Volume 6, Issue 3, March 2016 ISSN: 2277 128X
International Journal of Advanced Research in Computer Science and Software Engineering
Research Paper
Available online at: www.ijarcsse.com

Decryption using Neural Network and Decision Trees

Parijit Kedia
School of Computer Science and Technology,
VIT University, Tamilnadu, India

Abstract: ***There might be different techniques for decrypting an encrypted text. A system might take years to decode but what if the system learns from it and then predicts the encrypted text back. Neural Network and Decision Trees will help in learning and predicting the encrypted text respectively.***

Keywords: Decryption, Neural Network, Decision Trees, Prediction, Classification, Learning.

I. INTRODUCTION

Cryptography is the technique of hiding important message in a format that becomes very difficult for anyone to understand the context of the text. This is used to communicate between individual like spies, agents, and individuals who do not want anyone to eavesdrop on their conversation. There are people who are trying to decrypt the text using the current technologies but there are some that are unbreakable. So if the machine learns from the encrypted text and forms a pattern, then that can be used to form a decision pattern/tree for predicting the actual text.

II. LITERATURE SURVEY

One research paper shows how a text can be decrypted using neural network. It represented the amount of time that it has to perform for decryption. But in this method that is reduced reasonably since the neural network won't be able to give the correct output to be precise. In my method, decision tree will be used to remove that ambiguity. After neural network has learned the dataset provided and predicted that a particular outcome has reached, that new outcome will be given to the decision tree where which numbers should be modified so that the text is decrypted correctly and efficiently.

III. CONCEPTS USED

A. Neural Networks

Artificial Neural Network or ANN are inspired from the nervous system of the human beings, where the nodes represent the neurons which are interconnected by edges. These edges contain adaptive weights which change depending on the learning factor of the neural net algorithm.
Their function is similar to biological neural network in the way they perform and work.
These are mathematical model which is used to define a function F: X -> Y. The simple neural network has 3 layers – input layer, hidden layer, output layer.
These layers interact with the edges that connect these layers together.
The neural network has a definite patter of interconnection which defines the workflow of the input to the output. This interconnection helps in the calculation among the hidden layers.
It has learning process which means how fast the system should learn. This means that the learning factor should not be too high or too low so that the algorithm shoots from the stable position to unstable one.
The activation function helps in converging the output to the desired output i.e. the output of the neural network may be any random value but to map this random value to the desired output, an activation function is used.

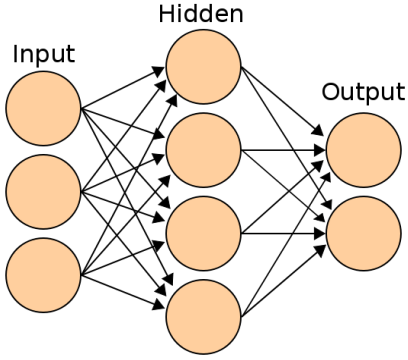

Figure 1.1 – basic neural network example

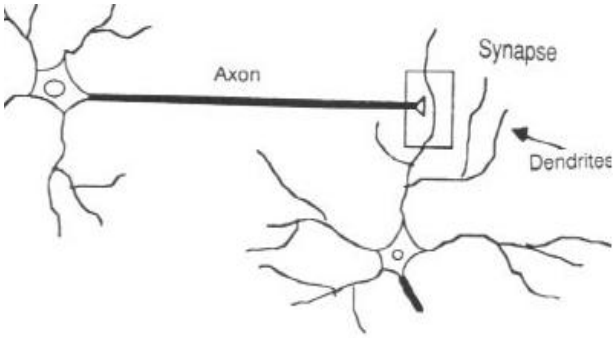

Figure 1.2 – Biological Neuron

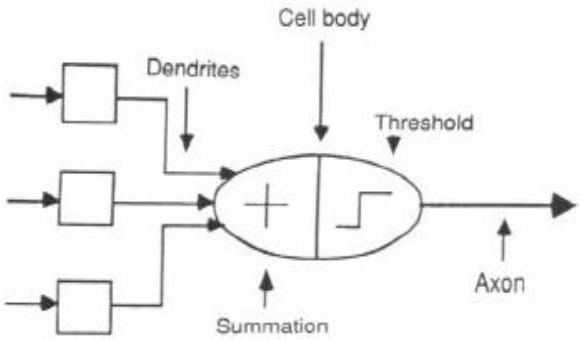

Figure 1.3 – Mathematical model

B. Decision Trees

Decision tree is a tool that uses a tree like model or graph for decision making purposes. This helps in predicting the chance of the event occurring, costs and utility.

The root node represents the main query which is subdivided into many children depending on the options available for that particular query.

Now, suppose if the query has 2 options then the root node will have 2 children and each children will represent another query. This new query is formed by redefining the main query given a particular condition. This process repeats till the leaf node where a complex query is broken down to a simple query.

The leaf nodes represents the solutions to the query. The edges represents the decisions taken. So, there is only one path from root to the leaf node and that path tells us what decisions have been taken to arrive that particular conclusion.

