# ML based Improved Differential Distinguisher with High Accuracy: Application to GIFT-128 and ASCON

Tarun Yadav<sup>[0000-0002-5462-8253]</sup> and Manoi Kumar<sup>[0000-0001-6900-4075]</sup>

Scientific Analysis Group, DRDO, Metcalfe House Complex, Delhi-110 054, INDIA {tarunyadav,manojkumar}.sag@gov.in

Abstract. In recent years, ML based differential distinguishers have been explored and compared with the classical methods. Complexity of a key recovery attack on block ciphers is calculated using the probability of a differential distinguisher provided by classical methods. Since theoretical computations suffice to calculate the data complexity in these cases, so there seems no restrictions on the practical availability of computational resources to attack a block cipher using classical methods. However, ML based differential cryptanalysis is based on the machine learning model that uses encrypted data to learn its features using available compute power. This poses a restriction on the accuracy of ML distinguisher for increased number of rounds and ciphers with large block size. Moreover, we can still construct the distinguisher but the accuracy becomes very low in such cases. In this paper, we present a new approach to construct the differential distinguisher with high accuracy using the existing ML based distinguisher of low accuracy. This approach outperforms all existing approaches with similar objective. We demonstrate our method to construct the high accuracy ML based distinguishers for GIFT-128 and ASCON permutation. For GIFT-128, accuracy of 7-round distinguisher is increased to 98.8% with  $2^9$  data complexity. For ASCON, accuracy of 4-round distinguisher is increased to 99.4% with  $2^{18}$  data complexity. We present the first ML based distinguisher for 8 rounds of GIFT-128 using the differential-ML distinguisher presented in Latincrypt-2021. This distinguisher is constructed with 99.8% accuracy and  $2^{18}$  data complexity.

**Keywords:** ASCON, Block Cipher, Differential Cryptanalysis, GIFT, Machine Learning

# 1 Introduction

Application of machine learning (ML) in cryptanalysis of symmetric ciphers is trending in recent years [11] [13] [22] [23]. Cryptanalysts are experimenting with the different types of machine learning architectures to construct ML based distinguishers for symmetric ciphers [3] [21]. High probable characteristics are required to mount an attack on a block cipher but searching such characteristics

with large block size is computationally intensive task. The automated techniques [18] reduces the effort of cryptanalysts and help to rule out or propose the existence of high probability differential characteristics in a block cipher. Since the first proposal of differential attack in 1993 [5], numerous techniques have been devised to solve the differential characteristic search problem. Solution to these problems are automated by modeling with MILP [15], SAT/SMT, constraint programming [17] to get a solution using an appropriate solver.

In a classical differential attack, we search for an input difference  $\Delta_i$  that leads to an output difference  $\Delta_o$  with a probability  $2^{-p}$  larger than  $2^{-n}$  for a block cipher with *n*-bit block size. If we can find multiple paths connecting these input and output differences then it is called a differential and its probability  $2^{-\sum p_j}$  is calculated by adding the probabilities of j individual paths. The multiple differential is a generalization of classical differential where we combine the differential characteristic with multiple input and multiple output differences. The multiple differentials work with lower complexity than any single differential characteristic of the differential.

The current trends in AI and ML has improvised its usage in cryptanalysis of block cipher [9]. The first application of ML in this direction was presented by Gohr in 2019 through an ML based differential distinguisher for SPECK32/64 [10]. The ML based differential distinguisher was searched using machine learning algorithm and key recovery mechanism was also proposed using ML by Gohr for the first time. The distinguisher is trained on the data with single input difference  $\Delta_i$  but it tends to learn the multiple differences in the outputs. The capability of learning the multiple differentials in the output provides an edge to the ML based differential distinguisher. This distinguisher achieved higher accuracy than the classical distinguisher and covered more rounds for SPECK. The labeled data was used by Gohr to train a deep neural network where half of the data was from a random source and half was taken from actual cipher. The trained ML model was used to predict the cipher with an accuracy. Baksi *et.al.* extended the Gohr's approach on Gimli using multi layer perceptron and other architectures available in deep learning networks [1].

Yadav *et.al.* proposed the first extension of ML with classical differential distinguisher at Latincrypt-2021 [24]. This distinguisher was called as differential-ML distinguisher which covered more rounds then the ML and classical alone. The high accuracy ML distinguisher was trained for *s*-rounds on the data with fixed difference. This distinguisher was used for prediction on (r+s)-rounds with an input difference after appending the *r*-round classical differential characteristic on the top. The complexity of differential-ML distinguisher was calculated experimentally based on the cutoff for number of high accuracy predictions by the authors. Differential-ML distinguisher was able to cover more rounds with high accuracy than classical and ML based distinguisher proposed by Gohr.

