|  |  |  |
| --- | --- | --- |
|  |  | |
| Encryption Scheme Classification |  | |
|  |  | |
|  |  | |
|  | | Name: MayankDate: 28-Jan-24 |

|  |  |  |
| --- | --- | --- |
| Table of Contents | | |
| 1. | Introduction | 3 |
| 2. | Background | 4 |
| 3. | Encryption | 4 |
| 4. | Working of Encryption | 4-5 |
| 5. | Types of Encryptions | 5-6 |
| 6. | Importance of Encryption | 6-7 |
| 7. | Block Cipher | 7 |
| 8. | Working of Block Cipher | 7-8 |
| 9. | Block Cipher modes of Operation | 8-13 |
| 10. | Data encryption standard (DES) | 13-14 |
| 11. | Broad-level steps in DES | 14-15 |
| 12. | Double DES and Triple DES | 15-17 |
| 13. | Advanced Encryption Standard (AES) | 17-18 |
| 14. | Encryption and Decryption in AES | 18-19 |
| 15. | Deep Learning | 19 |
| 16. | Working of Deep Learning | 19-20 |
| 17. | Components of a Deep Learning network | 20 |
| 18. | Convolutional Networks | 21 |
| 19. | Working of CNN | 21-25 |
| 20. | Method | 25 |
| 21. | Data Preparation | 25 |
| 22. | Model Preparation | 25 |
| 23. | Evaluation | 25-28 |
| 24. | Conclusion | 29 |
| 25. | References | 29 |

### Introduction

In the ever-expanding digital realm where data has become a cornerstone, safeguarding sensitive information has never been more crucial. Encryption stands as a guardian, but encryption is diverse, comprising various schemes and methodologies designed to meet distinct security needs. The classification of encryption schemes plays a very important role in understanding the nuances of these techniques and tailoring security measures.

Encryption involves the use of special mathematical algorithms to convert data in its plain or readable form by software or by humans also known as plaintext to a form that is unreadable or non-interpretable form also known as ciphertext. The encrypted data or ciphertext is only readable after a matching decryption process converts the unreadable form back to its original readable form. Generally, valid decryption entails the use of matching decryption algorithm to the encryption algorithm with the needed decryption key.

By using different techniques, someone might identify the type of encryption algorithm we're using. Once they know that, encrypted data could be deciphered through matching of decryption algorithm and subsequent search for the valid decryption key. The key may be determined through brute force or forms of heuristic determination.

Convolution Neural Networks (CNN) may be used to classify encryption schemes over encrypted images without any feature engineering done. We trained YOLOv8x-cls CNN model for the task.

### Background

The following details are the introduction to encryption and a brief introduction to Convolutional Neural Network deep learning algorithm.

### Encryption

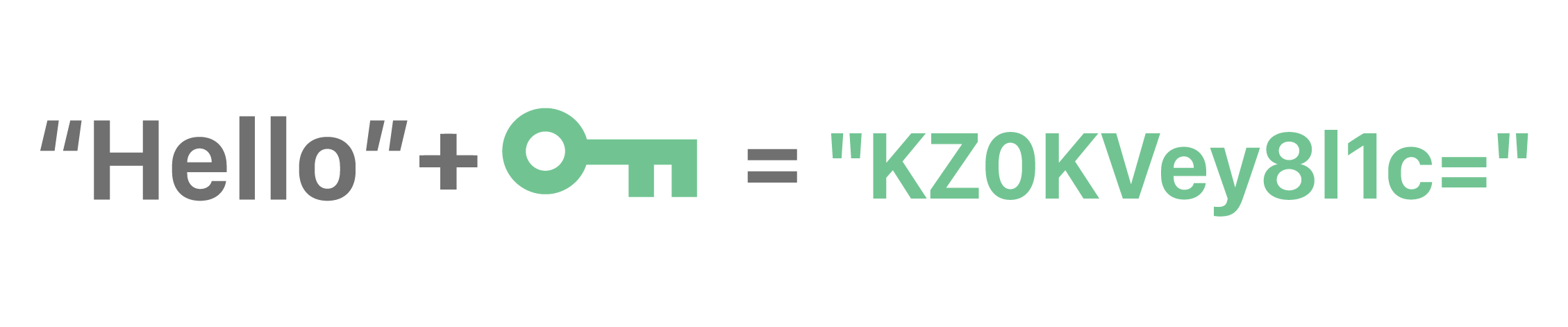
Encryption serves as a crucial method for securing data by transforming human-readable plaintext into incomprehensible ciphertext, ensuring that only authorized parties can decipher the information.

This technical process involves the utilization of a **cryptographic key**—a set of mathematical values that is provided to both the sender and the recipient of an encrypted message. Through this key, the encryption algorithm systematically rearranges the data, making it inaccessible to unauthorized individuals.

The use of cryptographic keys not only facilitates secure communication but also plays a pivotal role in safeguarding sensitive information across various digital platforms and communication channels. In essence, encryption acts as a fundamental pillar in modern cybersecurity, offering a robust defense against unauthorized access and ensuring the confidentiality and integrity of transmitted data.

### Cryptographic key

In cryptography, a key is a string of characters used within an encryption algorithm for altering data so that it appears random. Like a physical key, it locks (encrypts) data so that only someone with the right key can unlock (decrypt) it.



The original data is known as the plaintext, and the data after the key encrypts it is known as the ciphertext.

