

Development of Spiking Neural Network Methods for Spatio-Temporal Brain Data Analysis

Maryam Doborjeh

A thesis submitted to

Auckland University of Technology

in fulfilment of the requirements for the degree of

Doctor of Philosophy (PhD)

2018

School of Engineering, Computer and Mathematical Sciences

Abstract

This thesis proposes methods employing an evolving Spiking Neural Network (SNN) architecture for the analysis of spatio-temporal neuroimaging data. Multivariate Spatio-Temporal Brain Data (STBD) is intrinsically complex as it contains both *time* and *space* dimensions that represent the patterns of cognitive processes in the brain. Scrutinising the spatio-temporal interactions between variables in such complex data demands incorporating the spatial and temporal aspects into the model's computations.

To this end, first an SNN architecture was used for modelling, learning, mapping and classifying of STBD, including Electroencephalogram (EEG) and Functional Magnetic Resonance Imaging (fMRI) data. I designed SNN models that allowed for a better understanding of cognitive processes by capturing the spatio-temporal interactions between variables when compared with extant reservoir computing systems. The models enhanced the classification performance by achieving up to 92% accuracy which represents an average improvement of 20% when compared with different machine learning methods.

Further, I proposed and developed a new dynamic spatio-temporal clustering approach which allowed for the assessment of the evolving learning patterns in SNN models. This study led to knowledge discovery in SNN evolutionary learning patterns and resulted in feature selection that improved the classification accuracy by up to 10%. It also revealed the

trajectory of brain areas involved in response to a cognitive task. The proposed clustering configuration was evaluated using a validity measurement method based on cohesion and separation that represented a high goodness of the clustering structure.

Finally, I proposed a new personalised modelling approach for integrated static and spatio-temporal data using SNN models. To build a personalised SNN model (PSNN), I developed a new clustering method, named Dynamic Weighted-Weighted Distance K-nearest Neighbours (DWWKNN). The developed PSNN improved the classification accuracy by 12% when compared with the global SNN models. This also resulted in creating a profile for an individual.

Overall, this research has scrutinised the hidden evolutionary learning patterns in SNN architecture, which resulted in an identification of neural areas activated by different input neurons. Furthermore, it has demonstrated an original personalised modelling that resulted in an improvement in classification accuracy.

Table of Content

Abstract.....	I
Table of Content.....	III
List of Figures	VI
List of Tables	XVI
List of Abbreviations.....	XXI
Attestation of Authorship	XXIII
Acknowledgements	XXIV
Chapter 1 Introduction.....	1
1.1 Rationale and Motivation	1
1.2 Aims of this Thesis and Research Questions.....	3
1.3 Thesis Structure.....	4
Chapter 2 Spatio-temporal Brain Data and Analytical Methods	8
2.1 Introduction.....	8
2.2 Information Processing in the Human Brain	8
2.3 EEG Data.....	12
2.4 FMRI Data.....	14
2.5 Analytical Methods for STBD	15
2.6 Chapter Summary.....	30
Chapter 3 Spiking Neural Networks	31
3.1 Introduction.....	31
3.2 Computational Model of a Spiking Neuron.....	33
3.3 Information Encoding in a Spiking Neuron.....	40
3.4 Learning in SNN Models.....	42

3.5	SNN Reservoir Computing Systems	43
3.6	NeuCube for STBD Modelling	45
3.7	Chapter Summary.....	53
Chapter 4	SNN Feasibility Study on STBD	57
4.1	Introduction.....	57
4.2	NeuCube Architecture for STBD Analysis	57
4.3	A New SNN Study on fMRI.....	61
4.4	SNN Application on EEG.....	79
4.5	Chapter Summary.....	89
4.6	Contribution	91
Chapter 5	A New Spatio-temporal Clustering of SNN Patterns.....	92
5.1	Introduction.....	92
5.2	Clustering of SNN Evolutionary Patterns	95
5.3	Clustering Validation	100
5.4	Chapter Summary.....	103
5.5	Contribution	104
Chapter 6	Feasibility Study of Spatio-temporal Clustering	105
6.1	Introduction.....	105
6.2	Clustering Patterns in SNN Models	106
6.3	Chapter Summary.....	141
6.4	Contribution	142
Chapter 7	A New Personalised Modelling using SNN	143
7.1	Introduction.....	143
7.2	Personalised Modelling for Integrated Static and Dynamic Data.....	144
7.3	Personalised Modelling for EEG Data	149
7.4	Chapter Summary.....	159
7.5	Contribution	160
Chapter 8	Conclusions and Recommendations for Future Work	161
8.1	Introduction.....	161

8.2 Aims and Methodological Approach	161
8.3 Key Findings.....	162
8.4 Empirical and Theoretical Contributions	164
8.5 Limitations of the Thesis	166
8.6 Future Direction and Implications	166
References.....	169
Appendix A Talairach Mapping	195
Appendix B Spatio-Temporal fMRI Study	197
B.1 Spatio-temporal fMRI Data Description	197
B.2 Statistical Analysis of the SNN Models of fMRI.....	199
B.3 SNN Parameter Optimisation	200
Appendix C EEG Study	202
C.1 Participants.....	202
C.2 Cognitive GO/NOGO Task	204
C.3 EEG Data Acquisition	205
C.4 Event-Related Potentials (ERPs) Processing	206
C.5 T-Test Results	207
C.6 Feature Interaction in SNN Models	208
C.7 Test Accuracy Evaluation using F-Score Measurement	216
C.8 SNN Parameter Optimisation	217
C.9 Parameter Setting for Conventional Methods.....	218
Appendix D Dynamic Clustering Patterns	221
Appendix E Co-authored Publications	232