The accuracy of neural differential distinguisher becomes very low as the number of rounds are increased for ciphers with large block size. Shen *et.al.* [19] proposed paired ML models based approach and uses a low accuracy neural distinguisher to construct the high accuracy distinguisher through a score distri-

bution of predictions for multiple ciphertext differences. This approach is used to improve the prediction accuracy of 7-round GIFT-128 from 55.42% to 99.36% and 4-round ASCON form 50.69% to 69.25%. We have found some discrepancies in the implementation of GIFT-128 encryption given by authors at Github and therefore, the results claimed by the authors in [19] for GIFT-128 could not be validated.

**Our Contribution.** In this paper, we address a major challenge that have become a roadblock for ML based distinguishers. The challenge is to construct ML based distinguisher with high accuracy and lesser data complexity than the classical distinguisher. To overcome this challenges, we propose a new approach to construct the ML based differential distinguisher with high accuracy and its application to GIFT-128 and ASCON. A comparison of our results with the existing work is presented in Table 1. It can be inferred from the results that the approach presented in this paper constructs a better ML based differential distinguisher in terms of accuracy than existing approaches. We also present the first ML based distinguisher for 8 rounds of GIFT-128 with 99.8% accuracy and  $2^{18}$  data complexity.

**Organization.** This paper is divided into 5 sections. We provide a brief description of GIFT-128 and ASCON permutation in Section 2. We present a new approach to construct ML based differential distinguisher with high accuracy and its application to GIFT-128 and ASCON in Section 3. The construction of 8-round distinguisher for GIFT-128 with modified differential-ML approach and experimental results are discussed in section 4. We conclude the paper in Section 5.

Ciphor	Dounda	Classical Distinguishers	1	ML based Distingu	ishers	Sourco	
Cipiter	nounus	Data Complexity [4] [7]	Bits	Data Complexity	Accuracy	Source	
GIFT-128	7	$2^{29.415}$	128	2 <sup>9</sup>	99.36 <sup>1</sup>	[19]	
GIFT-128	7	$2^{29.415}$	128	$2^{9}$	98.8	This paper	
GIFT-128	8	$2^{41}$	128	$2^{18}$	99.8	This paper	
ASCON	4	$2^{107}$	320	$2^{12}$	69.25	[19]	
ASCON	4	$2^{107}$	320	2 <sup>18</sup>	99.4	This paper	

Table 1: Summary of Results

## 2 Block Ciphers: GIFT and ASCON

The block ciphers GIFT [4] and ASCON [6] were among the finalist of NIST lightweight cryptography competition concluded in 2023 [16] and ASCON remained a winner in this competition. The base permutation of these two ciphers are briefly discussed in this section. For more details on the design specifications and key scheduling of these ciphers, the papers [4] and [6] can be referred.

 $<sup>^{1}</sup>$  Results could not be validated due to discrepancies in GIFT-128 implementation

#### 2.1 Specifications of GIFT-128

GIFT is a family of two lightweight block ciphers GIFT-64 and GIFT-128 which was proposed by Banik *et.al.* in 2017. Two lightweight authenticated encryption schemes namely GIFT-COFB and SUNDAE-GIFT were submitted to NIST lightweight cryptography competition and both of these use GIFT-128 block cipher as their base permutation [2]. GIFT-128 is based on SPN structure that applies a round function 40 times iteratively to encrypt the 128-bit plaintext using a 128-bit secret key. In each round, it applies round keys and constant addition operation on selective bits, substitution using 4-bit S-box 32 times in parallel and bit-wise permutation operation on 128 bits. The 4-bit S-box and 128-bit permutation are given in the Table 2 and 3 respectively. Encryption function of GIFT-128 is described in Algorithm 1.

Algorithm 1: GIFT-128 Permutation **1 Input:**  $X_1 = (x_{127}, x_{126}, \dots, x_0)$  and  $RK_i; 1 \le i \le 41$ **2 Output:**  $C = X_{41} \oplus RK_{41} \oplus RC_{41}$ **3 for**  $i \leftarrow 1$  to  $4\theta$  **do**  $T_i = X_i \oplus RK_i \oplus RC_i$  $\mathbf{4}$  $\mathbf{5}$  $T_i = (t_{127}, t_{126}, \cdots, t_0)$ 6 for  $j \leftarrow 0$  to 31 do  $(u_{4*j+3}||u_{4*j+2}||u_{4*j+1}||u_{4*j}) = S[t_{4*j+3}||t_{4*j+2}||t_{4*j+1}||t_{4*j}]$ 7 end 8  $U_i = (u_{127}, u_{126}, \cdots, u_0)$ 9  $V_i = PN(U_i)$ 10  $X_{i+1} = V_i = (x_{127}, x_{126}, \cdots, x_0)$ 11 12 end

x	0	1	2	3	4	5	6	7	8	9	А	В	С	D	Е	F
S(x)	1	a	4	$\mathbf{c}$	6	f	3	9	<b>2</b>	d	b	7	5	0	8	е

