Autoencoder Neural Networks for Digital Cryptography

Jaime Tenorio

Óbuda University
John von Neumann Faculty of Informatics
Information and Coding Theory
May 2023

1 Introduction

1.1 Cryptography

Cryptography is the study of techniques for secure communication, so that third parties cannot read messages sent between the sender and the receiver. An encryption algorithm is a mathematical function that transforms a message into a form that is difficult to predict or duplicate. Several encryption methods have been developed over the years, the most popular and widely used involve the use of a key, which defines how the message is encoded and decoded, for example, the Advanced Encryption Standard (AES) [1]. Cryptography has many applications, such as authentication, confidentiality, integrity, and non-repudiation.

1.2 Autoencoder Neural Networks

Autoencoders are a type of neural networks whose purpose is to learn efficient representations of data by undergoing unsupervised training [2]. When training an autoencoder, the network learns to reconstruct the original data by encoding the information in a latent space, which is a representation that captures the essential features of the data. These neural networks have a symmetric architecture, in other words, their input is the same shape as their output, and the middle layer is the one responsible for the representation of the data. The middle layer is usually called the code layer.

Autoencoder neural networks are used in tasks such as dimensionality reduction, anomaly detection, signal processing, data compression, and cryptography.

A basic autoencoder architecture is shown in 1. The autoencoder may be

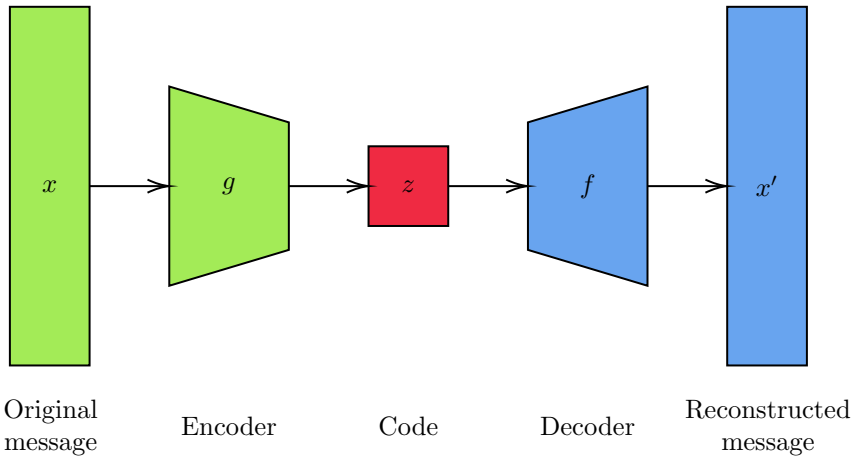

Figure 1: Autoencoder neural network.

expressed in mathematical terms as follows:

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

where x is the input, z is the code, x' is the output, g is the encoder, and f is the decoder.

2 Autoencoders for Cryptography

The nature of the autoencoder architecture makes it a suitable technique for cryptography, since the code layer may be used as an encrypted message. Essentially, a message can be encrypted by passing it through the trained encoder part of the network and extracting the output from the code layer. Then, the encrypted message is decoded by passing it through the decoder part of the network. The encoded message can then be transmitted securely, because the latent representation may not reveal sensitive information, even if it may be intercepted.

One advantage of autoencoder networks for cryptography is that they are versatile. Autoencoders can be adapted to any kind of data, such as images, text, numerical data, etc., and can be designed to learn codes of any size. Another advantage is their ability to filter noise and redundancies while encoding, due to the efficient representation of the data in the latent space.

Autoencoder architectures may vary depending on the cryptographic use case. Some examples include:

- **Fully connected autoencoder.** This is the most basic autoencoder, which consists of fully connected layers. This architecture can be used in general cryptography for encoding and decoding messages.
- **Convolutional autoencoder.** The use of convolutional layers in autoencoder architecture allows for the capture of spatial information, therefore this can be used for image or video cryptography.
- **Recurrent autoencoder.** Recurrent neural networks can learn sequential data efficiently. An autoencoder with this architecture may be useful for text encryption.

3 Case Study

We can demonstrate an autoencoder for cryptography by attempting to encrypt binary representation of characters. Previous studies show the viability of such autoencoders for 8-bit ASCII characters [3], so we will expand on this example and do it for 16-bit Unicode characters.

Firstly, the training data is determined by generating all Unicode characters, that is, binary representations of all integers from 32 to 65535. If we define the training data as X, then it can be defined as follows:

$$X = \{x \in \mathbb{R}^{16} \mid x = \text{bin}(i), i \in [32, 65535]\} \tag{2}$$

As it was discussed previously, the output of the neural network will be of the same dimension as its input. The depth of the neural network depends on the specific application, and the complexity of the system. Many sources suggest that a hidden encoder layer and a hidden decoder layer are sufficient for general purposes [4]. In this case, we will use a hidden layer of 32 neurons for both the encoder and the decoder. The activation function for the hidden layers will be the rectified linear unit (ReLU) function, and the output layer will use the sigmoid function. The sigmoid function is used because it is a good choice for binary classification problems. The ReLU function is used because it is a good choice for hidden layers in general, and it is the most common activation function for autoencoders [5].

Since the encoding dimension will represent the characters, it is the most important parameter of the autoencoder. To determine an appropriate encoding dimension, we may train multiple autoencoders with different encoding dimensions, and compare their performance. The performance of the autoencoder will be measured by the percentage of characters that are correctly reconstructed. The encoding dimensions to be tested are 8, 16, and 32.

4 Conclusion

Autoencoder neural networks are powerful and versatile tools that can be used for a variety of applications. It is possible to apply this technology in the field

of cryptography. As we discussed previously, the advantages of autoencoders are that they are easy to implement, and they can be trained with either small or large datasets.

References

[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] F. Q. Socasi., R. Velastegui., L. Zhinin-Vera., R. Valencia-Ramos., F. Q. Ortega-Zamorano., and O. Chang., "Digital cryptography implementation using neurocomputational model with autoencoder architecture," in *Proceedings of the 12th International Conference on Agents and Artificial Intelligence - Volume 2: ICAART,*, INSTICC, SciTePress, 2020, pp. 865–872, ISBN: 978-989-758-395-7. DOI: 10.5220/0009154908650872.

[4] 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.

[5] Y. Song, S. Hyun, and Y.-G. Cheong, "Analysis of autoencoders for network intrusion detection," *Sensors*, vol. 21, no. 13, 2021, ISSN: 1424-8220. DOI: 10.3390/s21134294.