



The Random Neural Network with a Genetic Algorithm and Deep Learning Clusters in Fintech: Smart Investment

Will Serrano^(✉)

Intelligent Systems and Networks Group, Electrical and Electronic Engineering,
Imperial College London, London, UK
g.serrano11@imperial.ac.uk

Abstract. This paper presents the Random Neural Network in a Deep Learning Cluster structure with a new learning algorithm based on the genetics according to the genome model, where information is transmitted in the combination of genes rather than the genes themselves. The proposed genetic model transmits information to future generations in the network weights rather than the neurons. The innovative genetic algorithm is implanted in a complex deep learning structure that emulates the human brain: Reinforcement Learning takes fast local current decisions, Deep Learning Clusters provide identity and memory, Deep Learning Management Clusters take final strategic decisions and finally Genetic Learning transmits the information learned to future generations. This proposed structure has been applied and validated in Fintech; a Smart Investment application: an Intelligent Banker that performs Buy and Sell decisions on several Assets with an associated market and risk. Our results are promising; we have connected the human brain and genetics with Machine Learning based on the Random Neural Network model where biology; similar as Artificial Intelligence is learning gradually and continuously while adapting to the environment.

Keywords: Genetic learning · Deep Learning clusters
Reinforcement Learning · Random Neural Network · Smart Investment
Fintech

1 Introduction

Biology is gradually and continuously learning while adapting to the environment using genetic changes to generate new complex structures in organisms [1], the current structure of the organisms defines the type and level of future genetic variation that will provide a better adaption to the environment or increased reward to a goal function. Random genetic changes have more probability to be successful in organisms that change in a systematic and modular manner where the new structures acquire the same set of sub goals in different combinations; therefore they not only remember their reward evolution but also generalize goal functions to successfully adapt future environments [2]. The adaptations learned from the living organisms affect and guide evolution even though the characteristics acquired are not transmitted to the genome [3], however, its

gene functions are altered and transmitted to the new generation; this enables learning organisms to evolve much faster.

The genome is the genetic material of an organism; it consists of 23 pairs of chromosomes (1-22, X and Y) for a human cell formed of genes (approximately 21,000 in total) that code for a molecule that has a function or instruction to make proteins [4], furthermore genes are formed of base pairs (approximately 3 billion in total). The DNA is a double helix formed by the combination of only four nucleotides (cytosine [C], guanine [G], adenine [A] or thymine [T]) where each base pair consists of the combination of two nucleotides G-C and A-T. The genetic code is formed of codons, a sequence consisted of three nucleotides or three-letter words. Proteins that have similar combination of base pairs tend to have a related functionality determination of protein functions from genetic sequences [5].

Successful Machine Learning and Artificial Intelligence models have been purely based on biology emulating the structures provided by nature during the learning, adaptation and evolution when interacting with the external environment. Neural networks and deep learning are based on the brain structure which is formed of dense local clusters of same neurons performing different functions which are connected between each other with numerous very short paths and few long distance connections [6]. The brain retrieves a large amount of data obtained from the senses; analyses the material and finally selects the relevant information [7] where the cluster of neurons specialization occurs due to their adaption when learning tasks.

This paper proposes a new genetic learning algorithm on Sect. 3 based on the genome and evolution; where the information transmitted to new generations is learned when interacting and adapting to the environment using reinforcement and deep learning respectively. Information in the proposed genetic algorithm is transmitted in the network weights through the different combinations of four different nodes (C, G, A, T) rather than the value of nodes themselves where the output layer of nodes replicates the input layer as the genome. This innovative genetic algorithm is inserted in a complex deep learning structure that emulates the human brain on Sect. 4: Reinforcement Learning takes fast local current decisions, Deep Learning clusters provide identity and memory, Deep Learning Management Clusters takes final strategic decisions and finally Genetic Learning transmits the information learned to future generations. This innovative model has been applied and validated in Fintech, a Smart Investment application on Sect. 5; an Intelligent Banker that performs Buy and Sell decisions on several assets with an associated market and risk. The results shown on Sect. 6 are promising; the Intelligent Banker takes the right decisions, learns the variable asset price, makes profits on specific markets at minimum risk and finally it transmits the information learned to future generations.

2 Related Work

Artificial Neural Networks have been applied to make financial predictions. Leshno et al. [8] evaluate the bankruptcy prediction capability of several neural network models based on the firm's financial reports. Chen et al. [9] uses Artificial Neural Networks for a financial distress prediction model. Kara et al. [10] apply an Artificial Neural Network to

predict the direction of Stock Market index movement. Guresen [11] evaluates the effectiveness of neural network models in stock market predictions. Zhang et al. [12] analyse Artificial Neural Networks in bankruptcy prediction. Kohara et al. [13] investigate different ways to use prior knowledge and neural networks to improve multivariate prediction ability. Sheta et al. [14] compares Regression, Artificial Neural Networks and Support Vector Machines for predicting the S&P 500 Stock Market Price Index. Tung et al. [15] includes Artificial Neural Networks and Fuzzy Logic for market predictions. Pakdaman et al. [16] use a feed forward multilayer perceptron and an Elman recurrent Network to predict a company's stock value. Iuhasz et al. [17] create a hybrid system based on a multi Agent Architecture to analyse Stock Market behaviour to improve the profitability in a short or medium time period investment. Nicholas et al. [18] examine the use of neural networks in stock performance modelling.

