HALMSTAD UNIVERSITY

Brain Emotional Learning-Inspired Models

Mahboobeh Parsapoor

LicentiateThesis| Halmstad University Dissertations no. 8

Abstract

In this thesis the mammalian nervous system and mammalian brain have been used as inspiration to develop a computational intelligence model based on the neural structure of fear conditioning and to extend the structure of the previous proposed amygdala-orbitofrontal model. The proposed model can be seen as a framework for developing general computational intelligence based on the emotional system instead of traditional models on the rational system of the human brain. The suggested model can be considered a new data driven model and is referred to as the brain emotional learning-inspired model (BELIM). Structurally, a BELIM consists of four main parts to mimic those parts of the brain's emotional system that are responsible for activating the fear response. In this thesis the model is initially investigated for prediction and classification. The performance has been evaluated using various benchmark data sets from prediction applications, e.g. sunspot numbers from solar activity prediction, auroral electroject (AE) index from geomagnetic storms prediction and Henon map, Lorenz time series. In most of these cases, the model was tested for both long-term and short-term prediction. The performance of BELIM has also been evaluated for classification, by classifying binary and multiclass benchmark data sets.

Acknowledgements

My emotions are attuned in such a way as to make me happy with the simple pleasures in my life. I started working on this research on 11 May 2011 and now, on 14 May 2014, and with great joy, I am writing the acknowledgment part of my licentiate thesis.

First and foremost, I would like to express my sincere thanks to my main supervisor, Professor Bertil Svensson for all of help that he has given me in completing this thesis. I remember very clearly that day that I was trying to convince myself to give up working on the emotional learning aspect because of the numerous negative comments I had received. A couple of minutes after entertaining these thoughts, he talked with me about the possibility of writing my licentiate thesis on this research topic. I felt very happy to have an opportunity to do so and it has been an honour to be one of his research students.

No words can be adequate to show the degree of my appreciation to my co-supervisor, Dr. Urban Bilstrup for giving me more than enough attention and support during these three years. I vividly remember the first time that we discussed the topic of cognitive radio networks and I outlined my thoughts about developing an emotion-based model in this context, and I recall his broad smile that gave me confidence to explain my idea. Urban's kindness, sense of humour and intelligence always impressed me. I am truly proud to have been one of his first PhD students. I shall never forget his considerable efforts, and the time that he devoted in encouraging, tutoring and counseling me both emotionally and rationally in order to make me a smart and creative researcher.

I am especially grateful for the members of my support committee, Professor Antanas Verikas and Dr. Emil Nilsson, for their time, interest, and helpful suggestions and comments. I appreciate Dr. Jörgen Carlsson and professor, Magnus Jonsson the current and former CC-lab (Computing and Communication laboratory) leaders for accepting me as a member of CC-lab and for financing me. I would like to say thanks to the senior researchers of CC-Lab: Professor Tony Larsson, Professor Tomas Nordström, Professor Walid Taha, Professor Mohammad Reza Mousavi, Professor Alexey Vinel, Dr. Kristina Kunert and Dr. Annette Böhem for their valuable feedback during my oral presentation.

Being a researcher at IDE has provided me with a unique opportunity to meet many kind people and it has been a pleasure to work alongside them as colleagues and benefit from their friendship. I would specifically like to mention Per-Arne Wiberg whose enthusiasm and insights have inspired my research work. I would appreciate Stefan Byttner (Director of doctoral education) and Eva Nestius (research administrator) for helping me out with different administrative matters. I wish to thank all my wonderful friends: Erik, Wagner, Saeed, Adam, Nam, Sid, Süleyman and Essayas for cheering up me whenever I was depressed and lifting my mood.

It's my fortune to gratefully acknowledge the opponent of this thesis, Professor Christian Balkenius, for his time and his precious comments on this thesis.

I would like to mention my sincerest gratitude to the late Professor Caro Lucas for his inspiration and kind support in making me familiar with brain emotional learning, and for supervising me throughout my research process in my home country.

Lastly, I wish to extend my appreciation to my parents for all their love and endless support. My mother has raised me with a love of science and encouraged me in all my pursuits and inspired me to follow my dreams. A thousand thanks to 'ma rose' for training my emotional brain in how to offer unconditioned love.

This work was mainly funded by the Knowledge Foundation (KK-stiftelsen) through the profile Centre for Research on Embedded Systems (CERES) at Halmstad University and MaC^{2}WiN project.

Contents

1 Introduction

A challenge within the computational intelligence research community is to develop and evaluate bio-inspired data-driven models. Earlier studies in developing bio-inspired data-driven methods are related to developing an artificial neural network (ANN) by mimicking the mammalian nervous system [1][2]. Another well-known bio-inspired model is the neuro-fuzzy method, which combines an adaptive network and a fuzzy inference system that has been inspired by the human decision-making process [3]. Neural networks and neuro-fuzzy methods have a high generalization capability to model the nonlinear and chaotic behavior of complex systems [1]-[13] and they have wide applications in different areas, e.g. economic, healthcare, communication, space weather etc. [2], [3].

This thesis suggests the development of a bio-inspired data-driven model by combining the neural structure of fear conditioning and the amygdala-orbitofrontal system. Fear conditioning is a mechanism by which one learns fearful stimuli to predict aversive events and it is common to men and animals. The neural structure of fear conditioning was proposed by Ledoux based on laboratory experiments on mice. Ledoux presented the neural structure, emphasizing the role of the amygdala in activating a fear response [14]. Another important part of the brain is the orbitofrontal cortex, which has a main role in processing a fearful stimulus, in particular in preventing a fear response. The amygdala-orbitofrontal system is a computational model of emotional learning and consists of four parts, sensory cortex, thalamous, amygdala and orbitofrontal [15], to copy the corresponding parts of the brain that are responsible for emotional learning.

The suggested bio-inspired data-driven model is referred to as the Brain Emotional Learning Inspired Model (BELIM). The outer structure of BELIM consists of four main modules that are similar to the parts of the amygdala-orbitofrontal system. The internal structure of each part, its input and its interaction with other parts have been imitated by the interaction between the corresponding parts of the neural structure of fear conditioning. The function of each part of this model has been defined by using an appropriate adaptive network to imitate the role of the corresponding part involved in fear conditioning. The overall function of a BELIM would be implemented by merging the functions of the adaptive networks in an appropriate way.

1.1 Research Backgrounds

Psychologists, neuroscientists and philosophers have made efforts to analyze the brain's emotional system and define different hypotheses and theories (e.g., appraisal, cognitive, central etc.) to describe emotional behavior [14]-[20]. Cognitive neuroscientists have also tried to describe the neural system underlying the emotional process. One of the earlier works is the 'Papez circuit' (See Fig. 1.1) that was proposed by Jamez Papez and includes the 'hypothalamus, anterior thalamus, cingulate gyrus and hippocampus' [17].

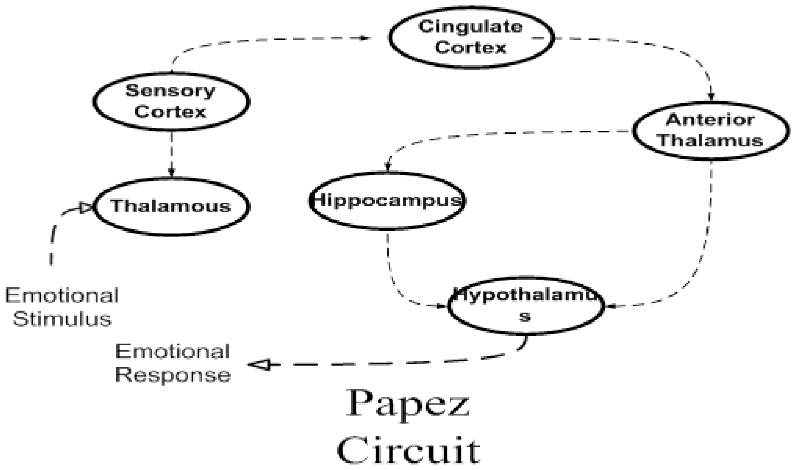


Figure 1.1: The components of Papez Circuit and their connections.

Cannon suggested another structure, emphasizing the role of the hypothalamus and the cingulate cortex in emotional processing [17][18]. MacLean suggested the limbic system theory to describe the regions of the brain which are responsible for emotional processing. The so-called limbic system theory was proposed, based on Cannon's structure and the Papez circuit emphasizing the role of the limbic system (i.e. a group of the brain regions including the thalamus and amygdala) for emotional processing. The limbic system includes the hippocampus, amygdala, thalamus and sensory cortex [14]-[20].

The earlier neuroscience studies concentrated on localizing the emotional system in the brain; however, the limbic system theory has been rejected on the basis of laboratory experiments on mice, which have verified that different parts of the brain are responsible for different emotional behaviors [20]. Thus, neuroscientists started to do research and find neural circuits underlying specific emotional behaviors. One well-known neural circuit is based on fear conditioning, which means learning fearful stimuli to predict aversive events, and was proposed by Ledoux [14]. Fear conditioning defines why mammals not only predict the occurrence of fearful stimuli but also learn to avoid from the origination of fearful stimuli and the importance of the fearful stimuli received. Ledoux presented a neural structure for fear conditioning by focusing on the role of the amygdala in activating a fear response [14].

The neural structure of fear conditioning was proposed by Ledoux on the basis of laboratory experiments on mice. This structure is based on the amygdala, i.e. components that have important roles in fear conditioning. Figure 1.2 displays how the amygdala and its components operate when receiving a fearful stimulus. As the diagram indicates, the amygdala consists of three parts, lateral, basolateral and central parts. The lateral part is 'the gateway to the amygdala [14]. The lateral part receives the emotional stimulus, processes it and sends it to basolateral and central parts. The central is responsible for the provision of an emotional response.

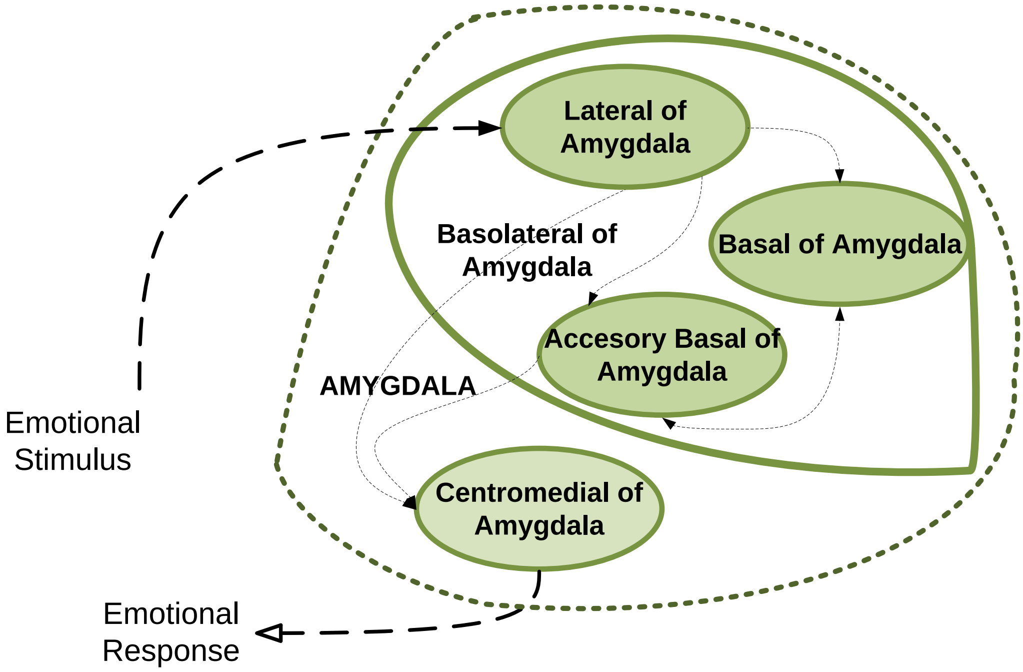


Figure 1.2: The components of the amygdala and their connections in fear conditioning.

Another important part that has important roles in emotional processing is the orbitofrontal cortex, which is located very close to the amygdala [16].