### Working of Encryption

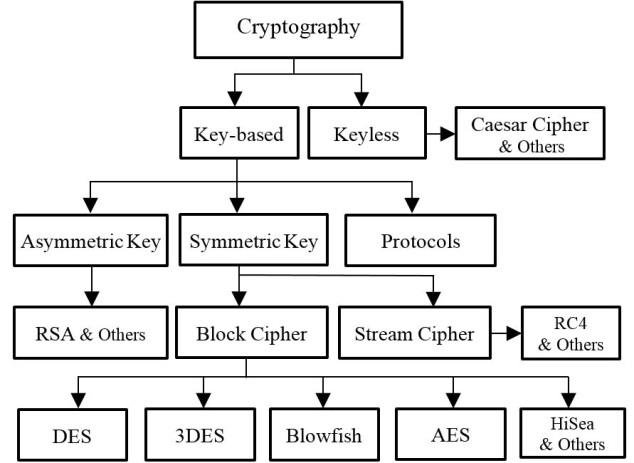
Encryption is a mathematical process which operates through the application of an encryption algorithm and a corresponding key to alter data. Despite the seemingly random appearance of encrypted data, the encryption process adheres to a logical and predictable sequence. This predictability allows an authorized party possessing the correct key to decrypt the data, reverting it back to its original plaintext form.

The strength of encryption lies in the complexity of the keys employed, making it highly improbable for a third party to decrypt or break the ciphertext through brute force methods. Secure encryption is designed to withstand such attempts, ensuring the confidentiality and integrity of the information.

Data can be encrypted in two primary states— "at rest," when stored, and "in transit," during transmission to another location. This dual application of encryption serves as a versatile safeguard, protecting sensitive information both when stored and during its journey across digital networks.

### Types of Encryptions

Encryption methods are broadly categorized into two main types: **Symmetric Encryption** and **Asymmetric Encryption**.



### Symmetric Encryption

In symmetric encryption**, a single key** is utilized for both the encryption and decryption processes. This secret key is provided to all parties involved in communication.

Despite its simplicity, symmetric encryption requires a secure method for distributing and managing the shared key to prevent unauthorized access. The strength of symmetric encryption lies in the confidentiality of the key itself, emphasizing the need for a robust key distribution mechanism.

### Asymmetric Encryption

Asymmetric encryption is also referred to as public key encryption, **has two distinct keys** for its operations. One key is used for encryption, and a separate corresponding key is used for decryption.

This dual-key system enhances security by eliminating the necessity of sharing a common secret key among communicating parties. The key used for encryption is made public, allowing anyone to send encrypted messages to the owner of the paired private key. Asymmetric encryption is particularly advantageous in scenarios where secure key exchange poses a challenge, offering a more flexible and secure approach to data protection.

Commonly used symmetric encryption algorithms include:

* **AES**
* **3-DES**
* **SNOW**

Commonly used asymmetric encryption algorithms include:

* **RSA**
* **Elliptic curve cryptography**

### Importance of Encryption

Encryption plays an important role in modern information security, serving various crucial purposes that secures the sensitive data. One primary significance of encryption lies in the preservation of **privacy**. By converting data into an incomprehensible format, encryption ensures that only the intended recipient or the rightful owner of the data can access and decipher the information. This measure is particularly critical in maintaining the confidentiality of communications and protecting data at rest from unauthorized access.

Another aspect is **security**, where encryption acts as defence against data breaches. Encryption serves as a barrier, preventing unauthorized parties from gaining access to the underlying information. This proactive approach helps eliminate cyber threats and ensures that even if data falls into the wrong hands, it remains indecipherable without the appropriate decryption key.

Encryption also plays a key role in ensuring **data integrity**. By securing data from on-path attacks and malicious tampering, encryption helps maintain the accuracy and reliability of the information. This aspect is crucial in preventing unauthorized alterations to data during transmission, providing a trustworthy foundation for digital communication.

Moreover, encryption aligns with **regulatory requirements** imposed by both industry standards and government authorities. Recognizing the importance of data protection, many regulations mandate companies handling user data to implement encryption measures. Compliance with these regulations not only helps organizations avoid legal consequences but also reinforces their commitment to safeguarding user privacy and maintaining the security of sensitive information.

### Block Cipher

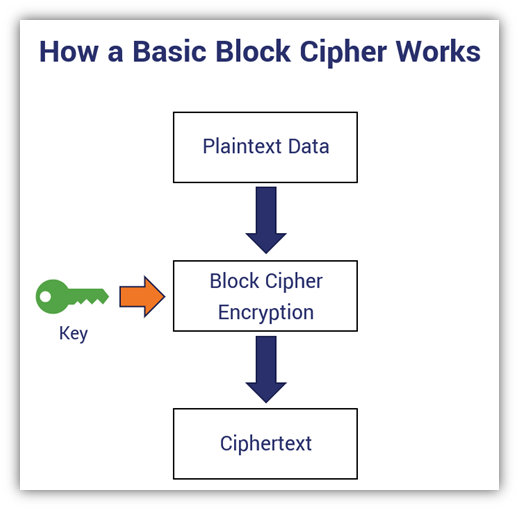
A block cipher is an encryption technique employed to secure data by processing it in fixed-size blocks, generating ciphertext through a combination of a cryptographic key and a designated algorithm.

In contrast to stream ciphers, which encrypt data bit by bit, block ciphers operate on fixed-size blocks concurrently, enhancing their efficiency and security. The fixed-size block approach simplifies the encryption process, as the algorithm can consistently handle predetermined chunks of data. In the realm of modern cryptography, prevalent block ciphers are typically engineered to encrypt data in blocks of either 64 or 128 bits. This standardized block size ensures compatibility and interoperability across various systems and applications, establishing a robust foundation for secure data transmission and storage.

