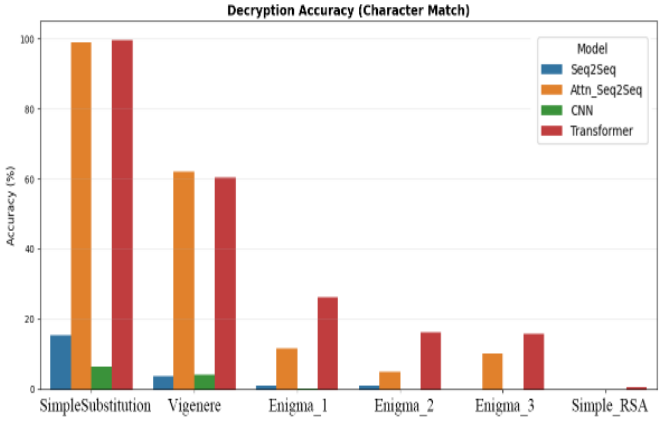


Fig. 3. Decryption Accuracy: Character Match

sharply to 3.7% and 4.2%, respectively. The attention mechanism appears necessary for handling periodic key structures.

Enigma proved considerably harder. With one rotor, only the Transformer showed a notable result (26.3%), while the others remained below 12%. Adding more rotors (Enigma2 and Enigma3) further degraded the Transformer's performance to around 16%. Basic Seq2Seq and CNN essentially failed ($<$ 2%), revealing the challenge for models with limited memory capacity.

Simple RSA presented a categorical failure. All models achieved near-zero accuracy ($<$ 1.5%), essentially random guessing. This stark contrast with Enigma establishes RSA as the empirical limit for pattern-based deep learning approaches.

B. Discrimination Accuracy

Next, we examined the discrimination to evaluate whether the AI models understand semantic context. Fig. 4 shows the results.

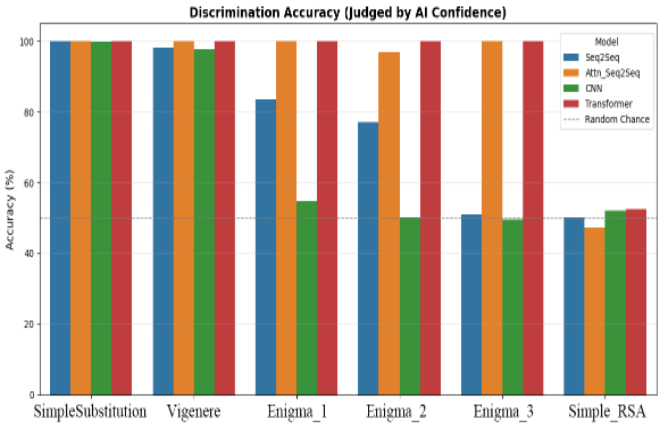


Fig. 4. Decryption Accuracy: Discrimination

For Simple Substitution and Vigenère, all models achieved very high scores (over 97%). This indicates that when character-level decryption succeeds, models also develop semantic understanding. They can easily distinguish meaningful text from random character sequences.

The Enigma results were particularly revealing. Even though the character accuracy was low (less than 27%), the Attention-based Seq2Seq and Transformer maintained very high discrimination scores (over 97%). These models had internalized English linguistic structure, even when unable to produce character-perfect outputs. They learned what English "looks like," not just how to match patterns.

In contrast, the basic Seq2Seq and CNN showed different results. Their scores dropped on Enigma (55 – 85%). This is likely because they have limited memory. They struggle to keep track of the complex changes in the Enigma machine, which requires remembering the whole sequence.

Finally, for Simple RSA, the result was completely different. All models fell to around 50%, which is exactly the same as random guessing. This confirms that RSA eliminates not just the spelling but also any understanding of the meaning. The models lost both the ability to spell and the ability to understand context.

V. DISCUSSION

A. Empirical Boundaries of AI Cryptanalysis

Our results establish empirical boundaries for AI-based cryptanalysis.

Models succeeded on classical ciphers like Simple Substitution. This is because these ciphers keep the statistical patterns of the original language. For example, if 'e' is common in English, it will still be common in the encrypted text. Models could easily detect these statistical regularities to break the code.

