Cryptanalysis of Hagelin M-209 Cipher Machine with Artificial Neural Networks: A Known-Plaintext Attack

Vasily Mikhalev[1*], Nils Kopal[1], Bernhard Esslinger[1],
Harald Lampesberger[2], Eckehard Hermann[2]
[1]University of Siegen, Germany
[2]University of Applied Sciences Upper Austria, Hagenberg, Austria
[*]vasily.mikhalev@gmail.com

Abstract

This paper introduces a machine learning (ML) approach for cryptanalysis of the cipher machine Hagelin M-209[1]. For recovering the part of the secret key, represented by the wheel pins, we use Artificial Neural Networks (ANN) which take as input the pseudo-random displacement values generated by the internal mechanism of the machine. The displacement values can be easily obtained when ciphertext and plaintext are known. In particular, we are using several distinct ANNs, each recovering exactly one pin. Thus, to recover all the 131 pins, we utilize 131 models each solving a binary classification problem. By experimenting with various ANN architectures and ciphertext lengths, ranging from 52 to 200 characters, we identified an ANN architecture that outperforms others in accuracy. This model, inspired by the architecture by Gohr used for attacking modern ciphers, achieved the following accuracies in recovering the pins of the first wheel of the machine: approximately 71% for 52-characters sequences, 88% for 104-characters, 96% for 200-characters. The first wheel has the largest size and hence represents the most complicated case. For the other wheels, these accuracies are slightly higher. To the best of our knowledge, this is the first time when ANNs are used in a key-recovery attack against such machines.

[1]The code used for this paper and instructions of how to download the data are available at `https://github.com/CrypToolProject/M209KnownPlaintextAttackML`

1 Introduction

The Hagelin M-209 is a mechanical cipher machine designed by Boris Hagelin and used by the United States military extensively during World War II. This device encapsulates the era's challenges and advancements in secure communications. The M-209, based on a stream cipher mechanism, encrypts messages using several wheels and bars whose interaction produces pseudo random numbers that are combined with plaintext for encryption.

Despite its historical significance, the M-209 has been the subject of various cryptanalytic efforts, showcasing vulnerabilities typical for mechanical encryption devices. ML presents new opportunities for cryptanalysis of such historical ciphers. This paper explores the application of ANNs in a known-plaintext attack (KPA) scenario, where the plaintext and corresponding ciphertext are known, enabling the study of the machine's internal key generation mechanism.

Our research aims to demonstrate the efficacy of using ANNs to cryptanalyze the Hagelin M-209. By treating the sequences of pseudo random numbers generated by the machine as data inputs for ML models, we train each of them to recover only one specific bit of the key represented by the wheel pin. This approach establishes a novel technique in cryptanalysis that could be extended beyond the M-209 to other cipher machines.

The remainder of this paper is organized as follows: Section 2 reviews related work in the field of attacks against M-209. Section 3 gives a short description of the machine. Section 4 details our methodology, including data preparation and neural network architectures used. Section 5 presents the results of our experiments. Finally, Section 6 concludes the paper with a summary of our findings and potential future research directions.

2 Related Work

Modern cryptanalytic efforts on the Hagelin M-209 used a variety of methodologies to break its cipher mechanism.

Morris's approach (Morris, 1978) involved a manual KPA. By analyzing and refining the displacement sequence, Morris's method allows to recovery of the internal key from 100-character-long messages and corresponding ciphertexts. Barker (Barker, 1977) proposed a ciphertext-only attack (COA) based on analysis of letter frequency distribution patterns. It requires from 2000 to 4000 letters to recover the key. In the work (Beker and Piper, 1982) a COA was presented using techniques to classify pins and solve ambiguities. Their method is claimed to be effective if 2500 characters are available. Sullivan's approach (Sullivan, 2002) for ciphertext-only recovery of M-209 key used a divide-and-conquer approach, incrementally recovering pin and lug settings, requiring 2500 letters to obtain the key.

Lasry, Kopal, and Wacker made notable advancements in applying heuristic algorithms to the cryptanalysis of M-209. Their initial contribution was an automated approach for known-plaintext cryptanalysis of short messages, having only 50 characters, in (Lasry et al., 2016a). They later proposed a COA in (Lasry et al., 2016b), which was improved in (Lasry et al., 2018), enabling the recovery of the machine's internal settings from approximately 500 characters of ciphertext.

Most recently, the exploration of ML techniques in the classification of World War II era ciphers, including the M-209 was done by Dalton and Stamp (Dalton and Stamp, 2023).