Table 2: S-box for GIFT-128

i	0	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15
PN(i)	0	33	66	99	96	1	34	67	64	97	2	35	32	65	98	3
i	16	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31
PN(i)	4	37	70	103	100	5	38	71	68	101	6	39	36	69	102	$\overline{7}$
i	32	33	34	35	36	37	38	39	40	41	42	43	44	45	46	47
PN(i)	8	41	74	107	104	9	42	75	72	105	10	43	40	73	106	11
i	48	49	50	51	52	53	54	55	56	57	58	59	60	61	62	63
PN(i)	12	45	78	111	108	13	46	79	76	109	14	47	44	77	110	15
i	64	65	66	67	68	69	70	71	72	73	74	75	76	77	78	79
PN(i)	16	49	82	115	112	17	50	83	80	113	18	51	48	81	114	19
i	80	81	82	83	84	85	86	87	88	89	90	91	92	93	94	95
PN(i)	20	53	86	119	116	21	54	87	84	117	22	55	52	85	118	23
i	96	97	98	99	100	101	102	103	104	105	106	107	108	109	110	111
PN(i)	24	57	90	123	120	25	58	91	88	121	26	59	56	89	122	27
i	112	113	114	115	116	117	118	119	120	121	122	123	124	125	126	127
PN(i)	28	61	94	127	124	29	62	95	92	125	30	63	60	93	126	31

Table 3: Bit Permutation for GIFT-128

#### 2.2 Specifications of ASCON

ASCON was declared winner in the lightweight cryptography competition by NIST and it was also selected in the final portfolio of CAESAR competition. ASCON was designed by Dobrauing *et.al.* [6] with an input state size of 320 bits. The 320-bit input is divided into 5 64-bit words  $w_i$ . The round function is composed of adding 8-bit round constants (Table 4) to the  $w_2$ , application of 5-bit S-box (Table 5) in columns and diffusion layer on each 64-bit word. A total of 12 rounds are used in the ASCON permutation. ASCON permutation is described in Algorithm 2.

Algorithm 2: ASCON permutation

```
1 Input: X_0 = (x_{319}, x_{318}, \dots, x_0) = (w_4 ||w_3||w_2||w_1||w_0); RC_i
 2 Output: X_{12}
 3 for i \leftarrow 0 to 11 do
 \mathbf{4}
        w_2 = w_2 \oplus RC_i
         for j \leftarrow 0 to 63 do
 \mathbf{5}
              (w_4(j), w_3(j), w_2(j), w_1(j), w_0(j)) =
 6
               S[w_4(j)||w_3(j)||w_2(j)||w_1(j)||w_0(j)]
 7
         end
         w_4 = w_4 \oplus (w_4 \gg 19) \oplus (w_4 \gg 28))
 8
         w_3 = w_3 \oplus (w_3 \gg 61) \oplus (w_3 \gg 39))
 9
         w_2 = w_2 \oplus (w_2 \gg 1) \oplus (w_2 \gg 6))
\mathbf{10}
         w_1 = w_1 \oplus (w_1 \gg 10) \oplus (w_1 \gg 17))
11
         w_0 = w_0 \oplus (w_0 \ggg 7) \oplus (w_0 \ggg 41))
12
        X_{i+1} = (w_4 ||w_3||w_2||w_1||w_0)
13
14 end
```

i	0	1	2	3	4	5	6	7	8	9	10	11
$RC_i$	f0	e1	d2	c3	b4	a5	96	87	78	69	5a	4b

Table 4: Round Constants for ASCON

### 

Table 5: S-box for ASCON

# 3 Improved Differential Distinguishers: A New Approach

Finding relations between the input and output differences is the key idea behind differential cryptanalysis. These high probability relations are used as distinguishers which have better data complexity than exhaustive trials. Non-linear component of the cipher makes it difficult to find such relations with high probability. S-box is a non-linear component that is widely used to design the block ciphers. There are various approaches that are used to search the high probability differential distinguishers e.g. branch-and-bound based [14], constraint programming [17], and mixed integer linear programming [15]. In contrast to classical approaches, where such distinguishers are identified using difference propagation, machine learning based distinguishers learn these relations on the difference of encrypted data.

#### 3.1 Machine Learning based Differential Distinguishers

Gohr proposed an approach to model the distinguisher using real and random differences [10]. Real difference is the input/plaintext difference for which the distinguisher is designed. The approach is a two class problem where the machine tries to classify the given data in any one of the two classes. The main benefit of ML based approach is that the classification can be done for a single data point unlike the classical approach. In this approach, we generate half of the data with fixed input difference ( $\Delta_0$ ) and remaining data with random differences ( $\Delta_R$ ). The data is encrypted and the corresponding output differences are computed. Differences in the output that belong to input difference  $\Delta_0$  are part of the class 1 while remaining data belongs to class 0. Once this model is trained, it is used for prediction and classification on the output difference. If the probability of prediction is greater than 0.5, then it is classified as class 1 data. If probability of prediction is less than or equal to 0.5, then it is classified as class 0 data. This approach is widely used with various kinds of neural networks to construct the ML based differential distinguishers.