Several survey papers have been published. Bahrammirzaee [19] presents a comparative survey of Artificial Intelligence Applications in Finance: Artificial Neural Networks, Expert System and hybrid intelligent systems. Coakley et al. [20] reviews the use of Artificial Neural Networks in Accounting and Finance including modeling issues and guidelines. Fadlalla et al. [21] analyses the applications of Neural Networks in Finance. Huang et al. [22] reviews the use of neural networks in finance and economics forecasting. Li et al. [23] summarize different applications or artificial intelligence technologies in several domains of business administration including finance, retail, manufacturing and management consultancy.

Machine learning has been applied to solve nonlinear models in continuous time in economics and finance by Duarte [24] and forecasting the volatility of asset prices by Stefani et al. [25]. Deep Learning has also recently incorporated in long short term memory Neural Networks for financial market predictions by Fischer et al. [26] and Hasan et al. [27].

Genetic Algorithms have been proposed as method to increase learning. Arifovic [28] analyses genetic algorithms in inflationary economies. Kim et al. [29] uses a genetic Algorithm to feature discretization in artificial neural networks for the prediction of stock market index. Ticona et al. [30] applies a hybrid model based on Genetic algorithm and Neural Networks to forecast Tax Collection. Hossain et al. [31] present a Genetic Algorithm based Deep Learning Method. Tirumala [32] and David et al. [33] review of the latest deep learning structures and evolutionary algorithms that can be used to train them.

3 The Random Neural Network Genetic Deep Learning Model

3.1 The Random Neural Network

The RNN [34–36] represents more closely how signals are transmitted in many biological neural networks where they travel as spikes or impulses, rather than as analogue signal levels. The RNN is a spiking recurrent stochastic model for neural networks. Its main analytical properties are the “product form” and the existence of the unique network steady state solution. The Random Neural network has been Genetics [37–46] (Fig. 1).

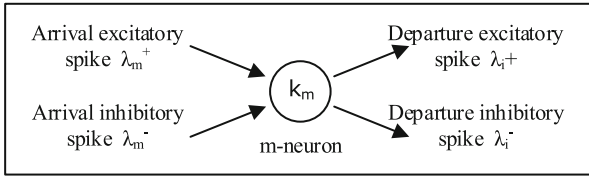


Fig. 1. The Random Neural Network

3.2 The Random Neural Network with Multiple Clusters

Deep Learning with Random Neural Networks is described by Gelenbe and Yin [47–49]. This model is based on the generalized queuing networks with triggered customer movement (G-networks) where customers are either “positive” or “negative” and customers can be moved from queues or leave the network. G-Networks are introduced by Gelenbe [50, 51]; an extension to this model is developed by Gelenbe et al. [52] where synchronised interactions of two queues could add a customer in a third queue (Fig. 2).

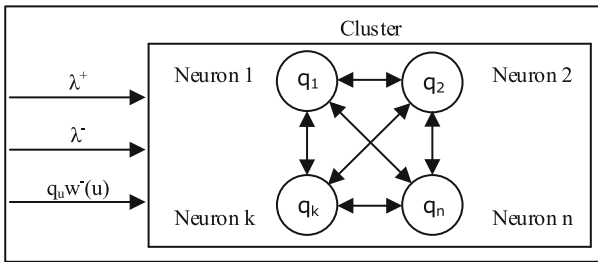


Fig. 2. Clusters of Neurons

3.3 Deep Learning Management Cluster

The Deep Learning management cluster was proposed by Serrano et al. [53–56]. It takes management decisions based on the inputs from different Deep Learning clusters.

3.4 Genetic Learning Algorithm Model

The proposed Genetic learning algorithm is based on the auto encoder presented by Gelenbe and Yin [47–49] based on two instances of the Network shown on Fig. 3, the auto encoder models the genome as it codes the replica of the organism that contains it. Network 1 is formed of U input neurons and C clusters and Network 2 has C input neurons and U clusters. The organism is represented as a set of data X which is a U vector $X \in [0, 1]^U$. The proposed Genetic learning algorithm fixes C to 4 neurons that represent the four different nucleoids G, C, A and T and it also fixes W_1 to generate 4 different types of neurons rather than random values.