3 The Hagelin M-209 Machine

M-209 operates as a stream cipher[2]. The main component is a pseudo-random keystream generator (KSG) which is used to produce sequences of displacement values, which modify the plaintext to generate the ciphertext, as illustrated by Eq 1:

$$c_i = (25 - p_i + d_i) \; mod \; 26 \qquad (1)$$

In this equation, p_i and c_i represent the positions of the plaintext and ciphertext characters in the Latin alphabet[3], respectively, while d_i is the displacement value generated for encryption of the i-th character of the plaintext.

The KSG comprises a 27-bar cage and six wheels with varying numbers of letters (26, 25, 23, 21, 19, 17). Each letter on a wheel has a pin that can be set into active or inactive state. Every bar in the cage has two adjustable lugs, which can be set against any of the six wheels or remain in a neutral (zero) position. When both lugs on a bar are placed in an active position, it results in a *lug overlap* involving the bar and two wheels. The number of total overlaps is an important property for analysis of the key complexity as shown in the Section 5,

With each encryption step, the cage makes the full rotation around its 27 bars. If a bar lug interacts with an active wheel pin, the bar shifts to the left. The displacement value d_i is determined by the number of shifted bars and can range from 0 to 27. Before the next encryption cycle, all wheels advance by one position.

The machine's secret key consists of the initial wheel positions, the pin settings (active/inactive), and the lug positions. These configurations are not arbitrary and followed the specific operating instructions, which were changing over time. This study focuses on the keys that correspond to 1944 operating instructions (War Department, 1944).

4 Methodology

This study focuses on the KPA scenario which allows to reverse Equation 1 to derive sequences of displacement values, which are utilized as inputs for our ANN. The ANN's goal is to predict M-209 pin values from a sequence of displacement values[4]. This problem can also be regarded as a binary classification problem. Instead of predicting all 131 pins at once with a single model, we consider each pin as an individual target. Therefore, 131 binary classifier models that share the same ANN architecture are used to predict the states of all pins.

If different keys used in a cipher for encrypting the same plaintext lead to the same ciphertext, these keys are called equivalent. An important M-209 property used in our methodology is that for any initial position of the wheels, there are pin settings that result in the equivalent keys (Lasry

[2]We only discuss the most important parts of the machine, directing readers to Lasry's PhD thesis (Lasry, 2018) for detailed explanations and to Chapter 2.5 in (Esslinger, 2024) for additional Hagelin Machine models.

[3]A is encoded by 0, B by 1 etc..

[4]The ANN doesn't get information about the lugs settings.

et al., 2018). This property is crucial as it implies that we can assume any initial position of the wheels, such as the default position "AAAAAA", and find proper pin configurations which constitute an equivalent key to the one used during the actual encryption process. When such an equivalent key is found, we can decrypt the remainder of the ciphertext without the need for plaintext.

For training the models, we generated millions of random keys following the operational instructions provided in the 1944 technical manual. Utilizing these keys, we created sequences of displacement values which are 52, 104, and 200 numbers long. The choice of length 52 ensures that the first wheel completes two full rotations, such that every pin contributed at least twice in the displacement generation process. Similarly, length 104 corresponds to four full rotations. In experiments, this approach led to a slight but noticeable improvement in accuracy compared to sequences of 50 and 100 characters. Extending the sequence length to 208 did not bring a notable difference in the results but challenges in the training process; therefore, a maximum length of 200 was considered in the experiments.

We conducted experiments with several ANN architectures using a range of hyperparameters, including Feedforward Neural Networks, LSTM, and Transformer models. The best results were obtained using a residual-networks-like architecture, similar to the one employed by Gohr in his work on cryptanalysis of modern ciphers through ML (Gohr, 2019). This specific architecture incorporates a dual-layer convolutional residual tower (DLCRT), having a dense-prediction layer as shown in the Figure 1.

Residual networks (ResNets) use shortcut connections, allowing data to skip layers and directly connect to deeper layers. This addresses the so-called vanishing gradient problem and enables training of deeper networks efficiently.

The implemented DLCRT model processes input displacement values by first normalizing them through division by 25 during preprocessing. These values are then fed into the initial convolution layer, which is succeeded by five residual blocks, each comprising two convolutional layers. Subsequently, a dense layer, in which every input is connected to every output, transmits the values to the output layer which is responsible for generating the final prediction for a given pin.