The thalamus, sensory cortex, amygdala and orbitofrontal cortex have roles in fear conditioning. The general role of these regions in the emotional processing can be summarized as follows [21]-[30]:

1) The thalamus is the entrance gate of emotional stimuli. It determines the effective value of the stimulus to be passed to the amygdala and the sensory cortex.

2) The sensory cortex is a part of the sensory area of the brain and is responsible for analysis and processing of received signals.

3) The amygdala is the central part of the emotional system of mammals and has a principal role in emotional learning [14]-[30]. The amygdala consists of several parts with different functional roles and it connects through these parts to other regions of the brain (e.g. the insular cortex, orbital cortex and frontal lobe). The amygdala (see Figure 1.3) has connections to the thalamus, orbitofrontal cortex and hypothalamus and participates during emotional learning in reacting to emotional stimuli, storing emotional responses, evaluating positive and negative reinforcement, and learning the association between unconditioned and conditioned stimuli. It also has a role in predicting the association between stimuli and future reinforcement and forming an association between neutral stimuli and emotionally charged stimuli [21]-[30].

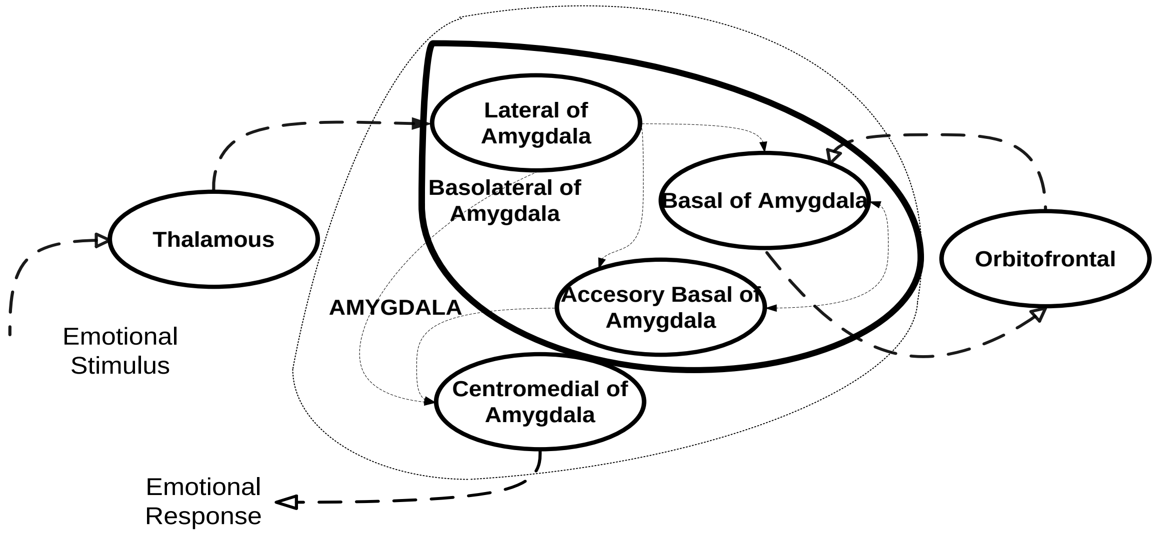


Figure 1.3: The parts of amygdala and their connections with the thalamus and orbitofrontal cortex.

The two main parts of the amygdala are the basolateral part (the largest portion of the amygdala) and the centeromedial part. The basolateral part has a bidirectional link to the insular cortex and orbital cortex and performs the main role in mediating memory consolidation and providing the primary response. The basolateral part is divided into three parts: the lateral, basal and accessory basal. The lateral is the part through which stimuli enter the amygdala. The lateral region not only passes the stimuli to other regions but also memorizes them to form the stimulus–response association. This part also takes some roles in spreading the information received to other parts of amygdala, forming the association between the conditioned and unconditioned stimuli, inhabiting and reflecting the external stimuli and memorizing emotional experiences. The basal and accessory basal parts participate in mediating the contextual conditioning. The centeromedial part, which is the main output for the basaloteral part, is divided into the central and medial parts. It is responsible for the hormonal aspects of emotional reactions or for mediating the expression of the emotional responses [20]-[27].

4) The orbitofrontal cortex is located close to the amygdala and has a bidirectional connection to the amygdala. This part plays roles in processing stimuli, decoding the primary reinforcement, representing the negative reinforcement and learning the stimulus–reinforcement association. It also evaluates and corrects reward and punishment, selects goals, makes decisions for a quick response to punishment and prevents inappropriate responses of the amygdala. The orbitofrontal cortex also encompasses two parts: the medial and the lateral. The medial part forms and memorizes reinforcement–stimulus associations and also has a role in providing responses and monitoring them, whereas the lateral part evaluates the response and provides punishment [27]-[30].

Computational models of emotion [31]-[34] are computer-based models that have been developed to simulate different aspects of the emotional system, e.g. an amygdala-orbitofrontal system to simulate emotional learning of the amygdala. The computational model of emotions can be considered as a way to validate the suggested theories, e.g. appraisal theory, central theory and cognitive theory and the underlying neural structure of the theories. A good example of a computational model is the Cathexis model [35], which was inspired by the human emotional decision-making process. The Cathexis model imitated the connection between the prefrontal lobe and other regions of the brain that are responsible for emotional response and decision making. The amygdala-orbitofrontal system [14] is another computational model with a simple structure that imitates the interaction between those parts of the limbic systems that have roles in emotional learning. The amygdala-orbitofrontal subsystem consists of four parts that are known as the thalamus, sensory cortex, amygdala and orbitofrontal cortex and imitates the role of the corresponding parts in emotional learning and making associations between the conditioned and the unconditioned stimuli (see Figure.1.4).

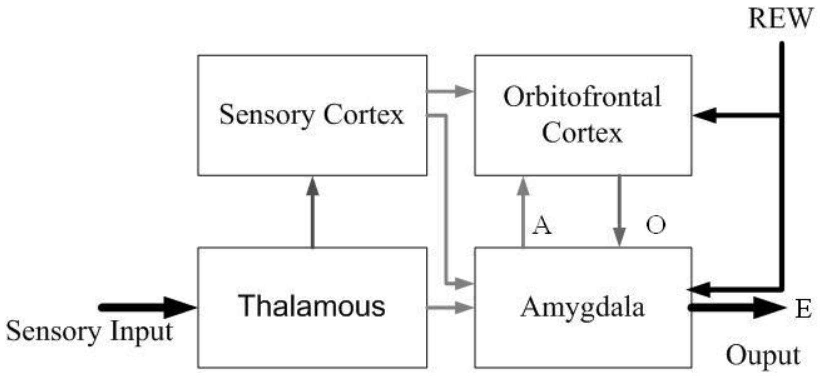


Figure 1.4: The components of the amygdala-orbitofrontal system.

Figure 1.4 shows how the parts of the amygdala-orbitofrontal model interact with each other to form the association between the conditioned and the unconditioned stimuli [15]. The amygdala-orbitofrontal model [14] has a simple structure to simulate emotional learning; it has been used as a foundation for several emotion-based machine learning approaches. The hippocampus-neocortex model [31] is another computational model that aims to predict time-series. A modified version of this model was also proposed, the amygdala hippocampus model [32], which combines associative memory and the amygdala-orbitofrontal system.

The computational models of emotional learning have been the foundation for developing emotion-based decision making, emotion-based controllers and emotion-based machine learning approaches. A good example is the emotional agent that was proposed using the Cathexis model [35]; the agent is a type of emotion-based decision making model that reacts to the stimuli being received.

The first practical implementation of an emotion-based controller is BELBIC (Brain Emotional Learning-Based Intelligent Controller) [36]. It was developed on the basis of an amygdala-orbitofrontal system. The BELBIC has been successfully employed for a number of applications: controlling heating and air conditioning [37], aerospace launch vehicles [38], intelligent washing machines [39] and trajectory tracking of stepper motors [40]. The results of applying emotion-based controllers have shown that they have the capability to overcome uncertainty and complexity issues of control applications. Specifically, the BELBIC has been proven to outperform other controllers in terms of simplicity, reliability and stability [37]-[40].

The Emotion-based machine-learning approaches [41]-[47] have also been developed by imitating the structure of computational models of emotions, in particular the amygdala-orbitofrontal model that was proposed by Moren and Balkenius [14]. Most studies have been carried out to develop new neural networks and have been applied for different benchmark data sets, e.g., auroral electroject (AE) index, solar activity and chaotic systems.

1.2 Research Questions

In the computational intelligence community, many efforts have been made to develop bio-inspired data-driven models, e.g. neural networks, neuro-fuzzy models and genetic algorithms, for prediction, classification and optimization applications. Developing a high performance bio-inspired model involves: choosing a suitable biological model, selecting an appropriate theory to explain the biological system and using simple mathematical formulations to implement the selected theory.

The research questions of this thesis can be divided into three categories: investigating the structural aspect of the emotion-based prediction model, investigating the functional aspect of the emotion-based prediction models and, finally, evaluating the performance of the suggested model.

To define the structural aspect of the emotion-based model, the following questions must be investigated in order to find out how the structure of an emotion-based prediction model can be developed; which computational model can be used as a foundation for the brain emotional learning-inspired model?; how the selected computational model can be extended to a data-driven model?; and finally which cognitive theory and neural structure of the emotional system have been utilized to describe the selected computational model?

To implement the functional aspect of the emotion-based model, the following questions must be answered: How can the functionality of the selected computational model be extended to develop the function of the emotion-based model?; What changes can be made to improve the performance of the data-driven models developed?; How may the learning algorithm and the adjusting rules be defined?

The performance of the emotion-based model is evaluated by examining it on the benchmark data sets and comparing the results obtained with the results of other methods. Before making the performance evaluation, it is essential to answer these questions: which data-driven models can be used to compare the results of the model?; on which benchmark data sets can the model be evaluated?

1.3 Research Contributions

The main contribution of this thesis is the development of an emotion-based model that can be categorized as a computational intelligence model. It has been verified that this computational intelligence model can be utilized for prediction and classification. This research specifically contributes by:

1) Presenting an extension of the amygdala-orbitofrontal model that is a simple computational model of emotional learning.

2) Using the neural structure of fear conditioning to change the structure of the amygdala-orbitofrontal.

3) Presenting new internal structures to the parts that mimic the amygdala and the orbitofrontal cortex of the brain.

4) Implementing the functional aspects of the emotion-based prediction model using adaptive networks and defining how the adaptive networks can be merged to provide the overall output of the emotion-based model.

5) Introducing an additional learning phase to increase the generalization capability of the emotion-based model.

6) Providing comparative results by extensively testing BELIMs to predict the benchmark data sets from chaotic systems, e.g. solar activity and geomagnetic storms. The results obtained have been compared with the results of powerful models such as an adaptive neuro-fuzzy inference system and locally linear model tree learning algorithm (LoLiMoT).

1.4 List of Publications

Publications related to the scope of this thesis are listed below. (Main authorship is indicated by underlining.)

1.4.1 Appended Papers

Paper A:

M. Parsapoor and U. Bilstrup, "Brain Emotional Learning Based Fuzzy Inference System (BELFIS) for Solar Activity Forecasting," *Tools with Artificial Intelligence (ICTAI), 2012 IEEE 24th International Conference on*, pp.532-539, Nov.2012. DOI: 10.1109/ICTAI.2012.78.

Paper B:

M.Parsapoor and U. Bilstrup, "Chaotic Time Series Prediction Using Brain Emotional Learning Based Recurrent Fuzzy System (BELRFS)," *International Journal of Reasoning-based Intelligent Systems(IJRIS)*, vol.5,no.2, pp.113 – 126.2013. DOI: 10.1504/IJRIS.2013.057273.

Paper C:

M. Parsapoor, U. Bilstrup, "An emotional learning-inspired ensemble classifier (ELiEC)," *Computer Science and Information Systems (FedCSIS), 2013 Federated Conference on*, pp.137-141,2013.

1.4.2 Related Publications

M. Parsapoor, U. Bilstrup, "Neuro-fuzzy models, BELRFS and LoLiMoT, for prediction of chaotic time series," *Innovations in Intelligent Systems and Applications (INISTA)*, pp.1-5, 2012. DOI: 10.1109/INISTA.2012.6247025.

M. Parsapoor, U. Bilstrup, "Brain emotional learning based fuzzy inference system (Modified using radial basis function)," *Digital Information Management (ICDIM)*, *2013 Eighth International Conference on*, pp.206-211, 2013. DOI: 10.1109/ICDIM.2013.6693994.