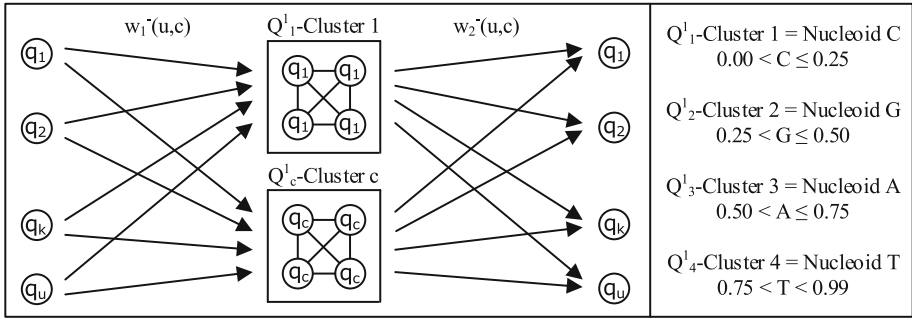


Fig. 3. Genetic Learning Algorithm

Network 1 encodes the organism, it is defined as:

- $q_1 = (q_1^1, q_2^1, \dots, q_u^1)$, a U -dimensional vector $q_1 \in [0, 1]^U$ that represents the input state q_u for neuron u ;
- W_1 is the $U \times C$ matrix of weights $w_1^-(u, c)$ from the U input neurons to the neurons in each of the C clusters;
- $Q^1 = (Q_1^1, Q_2^1, \dots, Q_c^1)$, a C -dimensional vector $Q^1 \in [0, 1]^C$ that represents state q_c for the cluster c where $Q^1 = \zeta(W_1 X)$.

Network 2 decodes the genome, as the pseudo inverse of Network 1, it is defined as:

- $q_2 = (q_1^2, q_2^2, \dots, q_c^2)$, a C -dimensional vector $q_2 \in [0, 1]^C$ that represents the input state q_c for neuron c with the same value as $Q^1 = (Q_1^1, Q_2^1, \dots, Q_c^1)$;
- W_2 is the $C \times U$ matrix of weights $w_2^-(u, c)$ from the C input neurons to the neurons in each of the U cells;
- $Q^2 = (q_1^2, q_2^2, \dots, q_u^2)$, a U -dimensional vector $Q^2 \in [0, 1]^U$ that represents the state q_u for the cell u where $Q^2 = \zeta(W_2 Q^1)$ or $Q^2 = \zeta(W_2 \zeta(XW_1))$.

The learning algorithm is the adjustment of W_1 to code the organism X into the four different neurons or nucleoids and then calculate W_2 so that resulting decoded organism Q_2 is the same as the encoded organism X :

$$\min \|X - \zeta(W_2 \zeta(XW_1))\| \text{ s.t. } W_1 \geq 0 \tag{1}$$

Following the Extreme Learning Machine on [57]; W_2 is calculated as:

$$W_2 = \text{pinv}(\zeta(XW_1))X \tag{2}$$

Where pinv is the Moore-Penrose pseudoinverse:

$$\text{pinv}(x) = (x^T x)^{-1} x^T$$

4 Smart Investment Model

The Smart Investment model, called “GoldAI Sachs”, combines three different learnings: Reinforcement Learning, Deep Learning and Genetic Learning. “GoldAI Sachs” is formed of clusters of Intelligent Bankers that take local fast local binary decisions “Buy or Sell” on a specific assets based on Reinforcement Learning through the interactions and adaptations with the environment where the Reward is the profit made. Each Asset Banker has an associated Deep Learning cluster that memorizes asset identity such as price and reward properties. Asset bankers are dynamically clustered to different properties such as investment reward, risk or market type and managed by a Market Banker Deep Learning Management Cluster that selects the best performing Asset Bankers. Finally, a CEO Banker Deep Learning Management Cluster manages the different Bankers and takes the final investment decisions based on the Market Reward and associated Risk prioritizing markets that generate more reward at a lower Risk as every banker would do. This approach enables decisions based on shared information where Intelligent Bankers work collaborative to achieve a bigger reward (Fig. 4).