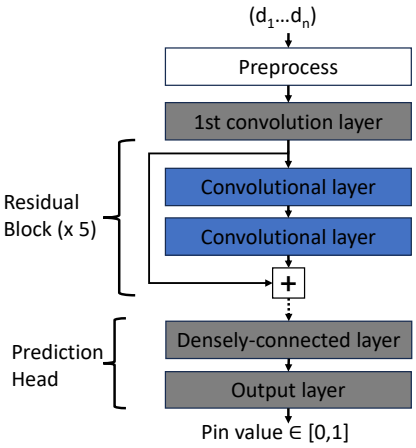

Figure 1: Used DLCRT architecture

5 Results

We now present the results[5] of testing our trained models with the randomly generated data, distinct from the data utilized during the training phase. For brevity, our discussion centers on the scenario of reconstructing the pins of the first wheel, which comprises 26 pins. This wheel is the largest among its counterparts, thus posing the most challenging scenario. The results for the other wheels are marginally better.

For testing, we generated 1000 random keys for each of the key classes based on their complexity which depends on 2 parameters. The first one is the number of lugs not involved in the lug overlap (non-shared lugs) positioned against the wheel, the pins of which are being recovered. The second one is the number of overlaps with the other wheels. To get an intuition on the significance of the number of non-shared lugs, consider a scenario where a wheel is set against 13 lugs. In this case, if the total displacement value is between[6] 15 and 25, it guarantees that the current pin of this wheel is active; otherwise, it's inactive. Such a property is easy to detect for an ANN, as opposed to situations where only one or two lugs are set against a wheel, which introduces a greater level of uncertainty.

For a single test, accuracy is defined as the ratio of correctly identified pins out of the total 26.

The accuracy distributions for sequence lengths n=52, n=104 and n=200 are depicted in figures: Figure 2, Figure 3 and Figure 4 respectively. In

[5]Initial measurements were done in (Landrichinger and Mikhalev, 2023)

[6]The values 26 and 27 are reduced modulo 26 in accordance with the Equation 1.

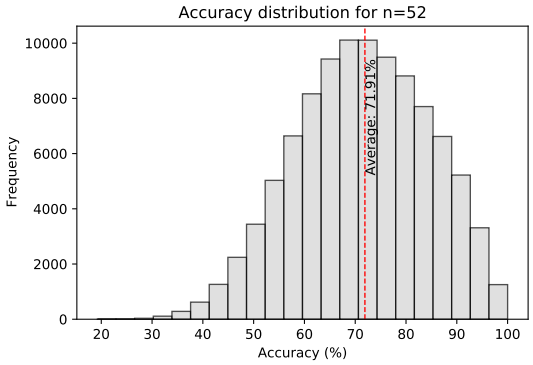

Figure 2: Accuracy distribution for displacement sequence length 52 (n= 52)

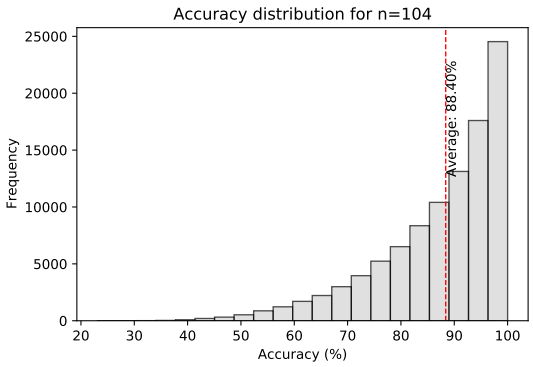

Figure 3: Accuracy distribution for displacement sequence length 104 (n= 104)

terms of average accuracy, sequences of 52 characters yield 71%, those with 104 characters reach 88%, and for sequences encompassing 200 characters, the accuracy stands at 96%. This behavior was expected because longer displacement sequences carry more information and the average accuracy therefore increases.

Additionally, our study includes an examination of the ANN's performance relative to the varying complexities of the keys.

The results related to the most difficult scenario, the 52-letter sequences, are shown in Table 1. These results suggest that in simple cases when a key has many non-shared lugs, the ANN can accurately recover the pins using only 52 characters of ciphertext and corresponding plaintext.

6 Conclusion and Future Work

This study showcased the potential of ANNs in cryptanalysis of the Hagelin M-209 cipher machine using a KPA. Our experiments with various ANN architectures, particularly a custom ResNet-like model, highlight the effectiveness of ML in the cryptanalysis of such machines.