Block ciphers are essential in information security for encrypting data, securing files and disks, and enabling secure communications through VPNs, SSL, and TLS. They also play a key role in implementing digital signatures for document authenticity. In brief, block ciphers are crucial for diverse applications, ensuring the confidentiality and integrity of sensitive information.

### Working of Block Cipher

A block cipher uses a symmetric key and algorithm to perform encryption and decryption on fixed-size blocks of data. To enhance security and eliminate brute-force attacks, a block cipher incorporates an **Initialization Vector (IV)** into the input plaintext. This IV, derived from a random number generator, is combined with the text in the initial block and the key. The inclusion of the IV ensures that subsequent blocks produce ciphertext distinct from that of the first encryption block, contributing to increased key space and making it more challenging for adversaries to decipher the key through brute force methods. The block size of a block cipher refers to the number of bits processed together in each encryption or decryption operation, a critical factor in determining the cipher's efficiency and overall security.



### Block Cipher modes of Operation

A block cipher is an encryption algorithm designed to process a fixed-size input, typically denoted as "b" bits, and generate a ciphertext of the same size. If the input exceeds this specified size, it can be divided into smaller blocks for processing. Various modes of operation have been developed to adapt block ciphers to different applications and use cases. These modes include **Electronic Code Book (ECB), Cipher Block Chaining (CBC), Cipher Feedback Mode (CFB), Output Feedback Mode (OFM) and Counter Mode.**

### Electronic Code Book (ECB)

The Electronic Code Book (ECB) mode is one of the simplest block cipher modes due to its direct and straightforward approach to encryption. In ECB mode, each block of input plaintext is independently encrypted, resulting in an output of blocks of encrypted ciphertext. This simplicity arises from the absence of intricate dependencies between blocks, making it easier to implement and understand.

A diagram of a computer program

Description automatically generated

If a message surpasses the specified size of "b" bits, it is divided into smaller blocks, and the encryption procedure is iteratively applied to each block. Despite its simplicity, the independent encryption of blocks in ECB mode makes it susceptible to certain vulnerabilities, particularly when identical plaintext blocks result in identical ciphertext blocks, potentially revealing patterns in the data. As a result, while ECB mode is straightforward, it may not be the most secure choice for certain applications, prompting the use of more sophisticated block cipher modes with enhanced security features.

**Advantages of using ECB:**

* Parallel encryption of blocks of bits is possible, thus it is a faster way of encryption.
* Simple way of the block cipher.

**Disadvantages of using ECB:**

* Prone to cryptanalysis since there is a direct relationship between plaintext and ciphertext.

### Cipher Block Chaining (CBC)

Cipher Block Chaining (CBC) mode has an advancement over Electronic Code Book (ECB), addressing some of the security concerns associated with ECB. In CBC, the encryption process involves the XOR operation between the previous ciphertext block and the current plaintext block before it undergoes encryption. This introduction of an XOR operation adds a crucial element of feedback, enhancing the security of the encryption process.

A diagram of a computer system

Description automatically generated

The key innovation in CBC is the way each cipher block is produced. Instead of directly encrypting the present plaintext block, CBC encrypts the XOR output of the previous cipher block and the current plaintext block. This approach introduces a level of interdependence between adjacent blocks, making it more challenging for adversaries to discern patterns in the ciphertext. The chaining effect, where the output of one block influences the encryption of the next, significantly improves the overall security of the block cipher, making CBC a more robust choice for applications where data confidentiality is paramount.

**Advantages of CBC:**

* CBC works well for input greater than b bits.
* CBC is a good authentication mechanism.
* Better resistive nature towards cryptanalysis than ECB.

**Disadvantages of CBC:**

* Parallel encryption is not possible since every encryption requires a previous cipher.

### Cipher Feedback Mode (CFB)

Cipher Feedback Mode (CFB) is a block cipher mode of operation that introduces a feedback mechanism into the encryption process. In CFB mode, the output of the encryption for the preceding block (cipher) is utilized as feedback for the encryption of the next block. This mode operates with specific specifications designed to enhance its security.

To initiate the process, an initial vector (IV) is employed for the first encryption, providing a starting point for the feedback loop. The output bits of this initial encryption are then divided into two sets: the left-hand side (LHS) with "s" bits and the right-hand side (RHS) with "b-s" bits. The LHS bits are selected and combined with the corresponding plaintext bits through an XOR operation.

The result of this XOR operation serves as input to a shift register, which has "b-s" bits on the left-hand side and "s" bits on the right-hand side. The shift register undergoes a shift operation, and the process continues for each subsequent block of plaintext. This feedback mechanism introduces a dynamic and interdependent relationship between the blocks, enhancing the security of CFB mode and making it suitable for applications where a balance between security and efficiency is crucial.

A diagram of a computer system

Description automatically generated

**Advantages of CFB:**

* Since there is some data loss due to the use of shift register, it is difficult for applying cryptanalysis.

**Disadvantages of using CFB:**

* The drawbacks of CFB are the same as those of CBC mode. Both block losses and concurrent encryption of several blocks are not supported by the encryption. Decryption, however, is parallelizable and loss tolerant.

### Output Feedback Mode (OFB)

The Output Feedback (OFB) mode in block is like the Cipher Feedback (CFB) mode in its process, with a notable distinction in the feedback mechanism. In OFB mode, the encrypted output, rather than the XOR output (cipher), is sent as feedback to the next block of encryption. Unlike CFB mode, where only selected "s" bits are sent as feedback, OFB mode transmits all bits of the block.