List of Figures

Figure 1-1 A bird’s-eye view of the thesis structure.	7
Figure 2-1 Information flows between neurons through the axon, which receives information from a presynaptic neuron and generates an action potential that is sent down to the synapses of the postsynaptic neuron. This figure is modified from (Marcella, 2011).	11
Figure 2-2 Sodium Na^+ and potassium K^+ ions move across the soma membrane through ion channels, this accomplishes electrochemical powers as a result of the ion exchange through specific channels located along the neural membrane (Brady, Siegel, Albers, & Price, 2011). This figure is modified from (Khanacademy, 2017).	11
Figure 2-3 EEG recording that shows the electrical charges resulting from the activity of the brain cells.	13
Figure 2-4 EEG band frequency corresponding to signals recorded from 14 electrodes and scalp maps generated by EEGLAB toolbox (Neuroscience, 2016).	13
Figure 2-5 A block diagram showing components of an artificial neuron.	19
Figure 2-6 A tree diagram of a data space with 11 samples of letters (right) and the corresponding nested clusters (left).	26

Figure 3-1 (a) Structure of a biological neuron, figure was modified form (Pearson , 2005); (b) Artificial spiking neuron which receives input spike trains, processes them and produces output spikes.	35
Figure 3-2 The LIFM of a spiking neuron. (a) Schematic representation; (b) Showing an input train of spikes (top row), the emitted output spikes (second row) and the membrane potential changes over time. Figure from (Kasabov, 2014).	37
Figure 3-3 Representation of one synaptic connection in a pSNM.	39
Figure 3-4 The multinodular development architecture of the NeuCube. Figure from (Kasabov, et al., 2016).	47
Figure 3-5 A functional diagram of the NeuCube SNN architecture, consisting of: input spatio-temporal data encoding module, 3-D SNN model and the STDP learning, output module for classification/regression, and gene regulatory network (GRN) module. Figure is modified from (Kasabov, 2012b).	48
Figure 3-6 An example of encoding EEG data into sequence of positive (black) and negative spikes (red) using the TBR algorithm that is the format of the input data into the NeuCube SNN architecture. The image shows the first 500 data points of one EEG channel (the Cz channel) from (Capecci, et al., 2016).	50
Figure 4-1 A schematic representation of the NeuCube framework for STBD mapping, learning, visualising, and classifying.	59

Figure 4-2 Unsupervised training iterations in SNN models by the leave-one-out method. For n samples, the SNN model is initialised n times, and trained by a fold of different $(n - 1)$ samples.	60
Figure 4-3 The SNR ranking (on the y-axis) of top voxels (on the x-axis) related to (a) the affirmative <i>versus</i> negative sentences and (b) the pictures <i>versus</i> sentences. The top voxels were selected according to their SNR values that were greater than a threshold= 0.4.	66
Figure 4-4 The fMRI data dimension of one person is defined by the maximum value of x, y, and z coordinates of voxels that forms a volume size of $51 \times 56 \times 8$. In this dimension, 5062 voxels are captured. These voxel coordinates are mapped into an SNN model. The selected top-informative voxels in Table 4-1 for each experiment are used as input variables and their ROIs are shown in the text boxes. (a) Affirmative <i>versus</i> negative sentences; (b) Pictures <i>versus</i> sentences.	68
Figure 4-5 The coordinates of the top-informative voxels in Table 4-1 are transferred to the Talairach and used as input variables, shown as circles along with the ROIs (as text in the boxes) for (a) affirmative <i>versus</i> negative sentences and (b) pictures <i>versus</i> sentences.	69
Figure 4-6 (a) The initial connections in the SNN model; (b) The learnt connections (absolute connection weights > 0.08) after STDP unsupervised learning using fMRI data of both affirmative and negative sentences (20 fMRI samples for affirmative sentence and 20 fMRI samples for negative sentences). The SNN models are mapped using the Talairach atlas with an allocation of 20 input voxels (Experiment A).	70

Figure 4-7 (a) The learnt connections (absolute connection weights >0.08) in an SNN model when only the fMRI samples of affirmative sentences were used (20 fMRI samples); (b) The learnt connections in an SNN model when only the fMRI samples of negative sentences were used (20 fMRI samples). 71

Figure 4-8 (a) Connectivity of an SNN model trained on fMRI data related to seeing pictures; (b) Connectivity of an SNN model trained on fMRI related to reading sentences; (c) 2-D coronal projection of the connectivity of the SNN model from Figure 4-8 (a); (d) 2-D coronal projection of the SNN model from Figure 4-8 (b). Connection weights >0.08 72

Figure 4-9 2-D coronal projection of the adapted connections in the SNN models after unsupervised learning with two data sets related to affirmative sentences (a) and negative sentences (b). The positive connections are shown in blue and negative ones in red (absolute connection weights >0.08). 74