6

In this paper, multi layer perceptrons (MLP) are used to train the ML distinguisher. MLP consists of input layer, output layer and two hidden layers. Hidden layers contain same number of neurons as in the input layer with ReLu activation function. Output layer uses sigmoid function to predict the probabilities of belonging to class 1 or 0. We use  $2^{23}$  data for training the model and  $2^{20}$  for validation in each experiment.

The approach discussed in above paragraph works well but the benefits comes with a drawback that the classification is probabilistic and thus, the distinguisher predicts with an accuracy. This accuracy decreases drastically when the number-of-rounds or block-size is increased. After a threshold, the training data also becomes a constraint as it is limited by the computation power. Therefore, despite being a promising approach, a low accuracy distinguisher lacks the practical applicability in comparison to the classical distinguishers. To address this drawback, we present a new method that is used to increase the accuracy of ML based distinguisher of low accuracy. We use the existing ML based distinguisher as a subsystem of the new proposed distinguisher.

### 3.2 New Approach to Construct ML based Distinguishers with High Accuracy

Gohr's ML based differential distinguisher  $(D^{ML})$  works with an accuracy. This accuracy is comprised of two parts, true positive (TP) and true negative (TN). Both of these accuracy are important as predicting a correct class is necessary for a positive data point as well as for a negative data point. The accuracy of a model is the average of TP and TN accuracies. While distinguishing the data, it is much required that both accuracies should be high for correct predictions. If one of the accuracy is too high and other one is too low, then despite getting a good average accuracy, the distinguisher will not work as expected. Some examples indicating such cases are shown in Table 6.

Cipher	Rounds	<b>Total Accuracy</b>	<b>TP</b> Accuracy	TN Accuracy
GIFT-128	5	0.939	0.915	0.967
GIFT-128	6	0.731	0.608	0.848
GIFT-128	7	0.538	0.366	0.710

Table 6: TP and TN Accuracies of  $D^{ML}$  for GIFT-128

As shown in Table 6, the accuracy of 7-round GIFT-128 distinguisher is 0.538 where TP accuracy is 0.366 and TN accuracy is 0.710. It means that data belonging to real difference  $(\Delta_0)$  is classified with almost half of the accuracy than random differences  $(\Delta_R)$ . Therefore, model's accuracy may create a false perception that both the classes are predicted with the same accuracy in these cases. Such instance arises when a model's accuracy is low to distinguish the data correctly. To overcome this problem, we propose a new approach to construct the high accuracy distinguishers. This approach is motivated by our previous work on Differential-ML distinguisher [24]. It uses both TP and TN accuracies. The main aim is to increase the accuracy of a distinguisher in both the cases by increasing the data required for prediction. We define this new distinguisher as High Accuracy ML based Distinguisher  $(D^{HA-ML})$ . It uses  $D^{ML}$  as a subsystem

for prediction with other parameters viz. threshold probability (T), cutoff  $(C_T)$  and data complexity  $(\beta)$ . The approach to construct  $D^{HA-ML}$  is described in Algorithm 3.

An s-round  $D_{r+1\cdots r+s}^{HA-ML}$  distinguisher is developed in two phases (Algorithm 3). Construction phase uses s-round ML based distinguisher  $D_{r+1\cdots r+s}^{ML}$  and threshold probability T as inputs. It starts with two pairs ( $\delta = 1$ ) and generate  $2^{\delta}$  plaintext pairs  $(P_{\Delta_0}, P'_{\Delta_0})$  for fixed/real difference ( $\Delta_0$ ) and  $2^{\delta}$  plaintext pairs  $(P_{\Delta_R}, P'_{\Delta_R})$  for random difference ( $\Delta_R$ ). The plaintext data is encrypted with s-round cipher (CIPHER<sub>s</sub>) to get the corresponding ciphertext pairs  $(C_{\Delta_0}, C'_{\Delta_0})$  and  $(C_{\Delta_R}, C'_{\Delta_R})$ . We use the s-round distinguisher  $D_{r+1\cdots r+s}^{ML}$  to make predictions on the difference of encrypted data and get prediction probabilities  $p_{\Delta_0}$  and  $p_{\Delta_R}$ . Now, we count the number of elements in  $p_{\Delta_0}$  and  $p_{\Delta_R}$  above threshold probability T to get  $TP_{\Delta_0}$  and  $TP_{\Delta_R}$  respectively.

We plot the  $TP_{\Delta_0}$  and  $TP_{\Delta_R}$  points where x-axis represents the number of experiments and y-axis represents the number of true positive points. This experiment is repeated 50 times to get 100 points on the curve corresponding to  $TP_{\Delta_0}$  and  $TP_{\Delta_R}$ . If the curves intersect then we increase the value of  $\delta$  and repeat the experiment till we get curves that do not intersect. When the curves do not intersect,  $C_T$  is calculated as an average of ordinates of closest points on  $TP_{\Delta_0}$  and  $TP_{\Delta_R}$  curves and data complexity  $\beta$  is obtained as  $2^{\delta}$ .