This approach in OFB mode contributes to its high resistance against bit transmission errors. By sending the entire encrypted block as feedback, it ensures that any errors in transmission affect only the specific block and do not propagate through subsequent blocks. Furthermore, OFB mode reduces the dependency or relationship of the cipher on the plaintext, enhancing security by minimizing potential patterns in the ciphertext corresponding to patterns in the plaintext. This decreased dependency adds an additional layer of robustness to the encryption process, making Output Feedback mode a suitable choice in scenarios where error resilience and reduced plaintext-ciphertext correlation are crucial considerations.

A diagram of a computer program

Description automatically generated with medium confidence

**Advantages of OFB:**

* In the case of CFB, a single bit error in a block is propagated to all subsequent blocks. This problem is solved by OFB as it is free from bit errors in the plaintext block.

**Disadvantages of OFB:**

* The drawback of OFB is that, because of its operational modes, it is more susceptible to a message stream modification attack than CFB.

### Counter Mode (CTR)

The Counter Mode (CTR) is a block cipher implementation that utilizes a straightforward counter-based approach. It encrypts a counter-initiated value, and the resulting output is XORed with the plaintext to generate the corresponding ciphertext block. CTR mode operates independently of feedback, allowing for parallel implementation and making it efficient for scenarios where simultaneous encryption of multiple blocks is essential. The simplicity of its design and parallelization capabilities contribute to the widespread use of CTR mode in applications requiring fast and secure encryption.

A diagram of a computer program

Description automatically generated

**Advantages of CTR:**

* Since there is a different counter value for each block, the direct plaintext and ciphertext relationship is avoided. This means that the same plain text can map to different ciphertext.
* Parallel execution of encryption is possible as outputs from previous stages are not chained as in the case of CBC.

**Disadvantages of CTR:**

* The fact that CTR mode requires a synchronous counter at both the transmitter and the receiver is a severe drawback. The recovery of plaintext is erroneous when synchronization is lost.

### Data encryption standard (DES)

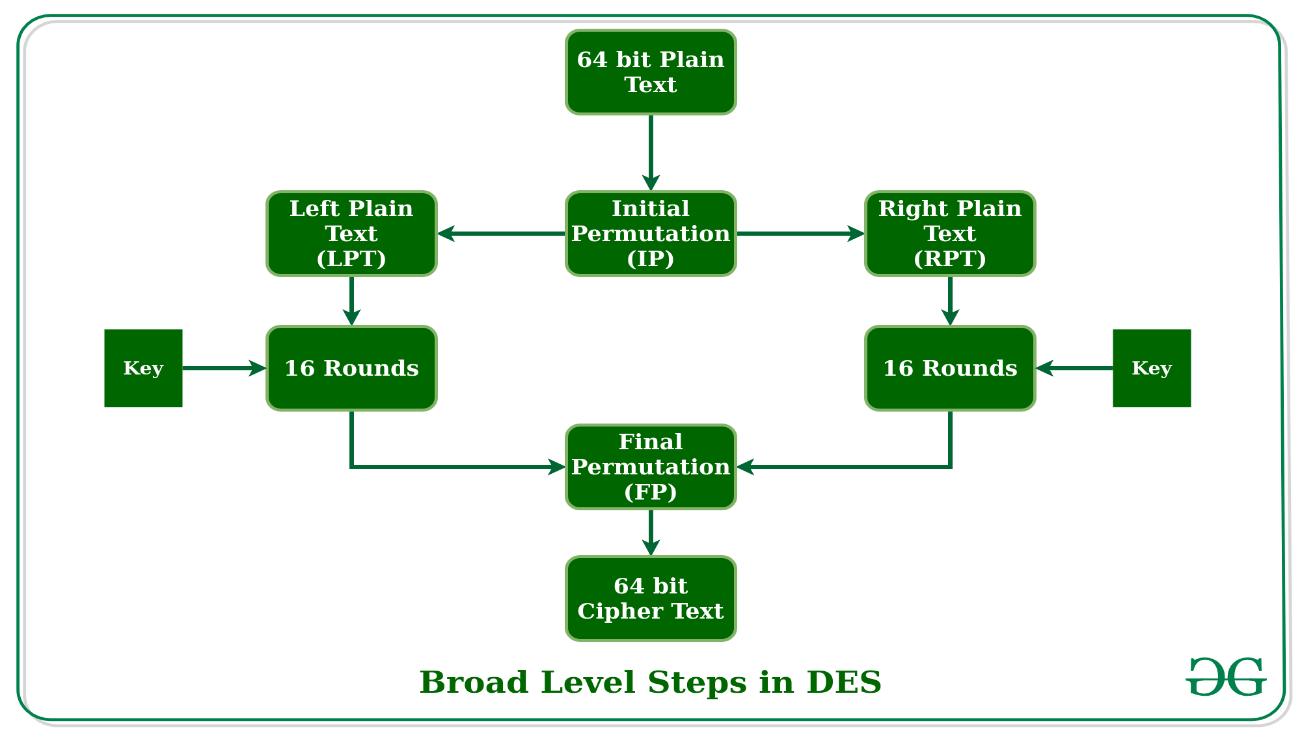
The Data Encryption Standard (DES) stands as a historically significant block cipher, contributing substantially to the field of data security. Employing a **56-bit key** length, DES encrypts data in fixed-size blocks of 64 bits each. This means that 64 bits of plaintext are processed by the DES algorithm, resulting in 64 bits of ciphertext. Notably, DES employs the same algorithm and key for both encryption and decryption, with minor distinctions in their application.