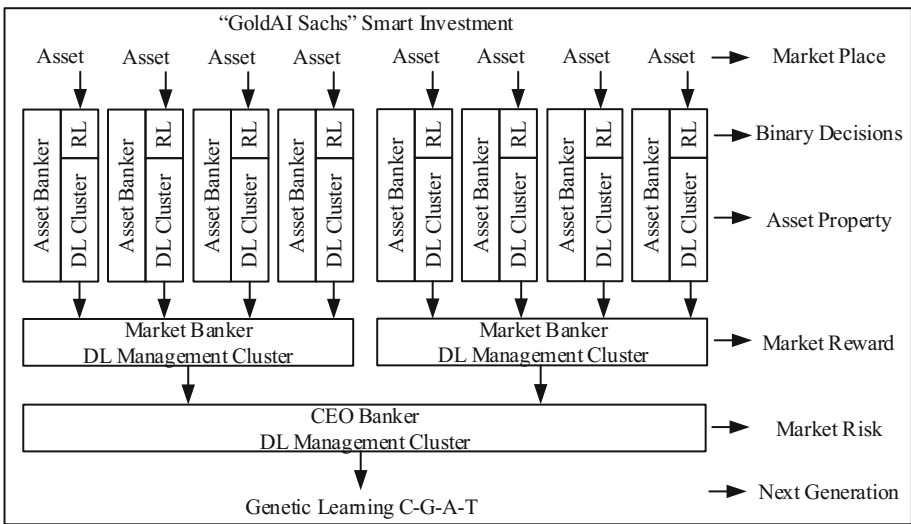


Fig. 4. “GoldAI Sachs” Smart Investment Model

4.1 Asset Banker Reinforcement Learning

The Reinforcement Learning algorithm is used to take fast binary investment decisions “Buy or Sell”, it is based on Cognitive Packet Network presented by Gelenbe [11–15]. The Intelligent Banker is formed of two interconnected neurons “ q_0 or Buy” and “ q_1 or Sell” where the investment decision is taken according to the neuron that has the

maximum potential. The state q_0 and q_1 is the probability that it is excited [11–15], these quantities satisfy the following system of non linear equations:

$$q_0 = \frac{\lambda^+(0)}{r(0) + \lambda(0)} \quad q_1 = \frac{\lambda^+(1)}{r(1) + \lambda(1)} \quad (3)$$

where:

$$\begin{aligned} \lambda^+(0) &= q_1 w_{10}^+ + \Lambda_0 & \lambda^+(1) &= q_0 w_{01}^+ + \Lambda_1 \\ \lambda^-(0) &= q_1 w_{10}^- + \lambda_0 & \lambda^-(1) &= q_0 w_{01}^- + \lambda_1 \\ r(0) &= w_{01}^+ + w_{01}^- & r(1) &= w_{10}^+ + w_{10}^- \end{aligned} \quad (4)$$

On the above equations, w_{ij}^+ is the rate at which neuron i transmits excitation spikes to neuron j and w_{ij}^- is the rate at which neuron i transmits inhibitory spikes to neuron j in both situations when neuron i is excited. Λ_i and λ_i are the rates of external excitatory and inhibitory signals respectively (Fig. 5).

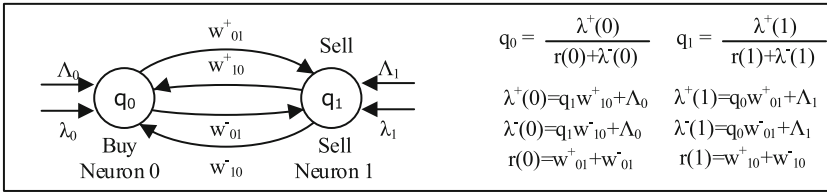


Fig. 5. Asset Banker Reinforcement Learning

The Reward R is based on the economic profit that the Asset Bankers achieve with the decisions they make, successive measured values of the R are denoted by R_l , $l = 1, 2, \dots$ these are used to compute the Predicted Reward:

$$PR_l = \alpha PR_{l-1} + (1 - \alpha)R_l \quad (5)$$

where α represents the investment reward memory.

If the observed measured Reward is greater than the associated Predicted Reward; Reinforcement Learning rewards the decision taken by increasing the network weight that point to it, otherwise; it penalises it.

5 Experimental Results

“GoldAI Sachs” is evaluated with eight different assets to assess the adaptability and performance of our proposed Smart Investment solution for eleven days. The assets are split into the Bond Market with low risk and slow reward and the Derivative Market with high risk and fast reward. Experiments are carried with very reduced memory $\alpha = 0.1$ where the Reinforcement Learning is first initialized with a Buy Decision (Fig. 6).

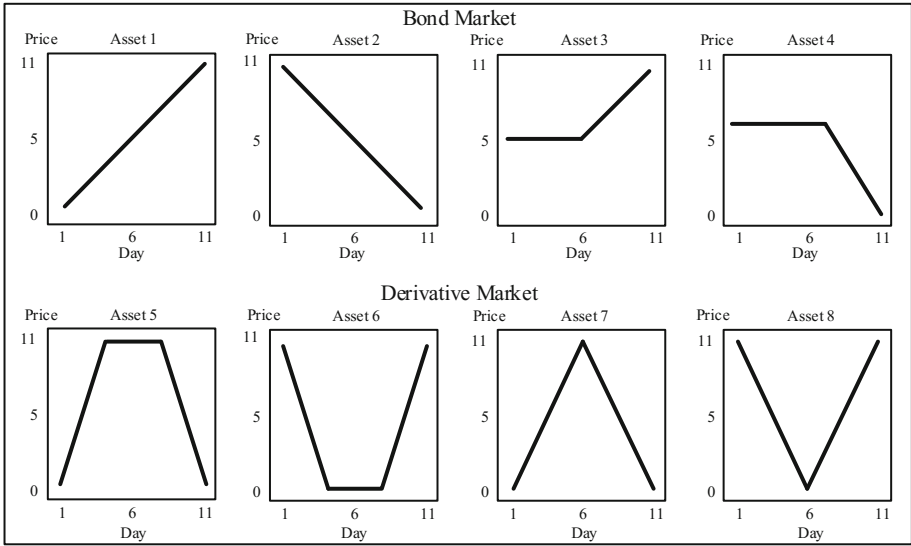


Fig. 6. “GoldAI Sachs” Smart Investment Model assets

5.1 Asset Banker Reinforcement Learning Validation

Table 1 represents the Profit that each Asset Banker makes when buying or selling 100 Assets for 11 days with the Maximum Profit, the number of winning decisions against the losing ones and the number of buy decisions against the sell.