Figure 4-10 2-D coronal projection of the adapted connections in the SNN models after unsupervised learning with two data sets related to (a) seeing pictures and (b) reading sentences. The positive connections are shown in blue and negative ones in red (absolute connection weights >0.08). 75

Figure 4-11 Visualisation of the SNN model connectivity of the H group (absolute connection weights >0.08) after the STDP learning with EEG data of 26 features (channels) for GO and NOGO trials. The average of the connection weights in each trained SNN model is also presented. The blue lines are positive (excitatory) connections, while the red lines are negative (inhibitory) connections. The thickness of the lines identifies the weight of the connection. The 1471 neurons of the SNN model are spatially mapped according to the

Talairach atlas (Koessler, et al., 2009). The SNN models are visualised in 3-D (top) and 2-D coronal projection (bottom).....	84
---	----

Figure 4-12 Visualisation of the SNN model connectivity of the MMT group (absolute connection weights>0.08) after the STDP learning with EEG data of 26 features (channels) for GO and NOGO trials. The average of the connection weights in each trained SNN model is also presented. The blue lines are positive (excitatory) connections, while the red lines are negative (inhibitory) connections. The thickness of the lines identifies the weight of the connection. The 1471 neurons of the SNN model are spatially mapped according to the Talairach atlas (Koessler, et al., 2009). The SNN models are visualised in 3-D (top) and 2-D coronal projection (bottom).....	85
---	----

Figure 4-13 Visualisation of the SNN model connectivity of the OP group (absolute connection weights>0.08) after the STDP learning with EEG data of 26 features (channels) for GO and NOGO trials. The average of the connection weights in each trained SNN model is also presented. The blue lines are positive (excitatory) connections, while the red lines are negative (inhibitory) connections. The thickness of the lines identifies the weight of the connection. The 1471 neurons of the SNN model are spatially mapped according to the Talairach atlas (Koessler, et al., 2009). The SNN models are visualised in 3-D (top) and 2-D coronal projection (bottom).....	86
--	----

Figure 5-1 A block diagram of the dynamic spatio-temporal clustering method using SNN.	97
---	----

Figure 5-2 Cohesion and separation of two neighbour clusters in an SNN model, where cluster centres are denoted by features 1 and 2.	101
---	-----

Figure 5-3 Silhouette measure exemplified on two clusters.	102
Figure 6-1 A step-wise visualisation of the dynamic cluster evolution corresponding to the 26 EEG channels of 21 healthy subjects in a GO task during unsupervised learning in an SNN model. The total number of time frames is $21 \times 75 = 1575$. The first two clusters are created at the 8 th time point of the EEG data that are associated with Fz and FCz channels.	109
Figure 6-2 A step-wise visualisation of the dynamic cluster evolution corresponding to the 26 EEG channels of 29 MMT subjects in a GO task. The total number of time frames is $29 \times 75 = 2175$. The first two clusters are created at the 11 th time point of the EEG data that are associated with Fp2 and Fp1 channels.	110
Figure 6-3 A step-wise visualisation of the dynamic cluster evolution corresponding to the 26 EEG channels of 18 OP subjects in a GO task. The total number of time frames is $18 \times 75 = 1350$. The first cluster is created at the 2 nd time point of the EEG that is associated with the Fp2 channel.	111
Figure 6-4 A step-wise visualisation of the dynamic cluster evolution corresponding to the 26 EEG channels of 21 healthy subjects in a NOGO task. The total number of time frames is $21 \times 75 = 1575$. The first cluster is created at the 2 nd time point of the EEG data that is associated with the FC3 channel.	112
Figure 6-5 A step-wise visualisation of the dynamic cluster evolution corresponding to the 26 EEG channels of 31 MMT subjects in a NOGO task. The total number of time frames is $31 \times 75 = 2325$. The first cluster is created at the 4 th time point of the EEG data that is associated with the C3 channel.	113

Figure 6-6 A step-wise visualisation of the dynamical cluster evolution corresponding to the 26 EEG channels of 18 OP subjects in a NOGO task. The total number of time frames is $18 \times 75 = 1350$. The first cluster is in the 9 th time point of the EEG data that is associated with the F8 channel.....	114
Figure 6-7 There are 26 clusters which are centred at the input neurons corresponding to the EEG channels. The size of the clusters changes while the SNN models are training on EEG data of the H, MMT and OP subjects in GO trials.	119
Figure 6-8 There are 26 clusters which are centred at the input neurons corresponding to the EEG channels. The size of the clusters changes while the SNN models are training on EEG data of the H, MMT and OP subjects in NOGO trials.....	123
Figure 6-9 The minimum, maximum and the mean of the number of neurons that belong to each cluster. The dynamic clustering was performed 10 times for each group of subjects using random order of the subject data presentations.	126
Figure 6-10 The standard deviation σ is reported for healthy (in blue), MMT (in red) and OP (in green). A higher σ value (mostly observed in MMT and OP groups) may represent less stability in cluster size evolved by different order of sample presentation. Blue, red and green colours represent respectively H, MMT and OP groups.....	127
Figure 6-11 Validity measurement of the clusters generated in the case study of EEG data with 26 channels from three groups of subjects (H group in blue, MMT group in red, and OP group in green). The Silhouette value was measured for every neuron in a cluster. Then the	