The initial key in DES comprises 64 bits, but before the encryption process commences, every 8th bit of the key is discarded. This operation transforms the original 64-bit key into a 56-bit key by discarding bits at positions 8, 16, 24, 32, 40, 48, 56, and 64. Despite its contribution to data security, DES has become somewhat outdated due to its relatively small key size, making it vulnerable to modern cryptographic attacks. As a result, more advanced encryption standards have been introduced to address the evolving landscape of cybersecurity.

It is based on the two fundamental attributes of **cryptography**: **substitution** (also called confusion) and **transposition** (also called diffusion). It consists of 16 steps, each of which is called a **round**. Each round performs the steps of substitution and transposition.

### Broad-level steps in DES

In the Data Encryption Standard (DES) encryption process, a 64-bit plaintext block undergoes an **initial permutation (IP)** through the Initial Permutation function. This IP function rearranges the bits of the plaintext block, producing two halves known as Left Plain Text (LPT) and Right Plain Text (RPT). These halves then progress through 16 rounds of the encryption process, each round involving a combination of substitution and transposition operations.



During each round, the LPT and RPT are transformed independently, and their results are then rejoined. After the 16 rounds are completed, a **Final Permutation (FP)** is applied to the combined block, resulting in a 64-bit ciphertext. The IP and FP functions serve as bookends to the encryption process, orchestrating the initial and final rearrangements of the bits.

The Initial Permutation (IP) is a one-time operation that occurs before the first round. It dictates how the transposition in IP should proceed, effectively juggling the bit positions of the original plaintext block. Following the IP, the resulting 64-bit permuted text block is divided into two 32-bit half-blocks, setting the stage for the subsequent rounds of encryption. This structured process, combining substitution and transposition in a series of rounds, forms the foundation of DES, showcasing its intricate design for secure data encryption.

### Step 1: Key transformation

* The initial 64-bit key is transformed into a 56-bit key by discarding every 8th bit of the initial key. Thus, for each a 56-bit key is available.
* From this 56-bit key, a different 48-bit Sub Key is generated during each round using a process called key transformation.
* For this, the 56-bit key is divided into two halves, each of 28 bits.
* These halves are circularly shifted left by one or two positions, depending on the round.
* A different subset of key bits is used in each round. That makes DES not easy to crack.

### Step 2: Expansion Permutation

* After the initial permutation, we had two 32-bit plain text areas called Left Plain Text (LPT) and Right Plain Text (RPT).
* During the expansion permutation, the RPT is expanded from 32 bits to 48 bits. Bits are permuted as well hence called expansion permutation.
* This happens as the 32-bit RPT is divided into 8 blocks, with each block consisting of 4 bits.
* Then, each 4-bit block of the previous step is then expanded to a corresponding 6-bit block, i.e., per 4-bit block, 2 more bits are added.
* This process results in expansion as well as a permutation of the input bit while creating output. The key transformation process compresses the 56-bit key to 48 bits. Then the expansion permutation process expands the **32-bit RPT** to **48-bits**.
* Now the 48-bit key is XOR with 48-bit RPT and the resulting output is given to the next step, which is the**S-Box substitution**.

### Double DES and Triple DES

The Data encryption standard (DES) uses 56 bits keys to encrypt any plain text which can easily be cracked by using modern technologies. To prevent this from happening double DES and triple DES were introduced which are much more secure than the original DES because it uses 112 and 168-bit keys respectively. They offer much more security than DES.

### Double DES

Double DES is an encryption technique which uses two instances of DES on the same plain text. In both instances it uses different keys to encrypt the plain text. Both keys are required at the time of decryption. The 64-bit plain text goes into first DES instance which then converted into a 64-bit middle text using the first key and then it goes to second DES instance which gives 64-bit cipher text by using second key. However double DES uses 112 bits key but gives security level of 2^56 not 2^112 and this is because of meet-in-the middle attack which can be used to break through double DES.

A diagram of a computer process

Description automatically generated

### Triple DES

Triple DES is an encryption technique which uses three instances of DES on the same plain text. It uses their different types of keys choosing technique in first all used keys are different and in second two keys are same and one is different and in third all keys are same. Triple DES is also vulnerable to meet-in-the middle attack because of which it gives total security level of 2^112 instead of using 168 bit of key. The block collision attack can also be done because of short block size and using same key to encrypt large size of text.

A diagram of a computer system

Description automatically generated

### Advanced Encryption Standard (AES)

The Advanced Encryption Standard (AES) represents a specification for electronic data encryption established by the U.S National Institute of Standards and Technology (NIST) in 2001. Employing the substitution-permutation network principle, AES encryption is carried out through a series of linked operations involving the replacement and shuffling of input data. Unlike its predecessor DES, AES offers key size options of 128, 192, or 256 bits, providing flexibility in tailoring security levels to specific needs.

AES operates by encrypting data in blocks of 128 bits each, enhancing its efficiency and security. This block size, coupled with the various key sizes, contributes to the adaptability of AES in diverse applications. Notably, AES has become widely adopted in contemporary cryptographic practices due to its superior strength compared to DES and triple DES, while still maintaining a reasonable level of implementation complexity. As a result, AES stands as a robust and trusted encryption standard, ensuring the confidentiality and integrity of electronic data in a variety of digital environments.

