Deep Learning Based Analysis of Key Scheduling Algorithms of Advanced Ciphers

Narendra Kumar Patel
School of CSE
VIT Bhopal University 466114
narendra.k91630@gmail.com

Hemraj Shobharam Lamkuche
School of CSE
VIT Bhopal University 466114
hemraj.lamkuche@gmail.com

Abstract :

The PRESENT cipher and the Advanced Encryption Standard (AES) are essential tools for protecting sensitive information and facilitating private transactions in the world of contemporary information technology. AES is well known for its adaptability and use in a variety of fields, whereas the PRESENT cipher performs well in situations involving light cryptography. This work performs a dual analysis, concentrating on the Key Scheduling Algorithms (KSAs) of the PRESENT cipher and the AES cipher, which are essential elements in charge of producing round keys in their corresponding encryption processes. Our goal is to shed light on the behaviors, strengths, and potential vulnerabilities of these KSAs by simplifying their study and using deep learning techniques, namely with a Neural Network model. Through the application of a deep learning neural network trained on extensive datasets, our study discerns patterns and weaknesses within both ciphers, providing insights essential for identifying potential avenues of exploitability by malicious entities, thereby emphasizing a proactive defense strategy against evolving threats.

This study advocates for a proactive strategy to secure cryptographic systems by proposing security modifications for both the PRESENT cipher and AES, going beyond vulnerability evaluation. A new viewpoint is offered by the use of machine learning, more especially deep learning, to find patterns and cryptographic keys in systems that use both the PRESENT cipher and AES. This thorough framework provides a proactive defense against possible vulnerabilities in a variety of security scenarios, in addition to furthering our understanding of various cryptographic methods.

Keywords: Advance encryption standards(AES), PRESENT Cipher , Deep learning , Neural Network, Key scheduling algorithm(KSA), Data security

Introduction

The story of the PRESENT cipher and Advanced Encryption Standards (AES) begins with the need to improve data security, which is consistent with the age-old proverb that "necessity is the mother of invention." The growing information flow over computer networks at the end of the 20th century made data privacy protection even more crucial. The National Institute of Standards and Technology (NIST) of the United States launched a competition in 1997 because the Data Encryption Standard (DES), which has been in widespread use since the 1970s, was becoming outdated due to contemporary cyber threats. Mathematicians and cryptographers were encouraged to contribute encryption methods that demonstrated exceptional security, efficiency, and versatility as part of this global challenge.

The Rijndael encryption algorithm, developed by Belgian cryptographers Vincent Rijmen and Joan Daemen, was the winner of a rigorous selection process and was formally adopted as the AES standard by NIST in 2001. Using a secret key, AES functions as a specific lock in computer networks and depends on the Key Scheduling Algorithm (KSA). This algorithm functions as a special formula, converting the secret key into a convoluted string of characters that forms the basis for protecting digital information. Parallel to this, the PRESENT cipher—which prioritizes performance over security in lightweight cryptography scenarios—was created in response to the shortcomings of the current encryption standards. Both AES and PRESENT employ secret keys and unique Key Scheduling Algorithms, pivotal components in their encryption processes.

This research study takes a step forward by using Deep Learning approaches to examine the security strengths and possible weaknesses of the AES and PRESENT Key Scheduling Algorithms. Similar to an extremely intelligent computer brain, deep learning can reveal complex patterns and hidden information in data that would be missed by traditional analytical techniques. The study intends to contribute to a better understanding of the cryptographic robustness of both AES and PRESENT's key scheduling algorithms by offering a thorough analysis of their security strengths, weaknesses, and potential patterns-based vulnerabilities through the application of deep learning techniques.

AES Cipher

Digital information transferred over the internet and in digital assets can be secured using the AES (Advanced Encryption Standard) Cypher. It protects our data whether it is being transmitted over the internet or is being stored on digital devices, much like a digital lock would. This cypher method treats our regular message as plain text, which AES translates into cipher text, which is a hidden code. Only the intended recipient must know the unique key needed to unlock this communication. The main reason AES is crucial for internet security is that it makes it extremely difficult or impossible for unauthorized parties to decrypt this shared private data. Any kind of information we may have, including passwords and messages, will only be accessed by those who are authorized. AES provides us very strong mechanism for securing and encrypting our data.

AES Keys and Rounds

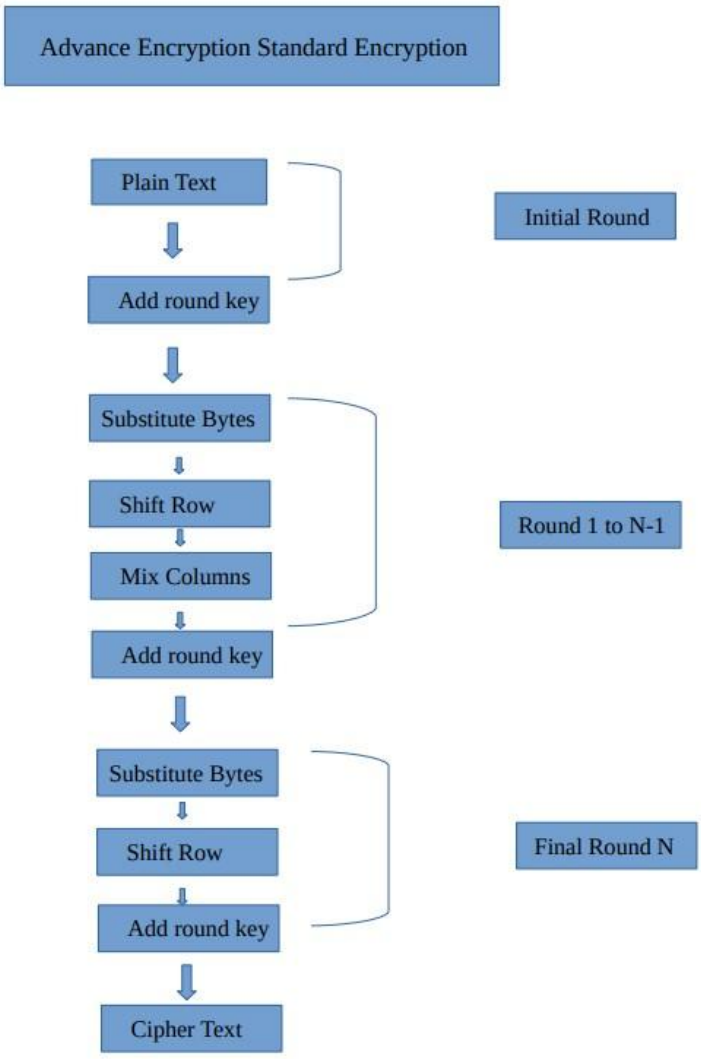
AES Encryption/ Decryption rounds depends on what type of key size we are using for encryption and decryption process . In generally we can say that If the key size is larger of encryption process then It is much more secure way compare to less size of key encryption process .

Based upon rounds and keys there are three type of AES Encryption and Decryption which are being used based upon selection and our need -

(i) 128 Bit key – 10 Rounds : It uses 128 bit block size of keys and total 10 rounds for encryption and decryption process .
(ii) 192 Bit key – 12 Rounds : It uses 192 bit block size of keys and total 12 rounds for encryption and decryption process .
(iii) 256 Bit key – 14 Rounds : It uses 256 bit block size of keys and total 14 rounds for encryption and decryption process.