Silhouette values were averaged over all the neurons in a cluster and represented as a validity measure for this cluster.	127
Figure 6-12 The spike rates and PSP rates of the neurons within the FP1 cluster during unsupervised learning of EEG data (blue: H group, red: MMT group, and green: OP).....	129
Figure 6-13 Dynamic patterns of one cluster (EEG channel Fz) against an input spike train (st) corresponding to 5 samples in class 1 (H subjects). Each sample constitutes of 75 time points.	130
Figure 6-14 Dynamic patterns of one cluster (EEG channel T4) against an input spike train (st) corresponding to 5 samples in class 1 (H subjects). Each sample constitutes of 75 time points.	131
Figure 6-15 The dynamic patterns of the average PSP rates from 26 clusters during the learning process with EEG samples from classes H (in red) and class OP (in blue).	135
Figure 6-16 For 26 clusters, the local maximum of the potential value, $P_{max}(t)$ are plotted as dots in time t for all samples in class H (red) and class OP (blue). The $P_{max}(t)$ values show discriminative patterns between class H and class OP in EEG variables with small p -value (measured by a t-test) as shown in Table 6-4.....	138
Figure 6-17 The area under the curve of PSP rates for 26 clusters for all the samples in class H (red) and class OP (blue). As shown in Table 6-4, different discriminative power between the samples that belong to class H <i>versus</i> OP have been observed.	139

Figure 6-18 The midrange of the PSP rates corresponding to 26 clusters for all samples in H (red) class and OP (blue) class. As shown in Table 6-4, the midrange values have shown different discriminative power between samples belong to class H versus OP.	139
Figure 7-1 A block diagram of the personalised modelling that I proposed for integrated static and dynamic data. An SNN is trained with STBD samples that are found using the proposed DWWKNN method (the new person x_i is excluded from training).....	145
Figure 7-2 The SNR values for the 20 variables of static data.....	151
Figure 7-3 The proposed personalised modelling user interface for integrated static-dynamic data, exemplified using a case study of EEG data and static clinical data. For a selected subject id: 4, the relevant samples to it is a cluster of the common samples between C_s and $CSTBD$ (green bar lines) defined using the DWWKNN.....	154
Figure 7-4 Data samples were ranked according to the integrated static-dynamic similarity to a new data from two H samples id: 1 and id: 15 in (a) and (b) respectively. The EEG data of the neighbour samples (shown in green) are used for training of the PSNN models.	154
Figure 7-5 Data samples were ranked according to the integrated static-dynamic similarity to the new data from two MMT samples id: 45 and id: 31 in (a) and (b). The EEG data of the neighbour samples (shown in green) are used for training of the PSNN models.....	155
Figure 7-6 Data samples were ranked according to the integrated static-dynamic similarity to the new data from two OP samples id: 58 and id: 61 in (a) and (b). The EEG data of the neighbour samples (shown in green) are used for training of the PSNN models.....	155

Figure 7-7 The PSNN models were created for 6 randomly selected persons from H (a-b), MMT (c-d), and OP (e-f) groups. Each PSNN model was trained by the closest samples to the corresponding person.157

List of Tables

Table 2-1 Different techniques for measuring the distance. 24

Table 4-1 Informative voxels are selected sing an SNR feature selection from two fMRI data sets. The voxels were selected due to their SNR values were higher than a threshold= 0.4. 67

Table 4-2 Average connection weights around each input neuron in the trained SNN models from Figure 4-7 related to affirmative (A) and negative (N) sentences. The sum of the average connection weights across all the input neurons in each SNN model is reported in the last column. 76

Table 4-3 Average connection weights of the trained SNN models for each iteration. Two SNN models were trained over 40 iterations (20 iterations for Affirmative (A) and 20 for negative (N) sentences) using different folds of samples. At each iteration, one sample was taken out from the training and the model was trained by the remaining samples. This procedure was repeated for all the 20 samples for both affirmative and negative sentences. The *p*-value represents that the trained SNN models of affirmative and negative sentences are statistically significant 76

Table 4-4 Average connection weights of the trained SNN model for each iteration. Two SNN models were trained over 80 iterations (40 iterations for pictures and 40 for sentences)

using different folds of samples. At each iteration, one sample was taken out from the training set and the model was trained by the remaining samples. This procedure was repeated for all the samples in both classes. The p -value represents that the trained SNN models of pictures and sentences are statistically significant..... 76

Table 4-5 Classification accuracy of the affirmative sentences (class C1) *versus* negative sentences (class C2) *via* an SNN model using the LOOCV method. The results of conventional machine learning methods along with the SVM classification results from (Behroozi & Daliri, 2014) are also reported. The experiment is done on a total number of 40 samples (20 samples per class)..... 78

Table 4-6 Classification accuracy of pictures (class C1) *versus* sentences (class C2) obtained *via* an SNN model using the LOOCV method. The results of the conventional machine learning methods are also reported. The experiment is done on a total number of 80 samples (40 samples per class). 78

Table 4-7 EEG data sets for the three experimental sessions to compare the brain activity patterns of the H, MMT, and OP subjects in a GO/NOGO task. Due to the quality of the data, some participants' data were omitted from the experimental sessions. 82