AES performs operations on bytes of data rather than in bits. Since the block size is 128 bits, the cipher processes 128 bits (or 16 bytes) of the input data at a time. The number of rounds depends on the key length as 128 bits key – 10 rounds, 192 bits key – 12 rounds and 256 bits key – 14 rounds.

A diagram of a diagram

Description automatically generated

A Key Schedule algorithm is used to calculate all the round keys from the key. So, the initial key is used to create many different round keys which will be used in the corresponding round of the encryption.

Some applications of AES are Wireless security Database Encryption, Secure communications, Data storage, Virtual Private Networks (VPNs), Secure Storage of Passwords and File and Disk Encryption.

### Encryption and Decryption in AES

### Encryption

AES considers each block as a 16-byte (4-byte x 4-byte = 128) grid in a column major arrangement. Each round comprises of **Sub Bytes, Shift Rows, Mix Columns** and **Add Round Key**.

In **Sub Bytes step** each byte is substituted by another byte. It is performed using a lookup table also called the S-box. This substitution is done in a way that a byte is never substituted by itself and not substituted by another byte which is a complement of the current byte.

In **Shift Rows step** each row is shifted a particular number of times.

**Mix Columns step** is basically matrix multiplication. Each column is multiplied with a specific matrix and thus the position of each byte in the column is changed as a result. This step is skipped in the last round.

Now, in the **Add Round Key step** the resultant output of the previous stage is XOR-ed with the corresponding round key. Here, the 16 bytes is not considered as a grid but just as 128 bits of data. After all these rounds 128 bits of encrypted data are given back as output. This process is repeated until all the data to be encrypted undergoes this process.

### Decryption

The stages in the rounds can be easily undone as these stages have an opposite to it which when performed reverts the changes. Each 128 blocks go through the 10,12 or 14 rounds depending on the key size. The stages of each round in decryption are **Add round key, Inverse Mix Columns, Shift Rows** and **Inverse Sub Byte.**

### Deep Learning

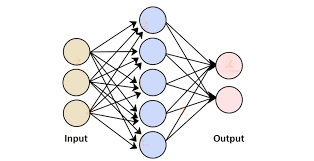
Deep learning is a method in **artificial intelligence (AI)** that teaches computers to process data in a way that is inspired by the human brain. Deep learning models can recognize complex patterns in pictures, text, sounds, and other data to produce accurate insights and predictions. You can use deep learning methods to automate tasks that typically require human intelligence, such as describing images or transcribing a sound file into text. Deep learning is used for Artificial Neural Networks for Regression and Classification, Convolutional Neural Networks for Computer Vision, Recurrent Neural Networks for Time Series Analysis, Self-Organizing Maps for Feature Extraction, Deep Boltzmann Machines for Recommendation Systems and Auto Encoders for Recommendation Systems.

**Advantages of Deep Learning:**

* Efficient processing of unstructured data
* Hidden relationships and pattern discovery
* Unsupervised learning
* Volatile data processing

### Working of Deep Learning

Deep learning algorithms are neural networks that are modeled after the human brain. For example, a human brain contains millions of interconnected neurons that work together to learn and process information. Similarly, deep learning neural networks, or artificial neural networks, are made of many layers of artificial neurons that work together inside the computer.



Artificial neurons are software modules called nodes, which use mathematical calculations to process data. Artificial neural networks are deep learning algorithms that use these nodes to solve complex problems.

### Components of a Deep Learning network

**Input layer:**

An artificial neural network has several nodes that input data into it. These nodes make up the input layer of the system.

**Hidden layer:**

The input layer processes and passes the data to layers further in the neural network. These hidden layers process information at different levels, adapting their behavior as they receive new information. Deep learning networks have hundreds of hidden layers that they can use to analyze a problem from several different angles.

**Output layer:**

The output layer consists of the nodes that output the data. Deep learning models that output “Yes" or “No" answers have only two nodes in the output layer. On the other hand, those that output a wider range of answers have more nodes.



### Challenges of Deep Learning

**Large quantities of high-quality data:**

Deep learning algorithms give better results when you train them on large amounts of high-quality data. Outliers or mistakes in your input dataset can significantly affect the deep learning process. To avoid such inaccuracies, you must clean and process large amounts of data before you can train deep learning models. Input data preprocessing requires large amounts of data storage capacity.

**Large processing power:**

Deep learning algorithms are compute-intensive and require infrastructure with sufficient compute capacity to properly function. Otherwise, they take a long time to process results.

### Convolutional Networks (LeCun, 1989)

Convolutional Neural Networks (CNNs) represent a specialized category of neural networks designed for processing data with a known, grid-like topology. Unlike conventional neural networks that employ general matrix multiplication in all layers, CNNs utilize convolution operations in at least one layer. The inspiration behind their name and structure draws from the human brain, specifically emulating the signalling process among biological neurons.

The grid-like topology of data, such as images, is well-suited for CNNs, as they excel in capturing spatial hierarchies and local patterns. Using convolutional layers, these networks can efficiently recognize features in a hierarchical manner, starting from basic patterns to more complex structures. This design makes CNNs particularly powerful in tasks related to image and pattern recognition, where spatial relationships play a crucial role in understanding and interpreting the data. As a result, CNNs have become a cornerstone in the field of deep learning, demonstrating their effectiveness in various applications, from image classification to object detection.

