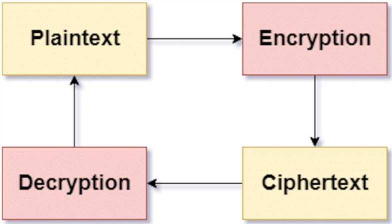
Autoencoder Neural Networks for Digital Cryptography

Jaime Tenorio

Óbuda University
John von Neumann Faculty of Informatics
Information and Coding Theory
Dr. Márta Takács
May 2023

Introduction

- Cryptography is the study of techniques for secure communication.
- An encryption algorithm transforms a message into one that is difficult to predict or duplicate.
- The most popular and widely used involve the use of a key, like Advanced Encryption Standard (AES) [1].

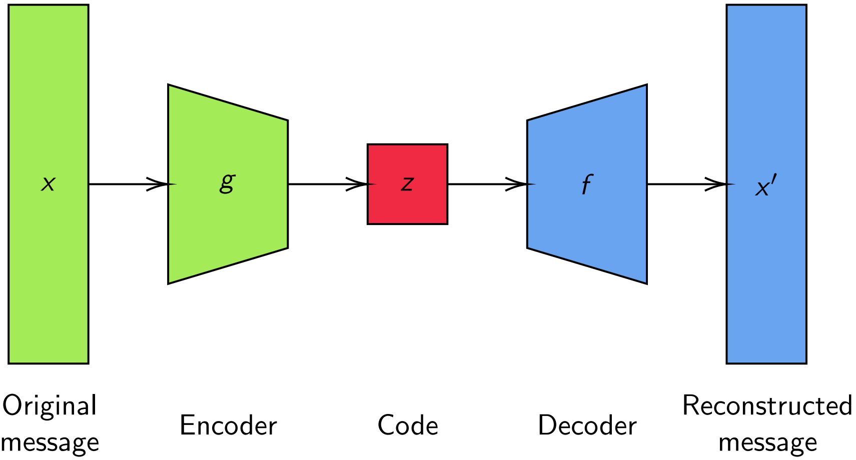


Introduction

- Autoencoders are neural networks whose that learn efficient representations of data by unsupervised training [2].
- The network learns to encode the information in a latent space, capturing the essential features.
- Symmetric architecture – the input has the same shape as the output. The middle layer is the code layer.
- Useful for dimensionality reduction, anomaly detection, signal processing, data compression, and cryptography.

$$\begin{aligned} z &= g(x) \\ x' &= f(z) \\ x &\approx x' \end{aligned} \tag{1}$$

Introduction



Autoencoders for Cryptography

- The code layer may be used as an encrypted message.
- The encoded message can then be transmitted securely – the latent representation may not reveal sensitive information.
- Can be adapted to any kind of data – images, text, numerical data, etc.
- Autoencoder architectures may vary depending on the cryptographic use case:
  - **Fully connected autoencoder:** general uses.
  - **Convolutional autoencoder:** images or video.
  - **Recurrent autoencoder:** text.

Case Study

Demonstration: encryption of binary representation of characters.

- Training data – all Unicode characters, at least 21 bits.
- Output dimension = input dimension.
- Depth of the neural network – 1 hidden layer for encoder and decoder [3] of 32 neurons each.

Hyperparameters:

- Activation: ReLU for hidden, sigmoid for output.
- Loss: binary cross entropy.
- Optimizer: Adam.
- Epochs: 50.

Case Study

- Encoding dimension = most important parameter.
- Encoding dimensions tested: 8, 12, and 16.

Encoding Dimension	Reconstruction Efficiency	Final Loss Value
8	4.43%	0.1715
12	33.83%	0.0794
16	100%	7.6340×10^{-5}

- Best performing model: **16-dimensional code, 100% reconstruction efficiency**.
- Autoencoder capable of reconstructing all characters without error.

Case Study

```python
sentence = "español 🇪🇸, русский 🇷🇺, français 🇫🇷, 한국인 🇰🇷"
encrypted_sentence = encrypt_sentence(sentence, encoder)
decrypted_sentence = decrypt_sentence(encrypted_sentence, decoder)

print(f"Original sentence: {sentence}")
print(f"Decrypted sentence: {decrypted_sentence}")

print(f"Encrypted sentence: {encrypted_sentence[:2]}")
```

```
2/2 [==============================] - 0s 3ms/step
2/2 [==============================] - 0s 4ms/step
Original sentence: español 🇪🇸, русский 🇷🇺, français 🇫🇷, 한국인 🇰🇷
Decrypted sentence: españoi 🇪🇸( русслйй 🇷🇺( fraoçais 🇫🇷( 한뷁인 🇰🇷
Encrypted sentence: [[ 3.7885313  26.431734   14.878041   12.408706   27.408964   29.216837
  21.249928   16.492983   20.69718    17.179825   16.881832   13.60826   ]
 [ 0.96823436 26.091682   20.463676   17.608208   29.640223   24.811924
  19.672276   15.238668   30.07502    12.234317   24.11402    22.007294  ]]
```

Case Study

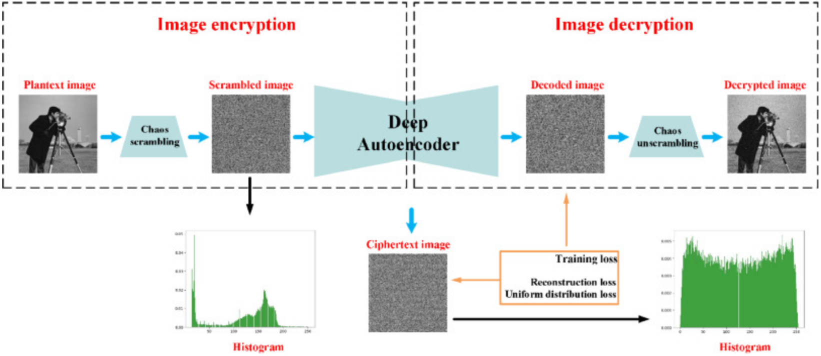
- **Issue**: transmission of the encoded messages.
- Encrypted characters consist of 16 neuron activations – 16 floating point numbers.
- Encrypted message is 20x larger than the original.
- Other encoding dimensions are not 100% efficient – they are not suitable.

Case Study

- **Solution**: smaller encoding dimension, at the cost of efficiency.
- **Other solution**: use a different activation at code layer – discretization.
- Different loss function, deeper neural network.
- Increases computational complexity.
- **AES complexity**: $O(n)$, where n is the number of bits in the message [1].
- **Autoencoder complexity**: $O(n^2)$, where n is the number of neurons in the network.

Conclusion

- Autoencoders are easy to implement, and they can be trained with either small or large datasets.
- Can be used to encrypt and decrypt messages without error.
- It would take a more sophisticated system to achieve lossless encryption with a reasonable message size.



[1] B. A. Forouzan,
Introduction to Cryptography and Network Security.
New York, NY, USA: McGraw-Hill Higher Education, 2008,
ISBN: 978–0–07–287022–0.

[2] M. Kramer, "Autoassociative neural networks," *Computers & Chemical Engineering*, vol. 16, no. 4, pp. 313–328, 1992, Neutral network applications in chemical engineering,
ISSN: 0098-1354. DOI: 10.1016/0098-1354(92)80051-A.

[3] U. Menon, A. R. Menon, A. Hudlikar, A. Sharmila, and P. Mahalakshmi,
"A hybrid autoencoder architecture for text encryption,"
in *2021 Innovations in Power and Advanced Computing Technologies (i-PACT)*, 2021, pp. 1–7.
DOI: 10.1109/i-PACT52855.2021.9696715.