Enigma's partial vulnerability to attention-based models can be attributed to three structural properties that persist despite its rotor-based complexity. First, spaces were left unencrypted in our implementation, thereby preserving word-boundary structure and word-length distributions in the ciphertext. This constitutes a statistically rich source of information about the underlying English text, as English word-length distributions are highly characteristic. Second, Enigma's rotor-stepping mechanism, while producing a large state space, follows entirely deterministic and periodic transition rules. The rightmost rotor advances with every keystroke (period 26), inducing periodic statistical correlations in the ciphertext that extend across the full sequence length. Third, because Enigma operates exclusively within a fixed 27-character alphabet, residual bigram and trigram frequency biases remain detectable across the ciphertext, even after polyalphabetic substitution.

These vulnerabilities are inherently global: detecting word-boundary patterns, rotor-period correlations, and character-frequency biases requires aggregating information across the entire sequence. Attention-based models are architecturally suited to exploit these properties. The Bahdanau attention mechanism enables dynamic reference to all encoder positions when generating each output token, while the Transformer's self-attention directly computes pairwise dependencies across

all sequence positions in parallel. Both architectures can therefore capture long-range statistical regularities corresponding to Enigma's periodic structure.

In contrast, CNN and basic Seq2Seq models fail to achieve comparable discrimination performance due to fundamental architectural limitations. The 1D-CNN's local receptive field (kernel sizes of 3–5) restricts feature extraction to neighboring characters, making it unable to detect word-length distributions or rotor-period correlations spanning dozens of positions. Although global max pooling aggregates features across the sequence, it discards positional information, destroying the ordered structure necessary for detecting periodic patterns. The basic Seq2Seq model processes the full sequence but compresses all information into a fixed-dimensional context vector, causing long-range statistical dependencies to dissipate before reaching the decoder. This information bottleneck is particularly severe for 100-character sequences, where early-position information is substantially attenuated by the time it reaches the final hidden state.

RSA encryption caused categorical failure across all models for reasons that are fundamentally distinct from those underlying Enigma's partial resistance. The failure stems from three compounding mechanisms in our RSA implementation. First, random padding adds an independent random value $r_i \in [1, 999]$ to each character before encryption, ensuring that identical plaintext characters produce different intermediate values on every encryption. This eliminates character frequency biases entirely: unlike Enigma, where the fixed alphabet constrains substitution targets, RSA maps each character to one of up to 999 distinct values before modular exponentiation. Second, the modular exponentiation operation $c_i = (p'_i)^e \bmod n$ distributes output values nearly uniformly across the range $[0, n)$, where $n \approx 2.5 \times 10^9$. This uniform distribution ensures that no statistical correlation between plaintext and ciphertext survives. Third, block expansion—where each plaintext character produces approximately eight ciphertext characters—destroys the positional correspondence that was preserved under Enigma's character-by-character substitution. Word boundaries, which remained detectable in Enigma ciphertext through unencrypted spaces, are completely obscured in RSA ciphertext.

From an information-theoretic perspective, these properties drive the mutual information between plaintext and ciphertext, $I(P; C)$, toward zero. Because the discrimination task requires a model to identify the true plaintext based on statistical evidence extracted from the ciphertext, a near-zero $I(P; C)$ renders the task equivalent to random guessing. The observed discrimination accuracy of approximately 50% across all models, including the Transformer, directly confirms this condition.

This outcome reveals the fundamental distinction between Enigma and RSA. Enigma's complexity is algorithmic: its rotor mechanics generate a large state space through deterministic, rule-based transitions. Neural networks, particularly attention-based architectures, can approximate these rules and exploit the statistical residue they leave. RSA's security, by contrast, is grounded in mathematical hardness: the one-way nature of modular exponentiation, whose inversion requires integer factorization, cannot be approximated through input-output pattern learning. No amount of architectural sophistication allows a neural network to recover information that has been structurally eliminated by mathematical randomness. This distinction—between algorithmic complexity and mathematical randomness—represents the empirical boundary identified in this study.

B. Context Understanding and Pattern Matching

The discrimination task revealed an important difference between simple pattern matching and true understanding. The advanced models—Attention-based Seq2Seq and Transformer—could understand linguistic context and meaning, not just match characters at the surface level.

The Enigma results were particularly revealing and warrant a mechanistic explanation. Despite low character accuracy ($< 27\%$), attention-based models achieved over 97% discrimination accuracy. This dissociation arises from two concurrent factors: the structural vulnerabilities of Enigma and the architectural capabilities of attention-based models.