Encryption Process of AES

There are multiple number of steps which AES use for Encrypting the our given input data. Following is Representation of AES Encryption Process. In the diagram N can be any value from the 10,12,14 which represents that which type of AES Encryption we are doing -



[Figure – AES Encryption Process]

For understanding in better way lets see each step one by one-

(i) Initial Round : In the initial round, AES takes the input data (plain text) and combines it with the first round key. This is the first step to create the cipher text. AES apply a substitution-permutation network (SPN) structure for encryption.
(ii) Sub-Bytes : This step involves replacing each byte of data with a corresponding byte from the AES S-box, which is a predefined substitution table and it is constant for any AES Encryption .
(iii) Shift Rows : This step rearranges the bytes within each row of the data matrix.

(iv) Mix Columns : In this operation, the columns of the data matrix are mixed using mathematical transformations.

(v) Add Round Key : The round key for the current round is XORed with the data.

(vi) Final Round : The final round is similar to the other rounds but leave the Mix Columns step. It includes Sub Bytes, Shift Rows, and Add Round Key operations. After all the rounds are completed, the plain text data is transformed into cipher text, which is the encrypted form of the plain text. This transferred Cipher text is decrypted on other side using same reverse process and finally recipient get the plain text in secure way.

Key Scheduling Algorithm of AES

The heart of AES is its Key scheduling algorithm which makes it much more secure than other ciphers. Following is process for generating AES-128 Bits round keys :

1. Key Expansion:

   a. Begin with the original 128-bit key.

   b. Initialize an array to store the round keys.

   c.The first round key is the original 128-bit key.

2. Round Constants:

   Initialize a list of round constants (Rcon). There are a total of 10 round constants for AES-128, which are used to generate the round keys.

3. Key Expansion Loop:

The key expansion loop consists of multiple iterations (10 in the case of AES-128).

4. Word Transformation:

For each iteration, take the last 4 bytes (32 bits) of the current round key. And

Perform a series of transformations

a. Rotate: Circularly shift the bytes in the word to the left by one byte.

b. Sub Bytes: Apply the S-box substitution to each byte.

c. XOR with R-con: XOR the first byte of the word with the corresponding round constant from R-con.

d. XOR the first byte of the word with the first byte of previous round key.

e. XOR the remaining three bytes with the corresponding bytes from the previous round key.

5. Expand Key:

The newly generated 32-bit word is XORed with the 32-bit word located 4 bytes earlier in the round key. This result is the next 32-bit word for the round key.

6. Store Round Key:

Add the newly generated 128-bit round key to the Round Keys array.

7. Repeat:

Repeat this process for a total of 10 iterations to generate 10 round keys for use in the encryption rounds.

This is how AES-128 Bits keys being generated

PRESENT Cipher

The present cipher is a cryptographic algorithm used to secure information by transforming it into a coded format. Unlike traditional methods, it operates on smaller units of data, typically one bit at a time. This unique approach enhances its efficiency and security. The present cipher is celebrated for its resistance to various attacks, ensuring robust protection for sensitive data. Its simplicity and effectiveness make it a popular choice in modern cryptography, offering a reliable means of safeguarding information in digital communication. As technology evolves, the present cipher stands as a key player in the ongoing quest for secure and efficient data encryption.

PRESENT Keys and Rounds

The present cipher employs a key schedule to generate a series of subkeys which are crucial for encoding our information. These subkeys, along with the chosen key size, contribute to the algorithm's security.The key size, typically set at either 80 or 128 bits, which determines the complexity of potential key combinations and it uses for enhancing resistance against brute-force attacks. With a standard of 31 rounds, where each involving key mixing and permutation operations, the present cipher achieves a delicate balance between security and computational efficiency.

Key Scheduling Algorithm of PRESENT

The Key Scheduling algorithm in PRESENT involves a series of precise operations to generate round keys. Following are general steps to generate 80 bit round keys for each round -

Initialization:

- The user inputs an 80-bit key into the key register (K).
- Let K=K79-K78-K77 …K1-K0.

· **Round Key Generation:**

- For each round i, the left-most 64-bits of the current key register are taken as the round key kr for that round.
- kr=k63k62…k0=K79K78…K16.

· **Key Rotation:**

- After every round i, the key register is rotated by 61-bit positions to the left.
- Let K′K′ be the rotated key register.

· **Key Update:**

- The rotated key register (K′K′) is updated by passing the leftmost 4 bits through the S-Box.
- The round-counter value i is XORed with bits K19-K18-K17-K16-K15 with the least significant bit of the round-counter on the right.
- The result of the XOR operation is then used to update the key register.

By this process every round key register get updated and it uses for next round.

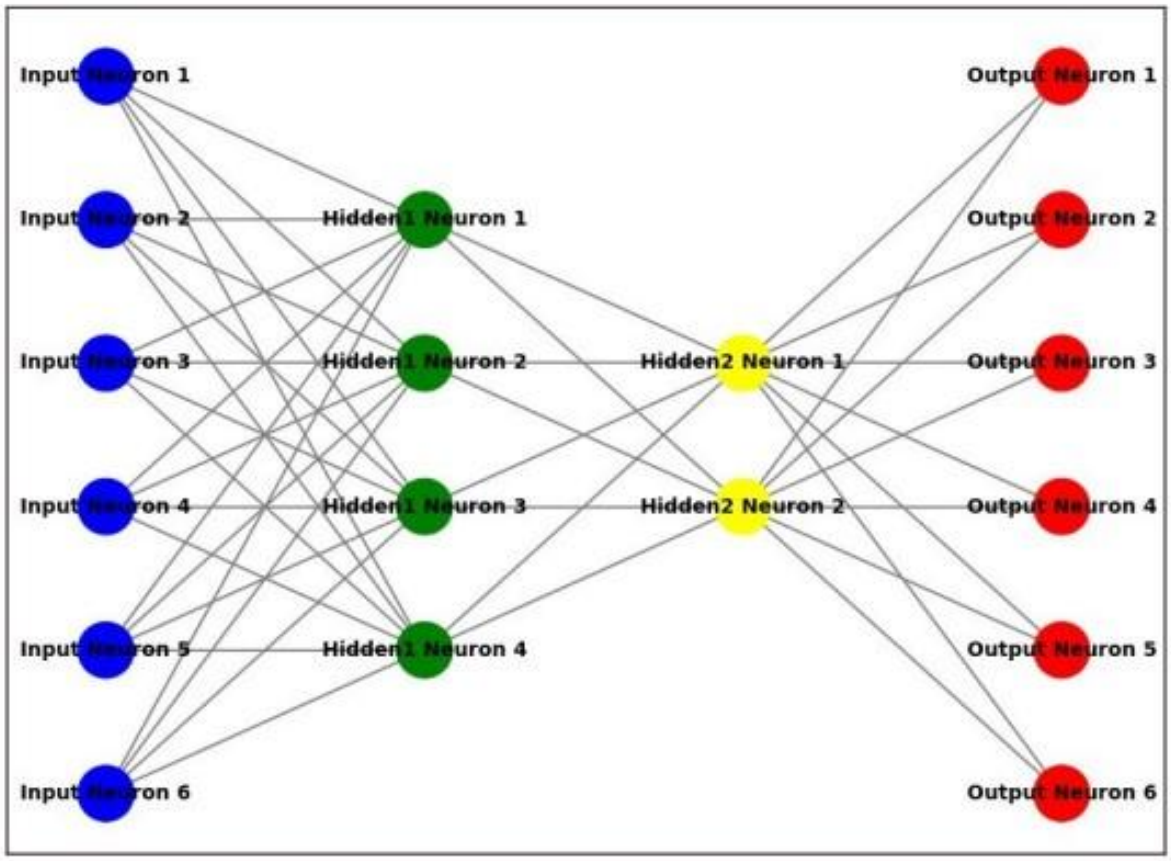
Deep Learning

In this modern technology world we can not ignore the power of AI and Deep Learning. Deep learning is like teaching computers to learn on their own, just like we teach kids. It is something like when we see any object again and again then our mind become able to identify similar type of object in very short time. In deep learning, we give lots of data in array and matrix form to model , like cat and dog pictures, and our deep learning models then tries to find patterns by itself. It uses artificial neural networks, which are like virtual brains. These networks have layers of tiny decision , makers that work together to understand the data. Once our model trained using various data and techniques then it can identify similar pattern and can generate results. This helps in many things, like recognizing our voice or faces in photos. Deep learning is an important part of artificial intelligence which makes our technology smarter and it also becoming more helpful in our daily life.