The Profit made in assets that start downwards such as Asset 2, Asset 4, Asset 6 and Asset 8 is worse than the upwards ones because the Asset Bankers are initialized with a buy decision. The Reinforcement Learning Algorithm adapts very quickly to variable asset prices.

Table 1. Asset Banker Reinforcement Learning Validation

Asset	Profit	Maximum Profit	Win	Loss	Buy	Sell
1	1000	1000	10	0	10	0
2	800	1000	9	1	1	9
3	600	600	6	0	10	0
4	300	500	4	1	6	4
5	2000	2000	8	0	4	6
6	1200	2000	6	2	4	6
7	1600	2000	9	1	6	4
8	800	2000	7	3	4	6

5.2 Market Banker Deep Learning Management Cluster Validation

The profits the Market Bankers can make are shown in Tables 2 and 3. The Market Bankers take market decisions rather than individual asset decisions from the Asset Bankers. Market Bankers invests 400 assets which is the combination of the four Asset Bankers purchasing power.

Table 2. Bond Market Banker Profits

Day	Total Asset Banker	Bond Market	I	Maximum Asset	Maximum Market	I
2	0	400	400.00%	200	400	100.00%
3	200	400	100.00%	200	400	100.00%
4	200	400	100.00%	200	400	100.00%
5	200	400	100.00%	200	400	100.00%
6	300	400	33.33%	300	400	33.33%
7	200	400	100.00%	400	400	0.00%
8	400	400	0.00%	400	400	0.00%
9	400	400	0.00%	400	400	0.00%
10	400	400	0.00%	400	400	0.00%
11	400	400	0.00%	400	400	0.00%
Total	2700	4000	48.15%	3100	4000	29.03%

Table 3. Derivative Market Banker Profits

Day	Total Asset Banker	Derivative Market	I	Maximum Asset	Maximum Market	I
2	0	800.0	400.00%	800	800.0	400.00%
3	1000	1200	20.00%	1000	1200	20.00%
4	1000	1200	20.00%	1000	1200	20.00%
5	800	800	0.00%	800	800	0.00%
6	400	0	-100.00%	400	0	-100.00%
7	-400	-800	100.00%	400	800	100.00%
8	0	800	800%	800	800	0.00%
9	1000	1200	20.00%	1000	1200	20.00%
10	1000	1200	20.00%	1000	1200	20.00%
11	800	800	0.00%	800	800	0.00%
Total	5600	7200	28.57%	8000	8800	10.00%

5.3 CEO Banker Deep Learning Management Cluster Validation

Table 4 represents the CEO Banker, “AI Morgan” profits at different Risks ratios with a total of investment of 800 assets. A risk value $\beta = 0.2$ represents 640 assets in the Bond Market and 160 is the Derivative Market whereas a risk value $\beta = 0.8$ is 160 assets in the Bond Market and 640 in the Derivative Market respectively.

Table 4. CEO Banker Profits

Day	Risk $\beta = 0.2$			Risk $\beta = 0.5$			Risk $\beta = 0.8$			Max profit
	B	D	Total	B	D	Total	B	D	Total	
2	640	320	960	400	800	1200	160	1280	1440	1440
3	640	480	1120	400	1200	1600	160	1920	2080	2080
4	640	480	1120	400	1200	1600	160	1920	2080	2080
5	640	320	960	400	800	2800	160	1280	1440	1440
6	640	0	640	400	0	400	160	0	160	640
7	640	-320	320	400	-800	-400	160	-1280	-1120	320
8	640	320	960	400	800	1200	160	1280	1440	1440
9	640	480	1120	400	1200	1600	160	1920	2080	2080
10	640	480	1120	400	1200	1600	160	1920	2080	2080
11	640	640	1280	400	800	1200	160	1280	1440	1440
Total	6400	2880	9280	4000	7200	11200	1600	11520	13120	15040

5.4 Genetic Algorithm Validation

The Genetic Algorithm validation for the four different Nucleoids (C, G, A, T) during the 11 different days is shown in Table 5 with the Genetic Algorithm Error.