### Working of CNN

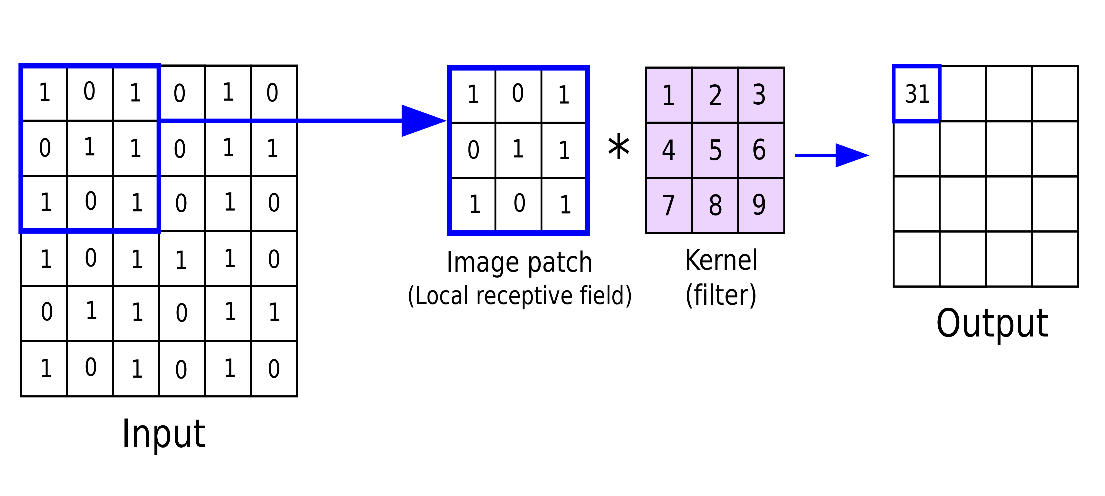
The convolutional layer serves as the inaugural component of a convolutional neural network (CNN), initiating the process of feature extraction. Following the convolutional layer, subsequent layers in a CNN architecture can consist of additional convolutional layers or pooling layers, each contributing to the network's increasing complexity. The final layer, known as the fully connected layer, integrates the extracted features and makes the network capable of producing meaningful predictions or classifications.



The progression of image data through the layers of a CNN is marked by an increasing focus on more intricate features. In the initial layers, the network identifies basic elements such as colours and edges, capturing low-level details. As the data traverses through subsequent layers, the network gradually recognizes larger and more complex structures or shapes within the input, ultimately leading to the identification of the intended object. This hierarchical feature extraction process, inspired by the human visual system, enables CNNs to excel in tasks like image recognition and classification, making them highly effective in various domains of machine learning and computer vision.

### Convolutional layer

A color image, which is made up of a matrix of pixels in 3D will have three dimensions— height, width, and depth. A **kernel** or a **filter** moves across the receptive fields of the image, checking if the feature is present. The **feature detector** is a two-dimensional (2-D) array of weights. The filter is then applied to an area of the image, and a dot product is calculated between the input pixels and the filter. This dot product is then fed into an output array. Afterwards, the filter shifts in a **stride**, repeating the process until the kernel has swept across the entire image.



The final output from the series of dot products from the input and the filter is known as a **feature map**, **activation map**, or a **convolved feature**. The weights in the feature detector remain fixed as it moves across the image, which is also known as parameter sharing. Some parameters, like the weight values, adjust during training through the process of **backpropagation** and **gradient descent**.

There are three hyperparameters which affect the volume size of the output that need to be set before the training of the neural network begins. These include **Number of filters, Stride** and **Padding**. There are three types of padding: **Valid padding, Same padding** and **Full padding.**

After each convolution operation, a CNN applies an **Activation Function** to the feature map, introducing nonlinearity to the model.

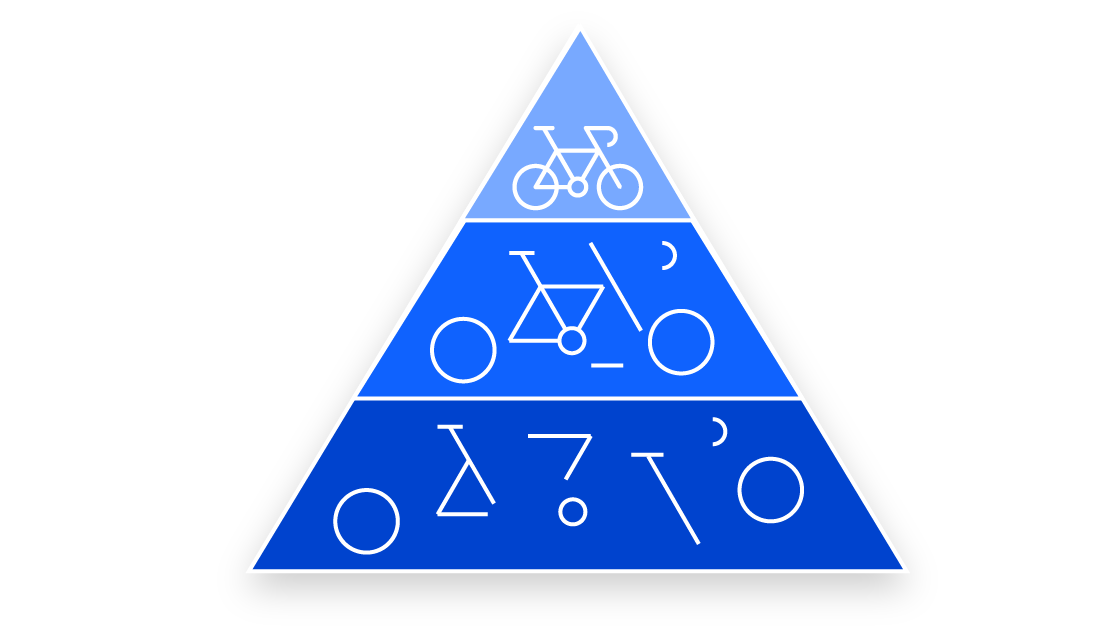
### Activation Layers

After each Convolutional layer in a CNN, we apply a nonlinear activation function, such as ReLU, ELU, or any of the other Leaky ReLU variants. We typically denote activation layers as RELU. In network diagrams as since ReLU activations are most used, we may also simply state ACT. Activation layers are not technically “layers” (since no parameters/weights are learned inside an activation layer).