M. Parsapoor, C. Lucas, S. Setayeshi, "Reinforcement _recurrent fuzzy rule based system based on brain emotional learning structure to predict the complexity dynamic system," *Digital Information Management (ICDIM), ICDIM 2008. Third International Conference on*, pp.25_32, 2008. DOI: 10.1109/ICDIM.2008.4746712.

2 Brain Emotional Learning-Inspired Models

Brain Emotional Learning-Inspired Models (BELIMs) present a new category of data-driven models that have inspired structures from the neural structure of fear conditioning. The general structure of BELIMs is an extension of the amygdala-orbitofrontal system by adding new internal parts that have been inspired by the neural structure of the brain emotional system, in particular the circuit of fear conditioning, which emphasizes the role of the amygdala and its components. To implement the internal structure of each part of a BELIM, the adaptive networks are assigned. This chapter presents the general structure, algorithm and the adjusting rules in BELIMs.

2.1 General Structure

The general structure of BELIMs consists of four main parts, referred to as the THalamus (TH), sensory CorteX (CX), AMYGdala (AMYG), and ORBItofrontal cortex (ORBI), respectively. These parts imitate the connection of those four regions, the thalamus, sensory cortex, amygdala and orbitofrontal cortex, that have roles in fear conditioning. Figure 2.1 describes the general structure of BELIMs and depicts how the input and output of each part has been defined and how these parts are connected to each other.

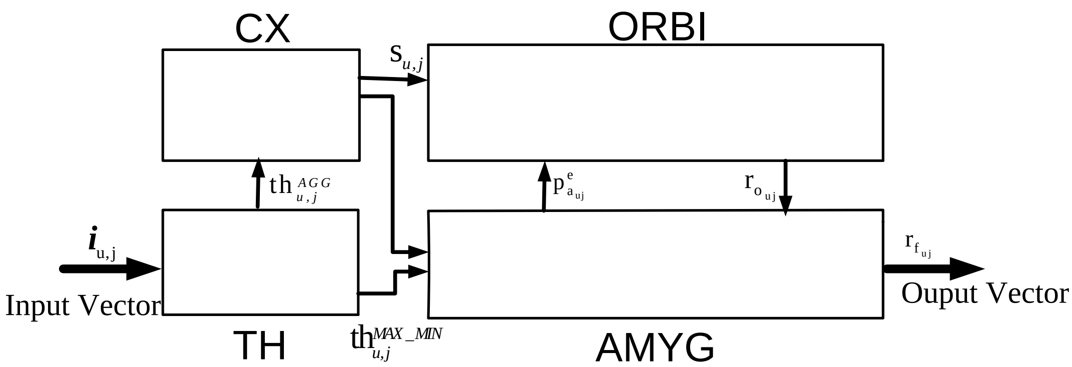
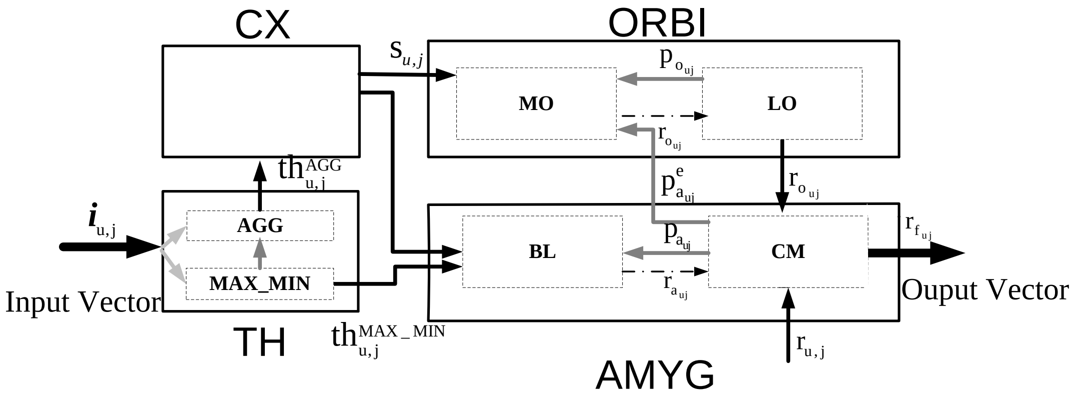
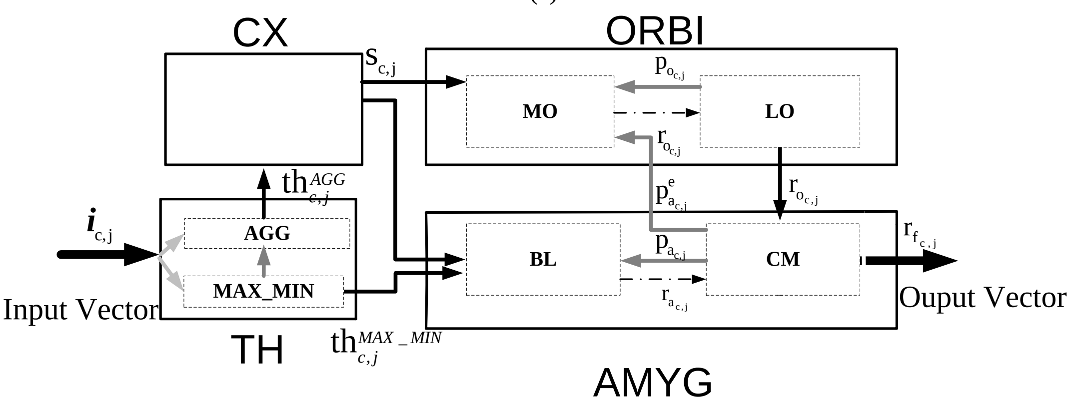


Figure 2.1: The general structure of a BELIM.

To imitate the roles and the connections between the thalamus, amygdala and orbitofrontal in more detail, in BELIM, the TH, AMYG and ORBI parts have been further divided into two internal parts. Figure 2.2 depicts the internal parts of the TH, the AMYG and the ORBI and describes the inputs and outputs of these subparts. The TH is divided into two subparts: the MAX_MIN and the AGG. As was mentioned, the AMYG imitates the amygdala regions (Lateral, Basal and Accessory Basal of the amygdala and their connections), so the AMYG is divided into two parts: BL (corresponds to the set of the Basal and Lateral of the amygdala) and CM (corresponds to the set of Accessory Basal and CentroMedial of the amygdala). The ORBI also mimics the role of the orbitofrontal and consists of two sub-parts: MO (corresponds to Medial of Orbitofrontal) and LO (corresponds to Lateral of Orbitofrontal). The input and output of MO are the expected punishment and the secondary output, while the input and the output of LO are the secondary output and the punishment, respectively. There is a bidirectional connection between the CM and ORBI to exchange the AMYG's expected punishment and ORBI's response.



(a)



(b)

Figure 2.2: The details of the architecture of BELIM showing the structure of each part and its connection to other parts. (a) An input from the training set enters the BELIM. (b) An input from the test set enters the BELIM.

The connection and the inputs and the outputs of each part during the first learning phase can be described as follows. To distinguish the inputs and outputs of each part in the first learning phase, the subscript u has been used. First, $\mathbf{i}_{u,j}$ the j^{th} input vector from $\mathbf{I}_u = \{\mathbf{i}_{u,j}\}_{j=1}^{N_u}$ (taking the assumption that the number of training samples is equal to N_u) enters the TH, which provides two outputs, $\mathbf{th}^{Max\_Min}_{u,j}$ and $\mathbf{th}^{AGG}_{u,j}$, that are the outputs of the MAX_MIN and the AGG, respectively. They are sent to the AMYG and the CX. The CX provides $\mathbf{s}_{u,j}$ as its output and sends it to both the AMYG and the ORBI. Receiving $\mathbf{th}^{Max\_Min}_{u,j}$ and $\mathbf{s}_{u,j}$, the AMYG provides the primary output, $r_{a_{u,j}}$, and expected punishment, $p^{e}_{a_{u,j}}$, that is sent to the ORBI (the subscript a has been used to show the outputs of AMYG). In more detail, the BL of AMYG receives both $\mathbf{th}^{Max\_Min}_{u,j}$, $\mathbf{s}_{u,j}$ and provides the primary output, $r_{a_{u,j}}$, that is sent to the CM, which is responsible for providing the reinforcement signal, i.e. the expected punishment, $p^{e}_{a_{u,j}}$. Receiving $\mathbf{s}_{u,j}$ and

$p^e_{a_{u,j}}$, the MO of ORBI provides the secondary output, $r_{o_{u,j}}$, and sends it to the LO of ORBI (the subscript o has been used to show the outputs of ORBI) and the CM of AMYG. The CM of AMYG is responsible for providing the final output, $r_{f_{u,j}}$ (the subscript f has been used to show the final outputs).

There is a slight difference in the connection and the inputs and the outputs of each part during the second learning phase (see Figure 2.2. (b)) that has been described as the following. To distinguish the inputs and outputs of each part in this learning phase, the subscript c has been used. First, $\mathbf{i}_{c,j}$ the j^{th} input vector from $\mathbf{I}_c = \{\mathbf{i}_{c,j}\}_{j=1}^{N_c}$ (taking the assumption that the number of training samples is equal to N_c) enters the TH, which provides two outputs, $\mathbf{th}_{c,j}^{Max\_Min}$ and $\mathbf{th}_{c,j}^{AGG}$, that are the output of the MAX_MIN and the AGG, respectively. They are sent to the AMYG and the CX. The CX, provides $\mathbf{s}_{c,j}$ as its output and sends it to both the AMYG and the ORBI. Receiving $\mathbf{th}_{c,j}^{Max\_Min}$ and $\mathbf{s}_{c,j}$, the AMYG provides the primary output, $r_{a_{c,j}}$, and expected punishment, $p^e_{a_{c,j}}$, that is sent to the ORBI. In more detail, the BL of AMYG receives both $\mathbf{th}_{c,j}^{Max\_Min}$, $\mathbf{s}_{c,j}$ and provides the primary output, $r_{o_{c,j}}$, that is sent to the CM, which is responsible for providing the reinforcement signal, i.e. the expected punishment, $p^e_{a_{c,j}}$. Receiving $\mathbf{s}_{c,j}$ and $p^e_{a_{c,j}}$, the MO of ORBI provides the secondary output, $r_{o_{c,j}}$, and sends it to the LO of ORBI and the CM of AMYG. The CM of AMYG is responsible for providing the final output, $r_{f_{c,j}}$.

2.2 General Algorithm

In order to adjust the model parameters of BELIMs, all types of BELIMs follow a general learning algorithm that is a combination of two well-known learning methods: Steepest Descent (SD) [1][2] and Least Square Estimator (LSE) [1]-[3]. The learning algorithm of BELIMs consists of two phases: the first learning phase and the second learning phase. The first learning phase is a supervised learning algorithm, while the second learning phase is an unsupervised learning algorithm.