Neural Network

As the name suggest neural network it is a computer system which is designed to behave like the human brain. It is made of many interconnected nodes, like how our human neurons are connected in our brain, and it can process information and learn from this information. These network nodes work together to analyse data, recognize patterns, and make decisions. For example neural networks can be used to understand our spoken language, it can be also used to identify different objects in images, or predict trends based on input data. Neural network used in various tasks that involve analysis of complex data and they also have become important in machine learning and artificial intelligence. Just like our brain neural network adapts and learns from the given data and perform task which can done by human in the given situation . Neural networks can also be trained to improve their performance which makes them powerful tools for various applications. Following is structure of neural network which have one input layer with 6 neurons and two

hidden layer with 4 and 2 neurons respectively and one output layer with 6 neurons. We can increase or decrease layers and neurons based upon need.



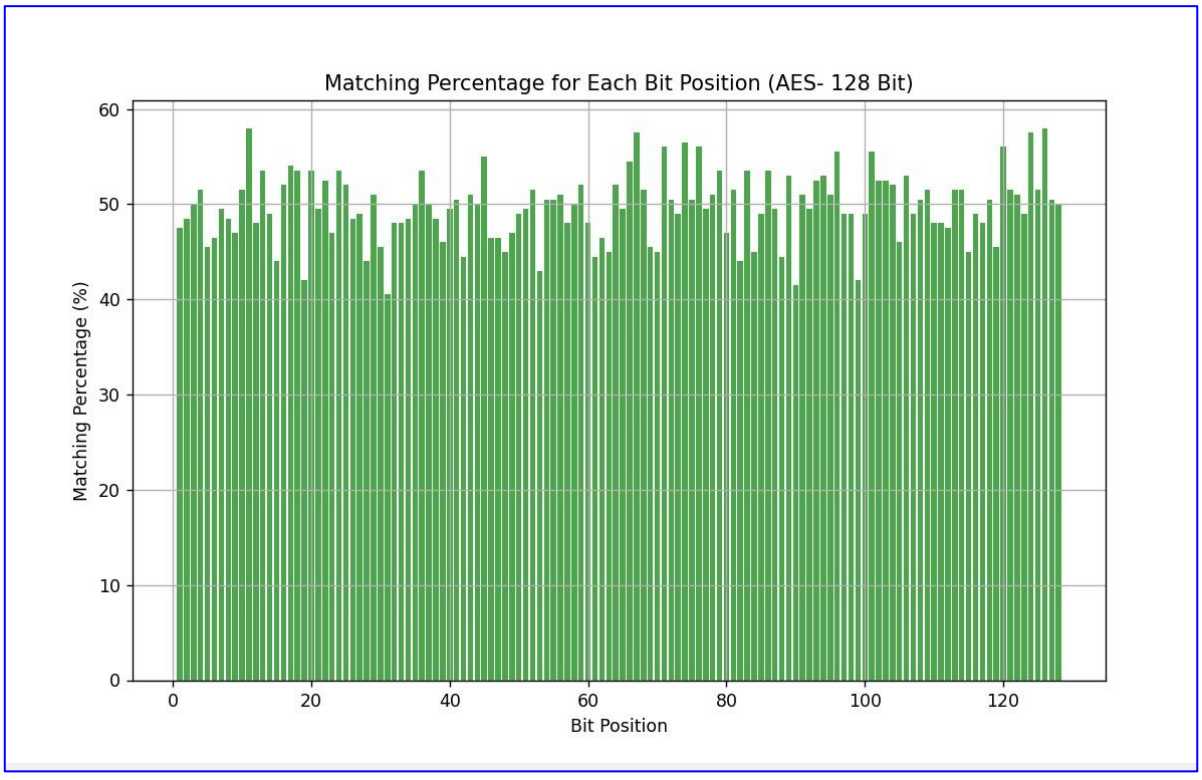
Analysis of AES & PRESENT Cipher KSA

In this research paper, we focus on enhancing the security analysis of the Advanced Encryption Standard (AES) and PRESENT cipher by using the power of Deep Learning, specifically Sequential Neural Networks (SNN). AES and PRESENT is mostly considered as most secure encryption algorithms, but they may not be immune to any deep learning based attacks. One area of vulnerability may lie in the key scheduling process, which generates round keys from the initial secret key. We propose using SNN to analyse and potentially strengthen this crucial component . We trained our model using 10,000 samples of AES and PRESENT Initial Round key and last Round key and after then we tested our Deep Learning trained model using 200 Initial Round Keys which tries to predict corresponding final round keys from Initial Keys. Following are Steps our Deep Learning Implementation:

1. Input Layer contains 128 neurons for AES and 80 neurons for PRESENT Cipher
2. There are 5 hidden layers of 64, 32 , 16 , 8, 4 bits of neurons.
3. Output layers activated using sigmoid function and it is of 128 bit neurons for 128 bit final Round key of AES Cipher and 80 bit neurons for 80 bit final round key of PRESENT Cipher.

By using above steps we tested our model and gain insights about the AES KSA and PRESENT KSA security and Vulnerabilities.

Following is Bar chart representation of the gained output which shows bit position to X axis and Matching percentage to Y Axis.