Table 4-8 Average connection weights of the SNN models trained over 42 iterations (21 iterations for class GO and 21 for class NOGO) using different folds of samples. At each iteration, one sample was taken out and the model was trained by the remaining samples. This procedure was repeated for all the 21 samples in both GO and NOGO. The p -value represents that the trained SNN models of GO and NOGO are statistically significant with a high confidence, greater than 99%. 87

Table 4-9 Average connection weights of the SNN models trained over 60 iterations (29 iterations for class GO and 31 iterations for class NOGO) using different folds of samples. At each iteration, one sample was taken out and the model was trained by the remaining samples. This procedure was repeated for all the samples in both GO and NOGO. The p -value represents that the trained SNN models of GO and NOGO are statistically significant with a high confidence, greater than 99%.	87
Table 4-10 Average connection weights of the SNN models trained over 36 iterations (18 iterations for class GO and 18 for class NOGO) using different folds of samples. At each iteration, one sample was taken out and the model was trained by the remaining samples. This procedure was repeated for all the samples in both GO and NOGO. The p -value represents that the trained SNN models of GO and NOGO are statistically significant with a high confidence, greater than 99%.	87
Table 4-11 The EEG data classification accuracy (in %) using the LOOCV method. Correctly classified samples are shown on the diagonal of the confusion matrix, shown in bold.	88
Table 4-12 The EEG data classification in conventional methods using the LOOCV method. The optimal parameter setting is reported in Appendix C, Tables C-12 and C-13.	88
Table 5-1 The dynamic spatio-temporal clustering algorithm, called at each time point t of the STDP learning.....	99
Table 6-1 EEG data samples used for dynamic clustering to study the activity patterns of H, MMT, and OP subjects in a GO/NOGO task.	106

Table 6-2 The size of all the EEG channel clusters (plotted in Figure 6-7) was averaged over all the time points during the STDP learning in SNN models of H, MMT and OP groups in GO task.....	124
Table 6-3 The size of all the EEG channel clusters (plotted in Figure 6-8) was averaged over all the time points during the STDP learning in SNN models of H, MMT and OP groups in NOGO task.	124
Table 6-4 A t-test measure was applied to the <i>Pmax</i> (left), the area under the curve of PSP (middle) and the midrange of the PSP (right) to identify how two classes H and OP are statistically significant. EEG channel 17 has the lowest <i>p</i> -value, representing the highest discriminative power between the samples that belong to different classes.	140
Table 6-5 The classification accuracy between EEG samples in H and OP in the GO task obtained when using the all the EEG variables <i>versus</i> using the 8 top-informative variables.	141
Table 7-1 The proposed DWWKNN algorithm.	148
Table 7-2 Five samples are randomly selected from each groups of subjects (in total 15 samples). V1: age; V2: gender (0 is male and 1 is female); V3: level of education; V4: life time nicotine consumption; V5: illness; V6: history of overdose; V7: times of hospitalised; V8: Legal charge; V9: days being in jail; V10: Methadone dose; V11: alcohol consumption in last 30 days; V12: sedative consumption in last 30 days; V13:level of anger; V14:cannabis consumption; V15: hallucinogens consumption; V16: taking ecstasy; V17:amphetamine	

consumption; V18: barbiturate consumption; V19: heroin; and V20: class label of subject groups (1 is H, 2 is MMT and 3 is OP).....150

Table 7-3 Classification accuracy obtained *via* SNN-based personalised modelling *versus* using a global SNN model. The number of correctly classified samples in each class is shown on the diagonal of the tables. For each person x , one PSNN model is trained by EEG of subjects who have similar integrated static and dynamic data to x and then tested by the EEG data of x , which was unseen during the learning.158

Table 7-4 Comparison of the classification accuracy (in %) obtained using a global SNN, PSNN and conventional methods including: SVM and MLP. The MLP optimal parameters that resulted the best classification accuracy were found after performing the experiments several times with different parameter setting (learning rate (LR) = [0.01, 0.5], momentum (M) = [0.1, 0.9], training iteration (TI) = [500, 1500], and number of hidden layer (HL) = [2, 6]).....158

Table 7-5 Classification accuracy obtained using the PSNN models with different clustering approaches (DWWKNN, WWKNN, WKNN and KNN) for selecting the nearest neighbour samples to an individual.....159

List of Abbreviations

AD	Alzheimer Disease
AI	Artificial Intelligence
ANN	Artificial Neural Network
BCI	Brain Computer Interface
BOLD	Blood Oxygenation Level Dependence
BSA	Ben's Spikes Algorithm
CALC	Calcarine Sulcus
CNN	Convolutional Neural Network
DBNN	Deep Belief Neural Network
DENFIS	Dynamic Evolving Neural-fuzzy Inference System
DNN	Deep Neural Network
DSL	Deep Structured Learning
DWWKNN	Dynamic Weighted-Weighted Distance K-nearest Neighbours
ECM	Evolving Clustering Method
EEG	Electroencephalogram
ESNN	Evolving Spiking Neural Network
ESOM	Evolving Self-organizing Maps
FIN	Feature Interaction Network
FMRI	Functional Magnetic Resonance Imaging
GLM	General Linear Method
H	Healthy
HDR	Hemodynamic Response Rate
HMM	Hidden Markov Models
KNN	K-Nearest Neighbours
LDA	Linear Discriminant Analysis
LDLPFC	Left Dorsolateral Prefrontal Cortex
LFEF	Left Frontal Eye Fields
LIFG	Left Inferior Frontal Gyrus
LIFM	Integrate-and-fire Model
LIPL	Left Inferior Parietal Lobule
LIPS	Left Intraparietal Sulcus