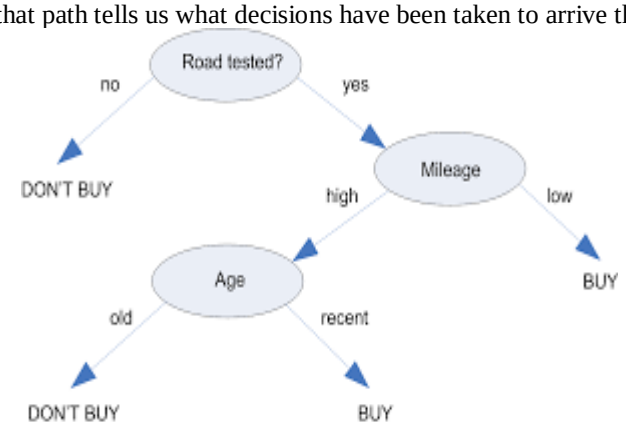

Figure 2.1 – A decision tree example

C. Input and Output of Neural Network

The input to the neural network is the encrypted text that is to be decoded. This is fed into the neural network either in bipolar or binary format. This then traverses through the hidden layer to the final output layer which is also in the bipolar or binary format (as given in the input). This is then converted back to the plain text for further process.

D. Decision Tree Learning

It uses a predictive model which maps values from the input to the predicted output. It is commonly used in data mining. The main objective is to predict the value of the target based on the input being given. It is an example of classification. It is of two types – Classification and Regression.

IV. ALGORITHM

Now we discuss our algorithm. There are phases that the algorithm works on.

PHASE I:

This step includes the encrypted text from any encryption algorithm like MD5, SHA1, bluefish, etc. This encrypted text is then used in subsequent phases.

PHASE II:

This is the decryption phase where the encrypted data is passed into the neural network for learning. The encrypted data is converted to either binary or bipolar format by some mechanism (direct conversion, bit wise conversion or by converting it after mathematical operation). The length of the binary data will be huge since the encrypted text will be of length around 20.

Thus, the binary format of the encrypted text is greater than the encrypted text thereby making the input size and the weight matrix for the neural network. This takes a bit of a computation time to learn and give a predicted output. We can increase the accuracy of the computation by increasing the hidden layer. But care should be taken so that the hidden layer is not increased to a level where the time taken to perform is increased exponentially. The computation time should be within the limits.

PHASE III:

The text obtained after learning may or may not be the correct decrypted text as needed. So, a decision tree is made with the data set where the nodes represent the decision that needs to be taken depending on the result obtained after PHASE II. This means that a decision tree is made from the huge data set. The nodes represent the values of the encrypted text and the edges represent the decision to take depending on its place value.

Firstly, we need to construct the decision tree. This is done by using various metrics option like GINI Index, Information Gain. This will be helpful since the different algorithms that are used for encryption have different mechanism and to incorporate the power of all we need to weigh out the factor of all the characters in the text like its position, relative position.

This is the pattern that will be learned from the PHASE II method. So, decisions are made such that the leaf nodes are obtained and the correct leaf nodes replace the irregularities in the decrypted text. Thus obtaining the correct result.

ASSUMPTIONS:

The data set provided is large enough. By large enough it contains almost all the passwords that are encrypted using the current technologies.

V. ANALYSIS OF ALGORITHM

1. Genetic algorithms can be used to improve the selection process since we need to find out the one with very less discrepancy. This can be solved by Genetic Algorithm since we need to select the one with high probability of occurring i.e. the best children out of all the possible outcomes. This then undergoes under mutation and fitness.
2. Decision tree learning algorithm can be improved using CART or C4.5 or MARS.

VI. CONCLUSION

By combining techniques of neural network and data mining a text can be decrypted. Even if the middle man gets the text, he won't be able to get the correct text thereby confusing the middle man

Using data mining techniques helps in easy data processing and extraction since we need to get rid of the ambiguities.

REFERENCES

[1] Wolfgang Kinzel and ldo Kanter, "Interacting neural networks and cryptography", Advances in Solid State Physics, Ed. by B. Kramer (Springer, Berlin. 2002), Vol. 42, p. 383 arXiv- cond-mat/020301 1

[2] Jacek M. Zurada, Introduction to Artificial Neural Systems.

[3] R. Metzler and W. Kinzel and 1. Kanter, Phys. Rev. E, 62, 2555 (2000).

[4] Kaniter, W. Kinzel and E. Kanter, Europhys. Lett, 57,141-147 (2002).

[5] D. R. Stinson, Cryptography: Theory and Practice (CRC Press 2002).

[6] C.P. Williams and S.H. Cleat-water, Explorations in Quantum Computing, Springer Verlag, 1998.

[7] G. K. Patra, Tahir Ali, Anil Kumar and R. P. Thangavelu, "Multiparty Secure Key Exchange Algorithm using Neural Cryptology", National Workshop On Cryptology, 2004.

[8] D. R. Stinson, Cryptography: Theory and Practice, CRC Press 1995.

[9] M. Rosen-Zvi, E. Klein, 1. Kanter and W. Kinzel, "Mutual leamning in a treepanty machine and its application to cryptography", Phys. Rev. E (2002)

[10] L. Breiman. Bagging predictors. Machine Learning, 24:123–140, 1996.

[11] N. V. Chawla, K. W. Bowyer, L. O. Hall, and W. P. Kegelmeyer. SMOTE: Synthetic minority over-sampling technique. Journal of Artificial Intelligence Research, 16:321– 357, 2002.

[12] P. Domingos. MetaCost: A general method for making classifiers cost-sensitive. In Proceedings of the ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, pages 155–164, 1999.

[13] R. E. Fan, K. W. Chang, C. J. Hsieh, X. R. Wang, and C. J. Lin. LIBLINEAR: A library for large linear classification. Journal of Machine Learning Research, 9:1871–1874, 2008.

[14] J. R. Quinlan. C4.5: Programs for machine learning. Morgan Kaufmann Publishers Inc., San Francisco, CA, 1993.

[15] J. Su and H. Zhang. A fast decision tree learning algorithm. In Proceedings of the AAAI National Conference on Artificial Intelligence, pages 500–505, Boston, MA, 2006.