Although Enigma's rotor-stepping mechanism creates a vast state space (up to 17,576 rotor configurations for three rotors), it does not eliminate linguistic structure from the ciphertext. Because Enigma operates as a character-level substitution constrained to a fixed alphabet, and because spaces were preserved without encryption in our implementation, word boundaries and residual character frequency biases remain detectable in the ciphertext. These statistical traces are insufficient for character-perfect decryption, which requires tracking the exact rotor state at each position—a task that exceeds the capacity of our model architectures. However, they provide enough global structure for a model to distinguish English text from random sequences.

Attention-based models exploit these traces by aggregating information across the entire input sequence. The Bahdanau attention mechanism allows the decoder to dynamically weight all encoder hidden states when generating each output token, enabling the model to capture global statistical regularities rather than local character patterns. The Transformer extends this further through parallel self-attention over all position pairs, directly modeling long-range dependencies that correspond to rotor-induced correlations across the sequence. As a result, these models effectively internalize a statistical model of English that is robust enough to support semantic discrimination, even when character-level reconstruction fails. In contrast, CNNs and basic Seq2Seq models—limited to local receptive fields and sequential hidden states, respectively—cannot aggregate sufficient global context to support this judgment, which explains their near-random discrimination scores on Enigma.

This dissociation between character accuracy and discrimination accuracy reveals a critical limitation of relying solely on character-level metrics. A model that achieves 97% discrimination accuracy on Enigma—despite character accuracy below

27%—has internalized enough of the English linguistic distribution to identify semantically coherent text. In a real-world adversarial scenario, this capability is more dangerous than high character accuracy because it implies that the model can leverage partial semantic leakage even without recovering the full plaintext. The discrimination task thus provides a stricter, more security-relevant evaluation criterion: a cipher should be considered AI-resistant only if it reduces discrimination accuracy to the random-guessing baseline of 50%.

C. Implications for Cryptographic Security

Our findings have important implications for future security. The experimental trajectory from Simple Substitution through Enigma to RSA reveals a consistent pattern: algorithmic complexity, measured by state-space size or rule intricacy, does not monotonically increase AI resistance as measured by the discrimination task. Attention-based models maintained near-perfect discrimination accuracy across all three algorithmically complex ciphers, while failing completely only when confronted with RSA's mathematical randomness. This dissociation between complexity and resistance can be understood through the lens of mutual information: algorithmically complex but deterministic ciphers preserve non-zero statistical dependencies between plaintext and ciphertext, whereas mathematically random ciphers structurally eliminate these dependencies. The implication for cipher design is that complexity is a necessary but insufficient condition for AI resistance. True AI resistance requires that the encryption function be probabilistic and grounded in mathematical hardness assumptions, such that no statistical correlation between plaintext semantics and ciphertext patterns survives for a learning-based adversary to exploit.

Complexity alone does not guarantee AI resistance. Even though Enigma had complex rotor mechanics, the AI models could still detect hidden statistical patterns. To effectively stop AI, we need mathematical operations that eliminate all patterns, just like RSA did. This challenges the conventional understanding of cipher strength.

Enigma is hard because of its complex rules (state changes), but RSA is hard because of a mathematical problem (integer factorization). Our results suggest that AI can learn "algorithmic rules," but cannot overcome "mathematical hardness." Future cipher design should prioritize mathematical randomness rather than just making the algorithm complex.

The discrimination task also provides a practical testing framework. If a neural network can distinguish the real decrypted text from the fake one better than random guessing (50%), it means the cipher is leaking information. This provides a simple, direct method for assessing AI resistance when designing new ciphers.

VI. CONCLUSION

This study empirically establishes the boundary of AI-based cryptanalysis by systematically evaluating four neural network architectures against four cipher types. The central finding is that algorithmic complexity and AI resistance are not equivalent properties. Attention-based models, particularly the Transformer, successfully exploit the statistical residue left by deterministic, rule-based ciphers, including Enigma's rotor mechanics, thereby achieving both character-level decryption and semantic discrimination. RSA, however, represents a categorical limit: its probabilistic padding and modular exponentiation drive the mutual information between plaintext and ciphertext toward zero, causing all models to collapse to the 50% random-guessing baseline.

Our work makes three key contributions. First, we identified the structural property that confers genuine AI resistance: mathematical randomness that eliminates all statistical dependencies between plaintext and ciphertext. Enigma's sophisticated state space proved insufficient because its transitions remain deterministic and rule-based; RSA's mathematical hardness provided complete protection where rule complexity could not.