LIT	Left Inferior Temporal Lobule
LOOCV	Leave-one-out Cross Validation
LOPER	Left Opercularis
LPPREC	Left Posterior Precentral Sulcus
LSGA	Left Supramarginal Gyrus
LSM	Liquid State Machines
LSPL	Left Superior Parietal Lobule
LT	Left Temporal Lobe
LTRIA	Left Triangularis
MCI	Mild Cognitive Impairment
MLP	Multilayer Perceptron
MLR	Multiple Linear Regression
MMT	Methadone Maintenance Treatment
OP	Opiate Users
PCA	Principle Component Analysis
PSNM	Probabilistic Spiking Neural Model
PSNN	Personalised Spiking Neural Networks
RDLPFC	Right Dorsolateral Prefrontal Cortex
RELU	Rectified Linear Units
RFEF	Right Frontal Eye Fields
RIPL	Right Inferior Parietal Lobule
RIPS	Right Intraparietal Sulcus
RIT	Right Inferior Temporal Lobule
RNN	Recurrent Neural Networks
ROPER	Right Opercularis
RPPREC	Right Posterior Precentral Sulcus
RSGA	Right Supramarginal Gyrus
RSPL	Right Superior Parietal Lobule
RT	Right Temporal Lobe
RTRIA	Right Triangularis
SMA	Supplementary Motor Areas
SNN	Spiking Neural Networks
SOM	Self-organizing Maps
STBD	Spatio-temporal Brain Data
STDM	Spatio-temporal Data Machine
STDP	Spike Time Dependent Plasticity
STING	Statistical Information Grid-based Method
SVM	Support Vector Machine
TBR	Threshold-based Representation

Attestation of Authorship

I hereby declare that this submission is my own work and that, to the best of my knowledge and belief, it contains no material previously published or written by another person (except where explicitly defined in the acknowledgements), nor material which to a substantial extent has been submitted for the award of any other degree or diploma of a university or other institution of higher learning.

Acknowledgements

I have the pleasure to acknowledge the support of many people who have supported, inspired and helped me throughout my PhD journey. First and foremost, I am very grateful to my primary supervisor *Professor Nikola Kasabov* who gave me the chance to undertake my PhD program and become a member of a globally-known and high-profile research group, the Knowledge Engineering and Discovery Research Institute “KEDRI” at Auckland University of technology. I am also very thankful to my second supervisor *Professor Jie Yang* and my additional supervisor *Professor Reinhard Klette* for their support for the progression of this thesis.

My sincere appreciation also goes to *Joyce D’Mello*, the Administration Manager and the glow of KEDRI. Her encouraging words and most cordial support helped me to overcome all the arduous moments of this journey.

A big acknowledgement also goes to my truelove *Dr Reza Enayatollahi* who has patiently and lovingly held my hand, supported me and kept me excited throughout my research and life.

I am also very grateful of having my sister *Zohreh Doborjeh* who brought me happiness to New Zealand, given me so many amazing memories and illuminated my days at KEDRI.

I acknowledge my first friend in New Zealand, *Dr Elisa Capecci*, who not only shared her knowledge and experiences with me, but also displayed her kindness like an amazing sister.

I would like to thank *Professor Robert Kydd* and *Dr Bruce Russell* from the University of Auckland, New Zealand and, *Dr Grace Wang* from Auckland University of Technology for providing me with EEG data, which was used as one of the case studies in my thesis.

I wish to thank my friends *Barry Dowdeswell* and *Michaela Selway* for proofreading my thesis. I believe their final contribution is of inestimable value to it.

Last but not least, I would like to thank my *parents* and my siblings *Mahsa* and *Milad* who are long away but always in my heart.

Chapter 1 Introduction

1.1 Rationale and Motivation

Large amounts of Spatio-temporal Brain Data (STBD) are being recorded in different areas of study including neuroscience, neurology, psychology, and so forth. STBD record cognitive brain functions, which are involved in the processing of afferent information produced by internal and/or external stimuli. Scrutinising such multivariate data by computational approaches has led to the proposal of neuroinformatics, where the models are inspired by neural systems.

The worldwide demand for modelling and understanding the underpinning of functional processes in STBD has propelled the development of various analytical methods. The majority of current statistical and Artificial Neural Network (ANN) machine learning techniques often create models by separately processing the *spatial* and *temporal* components. This usually results in losing some informative spatio-temporal correlations in real-life applications of STBD. Hence, the accuracy of the output prediction/classification might not be quite substantial.

Besides the model accuracy, the model interpretability is also of crucial importance in machine learning. This refers to understanding the relationships between the model features and the predicted outputs, which has not been investigated in depth. The higher the interpretability of a model, the easier it is for someone to comprehend why certain decisions

(output predictions) were made. This allows for knowledge discovery in the models and contributes to the understanding of interactions in the model that have controlled an output to occur. Nevertheless, the extant analytical methods develop models on data without investigating the model learning patterns itself. Hence, they act as black-box information processing systems that solve a problem without discovering the causal relationships that have triggered the output.