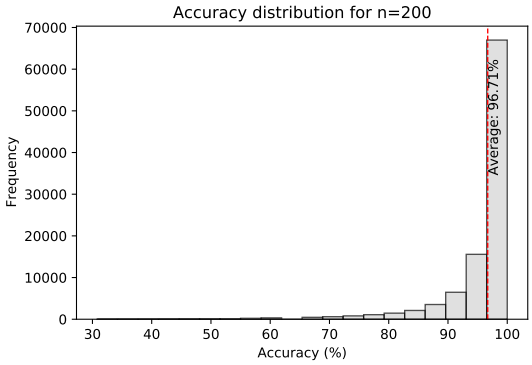

Figure 4: Accuracy distribution for displacement sequence length 200 (n= 200)

N\O	0	1	2	3	4	5	6	7	8	9	10	11	12
0	-	51	51	53	55	57	58	59	61	63	65	65	68
1	54	54	55	57	59	61	61	63	65	66	68	69	71
2	55	57	59	60	62	64	64	66	68	69	70	72	-
3	59	60	62	64	66	67	68	70	71	72	74	-	-
4	63	64	66	68	70	72	71	72	74	75	-	-	-
5	66	69	70	72	73	74	74	76	78	-	-	-	-
6	69	71	73	75	76	78	79	81	-	-	-	-	-
7	73	75	76	79	82	82	83	-	-	-	-	-	-
8	76	78	81	83	85	86	-	-	-	-	-	-	-
9	81	83	85	87	89	-	-	-	-	-	-	-	-
10	85	87	89	90	-	-	-	-	-	-	-	-	-
11	88	90	92	-	-	-	-	-	-	-	-	-	-
12	91	93	-	-	-	-	-	-	-	-	-	-	-
13	93	-	-	-	-	-	-	-	-	-	-	-	-

Table 1: Mean accuracy in % for the different number of non-shared lugs (represented by rows) and overlaps (represented by columns). These results are for sequences of 52 characters long.

The next steps of our research will include:

- Enhancing the accuracy of pin recovery, especially in shorter sequences, and designing the techniques to improve the errors.
- Recovery of the bar lugs.
- Investigating ciphertext-only cryptanalysis approaches.
- Applying ML to other cipher machines.

Acknowledgments

We are grateful to George Lasry who proposed the idea of this attack and for his valuable comments. This work was supported by the Swedish Research Council, grant 2018-06074, DECRYPT – Decryption of Historical Manuscripts.

References

Wayne G. Barker. 1977. *Cryptanalysis of the Hagelin Cryptograph*, volume 17. Aegean Park Press, Laguna Hills, CA.

Henry Beker and Fred Piper. 1982. *Cipher Systems: The Protection of Communications*. Northwood Books, London.

Brooke Dalton and Mark Stamp. 2023. Classifying World War II Era Ciphers with Machine Learning. *arXiv preprint arXiv:2307.00501*.

Bernhard Esslinger. 2024. *Learning and Experiencing Cryptography with CrypTool and SageMath*. Artech House, Norwood. https://us.artechhouse.com/Learning-and-Experiencing-Cryptography-with-CrypTool-and-SageMath-P2378.aspx.

Aron Gohr. 2019. Improving attacks on round-reduced speck32/64 using deep learning. In *Advances in Cryptology–CRYPTO 2019: 39th Annual International Cryptology Conference, Santa Barbara, CA, USA, August 18–22, 2019, Proceedings, Part II 39*, pages 150–179. Springer.

Robert Landrichinger and Vasily Mikhalev. 2023. Cryptanalysis of Cipher Machines with the Help of Artificial Neural Networks. Talk at Security Forum, Hagenberg im Mühlkreis, Austria. Retrieved from https://www.securityforum.at/cryptanalysis-of-cipher-machines-with-the-help-of-artificial-neural-networks-tag-2. Accessed: 25 January, 2024.

George Lasry, Nils Kopal, and Arno Wacker. 2016a. Automated Known-Plaintext Cryptanalysis of Short Hagelin M-209 Messages. *Cryptologia*, 40(1):49–69.

George Lasry, Nils Kopal, and Arno Wacker. 2016b. Ciphertext-only cryptanalysis of Hagelin M-209 pins and lugs. *Cryptologia*, 40(2):141–176.

George Lasry, Nils Kopal, and Arno Wacker. 2018. Ciphertext-only cryptanalysis of short Hagelin M-209 ciphertexts. *Cryptologia*, 42(6):485–513.

George Lasry. 2018. A Methodology for the Cryptanalysis of Classical Ciphers with Search Metaheuristics.

Robert Morris. 1978. The Hagelin Cipher Machine (M-209) Reconstruction of the Internal Settings. *Cryptologia*, 2(3):267–289.

Geoff Sullivan. 2002. Cryptanalysis of Hagelin Machine Pin Wheels. *Cryptologia*, 26(4):257–273.

War Department. 1944. TM-11-380, Technical Manual, Converter M-209, M-209A, M-209B (Cipher). https://deweger.net/apparaten/downloads/M209%20manual.pdf. Accessed: 25 January, 2024.