2.2.1 First Learning Phase

The first learning phase begins when the input is chosen from the training sets, $\mathbf{i}_{u,j} \in \mathbf{I}_u = \{\mathbf{i}_{u,j}\}_{j=1}^{N_u}$. In this phase, a hybrid learning algorithm [3], which is a combination of SD and LSE, is used to adjust the model's parameters, linear and nonlinear learning parameters, which are related to AMYG and ORBI. The nonlinear parameters of AMYG and ORBI are determined using two vectors, $\mathbf{b}_o$ and $\mathbf{b}_a$. Note that the two subscripts, a and o, are used to distinguish the parameters of AMYG and ORBI, respectively. The SD-based learning algorithms are applied to minimize the two loss functions, $f(\mathbf{p}_{a_u})$ and $f(\mathbf{p}_{o_u})$, which have been defined on the basis of $\mathbf{p}_{a_u}$ and $\mathbf{p}_{o_u}$ as equations (2.1) and (2.2). Equations (2.3) and (2.4) are the SD-based learning rules used to calculate the derivatives of the loss functions, with respect to $\mathbf{b}_a$ and $\mathbf{b}_o$. The parameter t denotes the current time, $\mathbf{b}_{a_u}^t$ and $\mathbf{b}_{o_u}^t$ denote the current values of learning parameters, where $\nabla\mathbf{b}_a^t$ and $\nabla\mathbf{b}_o^t$ are the gradients of the loss functions to the parameters $\mathbf{b}_a^t$ and $\mathbf{b}_o^t$, as in equations (2.5) and (2.6). Two learning rates, η_a^t and η_o^t, are defined as functions of $\mathbf{p}_{a_u}$ and $\mathbf{p}_{o_u}$. Note that ORBI and AMYG have their own loss functions to separately adjust their own learning parameters $\mathbf{b}_{a_u}$ and $\mathbf{b}_{o_u}$.

$$f(\mathbf{p}_{a_u}) = 1/\mathbf{p}_{a_u}^{2} \tag{2.1}$$

$$f(\mathbf{p}_{o_u}) = 1/\mathbf{p}_{o_u}^{2} \tag{2.2}$$

$$\mathbf{b}_a^{t+1} = \mathbf{b}_a^t - \eta_a^t \times \nabla\mathbf{b}_a^t \tag{2.3}$$

$$\mathbf{b}_o^{t+1} = \mathbf{b}_o^t - \eta_o^t \times \nabla\mathbf{b}_o^t \tag{2.4}$$

$$\nabla\mathbf{b}_a^t = \frac{\partial f(\mathbf{p}_{a_u})}{\partial \mathbf{b}_a} \tag{2.5}$$

$$\nabla\mathbf{b}_o^t = \frac{\partial f(\mathbf{p}_{o_u})}{\partial \mathbf{b}_o} \tag{2.6}$$

The LSE is used to update the linear parameters that have been used to define the punishment functions. As was discussed earlier, both the AMYG and ORBI have their own punishment functions. Equation (2.7) defines the general punishment function.

$$p_{u,j} = w_1 r_{f_{u,j}} + w_2 r_{a_{u,j}} + w_3 r_{o_{u,j}} + w_4 r_{u,j} + w_5 \tag{2.7}$$

The parameter $r_{u,j}$ is denoted the target output, $\mathbf{i}_{u,j}$, taking the assumption that $\{(\mathbf{i}_{u,j}, r_{u,j})\}_{j=1}^{N_u}$ is the training data set. Different punishment functions could be defined using specific values of $\mathbf{W}=[w_1,w_2,w_3,w_4,w_5]$. For example, the punishment function of AMYG as $p_{a_{u,j}}$ is defined by using $[0,w_2,0,w_4,w_5]$ and the expected punishment function of AMYG, $p^e_{a_{u,j}}$, is defined using $[0,w_2,0,w_4,0]$. The punishment function of ORBI as $p_{o_{u,j}}$ is defined by using $[0,w_2,0,w_4,w_5]$.

During the first learning phase, the learning parameters, linear and nonlinear parameters, are updated by using one of the following methods:

- All parameters can be adjusted using SD.
- The nonlinear parameters can be adjusted using SD and the initial values of linear parameters can be adjusted using LSE.
- The nonlinear parameters can be adjusted using SD and the linear parameters can be adjusted using LSE.

Certainly, these methods differ in terms of time complexity and prediction accuracy and a 'trade-off' between high accuracy and low computational time must be considered in choosing a feasible method. The batch mode or online mode of each method can be considered for the first learning phase.

2.2.2 Second Learning Phase

The second learning phase begins when the input is chosen from the test data sets, $\mathbf{i}_{c,j} \in \mathbf{I}_c = \{\mathbf{i}_{c,j}\}_{j=1}^{N_c}$. During the second learning phase, the nonlinear parameters of AMYG and ORBI are updated by using SD algorithms that minimize the loss functions, which are defined on the basis of the punishment functions $\mathbf{p}_{a_c}$ and $\mathbf{p}_{o_c}$. Note that in this phase, $\mathbf{p}_{a_c}$, is equal to $\{w_1 r_{f\,c,j} + w_2 r_{a\,c,j} + w_4 r_{o\,c,j}\}_{j=1}^{N_c}$. Similar to the first learning phase, the nonlinear parameters, $\mathbf{b}_o$ and $\mathbf{b}_a$, are adjusted using SD methods that calculate the derivatives of the loss functions, $f(\mathbf{p}_{a_c})$ and $f(\mathbf{p}_{o_c})$, with respect to $\mathbf{b}_a$ and $\mathbf{b}_o$. As was discussed earlier, gradient descent-based learning algorithms have been used to find the optimal values of the nonlinear learning parameters, i.e. minimize the loss functions which have been defined on the basis of the punishment functions.

Note that the learning algorithms of BELIM are mainly based on the SD, which is a simple method for finding a minimum value. In addition, SD has a linear memory complexity and computational time complexity. Nevertheless, this method has some disadvantages: it converges slowly and it is possible that it gets stuck in a local minimum.

3 Functional Aspect of BELIMs

BELIMs differ from other data-driven models (e.g. neural network and neuro-fuzzy models) in terms of structure, function and learning algorithms. As was mentioned previously, a BELIM consists of several parts and the connection between these parts has been inspired by the neural structure of fear conditioning. Each part can be described using an adaptive network. The function of each part is implemented by the underlying adaptive network. The overall function of a BELIM depends on the type of adaptive networks that are applied. Different types of BELIMs have been developed, for example, Brain Emotional Learning Fuzzy Inference System, BELFIS, using feed forward adaptive networks or Brain Emotional Learning Recurrent Fuzzy Inference System, BELRFS, merging feed forward and recurrent adaptive networks. The learning algorithm of a BELIM has been defined as a hybrid learning algorithm to adjust the learning parameters of the adaptive networks. This section explains how the functional aspects of types of BELIMs can be instantiated using different types of adaptive networks. This section also illustrates how BELIMs formulate the input-output mapping using adaptive networks.

3.1 Brain Emotional Learning Fuzzy Inference System (BELFIS)

One type of BELIM is the Brain Emotional Learning Fuzzy Inference System (BELFIS).

3.1.1 The Underlying Data-driven Model

The structure of an adaptive network consists of a number of nodes that are connected by directional links. The nodes of the adaptive network can be classified into circle and square nodes. A circle node has a function without adjustable parameters; in contrast, the square nodes have been defined by a function with the adjustable parameters. The learning parameters of an adaptive network are a combination of linear and nonlinear parameters and can be adjusted by using a learning algorithm.

Figure 3.1 depicts a simple adaptive neuro-fuzzy inference system (ANFIS) receiving a two-dimensional input vector, $\mathbf{i}_{u,j} = \{i_{u,j1}, i_{u,j2}\}_{j=1}^{N_u}$. An adaptive network can be adapted to the Sugeno fuzzy inference system or the Mamdani fuzzy inference system [3].

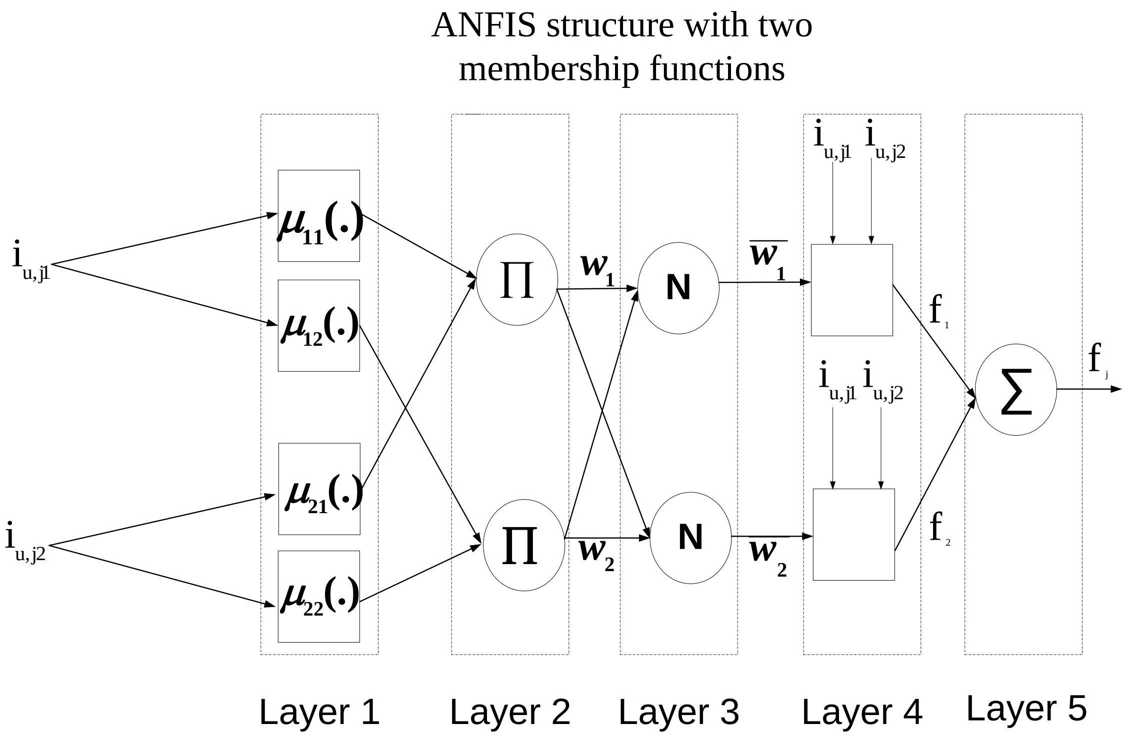


Figure 3.1.The third type of ANFIS with two rules.

The following steps explain the function of each layer of the adaptive network of Figure 3.1 with a two-dimensional input vector, $\mathbf{i}_{u,j} = \{i_{u,j1}, i_{u,j2}\}_{j=1}^{N_u}$ [3].

Layer 1: This layer consists of four square nodes, which are known as adaptive nodes. Each adaptive node can be assigned by a Gaussian or a bell-shaped function; the function is denoted by $\mu(.)$. In general, the first layer of an adaptive network has an R-dimensional input vector such as $\mathbf{i}_{u,j} = \{i_{u,j1}, ..., i_{u,jR}\}$ and m square nodes for each dimension of the input vector; thus this layer could consist of $m \times R$ square nodes. Equations (3.1) and (3.2) calculate a Gaussian or a bell-shaped function for the k^{th} node of the l^{th} dimension. The parameters c_{kl}, σ_{kl} are the center and sigma parameters of the Gaussian function, while a_{kl}, c_{kl}, b_{kl} are the learning parameters of the bell-shaped function.

$$\mu_{kl}(i_{u,jl}) = \exp\left(-\frac{1}{2}\frac{(i_{u,jl} - c_{kl})^2}{\sigma_{kl}^2}\right) \tag{3.1}$$

$$\mu_{kl}(i_{u,jl}) = \frac{1}{1 + \left|\dfrac{i_{u,jl} - c_{kl}}{a_{kl}}\right|^{2b_{kl}}} \tag{3.2}$$

Layer 2: This layer has two circular nodes that are labelled with $\prod$. The outputs of the first layers are the inputs of the nodes of this layer, and the outputs of these nodes are the multiplication of their inputs. In a general case, this layer could have $k_a = m^R$ circular nodes; assuming that the $k_2{}^{th}$ node receives the outputs of the second square node of each dimension. The output of this node is calculated as equation (3.3), where R is the dimension of an input vector.

$$w_{k_2 l}(\mathbf{i}_{u,j}) = \prod_{l=1}^{R} \mu_{2l}(i_{u,jl}) \tag{3.3}$$

Layer 3: This layer has two circle nodes with the normalization functions; each node is labelled N. In the general case, this layer could have $k_a = m^R$ circle nodes. Assuming that the $k_2{}^{th}$ node receives all $w_{k_2 l}$ s from the previous layer, the output of this node is calculated as equation (3.4), where R is the dimension of the input vector.

$$\bar{w}_{k_2 l}(\mathbf{i}_{u,j}) = \frac{\prod_{l=1}^{R} \mu_{2l}(i_{u,jl})}{\sum_{k_2=1}^{k_a} w_{k_2 l}(\mathbf{i}_{u,j})} \tag{3.4}$$

Layer 4: This layer has two square nodes. In the general case, the fourth layer could have $k_a = m^R$ square nodes; the function of $k_2{}^{th}$ node is calculated as f_{k_2}, equation (3.5). The parameters of the nodes of this layer are linear; each node receives the set of linear parameters as $\{q_{k_2 l}\}_{l=1}^{R}, q_{k_2 R+1}$.

$$f_{k_2}(\mathbf{i}_{u,j}) = \sum_{l=1}^{R} q_{k_2 l} i_{u,jl} + q_{k_2(R+1)} \tag{3.5}$$

Layer 5: The fifth layer has a single node (circle) that calculates the summation of its input vector, $\{f_{k_2}\}_{k_2=1}^{k_a}$. The output of $k_2{}^{th}$ node is f_k and has an important role to produce the final output F_j, equation (3.6).

$$F_j(\mathbf{i}_{u,j}) = \sum_{k_2=1}^{k_a} \bar{w}_{k_2}(\mathbf{i}_{u,j}) f_{k_2}(\mathbf{i}_{u,j}) \tag{3.6}$$

3.1.2 BELFIS in the First Learning Phase

The first learning phase begins when an input vector that is chosen from the training set, $\mathbf{i}_{u,j} \in \mathbf{I}_u = \{\mathbf{i}_{u,j}\}_{j=1}^{N_u}$, enters BELFIS; each part of BELFIS provides the corresponding output using the following steps:

1) The input vector $\mathbf{i}_{u,j}$ is fed to the TH, which is implemented using two adaptive networks. The output of the first adaptive network, the MAX_MIN, denoted $\mathbf{th}_{u,j}^{Max\_Min}$, is calculated using equation (3.7). It is a vector of the highest and the lowest values of the input vector with R dimensions (R is the dimension of $\mathbf{i}_{u,j}$; in Figure 3.1 the dimension of the input is equal to two).