Second, we demonstrated that attention-based models develop semantic understanding that persists even when character-level reconstruction fails. On Enigma, discrimination accuracy exceeded 97% despite character accuracy below 27%, indicating that semantic leakage poses a security risk independently of spelling-level decryption success.

Third, we established a practical criterion for evaluating AI resistance: a cipher should be considered secure against learning-based adversaries only if it suppresses discrimination accuracy to the 50% random-guessing baseline—confirming that mutual information between plaintext and ciphertext has been structurally eliminated.

Future work should address three open questions: determining the minimum degree of randomness sufficient to neutralize AI-based attacks, evaluating hybrid ciphers that trade efficiency for security, and extending the discrimination task to other languages and cipher families.

REFERENCES

[1] D. E. R. Denning, *Cryptography and Data Security*. Addison-Wesley Publishing Company, 1982.

[2] W. Stallings, *Cryptography and Network Security: Principles and Practices*, 4th ed. Pearson Education, 2006.

[3] E. Dunin and K. Schmeh, *Codebreaking: A Practical Guide*, 4th ed. No Starch Press, September 2023.

[4] E. Ahmadzadeh, H. Kim, O. Jeong, and I. Moon, "A Novel Dynamic Attack on Classical Ciphers Using an Attention-Based LSTM Encoder-Decoder Model," *IEEE Access*, vol. 9, pp. 60 960–60 970, April 2021.

[5] S. Greydanus, "Learning the Enigma with Recurrent Neural Networks," *arXiv*, 2017. [Online]. Available: https://arxiv.org/abs/1708.07576

[6] I. Sutskever, O. Vinyals, and Q. V. Le, "Sequence to Sequence Learning with Neural Networks," *arXiv*, 2014. [Online]. Available: https://arxiv.org/abs/1409.3215

[7] S. Hochreiter and J. Schmidhuber, "Long Short-Term Memory," *Neural Computation*, vol. 9, no. 8, pp. 1735–1780, 11 1997.

[8] D. Bahdanau, K. Cho, and Y. Bengio, "Neural Machine Translation by Jointly Learning to Align and Translate," in *International Conference on Learning Representations (ICLP 2014)*, CA, USA, May 2014. [Online]. Available: https://arxiv.org/abs/1708.07576

[9] R. Collobert and J. Weston, "A Unified Architecture for Natural Language Processing: Deep Neural Networks with Multitask Learning," in *25th International Conference on Machine Learning (ICML'08)*. Association for Computing Machinery, 2008, pp. 160–167.

[10] A. Krizhevsky, I. Sutskever, and G. E. Hinton, "ImageNet Classification with Deep Convolutional Neural Networks," *Communications of the ACM*, vol. 60, no. 6, pp. 84–90, 2017.

[11] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is All You Need," in *Proceedings of the Advances in Neural Information Processing Systems (NIPS 2017)*, I. Guyon, U. V. Luxburg, S. Bengio, H. Wallach, R. Fergus, S. Vishwanathan, and R. Garnett, Eds., vol. 30. CA, USA: Curran Associates, Inc., December 2017. [Online]. Available: https://proceedings.neurips.cc/paper_files/paper/2017/file/3f5ee243547dee91fbd053c1c4a845aa-Paper.pdf

[12] M. Rejewski, "How Polish Mathematicians Broke the Enigma Cipher," *Annals of the History of Computing*, vol. 3, no. 3, pp. 213–234, 1981.

[13] R. L. Rivest, A. Shamir, and L. M. Adleman, "A Method for Obtaining Digital Signatures and Public-Key Cryptosystems," *Communications of the ACM*, vol. 21, no. 2, pp. 120–126, February 1978.

[14] D. E. Rumelhart, G. E. Hinton, and R. J. Williams, "Learning Representations by Back-Propagating Errors," *Nature*, vol. 323, pp. 533–536, October 1986.

[15] S. Bird, E. Klein, and E. Loper, "NLTK Tutorial: Introduction to Natural Language Processing," *Creative Commons Attribution*, vol. 1037, April 2005. [Online]. Available: https://pages.ucsd.edu/ rbelew/courses/readings/bird05-NLTK-intro.pdf

[16] "Google Colab," Date of Access: 2026-03-25. [Online]. Available: https://colab.research.google.com/