In prediction phase, we use  $D_{r+1\cdots r+s}^{ML}$ ,  $C_T$ , and  $\beta$  to make predictions. We generate  $\beta$  plaintext pairs with difference  $\Delta_0$  and get the encrypted data from an ORACLE. We make the predictions on the difference of encrypted data and get the prediction probabilities (p). If number of elements with p greater than threshold probability (T) is greater than cutoff  $(C_T)$ , then we predict that ORACLE is the *s*-round CIPHER<sub>s</sub>. High accuracy differential distinguishers  $(D_{r+1\cdots r+s}^{HA-ML})$  for GIFT-128 and ASCON are constructed using the Algorithm 3.

$\mathbf{Al}$ T,	<b>Algorithm 3:</b> High Accuracy ML distinguisher $D_{r+1\cdots r+s}^{HA-ML} : (D_{r+1\cdots r+s}^{ML}, T, C_T, \beta)$									
1 <b>F</b>	Function Construction Phase( $D_{r+1\cdots r+s}^{ML}, T = 0.5$ ):									
2	$\delta \leftarrow 1$									
3	repeat									
4	for $k \leftarrow 1 \ to \ 50 \ do$									
5	$K \leftarrow$ Choose a random key									
6	$(P_{\Delta_0}, P'_{\Delta_0}) \leftarrow 2^{\delta}$ plaintext pairs with difference $\Delta_0$									
7	$(P_{\Delta_R}, P'_{\Delta_R}) \leftarrow 2^{\delta}$ plaintext pairs with random difference $\Delta_R$									
8	$(C_{\Delta_0}, C'_{\Delta_0}) \leftarrow (\text{CIPHER}_s(P_{\Delta_0}, K), \text{CIPHER}_s(P'_{\Delta_0}, K))$									
9	$(C_{\Delta_R}, C'_{\Delta_R}) \leftarrow (\text{CIPHER}_s(P_{\Delta_R}, K), \text{CIPHER}_s(P'_{\Delta_R}, K))$									
10	$p_{\Delta_0} \leftarrow \text{prediction probabilities for } (C_{\Delta_0} \oplus C'_{\Delta_0}) \text{ using}$ $D_{r+1}^{ML}$									
11	$p_{\Delta_R} \leftarrow \text{prediction probabilities for } (C_{\Delta_R} \oplus C'_{\Delta_R}) \text{ using}$ $D^{ML}$									
12	$TP_{A} \leftarrow \text{number of elements with } p_{A} > T$									
13	$TP_{\Lambda_{D}} \leftarrow \text{number of elements with } p_{\Lambda_{D}} > T$									
14	Plot the curve for $TP_{\Delta_0}$ and $TP_{\Delta_R}$ values									
15	end									
16	$\delta \leftarrow \delta + 1$									
17	<b>until</b> $(TP_{\Delta_0} \text{ and } TP_{\Delta_R} \text{ curves do not intersect});$									
18	$C_T \approx$ average of ordinates of closest points on $TP_{\Delta_0}$ and $TP_{\Delta_R}$ curves									
19	Data Complexity( $\beta$ ) $\leftarrow 2^{\delta}$									
20	$\mathbf{return}\ C_T,\ \beta$									
21 E	Ind Function									
22 F	Procedure Prediction Phase( $D^{ML}_{r+1\cdots r+s}, C_T, \ eta$ ):									
23	Test Data (TD) $\leftarrow$ (.)									
<b>24</b>	$\mathbf{for} \ i \leftarrow 1 \ to \ \beta \ \mathbf{do}$									
<b>25</b>	$P_i \leftarrow \text{Choose a random plaintext}$									
26	$P_i^{'}=P_i\oplus {\it \Delta}_0$									
<b>27</b>	$C_i \leftarrow \text{ORACLE}(P_i)$									
28	$C'_i \leftarrow \text{ORACLE}(P'_i)$									
29	Append TD by $C_i \oplus C_i'$									
30	end									
31	$p \leftarrow \text{prediction probabilities for elements in TD using } D_{r+1\cdots r+s}^{ML}$									
32	if ((number of elements with $p > T$ ) > $C_T$ ) then									
33	ORACLE = CIPHER <sub>s</sub>									
34	end									
35	else $\square OPACIE \rightarrow CIDUED$									
36 97	$  ORACLE \neq OPHER_s$									
37	nd Procoduro									
30 U										

9

# 3.3 Application of $D_{r+1\cdots r+s}^{HA-ML}$ to GIFT-128 and ASCON

For 7-round GIFT-128, the accuracy of  $D_{r+1...r+7}^{ML}$  is 0.55. To calculate  $C_T$  and  $\beta$ , curves are plotted in Fig. 1 as described in Algorithm 3. Although, the curves are almost separated at  $\delta = 2^8$  (Fig. 1 (c)) but a clear separation is visible in Fig. 1 (d) at  $\delta = 2^9$  and hence, the data complexity ( $\beta$ ) is 2<sup>9</sup>. The average of two closest points 171 and 181 on these curves is 176 which provides the value of  $C_T$  as 176. Using  $C_T$  and  $\beta$  values, we perform 10 experiments to validate the distinguisher's accuracy similar to previous case. Each experiment contains 50 TP and 50 TN samples and every sample contain 2<sup>9</sup> ciphertext differences. The results are presented in Table 10 which shows that the accuracy is higher than 97% in most of the cases. The source code for these experiments is available on GitHub<sup>2</sup>.