Table 5. Genetic Algorithm Validation

Day	Error	Nucleoid-C	Nucleoid-G	Nucleoid-A	Nucleoid-T
2	3.05*10E-31	0.2048	0.3893	0.6295	0.9268
3	5.85*10E-31	0.2026	0.3861	0.6263	0.9259
4	6.78*10E-32	0.2025	0.3859	0.6262	0.9259
5	1.17*10E-31	0.2029	0.3865	0.6267	0.9260
6	4.44*10E-31	0.2033	0.3870	0.6272	0.9262
7	1.29*10E-31	0.2049	0.3894	0.6296	0.9269
8	3.61*10E-31	0.2031	0.3868	0.6271	0.9261
9	2.96*10E-31	0.2021	0.3852	0.6255	0.9257
10	6.90*10E-31	0.2020	0.3851	0.6254	0.9256
11	1.36*10E-31	0.2023	0.3856	0.6259	0.9258

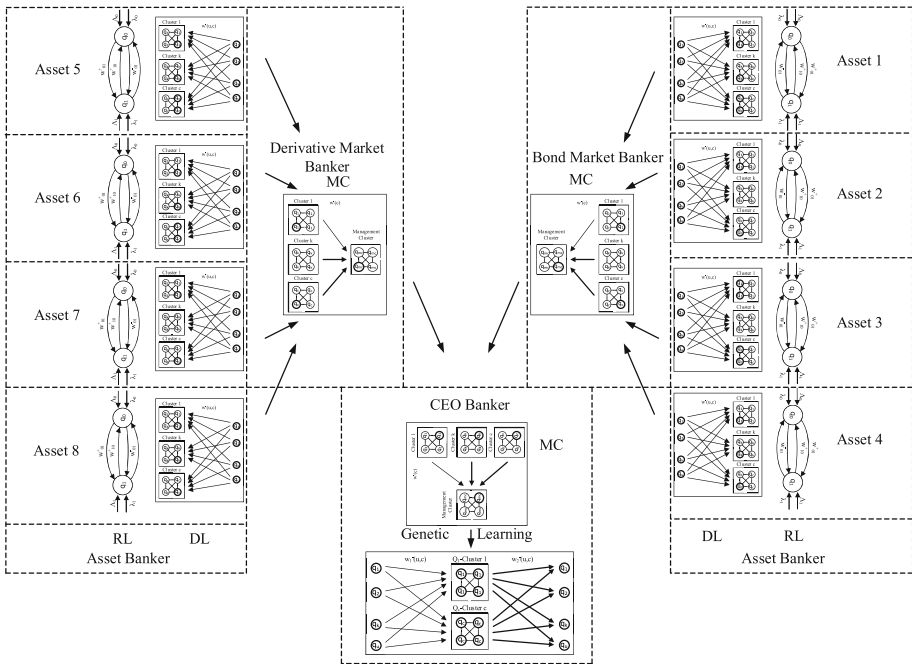
6 Conclusions

This paper has presented a new learning Genetic Algorithm based on the Genome where the information is transmitted in the network weights rather than the neurons. The algorithm has been incremented in an Smart Investment model that simulates the human brain with reinforcement learning for fast decisions, deep learning to memorize properties to create asset identity, deep learning management clusters to make global decisions and genetic to transmit learning into future generations.

In the Smart Investor Model, “GoldAI Sachs” Asset Banker Reinforcement Learning Algorithm takes the right investment decisions; with great adaptability to asset price changes whereas Asset Banker Deep Learning provides asset properties and identity. Market Bankers success to increase the profit by selecting the best performing Asset Bankers and the CEO Banker, “AI Morgan” increases the profits considering the associated market risks, prioritizing low risk investment decision at equal profit. Genetic learning algorithm has a minimum error and it exactly codes and encodes the CEO Banker, “AI Morgan”.

Future work will validate our model in a Fintech cryptocurrency environment with real market values. In addition the relevance of memory in investment with its optimum value will be analyzed.