The brain is a highly interactive and deep learning network. Understanding of the STBD is a complex task as the temporal features manifest complicated causal relations between the spatially distributed neural sources in the brain. To model such interactions, Spiking Neural Networks (SNNs) can be considered as suitable models that incorporate both spatial and temporal components into an operation. Therefore, their compact representation of space and time allows for learning of “hidden” spatio-temporal correlations in STBD.

This thesis is based on *brain-inspired*¹ SNN architecture for modelling and analysing of STBD towards improving the classification accuracy. This research also contributes to an improved level of interpretability of learning patterns in SNN models when compared with *conventional*² methods. The STBD case studies here are real-life Electroencephalogram (EEG) and Functional Magnetic Resonance Imaging (fMRI) data sets which were measured prior to this study by other institutions who are acknowledged in this thesis.

¹ Brain-inspired refers to the mathematical implementation of a method gets inspiration from neuroscience research on brain activity.

² Conventional methods refer to methods that have been proposed for data analysis, besides neural network techniques.

1.2 Aims of this Thesis and Research Questions

The primary aims of this thesis are summarised as follows:

- 1) Feasibility analysis of SNN architecture on case studies of real-life STBD.
 - To design optimal SNN models that can learn from STBD with respect to both space and time components.
 - To achieve an improved classification accuracy when compared with conventional AI and machine learning methods.
 - To interpret the spatio-temporal interactions, captured during the learning process in SNN models.
- 2) Development of new methods for knowledge discovery in SNN evolutionary learning patterns.
 - To develop a new method for dynamic spatio-temporal clustering of patterns generated during unsupervised learning in SNN models. This is to study the evolving patterns in SNN models, which has not been interpreted in depth (presented in Chapter 5).
 - Knowledge discovery through assessment of dynamic clustering patterns in SNN models, which results in detecting a set of discriminative features (feature selection). This contributes to improve the model interpretability and accuracy (presented in Chapter 6).
- 3) Proposal of a personalised SNN model.

- To develop a personalised modelling system based on integrated computational methods and SNN models, when both static data and spatio-temporal data from an individual are available.
- To improve the classification accuracy for an individual's outcome through personalisation of the SNN model.

During the progression of this thesis, the following research questions (RQ) will be addressed:

RQ 1. How to optimally design SNN architectures to model, learn and analyse different types of STBD and to precisely capture both spatial and temporal components?

RQ 2. How does spatio-temporal clustering of the evolving patterns in SNN models lead to knowledge discovery about the “hidden” dynamic behaviour (learning patterns) in SNN models during unsupervised learning from spatio-temporal streams over time?

RQ 3. When both static and dynamic datasets are available for an individual, how SNN models can be personalised towards the best possible diagnosis/prognosis outcomes of an individual?

1.3 Thesis Structure

This thesis consists of eight chapters which are outlined as follows:

Chapter 1 states the research motivations, goals, and research questions and outlines methods to address these questions.

Chapter 2 reviews the research about how a biological neuron functions and introduces two neuroimaging techniques for measuring the neural activities in the brain. This section is then followed by a review on some analytical methods for classification and clustering.

Chapter 3 discusses how the known mechanisms of neurons can be computationally modelled by artificial spiking neurons in machine learning. Next, this chapter introduces a brain-inspired SNN architecture, called NeuCube for modelling, learning, and understanding of STBD.

Chapter 4 demonstrates a feasibility analysis of the NeuCube SNN architecture on two case studies of cognitive data: fMRI and EEG. In this study, I designed SNN models of STBD that resulted in an improvement of the classification accuracy when compared with conventional machine techniques.

Chapter 5 represents an original contribution to the NeuCube SNN architecture by proposing a new method for dynamic spatio-temporal clustering of learning patterns in SNN models whilst training with STBD streams over time.

Chapter 6 investigates the proposed clustering approach through an empirical study on EEG data. This chapter represents knowledge discovery in SNN evolutionary patterns whilst incrementally learning from streaming EEG. The assessment of these evolutionary patterns allows us to identify the informative features (STBD variables) in SNN that lead to an improvement in classification accuracy.

Chapter 7 proposes a new personalised modelling system based on SNN architecture and a new clustering method for integrated static and dynamic STBD.

Chapter 8 summaries the thesis achievements, key findings and contributions, in particular, to SNN research. Future directions are also suggested.

Figure 1-1 illustrates a bird's-eye view of the thesis and its different components towards addressing the research questions. As illustrated here, my contributions are in two steps: (1) feasibility study of SNN on STBD and (2) new SNN-based method development. The first step is an optimal design of SNN models to better study the complex spatio-temporal interactions among the STBD variables. The designed models were tested using two types of STBD: EEG and fMRI. The second step refers to my original contributions to the NeuCube architecture by proposing two new approaches for scrutinising the SNN learning patterns. These approaches are (1) a new dynamic spatio-temporal clustering of the “hidden” learning patterns and (2) a new personalised modelling framework when both static and dynamic data are available for an individual.