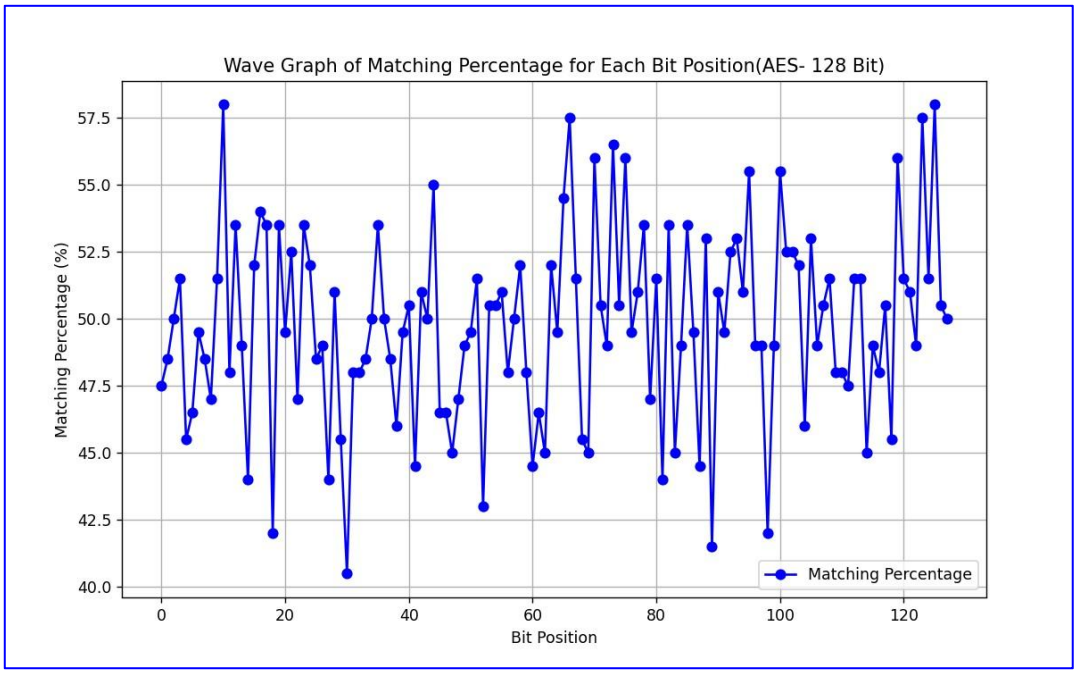
Results Table of AES

Deep Learning based Analysis of Key Scheduling Algorithm : AES Cipher							
Bit Position (1-32)	Matching % (1-32)	Bit Position (33-64)	Matching % (33-64)	Bit Position (65-96)	Matching % (65-96)	Bit Position (97-128)	Matching % (97-128)
1	47.5	33	48	65	49.5	97	49
2	48.5	34	48.5	66	54.5	98	49
3	50	35	50	67	57.5	99	42
4	51.5	36	53.5	68	51.5	100	49
5	45.5	37	50	69	45.5	101	55.5
6	46.5	38	48.5	70	45	102	52.5
7	49.5	39	46	71	56	103	52.5
8	48.5	40	49.5	72	50.5	104	52
9	47	41	50.5	73	49	105	46
10	51.5	42	44.5	74	56.5	106	53
11	58	43	51	75	50.5	107	49
12	48	44	50	76	56	108	50.5
13	53.5	45	55	77	49.5	109	51.5
14	49	46	46.5	78	51	110	48
15	44	47	46.5	79	53.5	111	48
16	52	48	45	80	47	112	47.5
17	54	49	47	81	51.5	113	51.5
18	53.5	50	49	82	44	114	51.5
19	42	51	49.5	83	53.5	115	45
20	53.5	52	51.5	84	45	116	49
21	49.5	53	43	85	49	117	48
22	52.5	54	50.5	86	53.5	118	50.5
23	47	55	50.5	87	49.5	119	45.5
24	53.5	56	51	88	44.5	120	56
25	52	57	48	89	53	121	51.5
26	48.5	58	50	90	41.5	122	51
27	49	59	52	91	51	123	49
28	44	60	48	92	49.5	124	57.5
29	51	61	44.5	93	52.5	125	51.5
30	45.5	62	46.5	94	53	126	58
31	40.5	63	45	95	51	127	50.5
32	48	64	52	96	55.5	128	50

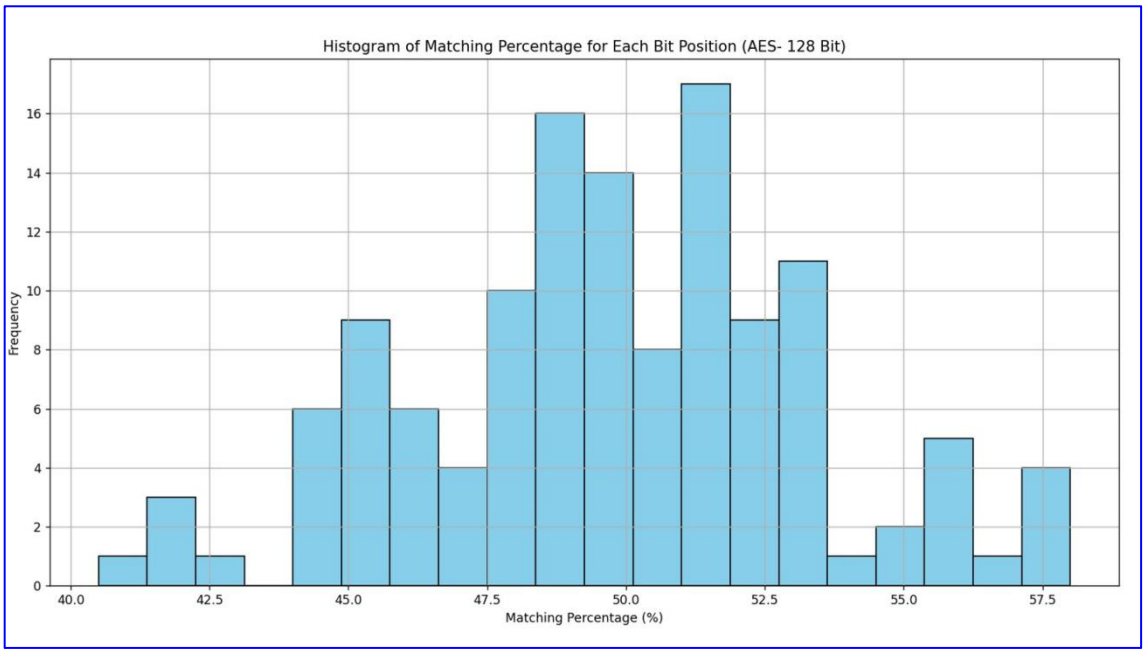
The above accuracy table shows about how much percentage of bits we are able to predict correctly using our trained model. In the result table we can see some bits have more than 50 percent accuracy and some have less than 50 precent , In the overall result we can say it have approximate 50 percent accuracy which is 1/2 accurate predictability of the 10th round key in the form of 128 bits 0s and 1s. In the other words 1 and 0 bit also have 1/2 predictability means in any place there will be either 0 or either 1 . So by analyzing our results we can say the AES Key

scheduling algorithm is secure and effective in order to face deep learning cryptanalysis by applying our used method. In further we can also used some more advance technique to test AES KSA Vulnerabilities.

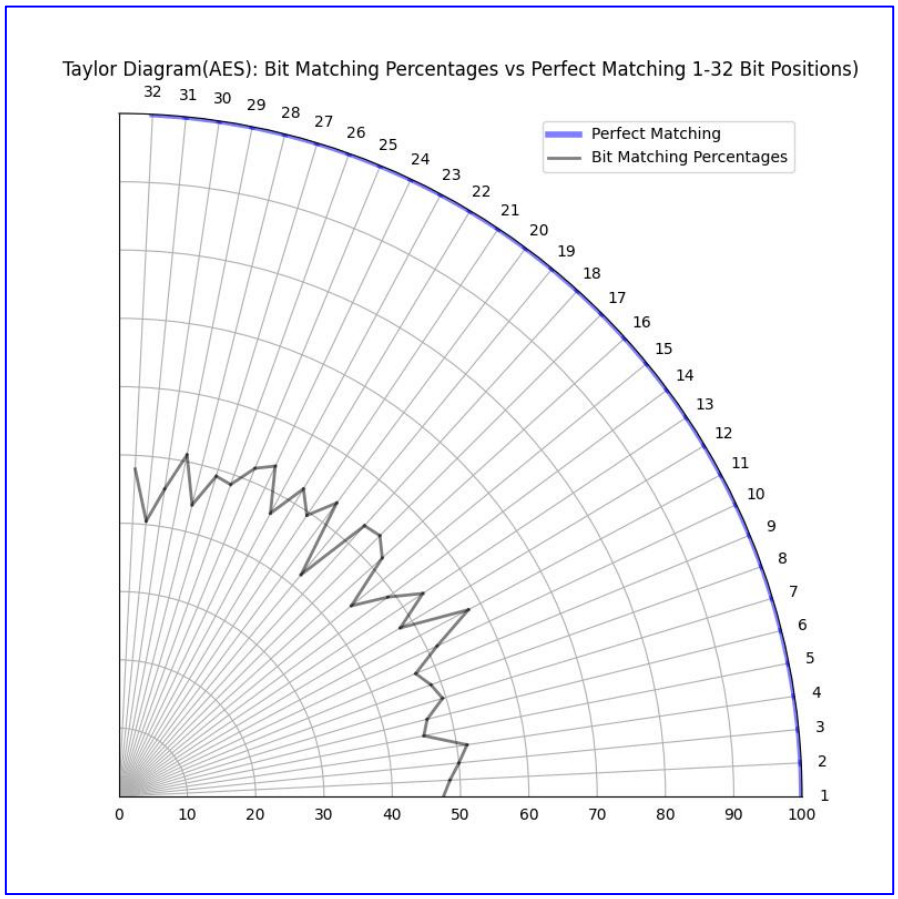
Following is Wave graph visualization of our gained output of AES Cipher.