For 4-round ASCON, the accuracy of  $D_{r+1\cdots r+4}^{ML}$  is 0.502 and it is too low for an ML based distinguisher. Even with such a low accuracy, we are able to construct  $D_{r+1\cdots r+4}^{HA-ML}$  as shown in Fig. 2. To find a clear separation of the  $TP_{\Delta_0}$  and  $TP_{\Delta_R}$  curves, data requirement increases due to the low accuracy of  $D_{r+1\cdots r+4}^{ML}$ . The curves are almost separated at  $\delta = 2^{18}$  (Fig. 2 (d)) and hence, the data complexity ( $\beta$ ) is  $2^{18}$  and calculated  $C_T$  is 132825. The experiments to validate this distinguisher are presented in the Table 10 and very high accuracy is obtained in all cases. The accuracy is equal to 100% in half of the experiments.

The values of  $C_T$  and  $\beta$  to construct the distinguishers with high accuracy for GIFT-128 and ASCON are summarized in Table 7.

Ciphor	Rounda	Size	Accuracy	Algorithm 3		
Cipiter	nounus		Accuracy	$C_T$	$\beta$	
GIFT-128	7	128	0.55	176	$2^{9}$	
ASCON	4	320	0.502	132825	$2^{18}$	

Table 7:  $C_T$  and  $\beta$  for Differential Distinguishers with High Accuracy

### 4 Differential-ML distinguisher for 8-round GIFT-128

Yadav et. al. [24] extended the classical differential distinguisher with the ML based distinguisher and it was called a Differential-ML distinguisher. In [24], an experimental approach to construct the differential-ML distinguisher was presented by the authors. In this section, we construct the 8-round Differential-ML distinguisher extending the 3 and 2 rounds of classical distinguisher with 5 and 6 rounds of ML distinguisher respectively. An Algorithm to construct the

 $<sup>^2</sup>$  https://github.com/tarunyadav/Improved-Differential-Distinguisher-GIFT128-ASCON

11



Fig. 1:  $C_T$  and  $\beta$  for 7-round GIFT-128 Distinguisher  $(D_{r+1\cdots r+7}^{HA-ML})$ 

differential-ML distinguisher  $(D_{1\cdots r+s}^{CD\to ML})$  is presented in [24]. Differential-ML distinguisher  $D_{1\cdots r+s}^{CD\to ML}$  is represented with five parameters namely, *r*-round classical distinguisher  $D_{1\cdots r}^{CD}$ , *s*-round ML distinguisher  $D_{r+1\cdots r+s}^{ML}$ , threshold probability *T*, cutoff  $C_T$ , and data complexity  $\beta$ . In this paper, threshold probability is taken as 0.5 instead of accuracy  $(\alpha_s)$  of ML based distinguisher used in [24].





Fig. 2:  $C_T$  and  $(\beta)$  for 4-round ASCON Distinguisher  $(D^{HA-ML}_{r+1\cdots r+4})$ 

<b></b>	T DIAM	
Round	Input Difference	Probability
(r)	$(\Delta_r)$	$(2^{-p_r})$
0	0000 0000 0000 0000 0000 0000 10c0 0000	0
1	$0000 \ 0000 \ 0000 \ 0000 \ 0000 \ 0000 \ 0000 \ 0000 \ 0000$	$2^{-5}$
2	$0000 \ 0001 \ 0000 \ 0000 \ 0000 \ 0000 \ 0000 \ 0000$	$2^{-2}$
3	0800 0000 0000 0000 0000 0000 0000 0000	$2^{-3}$
4	2000 0000 1000 0000 0000 0000 0000 0000	$2^{-2}$
5	4040 0000 2020 0000 0000 0000 0000 0000	$2^{-5}$
6	5050 0000 0000 0000 5050 0000 0000 0000	$2^{-8}$
7	0000 0000 0000 0000 0000 0000 a000 a00	$2^{-12}$
8	0000 0000 0000 0000 0000 0011 0000 0000	$2^{-4}$
9	0000 0800 0000 0800 0000 0000 0000 0000	$2^{-6}$
10	0202 0000 0101 0000 0000 0000 0000 0000	$2^{-4}$
11	0000 0000 5050 0000 0000 0000 5050 0000	$2^{-10}$
12	0000 0000 0000 0000 0000 0000 00a0 00a	$2^{-12}$
13	0000 0011 0000 0000 0000 0000 0000 0000	$2^{-4}$
14	0800 0000 0800 0000 0000 0000 0000 0000	$2^{-6}$
15	2020 0000 1010 0000 0000 0000 0000 0000	$2^{-4}$
16	5050 0000 0000 0000 5050 0000 0000 0000	$2^{-10}$
17	0000 0000 0000 0000 0000 0000 a000 a00	$2^{-12}$
18	0000 0000 0000 0000 0000 0011 0000 0000	$2^{-4}$
19	0000 0000 0000 0c00 0000 0600 0000 0000	$2^{-6}$
20	0002 0200 0000 0000 0000 0000 0000 0000	$2^{-4}$