The output of the second adaptive network is $\mathbf{th}_{u,j}^{AGG}$, which is calculated according to equation (3.8); it is the output of the AGG, the second part of TH, and is equal to $\mathbf{i}_{u,j}$ and is fed to the CX.

$$\mathbf{th}_{u,j}^{Max-Min} = [\mathrm{Max}(\mathbf{i}_{u,j}), \mathrm{Min}(\mathbf{i}_{u,j})] \tag{3.7}$$

$$\mathbf{th}_{u,j}^{AGG} = \mathbf{i}_{u,j} \tag{3.8}$$

2) The $\mathbf{th}_{u,j}^{AGG}$ is sent to the CX, which is implemented by a pre-trained neural network with one layer of linear functions. The number of neurons of the hidden layer is equal to R and the hidden layer weights can be represented by an $R \times R$ identity matrix. The function of the CX is to provide $\mathbf{s}_{u,j}$, which is sent to AMYG and ORBI. It should be noted that $\mathbf{i}_{u,j}$ and $\mathbf{s}_{u,j}$ have the same entity; however, they originate from different parts.

3) Both $\mathbf{s}_{u,j}$ and $\mathbf{th}_{u,j}^{Max-Min}$ are sent to AMYG, whose main role is to provide the primary and final outputs and is divided into two parts: BL and CM. The function of these parts is according to: BL is responsible for the provision of the primary output of AMYG and BL's function is implemented by using an adaptive network with five layers. Its output, $r_{a_{u,j}}$, is calculated according to equation (3.9), where F is a function, equation (3.6). Considering a structure similar to the adaptive network of Figure (3.1), the function of BL can be explained as equation (3.9).

$$r_{a_{u,j}} = F\left(\left[s_{u,j}, th_{u,j}\right]\right) \tag{3.9}$$

The primary output, $r_{a_{u,j}}$, is sent to the CM which is responsible for providing the expected punishment, $p^{e}_{a_{u,j}}$, and the final output, $r_{f_{u,j}}$. Before explaining the function of the CM during the first learning phase, the next step describes how the ORBI performs its function to provide the secondary output, $r_{o_{u,j}}$.

4) The MO of ORBI consists of a five-layer adaptive network whose output is calculated as $r_{oj} = F(\mathbf{s}_{u,j})$ and is sent to CM. Note that F is a function according to (3.6).

5) The CM also consists of an adaptive network with six layers; the output of the first five layers is calculated as $r_{f_{u,j}} = F\left(\left[r_{a_{u,j}}, r_{o_{u,j}}\right]\right)$.

The sixth layer consists of two circle nodes with the summation functions, providing $p^{e}_{a_{u,j}}$ and $p_{a_{u,j}}$. The former, $p^{e}_{a_{u,j}}$, represents the expected punishment and is formulated by equation

(3.10). The latter is the punishment, $p_{a_{u,j}}$, and is formulated as equation (3.11). The parameters $[w_1, w_2, w_3, w_4, w_5]$ are the corresponding weights.

$$p^e_{a_{u,j}} = w_2 r_{a_{u,j}} + w_4 r_{u,j} + w_5 \tag{3.10}$$

$$p_{a_{u,j}} = w_1 r_{f_{u,j}} + w_2 r_{a_{u,j}} + w_5 \tag{3.11}$$

3.1.3 BELFIS in the Second Learning Phase

The second learning phase begins when an input vector that is chosen from the test set, $\mathbf{i}_{c,j} \in \mathbf{I}_c = \{\mathbf{i}_{c,j}\}_{j=1}^{N_c}$, enters BELFIS, and BELFIS provides the corresponding output using the following steps:

1) The input vector $\mathbf{i}_{c,j}$ is fed to TH which provides two outputs: the first output is denoted by $\mathbf{th}^{Max-Min}_{c,j}$ and is calculated using equation (3.12). The second output of TH is $\mathbf{th}^{AGG}_{c,j}$, which is calculated according to equation (3.13); it is the output of AGG, that is the second part of the TH. Note that $\mathbf{th}^{AGG}_{c,j}$ is equal to $\mathbf{i}_{c,j}$ and is fed to the CX.

$$\mathbf{th}^{Max-Min}_{c,j} = [\mathrm{Max}(\mathbf{i}_{c,j}), \mathrm{Min}(\mathbf{i}_{c,j})] \tag{3.12}$$

$$\mathbf{th}^{AGG}_{c,j} = \mathbf{i}_{c,j} \tag{3.13}$$

2) The $\mathbf{th}^{AGG}_{c,j}$ is sent to the CX to provide $\mathbf{s}_{c,j}$, which is sent to AMYG and ORBI.

3) Both $\mathbf{s}_{c,j}$ and $\mathbf{th}^{Max-Min}_{c,j}$ are sent to the BL of AMYG, which is a trained adaptive network and provides the primary output for the test input; the primary output is calculated as $r_{a_{c,j}}$ using $r_{a_{c,j}} = F([s_{c,j}, th_{c,j}])$; where F is a fuzzy function that is equivalent to the trained adaptive network of the BL.

4) Receiving $\mathbf{s}_{c,j}$, the trained adaptive network of ORBI provides the secondary output, which is calculated as $r_{o_{c,j}} = F(\mathbf{s}_{c,j})$ and is sent to the CM of AMYG.

5) Receiving $r_{a_{c,j}}$ and $r_{o_{c,j}}$, the trained adaptive network of CM provides the final output, $r_{f_{c,j}} = F([r_{a_{c,j}}, r_{o_{c,j}}])$. The CM provides $p_{a_{c,j}}$, which is formulated as equation (3.14)

$$p_{a_{c,j}} = w_1 r_{f_{c,j}} + w_2 r_{a_{c,j}} + w_4 r_{o_{c,j}} \tag{3.14}$$

The fact is that the CM calculates, $\mathbf{p}_{a_c}$, in a different way from that which was formulated in the first learning phase. Similar to the first learning phase, the nonlinear parameters, $\mathbf{b}_o$ and $\mathbf{b}_a$ are adjusted using SD methods that calculate the derivatives of the loss functions, $f(\mathbf{p}_{a_c})$ and

$f(\mathbf{p}_{o_c})$ with respect to $\mathbf{b}_a$ and $\mathbf{b}_o$.

3.2 Brain Emotional Learning-based Recurrent Fuzzy System (BELRFS)

As was mentioned earlier, the structure of the Brain Emotional Learning Recurrent Fuzzy System (BELRFS) is quite like the structure of BELFIS. Figure 3.2 depicts the BELRFS's structure, which is slightly different from the general structure of BELIMs. The internal structure and function of the AMYG part of BELRFS consists of a recurrent adaptive network and a feed forward network. The structure and the function of other parts, TH, CX and ORBI, can be defined using neural networks and a feed forward adaptive network, respectively**.**

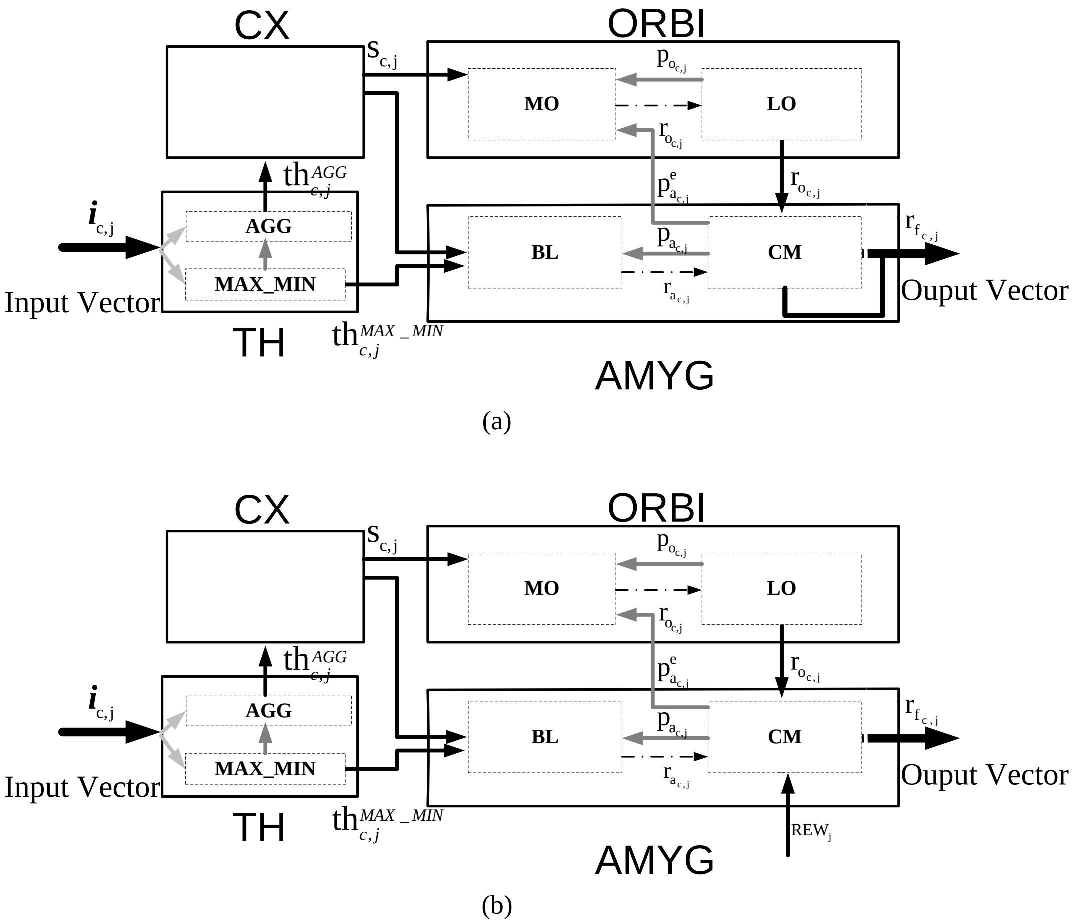


Figure 3.2: The structure of BELRFS and its connections in a) the first learning phase; b) second learning phase.

3.2.1 The Underlying Data-driven Model

In BELRFS, a recurrent adaptive network with fuzzy inference system is utilized for the BL of AMYG. In general, a recurrent adaptive network consists of feed forward layers and a recurrent layer with some delay nodes to form the recurrent signal and square nodes to form the feedback signal. Figure 3.3 depicts a simple recurrent adaptive network that consists of a recurrent layer and the feed forward layers that are similar to the feed forward adaptive network. The adaptive network consists of six layers: five feed forward layers and one recurrent layer with some unit-delay nodes and square nodes. The following steps explain the function of each layer of the adaptive network of Figure 3.3, with a two-dimensional input vector $\mathbf{i}_{u,j} = \{i_{u,j1}, i_{u,j2}\}_{j=1}^{N_u}$ and one recurrent signal.