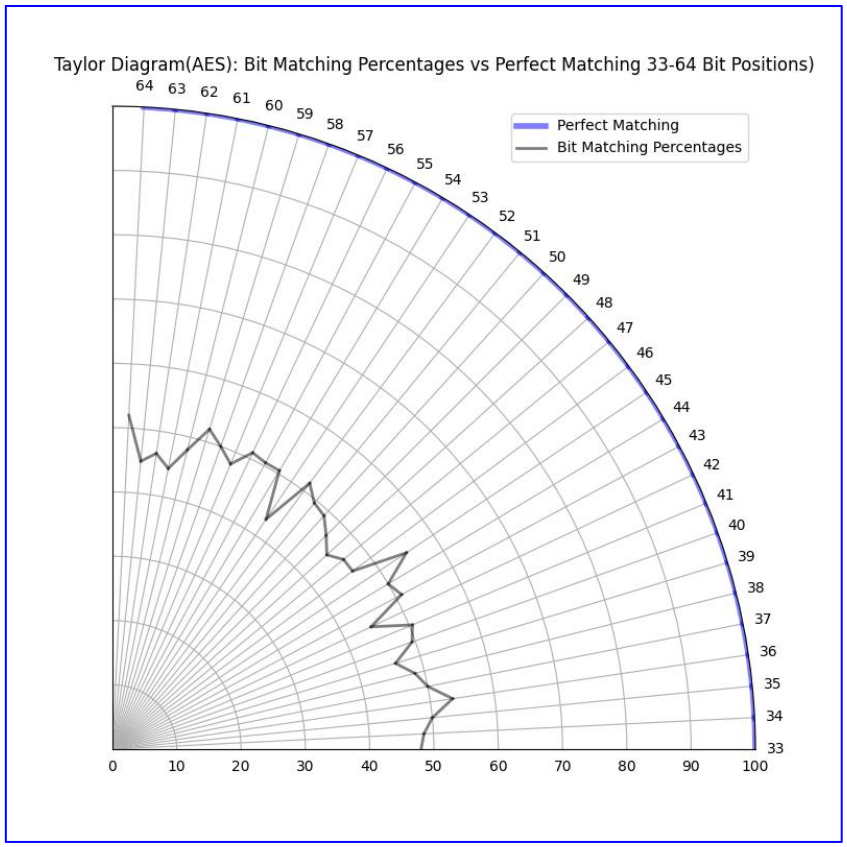


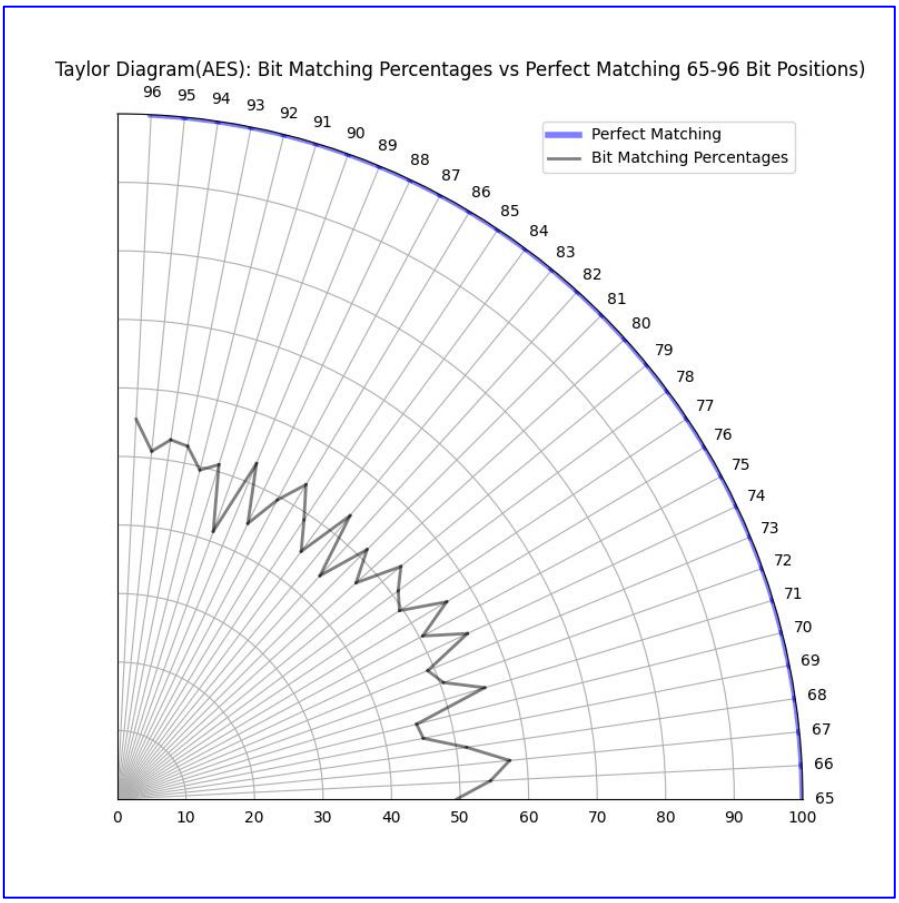
Following is histogram of our gained output of AES cipher where on X axis it showing Matching percentage and on Y axis it is showing the corresponding frequency.

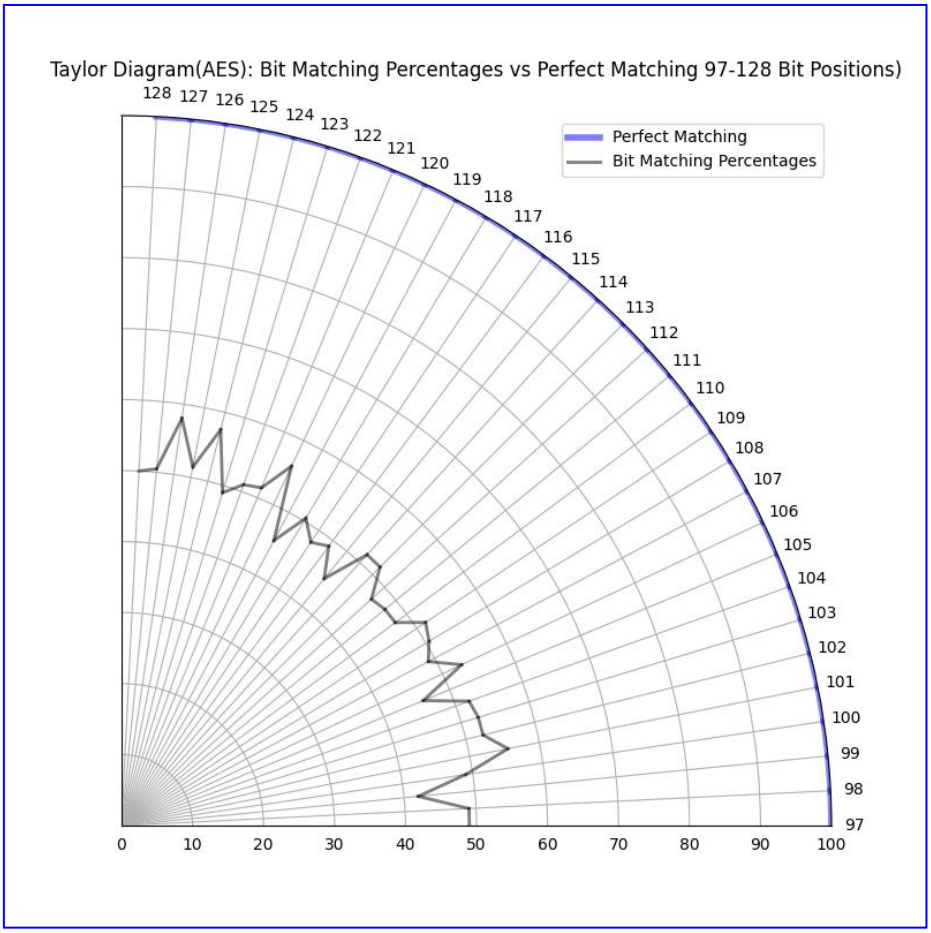


Following figures shows Taylor diagram of our gained output of 128 bit AES Cipher. There are four diagram where each shows for 32 bit part of 128 bit.