Appendix: Smart Investment Model - Neural Schematic



References

1. Kirschner, M., Gerhart, J.: *The Plausibility of Life Resolving Darwin's Dilemma*. Yale University Press, New Haven (2005)
2. Parter, M., Kashtan, N., Alon, U.: Facilitated Variation: How Evolution Learns from Past Environments To Generalize to New Environments. Department of Molecular Cell Biology, Weizmann Institute of Science
3. Hinton, G., Nowlan, S.: How learning can guide evolution. In: *Adaptive Individuals in Evolving Populations*, pp. 447–454 (1996)
4. Pellegrini, M., Marcotte, E., Thompson, M., Eisenberg, D., Yeates, T.: Assigning protein functions by comparative genome analysis: protein phylogenetic profiles. *Proc. Natl. Acad. Sci. USA* **96**, 4285–4288 (1999)
5. Suzuki, M.: A framework for the DNA protein recognition code of the probe helix in transcription factors: the chemical and stereo chemical rules. *Structure* **2**(4), 317–326 (1994)
6. Smith, D., Bullmore, E.: Small-World brain networks. *Neuroscientist* **12**, 512–523 (2007)
7. Sporns, O., Chialvo, D., Kaiser, M., Hilgetag, C.: Organization, development and function of complex brain networks. *Trends Cogn. Sci.* **8**(9), 418–425 (2004)
8. Leshno, M., Spector, Y.: Neural network prediction analysis: the bankruptcy case. *Neurocomputing* **10**(2), 125–147 (1996)
9. Chen, W., Du, Y.: Using neural networks and data mining techniques for the financial distress prediction model. *Expert Syst. Appl.* **36**, 4075–4086 (2009)
10. Kara, Y., Acar, M., Kaan, Ö.: Predicting direction of stock price index movement using artificial neural networks and support vector machines: the sample of the Istanbul Stock Exchange. *Expert Syst. Appl.* **38**, 5311–5319 (2011)
11. Guresen, E., Kayakutlu, G., Daim, T.U.: Using artificial neural network models in stock market index prediction. *Expert Syst. Appl.* **38**, 10389–10397 (2011)
12. Zhang, G., Hu, M., Patuwo, B., Indro, D.: Artificial neural networks in bankruptcy prediction: general framework and cross-validation analysis. *Eur. J. Oper. Res.* **116**, 16–32 (1999)
13. Kohara, K., Ishikawa, T., Fukuhara, Y., Nakamura, Y.: Stock price prediction using prior knowledge and neural networks. *Intell. Syst. Account. Fin. Manag.* **6**, 11–22 (1997)
14. Sheta, A., Ahmed, S., Faris, H.: A comparison between regression, artificial neural networks and support vector machines for predicting stock market index. *Int. J. Adv. Res. Artif. Intell.* **4**(7), 55–63 (2015)
15. Khuat, T., Le, M.: An application of artificial neural networks and fuzzy logic on the stock price prediction problem. *Int. J. Inform. Visual.* **1**(2), 40–49 (2017)
16. Naeini, M., Taremian, H., Hashemi, H.: Stock market value prediction using neural networks. In: *International Conference on Computer Information Systems and Industrial Management Applications*, pp. 132–136 (2010)
17. Iuhasz, G., Tirea, M., Negru, V.: Neural network predictions of stock price fluctuations. In: *International Symposium on Symbolic and Numeric Algorithms for Scientific Computing*, pp. 505–512 (2012)
18. Nicholas, A., Zaprani, A., Francis, G.: Stock performance modeling using neural networks: a comparative study with regression models. *Neural Netw.* **7**(2), 375–388 (1994)
19. Bahrammirzaee, A.: A comparative survey of artificial intelligence applications in finance: artificial neural networks, expert system and hybrid intelligent systems. *Neural Comput. Appl.* **19**, 1165–1195 (2010)
20. Coakley, J., Brown, C.: Artificial neural networks in accounting and finance: modeling issues. *Int. J. Intell. Syst. Account. Fin. Manag.* **9**, 119–144 (2000)

21. Fadlalla, A., Lin, C.: An analysis of the applications of neural networks in finance. *Interfaces* **31**(4), 112–122 (2001)
22. Huang, W., Lai, K., Nakamori, Y., Wang, S., Yu, L.: Neural networks in finance and economics forecasting. *Int. J. Inf. Technol. Decis. Mak.* **6**(1), 113–140 (2007)
23. Li, Y., Jiang, W., Yang, L., Wu, T.: On neural networks and learning systems for business computing. *Neurocomputing* **275**, 1150–1159 (2018)
24. Duarte, V.: *Macro, Finance, and Macro Finance: Solving Nonlinear Models in Continuous Time with Machine Learning*. Massachusetts Institute of Technology, Sloan School of Management, pp. 1–27 (2017)
25. Stefani, J., Caelen, O., Hattab, D., Bontempi, G.: Machine learning for multi-step ahead forecasting of volatility proxies. In: *Workshop on Mining Data for Financial Applications*, pp. 1–12 (2017)
26. Fischer, T., Krauss, C.: Deep learning with long short-term memory networks for financial market predictions. *FAU discussion Papers in Economics*, vol. 11, pp. 1–32 (2017)
27. Hasan, A., Kalıpsız, O., Akyokuş, S.: Predicting financial market in big data: deep learning. In: *International Conference on Computer Science and Engineering*, pp. 510–515 (2017)
28. Arifovic, J.: Genetic algorithms and inflationary economies. *J. Monetary Econ.* **36**, 219–243 (1995)
29. Kim, K., Han, I.: Genetic algorithms approach to feature discretization in artificial neural networks for the prediction of stock price index. *Expert Syst. Appl.* **19**, 125–132 (2000)
30. Ticona, W., Figueiredo, K., Vellasco, M.: Hybrid model based on genetic algorithms and neural networks to forecast tax collection: application using endogenous and exogenous variables. In: *International Conference on Electronics, Electrical Engineering and Computing*, pp. 1–4 (2017)
31. Hossain, D., Capi, G.: Genetic algorithm based deep learning parameters tuning for robot object recognition and grasping. *Int. Sch. Sci. Res. Innov.* **11**(3), 629–633 (2017)
32. Tirumala, S.: Implementation of evolutionary algorithms for deep architectures. In: *Artificial Intelligence and Cognition*, pp. 164–171 (2014)
33. David, O., Greental, I.: Genetic algorithms for evolving deep neural networks. In: *ACM Genetic and Evolutionary Computation Conference*, pp. 1451–1452 (2014)
34. Gelenbe, E.: Random neural networks with negative and positive signals and product form solution. *Neural Comput.* **1**, 502–510 (1989)
35. Gelenbe, E.: Learning in the recurrent random neural network. *Neural Comput.* **5**, 154–164 (1993)
36. Gelenbe, E.: G-Networks with triggered customer movement. *J. Appl. Probab.* **30**, 742–748 (1993)
37. Gelenbe, E.: A class of genetic algorithms with analytical solution. *Robot. Auton. Syst.* **22** (1), 59–64 (1997)
38. Gelenbe, E.: Steady-state solution of probabilistic gene regulatory networks. *Phys. Rev. E* **76** (1), 031903 (2007). Also in *Virtual Journal of Biological Physics Research*, 15 September 2007
39. Gelenbe, E., Liu, P., Laine, J.: Genetic algorithms for route discovery. *IEEE Trans. Syst. Man Cybern. B* **36**(6), 1247–1254 (2006)
40. Gelenbe, E.: Dealing with software viruses: a biological paradigm. *Inf. Secur. Tech. Rep.* **12**, 242–250 (2007)
41. Gelenbe, E.: Network of interacting synthetic molecules in equilibrium. *Proc. Royal Soc. A (Math. Phys. Sci.)* **464**, 2219–2228 (2008)
42. Kim, H., Gelenbe, E.: Anomaly detection in gene expression via stochastic models of gene regulatory networks. *BMC Genom.* **10**(Suppl 3), S26 (2009). <https://doi.org/10.1186/1471-2164-10-S3-S26>