Table 8: Differential Characteristic for 20-round GIFT-128

We searched for an optimal 20-round differential characteristics (Table 8) for GIFT-128 and used a difference 0x00020200000000000000000000000000 as  $\Delta_0$  for training the *s*-round  $D_{r+1\cdots r+s}^{ML}$ . We use 5-round ML distinguisher as  $D_{3\cdots 8}^{ML}$  and  $D_{2\cdots 8}^{ML}$  respectively. We construct high accuracy differential-ML distinguisher  $(D_{1\cdots 8}^{CD\to ML})$  for 8 rounds of GIFT-128. We extend 3 rounds of differential characteristics mentioned Table 8 ( $\Delta_{17} \to \Delta_{20}$ ) with 5-round  $D_{3\cdots 8}^{ML}$  of accuracy 0.83. The results are shown in Fig. 3. A separation of curves occur at  $\delta = 2^{18}$  and it becomes the data complexity ( $\beta$ ) of  $D_{1\cdots r+8}^{CD\to ML}$ . We perform 10 experiments containing 50 TP and 50 TN samples where each sample consists of  $2^{18}$  output differences. It is observed form the results that the differential-ML distinguisher  $D_{1\cdots r+8}^{CD\to ML}$  provides 100% accuracy in most of the cases.



Fig. 3:  $C_T$  and  $\beta$  for 8-round GIFT-128 Distinguisher  $(D_{1\cdots8}^{CD\rightarrow ML}: D_{1\cdots3}^{CD}; D_{4\cdots8}^{ML})$ 

Cipher	Rounds	Classic	cal Distinguisher	ML Dis	tinguisher	Differential-ML Distinguisher		
	Total	Rounds	Data Complexity	Rounds	Accuracy	Data Complexity	Accuracy	
GIFT-128	8	3	$2^{14}$	5	0.83	$2^{18}$	99.8%	
GIFT-128	8	2	$2^{10}$	6	0.58	2 <sup>18</sup>	99.2%	

Table 9: Differential-ML Distinguisher for GIFT-128

## 5 Conclusion

The accuracy of ML based differential distinguisher with increased number of rounds and block size becomes very low. In this paper, we have addressed this challenge and proposed a new technique to convert the existing low accuracy ML based differential distinguisher to a high accuracy ML based distinguishers. We obtained improved differential distinguishers for 7-round GIFT-128 and 4-round ASCON permutation with very high accuracy. Our approach provided best improvements as compared to all of the existing approaches proposed in the literature to increase the accuracy of ML based distinguishers. Further, differential distinguishers for 8 rounds of GIFT-128 are constructed using machine learning with accuracy more than 99%.

### References

- Baksi, A., Breier, J., Chen, Y., Dong, X.: Machine learning assisted differential distinguishers for lightweight ciphers. (extended version), Cryptology ePrint Archive, 2020/571, https://eprint.iacr.org/2020/571 (2020)
- Banik, S., Chakraborti, A., Inoue, A., Iwata, T., Minematsu, K., Nandi, M., Peyrin, T., Sasaki, Y., Sim, S.M., Todo, Y.: GIFT-COFB. Cryptology ePrint Archive, 2020/738, https://eprint.iacr.org/2020/738 (2020)
- Benamira, A., Gerault, D., Peyrin, T., Tan, Q.Q. : A deeper look at machine learning-based cryptanalysis. In: A., Standaert, FX. (eds) Advances in Cryptology-EUROCRYPT 2021, LNCS, vol. 12696, pp 805–835, Springer, Cham (2021)
- Banik, S., Pandey, S.K., Peyrin, T., Sasaki,Y., Sim, S.M., Todo,Y.: GIFT: A small present - towards reaching the limit of lightweight encryption. In: Cryptographic Hardware and Embedded Systems - CHES 2017, 19th International Conference, vol. 10529, pp 321-345, (2017)
- Biham, E., Shamir, A.: Differential cryptanalysis of DES-like cryptosystems. Journal of Cryptology, 4(1), pp 3–72 (1991)
- Dobraunig, C., Eichlseder, M., Mendel, F., Schläffer, M.: ASCONv1. 2: lightweight authenticated encryption and hashing. Journal of Cryptology 2021, 34, pp 1–42 (2021)
- Erlacher, J., Mendel, F., Eichlseder, M.: Bounds for the Security of Ascon against Differential and Linear Cryptanalysis. IACR Transactions on Symmetric Cryptology, 2022(1), pp 64-87. https://doi.org/10.46586/tosc.v2022.i1.64-87 (2022)
- Ebrahimi, A., Regazzoni, F., Palmieri, P.: Reducing the cost of machine learning differential attacks using bit selection and apartial ml-distinguisher. In: International conference on Foundations and Practice of Security, FPS 2022, LNCS, vol. 13877, pp 123-141, Springer, Cham, (2021)