Taylor Diagram(AES): Bit Matching Percentages vs Perfect Matching 65-96 Bit Positions)
Perfect Matching
Bit Matching Percentages
96 95 94 93 92 91 90 89 88 87 86 85 84 83 82 81 80 79 78 77 76 75 74 73 72 71 70 69 68 67 66 65
0 10 20 30 40 50 60 70 80 90 100

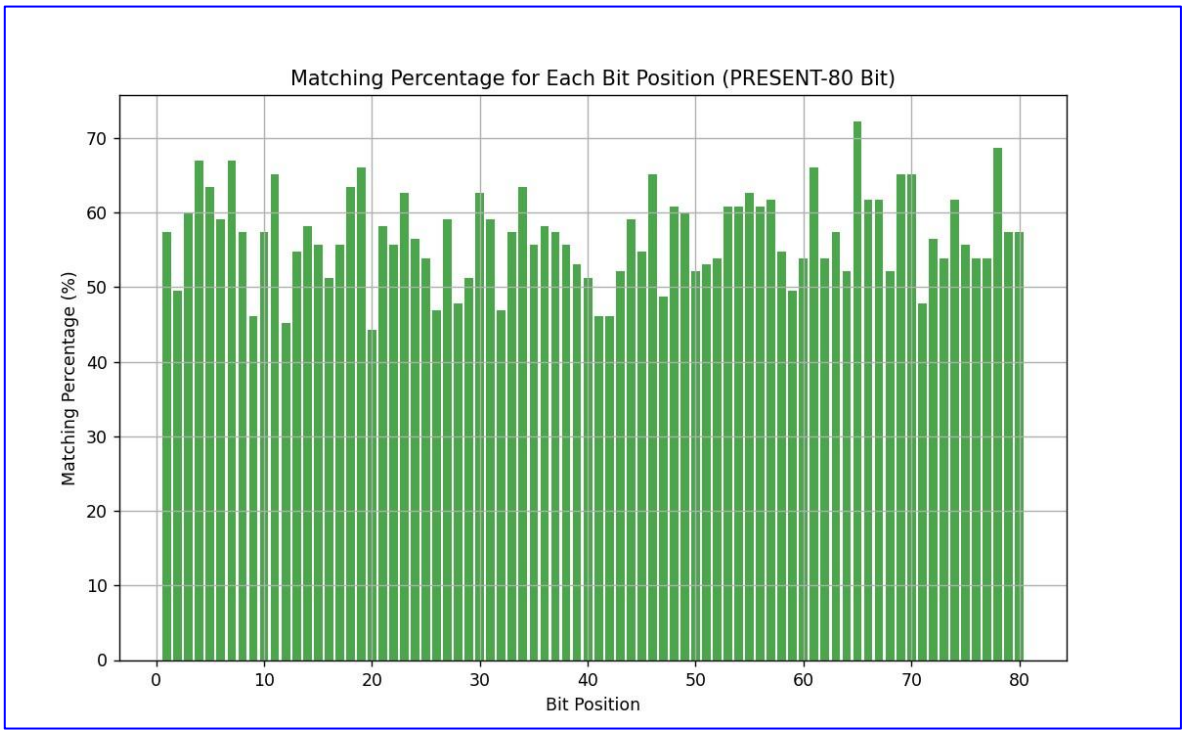

Taylor Diagram(AES): Bit Matching Percentages vs Perfect Matching 97-128 Bit Positions)
Perfect Matching
Bit Matching Percentages
128 127 126 125 124 123 122 121 120 119 118 117 116 115 114 113 112 111 110 109 108 107 106 105 104 103 102 101 100 99 98 97
0 10 20 30 40 50 60 70 80 90 100

Results Table of PRESENT

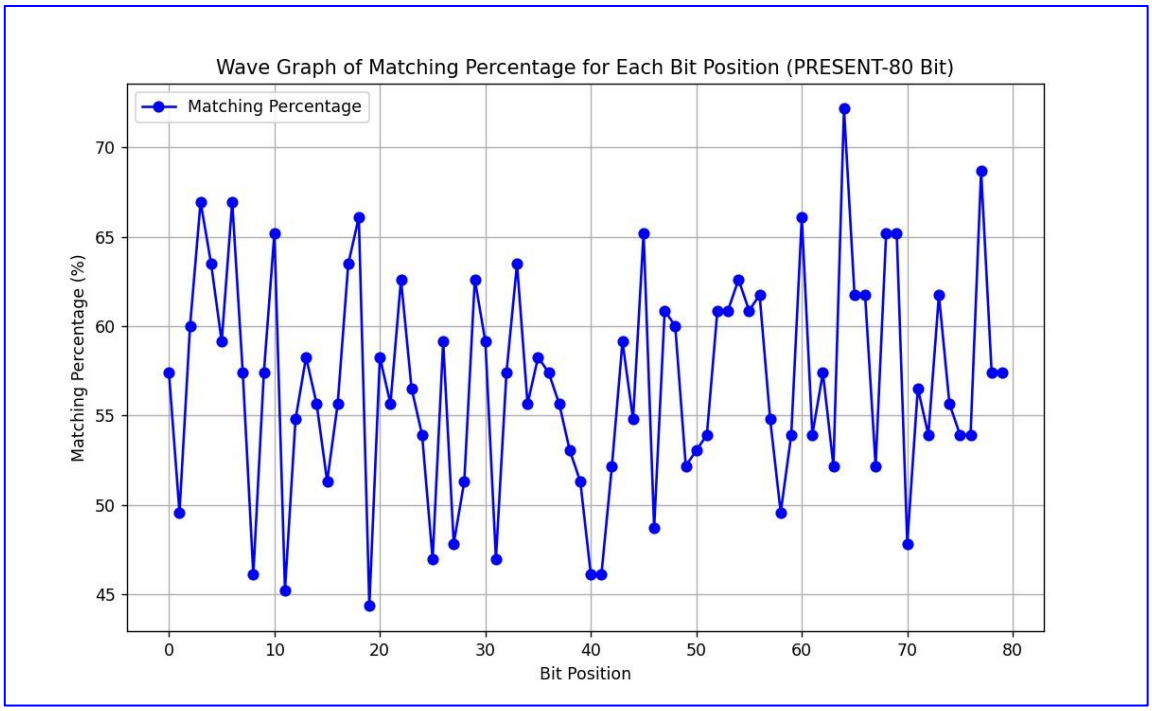
Deep Learning Based Analysis of Key Scheduling Algorithm : PRESENT Cipher							
Bit Position (1-20)	Matching % (1-20)	Bit Position (21-40)	Matching % (21-40)	Bit Position (41-60)	Matching % (41-60)	Bit Position (61-80)	Matching % (61-80)
1	57.39	21	58.26	41	46.09	61	66.09
2	49.57	22	55.65	42	46.09	62	53.91
3	60	23	62.61	43	52.17	63	57.39
4	66.96	24	56.52	44	59.13	64	52.17
5	63.48	25	53.91	45	54.78	65	72.17
6	59.13	26	46.96	46	65.22	66	61.74
7	66.96	27	59.13	47	48.7	67	61.74
8	57.39	28	47.83	48	60.87	68	52.17
9	46.09	29	51.3	49	60	69	65.22
10	57.39	30	62.61	50	52.17	70	65.22
11	65.22	31	59.13	51	53.04	71	47.83
12	45.22	32	46.96	52	53.91	72	56.52
13	54.78	33	57.39	53	60.87	73	53.91
14	58.26	34	63.48	54	60.87	74	61.74
15	55.65	35	55.65	55	62.61	75	55.65
16	51.3	36	58.26	56	60.87	76	53.91
17	55.65	37	57.39	57	61.74	77	53.91
18	63.48	38	55.65	58	54.78	78	68.7
19	66.09	39	53.04	59	49.57	79	57.39
20	44.35	40	51.3	60	53.91	80	57.39