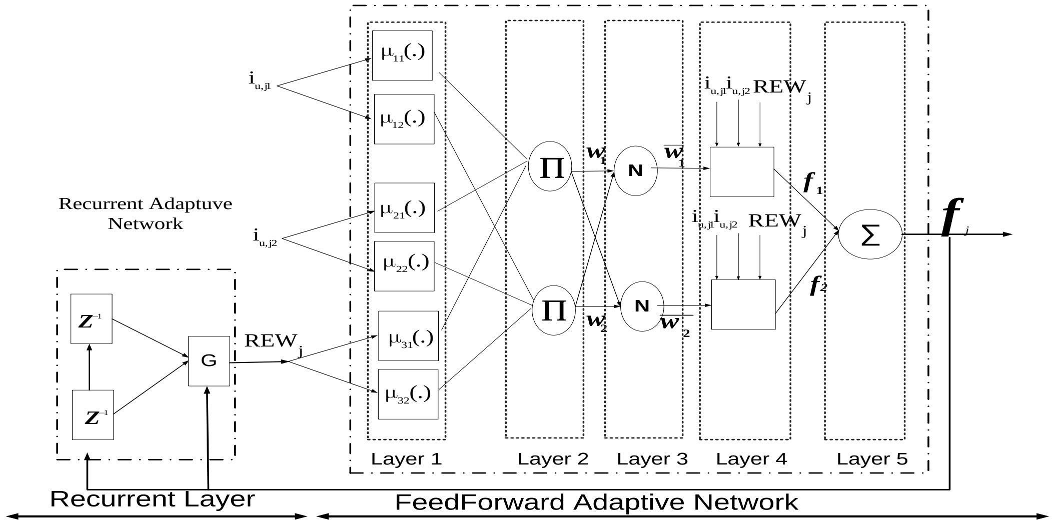


Figure 3.3: The structure of the recurrent adaptive network.

Layer 1: This layer receives a two-dimensional input vector $\mathbf{i}_{u,j} = \{i_{u,j1}, i_{u,j2}\}_{j=1}^{N_u}$, and a feedback signal that is denoted REW_j; the input of the network is $\mathbf{i}''_{u,j} = \{i_{u,j1}, i_{u,j2}, \mathrm{REW}_j\}_{j=1}^{N_u} = \{i''_{u,j1}, i''_{u,j2}, i''_{u,j3}\}_{j=1}^{N_u}$. For each dimension of the input vector, two square nodes with Gaussian or a Bell-shaped function are defined. In general, the first layer of this recurrent adaptive network with has an R-dimensional input vector as $\mathbf{i}_{u,j} = \{i_{u,j1}, \ldots, i_{u,jR}\}$ and m square nodes for each dimension of the input vector, consisting of $m \times (R+1)$ square nodes.

Layer 2: This layer has between two and eight circle nodes, which are labeled with $\prod$ receiving the outputs of the first layer; and the output vector of this layer is calculated by using multiplication operators between the inputs. Similar to the adaptive network in Figure 3.1, in the general case, this layer has $k_a = m^{R+1}$ circular nodes; assuming that the k_2 th node receives the outputs of the second square node of each dimension, the output of this node is calculated as (3.15) in a way similar to equation (3.3), where R is the dimension of input vector.

$$w_{k_2 l}(\mathbf{i}''_{u,j}) = \prod_{l=1}^{R+1} \mu_{2l}(i''_{u,jl}) \tag{3.15}$$

Layer 3: This layer has between two and eight circle nodes which are labeled with N and have normalization functions. Note that the number of nodes in this layer is equal to the number of nodes in the previous layer. In the general case, this layer has $k_a = m^{R+1}$ circle nodes, which are labeled N and are calculated according to equation (3.16).

$$\bar{w}_{k_2 l}(\mathbf{i}''_{u,j}) = \frac{\prod_{l=1}^{R+1} \mu_{2l}(i''_{u,jl})}{\sum_{k_2=1}^{k_a} w_{k_2 l}(i''_{u,jl})} \tag{3.16}$$

Layer 4: This layer has between two and eight square nodes. Equation (3.17) is the function of the k_2 th node. In the general case, the fourth layer has $k_a = m^{R+1}$ square nodes.

$$f_{k_2} = q_{k_2 1} i_{u,j1} + q_{k_2 2} i_{u,j2} + q_{k_2 3} REW_j + q_{k_2 4} \tag{3.17}$$

Layer 5: This layer has a single node (circle) that calculates the summation of its input vector, $\{f_{k_2}\}_{k_2=1}^{k_a}$. The final output is calculated as $F_j(\mathbf{i}''_{u,j}) = \sum_{k_2=1}^{k_a} \bar{w}_{k_2} f_{k_2}$

Recurrent layer: This layer is the recurrent layer and consists of the unit-delay nodes (in this case, two unit-delay nodes are included) to provide recurrent signals and a circle node with G function to provide the feedback signal. The function of G is as equation (3.18).

$$\begin{aligned} G(\hat{y}_j, y_{u,j}) &= 1 - (\hat{y}_j(t-1) - y_{u,j}(t-1)) \\ &+ \frac{\sum_{j=1}^{N} (\hat{y}_j(t-2) - y_{u,j}(t-2))^2 + \sum_{j=1}^{N} (\hat{y}_j(t-1) - y_{u,j}(t-1))^2}{\sum_{j=1}^{N} (\hat{y}_j - \bar{y})^2} \end{aligned} \tag{3.18}$$

The parameters: N, $\hat{y}_j$ and $y_{u,j}$, refer to as the number of samples, the observed values and desired targets, respectively. The parameter $\bar{y}$ is the average of the desired targets.

3.2.2 BELRFS in the First Learning Phase

The first learning phase of BELRFS begins by receiving an input vector from the training set, $\mathbf{i}_{u,j} \in \mathbf{I}_u = \{\mathbf{i}_{u,j}\}_{j=1}^{N_u}$, similar to the first learning phase of BELFIS. The only difference is in the functionality of the CM is that, in BELRFS, the CM consists of a recurrent adaptive network that provides the output of BELRFS as: $r_{f_{u,j}} = F([r_{au,j}, r_{ou,j}], G([r_{au,j}, r_{ou,j}]))$. Here, F is a fuzzy function as (3.6). CM has eight layers and the eighth layer consists of two circle nodes with the summation functions to provide $p^e_{a_{u,j}}$ and $p_{a_{u,j}}$. The former, $p^e_{a_{u,j}}$, represents the expected punishment and is formulated according to equation (3.10). The latter, $p_{a_{u,j}}$, is the punishment and is formulated as equation (3.11).

3.2.3 BELRFS in the Second Learning Phase

The second learning phase begins when BELRFS receives an input vector, $\mathbf{i}_{c,j}$, that is chosen from the test set, $\mathbf{I}_c = \{\mathbf{i}_{c,j}\}_{j=1}^{N_c}$. The BELRFS provides the corresponding primary and secondary outputs, $r_{a_{c,j}}$ and $r_{o_{c,j}}$, using steps similar to those in BELFIS. However, in BELRFS, the function of the CM is different; receiving $r_{a_{c,j}}$ and $r_{o_{c,j}}$ the trained recurrent adaptive network of CM provides the final output, $r_{f_{c,j}} = F([r_{a_{c,j}}, r_{o_{c,j}}], \text{feedbackvector})$. During the second learning phase, the feedback signal is calculated using the weighted k-nearest neighbor (Wk-NN) method. The following steps explain how the Wk-NN provides the feedback signal, REW_{test}:

1) For each $\mathbf{i}_{tset}$, the Euclidean distance $d_j = \|\mathbf{i}_{test} - \mathbf{i}_{u,j}\|_2$ is calculated, where $\mathbf{i}_{u,j}$ is a member of the training data set $\mathbf{i}_{u,j} \in \mathbf{I}_u = \{\mathbf{i}_{u,j}\}_{j=1}^{N_u}$.

2) For each test sample, e.g., $\mathbf{i}_{tset}$, a subset of k minimum values of $\mathbf{d} = \{d_1, d_2, ..., d_{N_u}\}$ is selected, $\mathbf{d}_{min}$. This is the set that corresponds to the k nearest neighbors of the test sample.

3) For these neighbors, a subset of $\mathbf{REW} = \{REW_1, ..., REW_{N_u}\}$ is selected, and this subset is referred to as $\mathbf{REW}_{min}$. For the test sample, $\mathbf{i}_{tset}$, the value of REW_{test} is according to equation (3.19).

$$REW_{test} = (\sum_{j=1}^{k} v_j \times REW_{min,j} / \sum_{j=1}^{k} v_j) \tag{3.19}$$

Where, v_j, is calculated as the kernel, equation (3.20).

$$v_j = K(d_{min,j}) \tag{3.20}$$

The kernel function $K(.)$ converts Euclidian distances to the weights, equation (3.21).

$$K(d) = \frac{\max(\mathbf{d}) - (d_j - \min(\mathbf{d}))}{\max(\mathbf{d})} \tag{3.21}$$

3.3 Emotional Learning-inspired Ensemble Classifier (ELiEC)

One instance of BELIMs has been tested for classification by using benchmark data sets obtained from the University of California, Irvine (UCI), machine learning repository. This model is referred to as the brain emotional learning-based ensemble classifier (ELiEC). Similar to all BELIMs, the ELiEC model consists of four main parts: TH, CX, AMYG and ORBI. The ELiEC model and the connection between these parts are depicted in Figure (3.4).

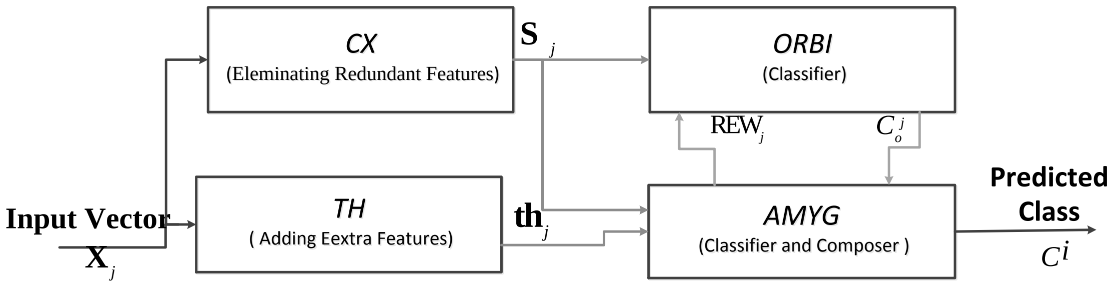


Figure 3.4: The structure of ELiEC.

3.3.1 The Function of ELiEC

Consider a set of training data with N_u samples as $\{(\mathbf{x}_j, c^j)\}_{j=1}^{N_u}$, where $\mathbf{x}_j$ is j^{th} instance with R dimension. In the classification context, the input vector and each element of the input vector are referred to as the instance and the features, respectively. The parameter $\mathbf{c}^j$ determines the target output of each input vector; in classification it has been referred to as the class label, so $\mathbf{c}^j$ is the class label of $\mathbf{x}_j$. One way to encode the label class, $\mathbf{c}^j$, is to use a binary representation as $\mathbf{y}^j = y_1^j,...,y_2^j,...,y_n^j$, where n is the number of class labels. If $\mathbf{c}^j$ is equal with the m^{th} class, the value of y_m^j will be equal to one and other values will be zero. The following steps explain the function of each part of ELiEC to provide the corresponding output of each input. Note that the TH and CX work as feature selection modules.

1) The TH evaluates $\mathbf{x}_j$ to provide $\mathbf{th}_j$ according to equation (3.22) and sends $\mathbf{th}_j$ to the AMYG and the CX.

$$\mathbf{th}_j = [\max(\mathbf{x}_j), \mathrm{mean}(\mathbf{x}_j), \min(\mathbf{x}_j)] \tag{3.22}$$

2) The CX evaluates $\mathbf{x}_j$ to select the most informative features and eliminate the redundant features. Thus, CX receives $\mathbf{x}_j$ with R features and provides $\mathbf{s}_j$ with l important features ($l \le R$).