43. Kim, H., Gelenbe, E.: Stochastic gene expression modeling with hill function for switch-like gene responses. *IEEE/ACM Trans. Comput. Biol. Bioinform.* **9**(4), 973–979 (2012). <https://doi.org/10.1109/tcbb.2011.153>. ISSN 1545-5963
44. Kim, H., Gelenbe, E.: Reconstruction of large-scale gene regulatory networks using Bayesian model averaging. *IEEE Trans. NanoBiosci.* **11**(3), 259–265 (2012). <https://doi.org/10.1109/tmb.2012.221>
45. Gelenbe, E.: Natural computation. *Comput. J.* **55**(7), 848–851 (2012)
46. Kim, H., Park, T., Gelenbe, E.: Identifying disease candidate genes via large-scale gene network analysis. *IJDMB* **10**(2), 175–188 (2014)
47. Gelenbe, E., Yin, Y.: Deep learning with random neural networks. In: *International Joint Conference on Neural Networks*, pp. 1633–1638 (2016)
48. Yin, Y., Gelenbe, E.: Deep Learning in Multi-Layer Architectures of Dense Nuclei. *CoRR abs/1609.07160*, pp. 1–10 (2016)
49. Yin, Y., Gelenbe, E.: Single-cell based random neural network for deep learning. In: *International Joint Conference on Neural Networks*, pp. 86–93 (2017)
50. Gelenbe, E.: G-Networks: a unifying model for neural nets and queueing networks. In: *Modelling Analysis and Simulation of Computer and Telecommunications Systems*, pp. 3–8 (1993)
51. Fourneau, J., Gelenbe, E., Suros, R.: G-Networks with multiple class negative and positive customers. In: *Modelling Analysis and Simulation of Computer and Telecommunications Systems*, pp. 30–34 (1994)
52. Gelenbe, E., Timotheou, S.: Random neural networks with synchronized interactions. *Neural Comput.* **20–9**, 2308–2324 (2008)
53. Serrano, W., Gelenbe, E.: An intelligent internet search assistant based on the random neural network. In: Iliadis, L., Maglogiannis, I. (eds.) *AIAI 2016. IAICT*, vol. 475, pp. 141–153. Springer, Cham (2016). https://doi.org/10.1007/978-3-319-44944-9_13
54. Serrano, W.: A big data intelligent search assistant based on the random neural network. In: Angelov, P., Manolopoulos, Y., Iliadis, L., Roy, A., Vellasco, M. (eds.) *INNS 2016. AISC*, vol. 529, pp. 254–261. Springer, Cham (2017). https://doi.org/10.1007/978-3-319-47898-2_26
55. Serrano, W., Gelenbe, E.: Intelligent search with deep learning clusters. In: *Intelligent Systems Conference*, pp. 254–267 (2017)
56. Serrano, W., Gelenbe, E.: The deep learning random neural network with a management cluster. In: Czarnowski, I., Howlett, Robert J., Jain, Lakhmi C. (eds.) *IDT 2017. SIST*, vol. 73, pp. 185–195. Springer, Cham (2018). https://doi.org/10.1007/978-3-319-59424-8_17
57. Kasun, L., Zhou, H., Huang, G.: Representational learning with extreme learning machine for big data. *IEEE Intell. Syst.* **28**(6), 31–34 (2013)