By applying same method as AES The above accuracy table shows about how much percentage of bits we are able to predict correctly using our trained model of PRESENT Cipher. We can see it is approximate 50% Accuracy and 1s and 0s predictability is also 50% separately .We can say that by using our model and method PRESENT Have resistance to face deep learning based Cryptography Analysis attack.

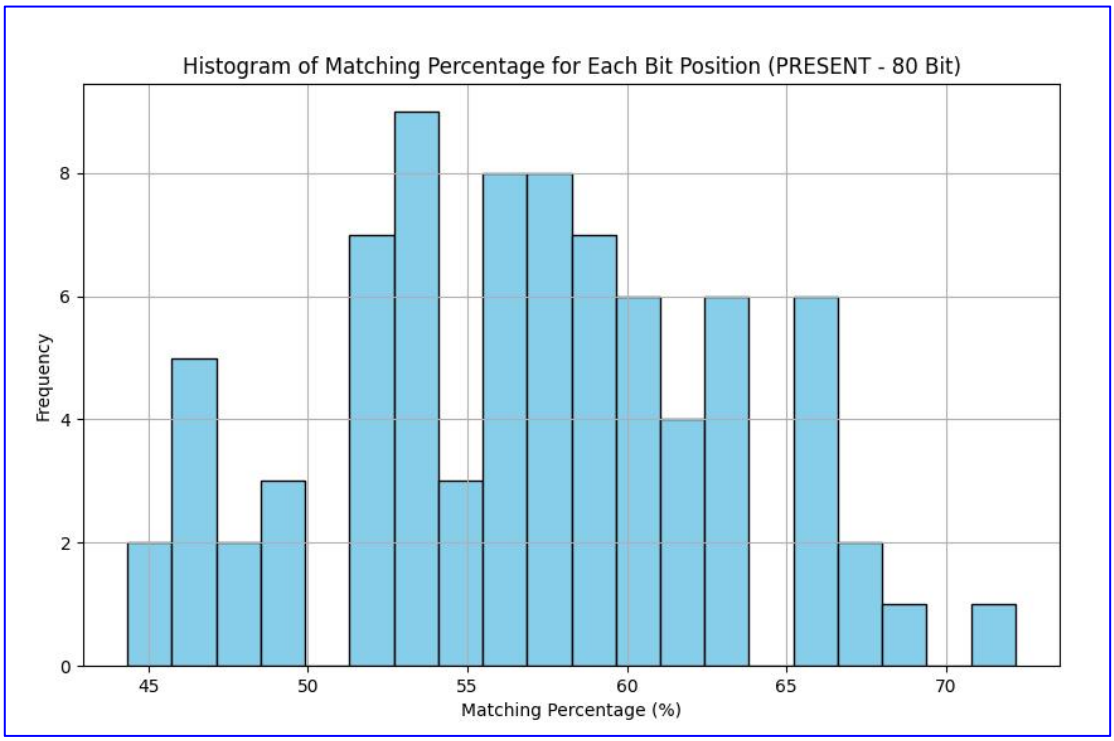
Following are bar chart and wave graph of our Analyzed results of PRESENT Cipher Key Scheduling Algorithm,



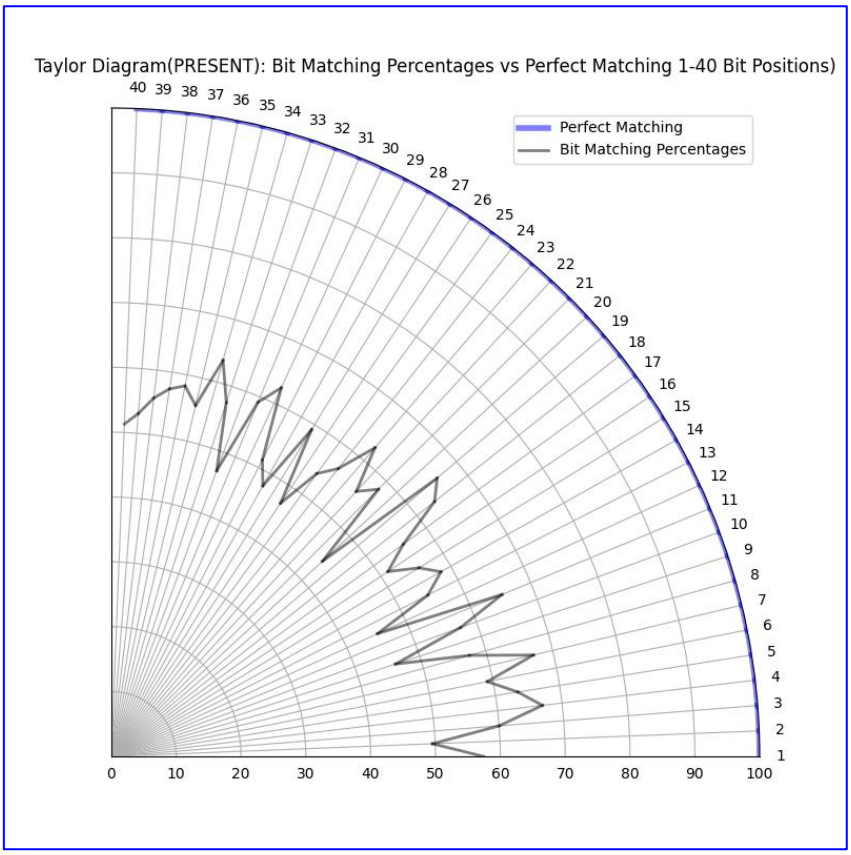
Following is wave graph of our gained result of PRESENT Cipher where we can see the variance of our matching percentage.