3) The AMYG consists of two parts: BL and CM. The function of BL is implemented using a classifier that denotes by $\mathrm{Class}(.)$. Thus, the BL of AMYG receives $\mathbf{x}_j^a = [\mathbf{th}_j, \mathbf{s}_j]$ and provides $C_a^j = \mathrm{Class}(\mathbf{x}_j^a)$. The other part of AMYG is the CM that is another classifier with a different instance, $\mathbf{x}_j^c = [\mathbf{x}_j^a, \mathbf{x}_j^o, C_a^j, C_o^j]$; it provides the final output $C^j = \mathrm{Class}(\mathbf{x}_j^c)$ using a classification method. It should be noted that the classifiers of AMYG and ORBI can be defined on the basis of any supervised classification method, e.g. decision tree, single or multilayer perception, and support a vector machine etc. For example, Weighted k-nearest neighbor (Wk-NN) is a type of instance-based algorithm that can be used as the classification methods for both the AMYG and the ORBI.

4) The ORBI can also be defined using a classification method. The input vector of the ORBI can be a combination of $\mathbf{s}_j$ and p_{aj}^e as $\mathbf{x}_j^o = [\mathbf{s}_j, p_{aj}^e]$; in this way, the ORBI classifier is a dependent classifier, which means that this classifier needs to receive the p_{aj}^e that is a punishment provided by the AMYG. By contrast, if the input vector is only equal to $\mathbf{s}_j$ as $\mathbf{x}_j^o = [\mathbf{s}_j]$, the ORBI classifier is an independent classifier. This means that it does not need to receive the punishment from the AMYG. The classification result of the ORBI is $C_o^j = \mathrm{Class}(\mathbf{x}_j^o)$, which is sent to the CM of the AMYG.

3.3.2 The Underlying Classification Models

As was mentioned earlier, the function of each part of ELiEC can be explained by adapting different classification methods. Weighted k-nearest neighbor (Wk-NN) is a type of instance-based algorithm that has been widely used as a classification and regression method. For a given training set: $(\mathbf{x}_1, c^1), ..., (\mathbf{x}_j, c^j) ..., (\mathbf{x}_{N_t}, c^{N_t})$, the Wk-NN determines the class of a test vector, $\mathbf{x}_{test}$,using the following steps:

1) The Euclidian distance between $\mathbf{x}_{test}$ and $\mathbf{x}_j$ is calculated, $d_j = \|\mathbf{x}_{test} - \mathbf{x}_j\|_2$, in which each $\mathbf{x}_j$ is a member of $\mathbf{x}_1, \mathbf{x}_2, ..., \mathbf{x}_{N_t}$, where N_t denotes the number of samples in the training data set.

2) The k minimum values of $\mathbf{d} = \{d_1, d_2, ..., d_{N_t}\}$ are selected as $\mathbf{d}_{min}$. The $\mathbf{x}_j$ s that correspond to $\mathbf{d}_{min}$ are the k nearest neighbors to $\mathbf{x}_{test}$ and define the local neighborhoods of the test vector, $\mathbf{x}_{test}$.

3) The class label of $\mathbf{x}_{test}$ is chosen from the class labels of the local neighborhoods. Using the weighted k-nearest neighbor (Wk-NN), a weight is assigned to each neighbor; the assigned weight is defined according the kernel function $K(.)$.

Any arbitrary function that holds the following properties can be considered as the kernel function.

1) For all d, $K(d) \geq 0$.
2) If $d = 0$ then $K(d)$ gets the maximum value.
3) If $d \to \pm\infty$ then $K(d)$ gets the minimum value.

The kernel function in this case is defined as equation (3.23); it gives that closer neighbors to $\mathbf{x}_{test}$ have higher weights to estimate the class label of $\mathbf{x}_{test}$.

$$K(d) = \frac{\max(\mathbf{d}) - (d_j - \min(\mathbf{d}))}{\max(\mathbf{d})} \tag{3.23}$$

4 Summary of Appended Papers

This chapter presents an abstract of each appended paper. It also describes the objective of each paper and how it contributes to the overall goal of this thesis.

4.1 PAPER A

This paper presented a type of BELIM that is referred to as the Brain Emotional Learning Based Fuzzy Inference System (BELFIS). The Brain Emotional Learning Based Fuzzy Inference System (BELFIS) is a type of neuro-fuzzy system. The structure of BELFIS is according to the general structure of BELIMs, although its functionality is implemented using adaptive networks with a fuzzy inference system. The learning algorithm of BELFIS is defined as a hybrid learning algorithm to adjust the learning parameters of the adaptive networks.

To evaluate the performance of BELFIS, this model has been tested to predict solar activity using sunspot numbers, which is one of the benchmark indices of solar activity. Solar activity as a chaotic natural system is measured using sunspots, solar flares and coronal mass ejections (CMEs) [48]. A good measure of solar activity through the solar cycle is the sunspot number that gradually increases to a highest number, which is known as the solar maximum, from the last lowest number, which is designated as the solar minimum.

The results of testing BELFIS on different data sets of a sunspot number were compared with various data-driven methods, e.g. neural networks (e.g.,MLP) and neuro-fuzzy methods (ANFIS and LoLiMoT . The graphs and tables verified that BELFIS could obtain very similar results to those of ANFIS for the short-term prediction and long-term prediction of solar activity. It should be emphasized that the execution time of the BELFIS in this case is lower than the execution time of ANFIS.

This paper contributed to this thesis by showing that using adaptive networks with the fuzzy inference system is one method for implementing BELIM. This paper also provided a comparative result for sunspot numbers, which is one of the benchmark data sets of machine learning.

4.2 PAPER B

This paper presented another type of BELIM that is referred to as the Brain Emotional Learning Based Recurrent Fuzzy System (BELRFS). Similar to BELFIS, this model of BELIM adopts adaptive networks to mimic the functionality of the brain's regions responsible for processing fearful stimulus. In contrast with BELFIS, BELRFS was developed by merging a recurrent adaptive network and feed forward adaptive networks with fuzzy inference systems.

BELRFS was examined on three benchmark data sets: a chaotic time series (Lorenz time series) and two real nonlinear time series (sunspot numbers and Auroral Electroject index).

BELRFS was tested for both long-term and short-term prediction of the Lorenz chaotic time series, a well-known benchmark time series, and the results obtained are compared with other new data drive models. To further evaluate the performance of the BELRFS and verify its robustness, Gaussian noise with a standard deviation 0.1 is added to the first data set. The results obtained in these case studies were compared with the results of different neural networks. It was noticeable that the normalized mean square error of BELRFS for predicting the noisy data samples was lower than most of the previous methods.

BELRFS was also applied to predict a non-smoothed monthly sunspots time series, a part of solar cycle 19, a well-known solar cycle for solar activity prediction. Solar cycle 19 started in 1954 and ended in 1964. The test data set contains the sunspots from 1950 to 1965. This set also includes the peak sunspot number of solar cycle 19, which occurred in 1957. The results obtained with BELRFS were compared with the results of LOLIMOT. The results showed that BELRFS could be used as a prediction model. To further investigate the performance of BELRFS, the model was applied for the smoothed sunspot number from November 1834 to June 2001. The results were compared with the results of previous methods.

As already stated, the AE index is a measure of geomagnetic storms and can be used to predict space storms. The AE index has been recorded by the World Data Center for Geomagnetism and Space Magnetism (Kyoto University). BELRFS was also examined for one minute-ahead prediction (a short-term prediction) of the AE index of the first seven days of March 1992 and was utilized as training data to predict the AE index of March 9th , 1992. It can be concluded that the BELRFS is more accurate than the LoLiMoT in short-term prediction of AE time series.

This paper contributes to this thesis by showing that BELIMs are not highly sensitive to noise. The accuracy of the BELIMs is in the same domain as other well-known data-driven models. The main contribution of this paper was to present a new type of adaptive network with the recurrent layer.

4.3 PAPER C

This paper demonstrated that the brain emotional learning based inspired model (BELIM) could also be used as a classification model. This model is known as the emotional learning-inspired ensemble classifier (ELiEC). The structure of this model is slightly different from the general structure of BELIMs, and the function of this model can be defined by adapting supervised classification models.

The performance of ELiEC was investigated by applying binary and multiclass benchmark data sets obtained from the University of California, Irvine (UCI), machine learning repository. The results obtained were compared with the results of other methods such as Support Vector Machines (SVMs).

The contribution of this paper was to show that a classification model based on a brain emotional learning model can classify the balanced and low dimensional data sets with an accuracy that is comparable with other well-known models such as a support vector machine and meta-cognitive neural network.

5 Discussion and Conclusion

A challenge for the computational intelligence community is to develop simple computational models for prediction, classification and optimization with low model complexity, high accuracy, fast training, high on-line adaptation and low noise sensitivity.

In this thesis, the proposed data driven model, BELIM, aimed to achieve the above goals by developing a model inspired from the neural structure of fear conditioning, implementing the functionality of this model by assigning adaptive networks to the different parts of the structure and defining how these adaptive networks are put together.

The main properties of BELIM can be highlighted as follows:

1) Accuracy: Accuracy is an important performance parameter of data driven models. One straightforward way to investigate this feature is to test the model with benchmark data sets. In this thesis, BELIM was tested by different benchmark data sets, Lorenz time series, sunspot numbers and the auroral electroject index. In comparison with other data-driven models, the results of BELIMs indicated that this model is capable of producing predictions with reasonable accuracy. In particular, for some cases, Paper B (e.g., long term and short-term prediction of Lorenz time series), BELRFS provides better prediction results than other well-known data-driven models such as ANFIS and LoLiMoT.

2) Curse of dimensionality: this feature reflects the dependence that exists between complexity and the dimension of the input space. Some data-driven models, such as fuzzy, suffer from increasing the model complexity when there is highly dimensional data. The suggested structure for BELIM could be considered a strategy to reduce the curse of dimensionality by considering different paths and different sizes of the input dimension. However, this problem is still not resolved in BELIM. Dependence on the underlying adaptive network can be a severe problem in BELIM. The structure of BELIMs is a way to combine a low model complexity adaptive network with a high model complexity to reduce the curse of dimensionality while increasing the accuracy of the model.

3) Sensitivity to noise: this feature reflects how noise sensitive the model's prediction is. It is important to note that a good data-driven model should have low noise sensitivity. Thus, this characteristic depends on the underlying adaptive networks, for example, BELFIS, which is based on adaptive networks with fuzzy inference system and is highly sensitive to noise.

4) Training time: measures the time needed to train the data-driven model, i.e. to optimize the learning parameters. It is clear that this parameter depends on the underlying adaptive network of BELIMs and the learning algorithm of BELIMs. In this thesis, the steepest descent algorithm is used as the main learning algorithm, and this algorithm is slow in finding the optimal parameters.

5) On-line adaptation: this characteristic reflects how the learning parameters of a data-driven model can be adjusted using on-line methods. The on-line adaptation in BELIM is efficient and robust if the nonlinear parameters of AMYG and ORBI are adjusted using the second learning phase. Using a robust adaptive network in terms of on-line adaptation gives an increase in the robustness of the BELIM.

6) Defining the second learning phase, the nonlinear learning parameters could be adjusted in an online manner; however, due to the use of the steepest descent algorithm as the optimization

algorithm, the convergence is slow.

BELIMs have a modular structure. However, they differ from modular neural networks and ensemble classification in terms of structure, function and learning algorithms. The main difference between the formal modular neural network and BELIMs is that, in the modular neural network, each module has different input vectors [2], and different modules do not have a connection with each other. However, BELIMs have been developed that are based on the connection between different parts. The overall output of the modular neural network is provided by an integrating unit, which has an important role in defining how the outputs of different parts should be combined and in deciding about which module should be trained with which training data samples. In contrast, in BELIM, the learning algorithm of each part could be defined at the initialization step and depends on the underlying adaptive networks. Ensemble classification is another modular model that might seem to be similar to BELIMs. Note that the idea of ensemble classification was inspired by the rational decision making theory. A rational person makes the optimal decision by seeking and combining other ideas. In contrast, the basic structure of BELIM is inspired from the neural structure of fear conditioning in the brain.

6 Future Works

This thesis investigated how brain emotional learning, in particular fear conditioning, can be utilized to develop a data-driven model. It also demonstrates how this model was able to be successfully tested for prediction and classification benchmark data sets. The proposed model can be improved in several aspects, e.g. in structural and functional perspectives and an application perspective.