- Gohr, A., Leander, G., Neumann P.: An assessment of differential neural distinguishers, 2022, Cryptology ePrint Archive, Paper 2022/1521, (2022) https://eprint.iacr.org/2022/1521
- Gohr, A.: Improving attacks on round-reduced speck32/64 using deep learning. In: Proceedings of Advances in Cryptology - CRYPTO 2019, SantaBarbara, USA, pp 150–179. Springer-verlag, Cham, (2019)
- 11. Hou, Z., Ren, J., Chen, S.: Improve neural distinguishers of Simon and Speck. Security and Communication Network, vol. 2021, Isuue 1, pp 1–11, (2021)
- Liu, J.S., Ren, J.J., Chen, S.Z.: A deep learning aided differential distinguisher improvement framework with more lightweight and universality, Cybersecurity, (2023)
- Lyu, L., Tu, Y., Zhang, Y.: Improving the deep-learning-based differential distinguisher and applications to simeck. In: 2022 IEEE 25th international conference on computer supported cooperative work in design, CSCWD, IEEE, pp 465–70 (2022)
- Matsui, M.: On correlation between the order of S-boxes and the strength of DES. Proceeding of International Conference. EUROCRYPT, Italy, May 1994, pp. 366–375 (1994)
- Mouha, N., Wang, Q., Gu, D. and Preneel, B.: Differential and Linear Cryptanalysis Using Mixed-Integer Linear Programming. In Chuankun Wu, Moti Yung, and Dongdai Lin, editors, Inscrypt 2011, vol. 7537, LNCS, pp 57–76, Springer (2011)
- National Institute of Standards and Technology. Lightweight Cryptography, Finalists, https://csrc.nist.gov/projects/lightweight-cryptography/finalists, NIST, (2021)
- Sun, S., Gerault, D., Lafourcade, P., Yang, Q., Todo, Y., Qiao, K., and Hu, L.: Analysis of AES, SKINNY, and Others with Constraint Programming. IACR Transaction on Symmetric Cryptology, vol. 1, pp 281–306, (2017)
- Sun, S., Hu, L., Wang, P., Qiao, K., Ma, X., Song, L.: Automatic security evaluation and (related-key) differential characteristic search: Application to SIMON, PRESENT, LBlock, DES and other bit-oriented block ciphers. In: International conference on the theory and application of cryptology and information security, Springer, 2014, pp 158–78 (2014)
- Shen, D., Song, Y., Lu, Y., Long, S., Tian, S.: Neural differential distinguisher for GIFT-128 and ASCON. Journal of Information Security and Applications, vol. 82, (2024)
- Tian, W., Hu, B.: Deep learning assisted differential cryptanalysis for the lightweight Cipher Simon. KSII Transaction on Internet and Information Systems, (TIIS) 2021, vol. 15, No 2, pp 600–616, (2021)
- Wang, G., Wang, G.: Improved differential-ML distinguisher: Machine learning based generic extension for differential analysis. In: International conference on information and communications security. Springer, 2021, pp 21–38, (2021)
- 22. Wang, G., Wang, G.: Keeping classical distinguisher and neural distinguisher in balance, Journal of Information Security and Applications, vol. 84, (2024)
- 23. Wang G, Wang G, He Y. Improved machine learning assisted (related-key) differential distinguishers for lightweight ciphers. In: 2021 IEEE 20th international conference on trust, security and privacy in computing and communications(trustCom). IEEE, 2021, pp 164–71 (2021)
- Yadav, T., Kumar, M.: Differential-ML Distinguisher: Machine Learning based Generic Extension for Differential Cryptanalysis. In Longa, P., Rafols, C., editors, Progress in Cryptology- LATINCRYPT 2021, vol. 12912, pp 191-212, (2021)

Experiment No.		GII	FT-128	ASCON					
	5	=7,	m = 128,		s=4, m=320,				
	$C_T$	= 1	76, $\beta = 2^9$	$C_T$	$C_T = 13825, \ \beta = 2^{18}$				
	$\mathrm{TP}$	TN	Accuracy	TP	TN	Accuracy			
1	49	50	99	50	50	100			
2	49	50	99	50	48	98			
3	47	50	97	49	50	99			
4	49	50	99	50	48	98			
5	50	50	100	50	50	100			
6	49	50	99	50	50	100			
7	48	50	98	50	50	100			
8	50	50	100	50	50	100			
9	50	50	100	50	46	96			
10	47	50	97	50	50	100			

# Appendix A - Experimental Accuracy of Distinguishers

Table 10: Accuracy of  $D_{r+1 \cdots r+s}^{HA-ML}$  for 7-round GIFT-128 and 4-round ASCON