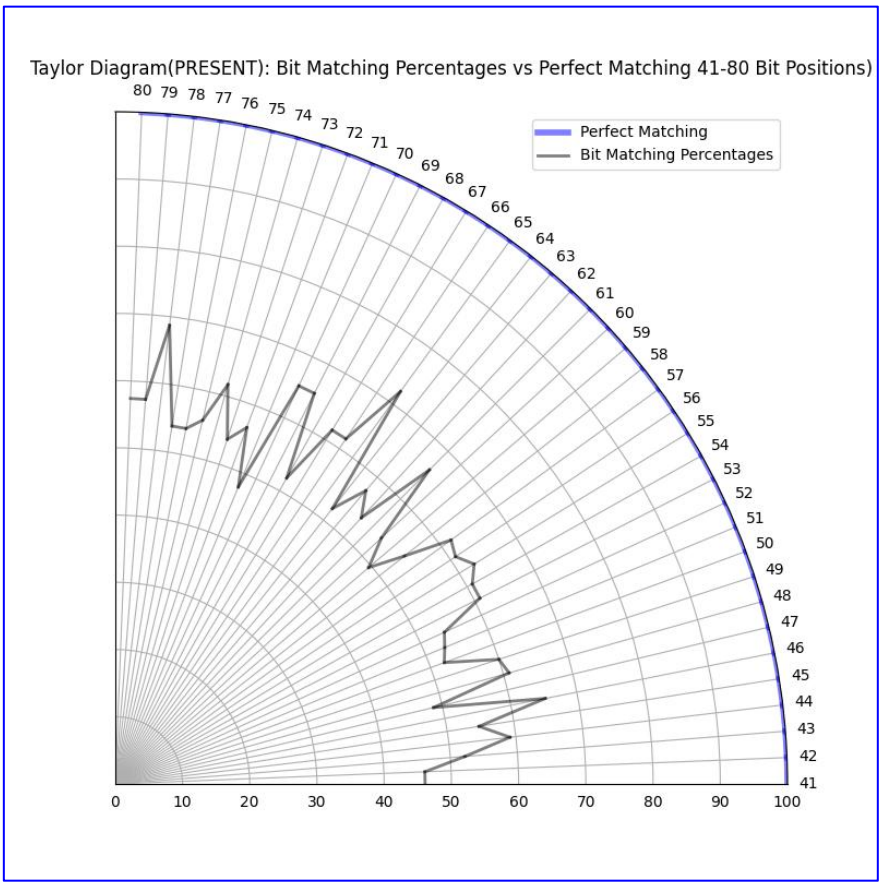


Following is Histogram Representation of our gained result of PRESNT Cipher with respect to Matching percentage and Frequency of matching percentage of bits.



Following are Taylor Diagram Representation of our gained results. In case of PRESENT 80 Bit here we have two Taylor diagram where each have 40 bit part of 80 bit PRESENT Cipher Analysis result.





Conclusion

In summary, we can say our paper focused on testing and ensuring that the Advanced Encryption Standard (AES) and PRESENT Cipher is secure by using a smart computer method which is also known as Deep Learning. We looked closely at a part of AES and PRESENT called the Key Scheduling Algorithm (KSA) and found that, despite being it strong, there are some areas where it could be safer. Using our Deep Learning model, we tried to find patterns and weaknesses in these KSA. The results, shown in the accuracy table, describes that we could predict about half of the final round key bits accurately. This means that the these KSA is quite good at resisting certain types of Deep learning analysis. Essentially, our study emphasizes that PRESENT and AES is secure and it provides ideas on how to make our cryptographic systems stronger against any AI based possible threats. It is also contributing to the ongoing improvement of data security and privacy issues.

References

[1] Chong, Bang & Salam, Iftekhar. (2021). Investigating Deep Learning Approaches on the Security Analysis of Cryptographic Algorithms. Cryptography. 5. 30.10.3390/cryptography5040030.

[2] Gupta, R.K., Lamkuche, H.S., Prasad, S. (2024). Enhancing the Security of Sensitive Data in Cloud Using Enhanced Cryptographic Scheme. In: Musleh Al-Sartawi, A.M.A., Al-Qudah, A.A., Shihadeh, F. (eds) Artificial Intelligence-Augmented Digital Twins. Studies in Systems, Decision and Control, vol 503. Springer, Cham. https://doi.org/10.1007/978-3-031-43490-7_29.

[3] D. Agarwal, S. Gurele and H. S. Lamkuche, "SAILFISH-I: A Lightweight Block Cipher for Cloud-Enabled Fog Devices," 2022 IEEE 6th Conference on Information and Communication Technology (CICT), Gwalior, India, 2022, pp. 1-6, doi: 10.1109/CICT56698.2022.9997844.

[4] Lamkuche, H. S., Pramod, D., Onker, V., Katiya, S., Lamkuche, G., & Hiremath, G. R. (2019). SAL–a lightweight symmetric cipher for Internet of Things. International Journal of Innovative Technology and Exploring Engineering, 8(11), 521-528

[5] Pareek, M., Mishra, G., & Kohli, V. (2020). Deep Learning based analysis of Key Scheduling Algorithm of PRESENT cipher. Cryptology ePrint Archive, Paper 2020/981.

]6] Sandhya Sarma, K.N., Chandra Blessie, E., Lamkuche, H.S. (2024). A Lightweight Cipher for Balancing Security Trade-Off in Smart Healthcare Application. In: Hassanien, A.E., Castillo, O., Anand, S., Jaiswal, A. (eds) International Conference on Innovative Computing and Communications. ICICC 2023. Lecture Notes in Networks and Systems, vol 731. Springer, Singapore. https://doi.org/10.1007/978-981-99-4071-4_42

[7] Bogdanov, A. *et al.* (2007). PRESENT: An Ultra-Lightweight Block Cipher. In: Paillier, P., Verbauwhede, I. (eds) Cryptographic Hardware and Embedded Systems - CHES 2007. CHES 2007. Lecture Notes in Computer Science, vol 4727. Springer, Berlin, Heidelberg. https://doi.org/10.1007/978-3-540-74735-2_31.

[8] So, Jaewoo. (2020). Deep Learning-Based Cryptanalysis of Lightweight Block Ciphers. Security and Communication Networks. 2020. 1-11. 10.1155/2020/3701067.

[9] Anees, Amir & Hussain, Iqtadar & Mujahid, Umar & Ahmed, Fawad & Shaukat,Sajjad. (2022). Machine Learning and Applied Cryptography. Security and Communication Networks. 2022. 1-3. 10.1155/2022/9797604.

[10] Abhishek Kumar Sinha , Jayaraj N, 2015, Performance Analysis of AES Cryptographic Algorithm, INTERNATIONAL JOURNAL OF ENGINEERING RESEARCH & TECHNOLOGY (IJERT) NCRTS – 2015 (Volume 3 – Issue 27)

[11] Abdullah, Ako. (2017). Advanced Encryption Standard (AES) Algorithm to Encrypt and Decrypt Data.

[12] Meraouche, Ishak & DUTTA, Sabyasachi & Tan, Haowen & Sakurai, Kouichi. (2021). Neural Networks Based Cryptography: A Survey. IEEE Access. PP. 1-1.10.1109/ACCESS.2021.3109635.ers published in translation journals, please give the English citation first, followed by the original foreign-language citation [6].

[13] Nitaj A, Rachidi T. Applications of Neural NetworkBased AI in Cryptography.Cryptography.2023;7(3):39.https://doi.org/10.3390/cryptography7030039

[14] Rivest, R.L. (1991). Cryptography and Machine Learning. International Conference on the Theory and Application of Cryptology and Information Security.

[15] Taye MM. Theoretical Understanding of Convolutional Neural Network: Concepts, Architectures, Applications, Future Directions. Computation. 2023; 11(3):52. https://doi.org/10.3390/computation11030052.

[16] O'Shea, Keiron & Nash, Ryan. (2015). An Introduction to Convolutional Neural Networks. ArXiv eprints.

[17] S. Albawi, T. A. Mohammed and S. Al-Zawi, "Understanding of a convolutional neural network," 2017 International Conference on Engineering and Technology (ICET), Antalya, Turkey, 2017, pp. 1-6, doi: 10.1109/ICEngTechnol.2017.8308186.

[18] Grossi, Enzo & Buscema, Massimo. (2008). Introduction to artificial neural networks. European journal of gastroenterology & hepatology. 19. 1046-54. 10.1097/MEG.0b013e3282f198a0.