6.1 Structural and functional perspective

The proposed model is a simple inspired model of the neural structure of fear conditioning. However, the internal structure of the thalamus, sensory cortex, amygdala and orbitofrontal cortex is more complicated than the proposed model. Thus, one improvement is to consider a more complex model than the one proposed in order to increase its accuracy. Extending another computational model, e.g. the hippocampus-neocortex, by the neural structure of fear conditioning may be another method that could improve the accuracy of the proposed data-driven models.

The training speed of the model could be enhanced by defining another optimization method; e.g. the Newton method can be utilized to update the nonlinear parameters. So far, the learning algorithm of BELIM is defined as a combination of a supervised learning and unsupervised learning algorithm. In the future, an unsupervised learning algorithm for BELIM will be used.

6.2. Perspective of the Application

The results obtained employing BELIMs for prediction and classification are a motivation for applying this model to identify nonlinear systems, clustering and optimization applications. This model will also apply as an intelligent controller to investigate its performance in comparison with a brain emotional learning-based intelligent controller. A next step is to apply the model as the fundament for an emotional decision making model in an attempt to develop a new emotional-based engine for cognitive radio networks. The suggested emotional decision making model may be applied for the purpose of data mining by utilizing the connection between subjective and objective decisions.

Finally, this thesis has been a starting point in discovering how the emotional system can be an inspiration in developing emotion-based models for different purposes. There are still many open questions related to this area.

References

[1] O. Nelles, *Nonlinear System Identification*:*From classicical Approches to Neural Networks and Fuzzy Models.* Berlin, Germany: Springer-Verlag, 2001.

[2] S. Haykin, *Neural Networks: A Comperhensive Foundation.*Upper Saddle River, NJ:Prentice Hall, 2nd ed., 1999.

[3] R. Jang, C. Sun and E. Mizutani, *Neuro-Fuzzy and Soft Computing: A computational approach to Learning and Machine Intelligence.*Upper Saddle River, NJ: Prentice Hall, 1997.

[4] R. J. Frank, N. Davey and S. P. Hunt, "Time Series Prediction and Neural Networks," *J. Intell Robot Syst.*, vol. 31, no. 1-3, pp. 91-103, 2001.

[5] Y. Chen, B. Yang, and J. Dong, "Time-series prediction using a local linear wavelet neural network," *J. Neurocomputing*, vol. 69, nos. 4–6, pp. 449–465, 2006.

[6] M. Ardalani-Farsa and S. Zolfaghari, S, "Chaotic time series prediction with residual analysis method using hybrid Elman–NARX neural networks,'' *J. Neurocomputing*, vol.73, issues 13–15, pp.2540-2553, 2010.

[7] G. Inoussa, H. Peng, and J. Wu, "Nonlinear time series modeling and prediction using functional weights wavelet neural network-based state-dependent AR model,'' *J. Neurocomputing Journal*, vol. 86, pp. 59-74,2012.

[8] D.T. Mirikitani and N. Nikolaev, "Recursive Bayesian Recurrent Neural Networks for Time-Series Modeling,", *IEEE Trans. Neural Netw*, vol.21, no.2, pp.262-274,2010.

[9] M. Qian-Li, Z. Qi-lun, P. Hong, Z. Tan-Wei and X. Li-Qiang, "Chaotic Time Series Prediction Based on Evolving Recurrent Neural Networks,'' in Proc. *Int. Conf. Machine Learning and Cybernetics (ICMLC.2007)*, vol.6,no.,pp.3496-3500, 2007.

[10] R. Chandra and M. Zhang, "Cooperative coevolution of elman recurrent neural networks for chaotic time series prediction,''*J. Neurocomputing*, vol. 86, pp.116- 123, 2012.

[11] P. Gómez-Gil, J.M. Ramírez-Cortes, S.E. Pomares Hernández, V. Alarcón-Aquino A neural network scheme for long-term forecasting of chaotic time series Neural Process. Lett., 33,pp.215-223,2011.

[12] P. Gomez-Gil, "Long term prediction, chaos and artificial neural networks Where is the meeting point?" Engineering Letters, vol. 15, no. 1, 2007.

[13] P. Gomez-Gil and M. Ramirez-Cortes, "Experiments with a hybridcomplex neural networks for long term prediction of electrocardiograms." Proceedings of the IEEE 2006 International World Congress of Computational Intelligence, IJCNN 2006. Vancouver. Canada. pp.4078-4083, 2006.

[14] J.E.Ledoux, *The emotional brain: the mysterious underpinnings of emotional life*, Simon & Schuster,NY ,1998.

[15] J.Moren, C.Balkenius,"A computational model of emotional learning in the amygdala,'', *in From Animals to Animats*, MIT, Cambridge, 2000.

[16] M. S. Gazzaniga, R. B, Ivry, G.R.Mangun, and Megan.S.Steven, *Gognative Nerosc in The Biology of the Mind.* W.W.Norton&Company, New York, 3rd ed., 2009.

[17] D.Reisbetg, R.College., *Cognetion: Exploring the science of the Mind, Newyork*, 4th edition, W. W. Norton & Company, New York 2006.

[18] T.Dalgleish, "The emotional brain,'' *J. NAT REV NEUROSCI.*, vol. 5, no. 7, pp. 583–589, 2004.

[19] E. R. Kandel., J. H. Schwartz., T. M. Jessell.: *Principles Of Neural Science*. 4th edition, McGraw-Hill Medical, 2003.

[20] J.M.Jenkins, K.Oatley, N.L.Stein, *Human emotions: A READER*, Blockwell publisher, U.K., 1998.

[21] D. H. Hubel, M. S. Livingstone, "Color and Contrast Sensitivity in the Lateral Geniculate Body and Primary Visual Cortex of the Macaque Monkey," *J., Neuroscience.* vol. 10, no.7, pp. 2223-2237, 1990.

[22] J. P. Kelly, "The Neural Basis of Perception and Movement, Principles of Neural Science," London: Prentice Hall. 1991.

[23] K. Amunts., O. Kedo., M. Kindler., P. Pieperhoff., H. Mohlberg., N. Shah., U. Habel., F. Schneider., K. Zilles., "Cytoarchitectonic mapping of the human amygdala, hippocampal region and entorhinal cortex : intersubject variability and probability maps,'' *J. Anatomy and Embryology.*, vol. 21, no. 5-6, pp. 343-352, 2005.

[24] C. I. Hooker., L. T. Germine., R. T. Knight., M. D. Esposito., “Amygdala Response to Facial Expressions Reflects Emotional Learning,’’ *Neuroscience. J.*, vol. 26, no.35, pp. 8915-8930, Aug. 2006.
[25] E. R. Kandel., J. H. Schwartz., T. M. Jessell.: *Principles Of Neural Science*. 4th edition, McGraw-Hill Medical, 2003.
[26] B. Ferry., B. Roozendaal., J. McGaugh., “Role of norepinephrine in mediating stress hormone regulation of long-term memory storage: a critical involvement of the amygdala,’’, *J. Biol Psychiatry.*, vol. 46, no. 9, pp. 1140-1152, 1999.
[27] M. L. Kringelbach., “The orbitofrontal cortex: linking reward to hedonic experience,’’ *J., Nat. Rev. Neurosci.*, vol. 6, pp. 691-702,2005.
[28] A. G. Phillips., “The Brain and Emotion by Edmund T. Rolls,’’ *J. TRENDS. COGN. SCI.*, http://www.sciencedirect.com/science/journal/13646613,vol. 3, pp. 281-282, 1999.
[29] C. Cavada., W. Schultz.,“The Mysterios of Orbitofrontal Cortex. Foreword. Cereb Cortex,’’, *J. Cerebr. Cortex.* vol. 10, no. 3, pp .205, 2000.
[30] C. A. Winstanley, D. E. H. Theobald, R. N. Cardinal, and T. W. Robbins, “Constracting Roles of Basolateral Amygdala and Orbitofrontal Cortex in Impulsive Choice,” *J. Neurosci.*, vol. 24, no. 20, pp. 4718-4722, 2004.
[31] T. Kuremoto, T. Ohta, K., Kobayashi, M., Obayashi, “A dynamic associative memory system by adopting amygdala model,” *J. AROB*, *v*ol.13, pp.478–482, 2009.
[32] T. Kuremoto, T. Ohta, K. Kobayashi, K., *M.* Obayashi, “A functional model of limbic system of brain,” in *Proc*. *Int. Conf. Brain informatics*, pp.135-146, 2009.
[33] S.Marsella, J.Gratch, and P.Petta, "Computational Models of Emotion", in Blueprint for Affective Computing (Series in Affective Science), 2010.
[34] M. Maçãs and L. Custódio, "Multiple Emotion-Based Agents Using an Extension of DARE Architecture," *J. Informatica (Slovenia),* vol. 27, no. 2, pp. 185-196, 2004.
[35] J. D. Velásquez., “When Robots Weep: Emotional Memories and Decision-Making,” in *Proc. Conf. on Artifitial Intelligence*, pp.70-75. 1997.
[36] C. Lucas, D. Shahmirzadi, N. Sheikholeslami, “Introducing BELBIC: brain emotional learning based intelligent controller,’’ *J. INTELL. AUTOM. SOFT. COMPUT.*, vol. 10, no. 1, pp. 11-22, 2004.
[37] N. Sheikholeslami, D. Shahmirzadi, E. Semsar, C. Lucas., "Applying Brain Emotional Learning Algorithm for Multivariable Control of HVAC Systems,", J. INTELL. FUZZY. SYST.vol.16, pp. 1–12, 2005.
[38] A. R. Mehrabian, C. Lucas, J. Roshanian,"Aerospace Launch Vehicle Control: An Intelligent Adaptive Approach", J. Aerosp. Sci. Technol.,vol.10,pp. 149–155, 2006.
[39] R. M. Milasi, C. Lucas, B. N. Araabi, "Intelligent Modeling and Control of Washing Machines Using LLNF Modeling and Modified BELBIC," in Proc. Int. Conf. Control and Automation., pp.812-817, 2005.
[40] A. M. Yazdani1, S. Buyamin1, S. Mahmoudzadeh2, Z. Ibrahim1 and M. F. Rahmat1., “Brain emotional learning based intelligent controller for stepper motor trajectory tracking,” *J. IJPS.*, vol. 7, no. 15, pp. 2364-2386, 2012.
[41] T. Babaie, R. Karimizandi, C. Lucas, “Learning based brain emotional intelligence as a new aspect for development of an alarm system,’’ *J. Soft Computing.*, vol. 9, issue 9, pp.857-873, 2008.
[42] M. Parsapoor, C. Lucas and S. Setayeshi, "Reinforcement _recurrent fuzzy rule based system based on brain emotional learning structure to predict the complexity dynamic system," *in Proc. IEEE Int. Conf. ICDIM*, pp.25-32, *2008*.
[43] M. Parsapoor, U. Bilstrup, "Brain Emotional Learning Based Fuzzy Inference System (BELFIS) for Solar Activity Forecasting," in *Proc. IEEE Int. Conf. ICTAI 2012*, 2012.
[44] M. Parsapoor and U. Bilstrup, “Chaotic Time Series Prediction Using Brain Emotional Learning Based Recurrent Fuzzy System (BELRFS),” *International Journal of Reasoning-based Intelligent Systems*, 2013.
[45] M. Parsapoor, “Prediction the price of Virtual Supply Chain Management with using emotional methods,’’ *M.S. thesis,* Dept. Computer. Eng., Science and research Branch, IAU., Tehran, Iran,2008.
[46] M. Parsapoor, U. Bilstrup, "Brain Emotional Learning Based Fuzzy Inference System (Modified using Radial Basis Function)," 8th IEEE International Joint Conference for Digital InformationManagement, 2013.
[47] M. Parsapoor, U. Bilstrup, "An Emotional Learning-inspired Ensemble Classifier (ELiEC)," In proceeding of: 8th International Symposium Advances in Artificial Intelligence and Applications (AAIA'13), At Kraków, Poland, 2013.

[48] A.J. Izeman and J.R.Wolf,“Zurich sunspot relative numbers,’’,*The Mathematical Intelligence Journal*,vol.7, pp.27-33,1998.