### Additional Convolutional Layer

Another convolution layer can follow the initial convolution layer. The structure of the CNN can become hierarchical as the later layers can see the pixels within the receptive fields of prior layers.

**Example:**



Let’s assume that we’re trying to determine if an image contains a bicycle. You can think of the bicycle as a sum of parts. It is comprised of a frame, handlebars, wheels, pedals, et cetera. Each individual part of the bicycle makes up a lower-level pattern in the neural net, and the combination of its parts represents a higher-level pattern, creating a feature hierarchy within the CNN. Ultimately, the convolutional layer converts the image into numerical values, allowing the neural network to interpret and extract relevant patterns.

### Pooling layer

Pooling layers, also known as down sampling, conducts dimensionality reduction, reducing the number of parameters in the input. The pooling operation sweeps a filter across the entire input, but the difference is that this filter does not have any weights. Instead, the kernel applies an aggregation function to the values within the receptive field, populating the output array. They help to reduce complexity, improve efficiency, and limit the risk of overfitting. There are two main types of pooling **Max pooling** and **Average pooling.**

### Max Pooling and Average Pooling

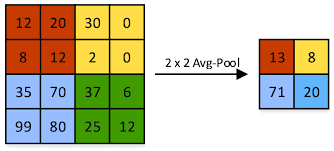
**Max pooling:**

 As the filter moves across the input, it selects the pixel with the maximum value to send to the output array. As an aside, this approach tends to be used more often compared to average pooling.



**Average pooling:**

As the filter moves across the input, it calculates the average value within the receptive field to send to the output array.



### Fully connected layer

This layer performs the task of classification based on the features extracted through the previous layers and their different filters. While convolutional and pooling layers tend to use ReLu functions, FC layers usually leverage a softmax activation function to classify inputs appropriately, producing a probability from 0 to 1.

### Method

The experiment started with the preparation of encrypted data needed to train, cross-validate and test the Convolutional Neural Network model. There was also the model preparation step to prepare the models through transfer learning from pre-trained models to accelerate model training.

### Data Preparation

The base dataset used to produce the model for encryption scheme classification consists of a set of images. They are a subset of the MNIST image dataset. The base dataset was further divided into two subsets. One subset of four image categories was used for training and validation of the deep learning models.

With the base dataset, the following encryption schemes were applied:

* TDES-ECB
* TDES-CBC
* AES-ECB
* AES-CBC

### Model Preparation

The Convolution Neural Network (CNN) deep learning algorithm was used in this experiment. Instead of constructing a new Convolution Neural Network from scratch, this experiment reused existing CNN models that have demonstrated its ability to classify complex images through YOLOv8x-cls.

### Evaluation

The evaluation or testing of the trained models was done with the testing dataset that used the same set of encryption schemes.

The classification evaluation of the base models was first done on the base CNN models using Confusion Matrix. Following which, all models including the depth modified models were evaluated using the measurements of TP (true positive) for valid classification of selected encryption scheme and FP (false positive) for wrong classification. The Precision, Recall, Accuracy and F1-Score (denoted as F1) measurements were then computed using the following formulas.

Precision = TP/ (TP + FP)

Recall = TP/ (TP + FN)

Accuracy = (TP + TN)/ (TP + TN + FP + FN)

F1 = 2\*((Precision \* Recall)/ (Precision + Recall))

The following is **Confusion Matrix assessment results** for the base models:

**AES ECB vs TDES ECB:**

A blue squares with white squares

Description automatically generated

**AES CBC vs TDES CBC:**

A blue and white diagram

Description automatically generated with medium confidence

**AES CBC vs TDES ECB:**

A blue squares with white text

Description automatically generated

**AES ECB vs TDES CBC:**

A blue squares with white dots

Description automatically generated

The following are the results of the experiment measured in **Precision, Recall, Accuracy and F1-Score values for the model**:

|  |  |  |  |  |
| --- | --- | --- | --- | --- |
| **Classification Test** | **Precision** | **Recall** | **Accuracy** | **F1-Score** |
| AES ECB vs TDES ECB | 93.8% | 92.1% | 93.0% | 0.93 |
| AES CBC vs TDES CBC | 56.4% | 13.8% | 51.5% | 0.22 |
| AES CBC vs TDES ECB | 95.6% | 95.1% | 95.4% | 0.95 |
| AES ECB vs TDES CBC | 52.1% | 63.6% | 52.6% | 0.57 |

### Conclusion

Deep learning algorithms specifically Convolutional Neural Networks could classify encryption schemes to a good level of accuracy based on entropic characteristics of the encrypted data. In this experiment YOLOv8x-cls were used.

### References

Cryptography and Network Security by William Stallings

Deep Learning by Ian Goodfellow Yoshua Bengio Aaron Courville

[Cryptography Introduction - GeeksforGeeks](https://www.geeksforgeeks.org/cryptography-introduction/?ref=lbp)

[Encryption - Wikipedia](https://en.wikipedia.org/wiki/Encryption)