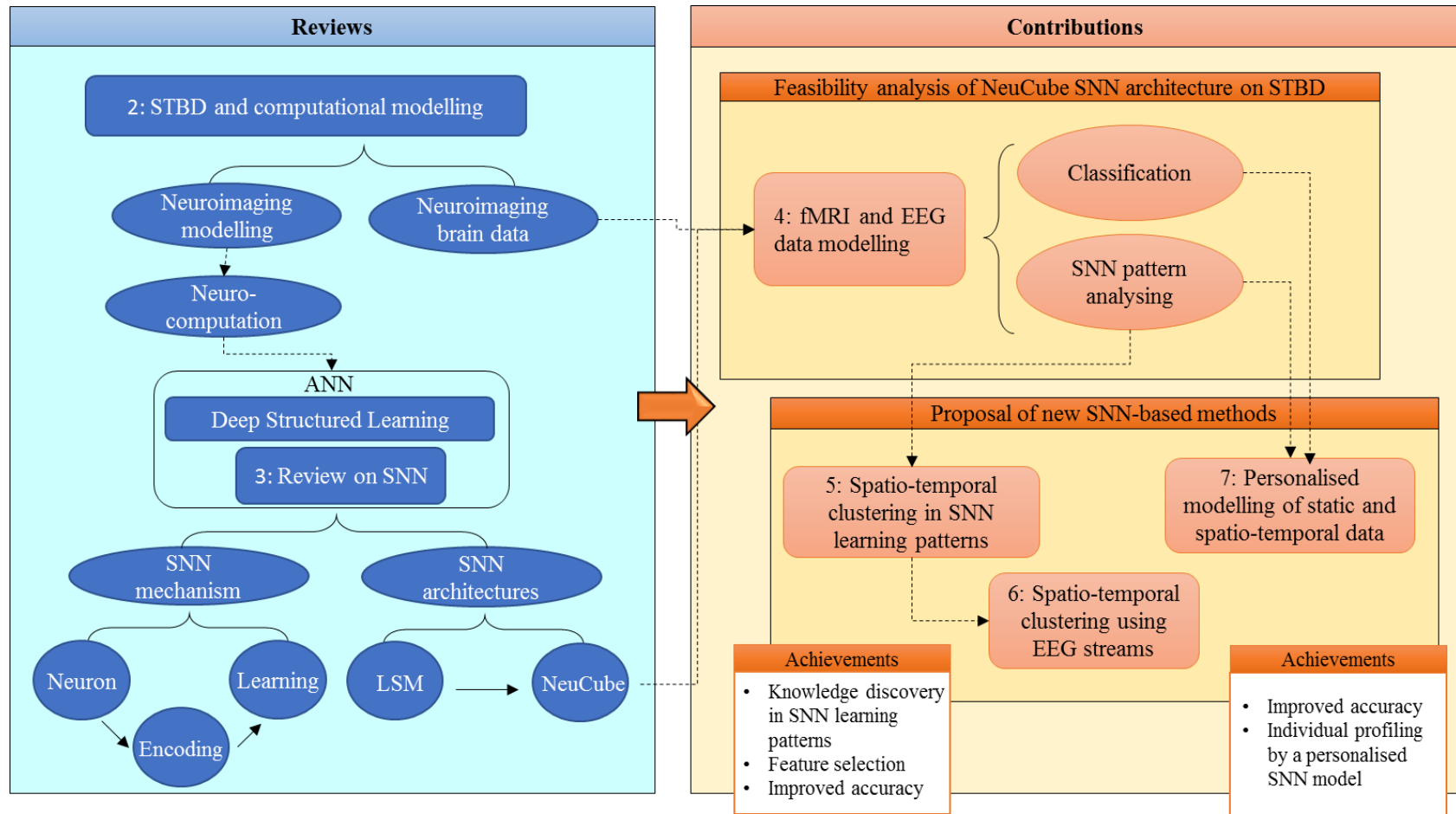


Figure 1-1 A bird's-eye view of the thesis structure.

Chapter 2 Spatio-temporal Brain Data and Analytical Methods

2.1 Introduction

This chapter will first review in Section 2.2 what is known in neuroscience research about the brain, which performs as a complex information processing system. Then, Sections 2.3 and 2.4 will review the two important types of STBD measuring techniques, Electroencephalogram (EEG) and functional Magnetic Resonance Imaging (fMRI) data that are used as case study problems in this thesis. Section 2.5 reviews some classification techniques and refers to their limitations when dealing with STBD. Then, clustering is discussed as one of the main techniques for understanding of the STBD.

2.2 Information Processing in the Human Brain

The brain is the most complex organ in the human body that contains approximately 86 billion nerve cells, known as neurons (Azevedo, Carvalho, Grinberg, & Farfel, 2009). Neurons are the fundamental information processing units that are interconnected to construct a complex neural network. Neurons use biochemical reactions to receive, process, store and transmit input information. A typical neuron consists of three major parts: the cell body, the dendrites and the axon. A neuron's cell body (also called soma) contains the nucleus and most of the main organelle. Soma accomplishes the continuous maintenance of the neuron's functionality. This retains a certain ion concentration in the membrane to

actively transfer sodium (Na^+) ions from intra-cellular fluid to the extra-cellular fluid. Potassium (K^+) ions flow in the opposite direction from the outside to the inside of the soma. For ion transportation across the soma membrane, several ion channels that contain specialised proteins are embedded in the membrane. These channels provide an outward flow of potassium to the extra-cellular fluid, while sodium moves inwards into the soma; thus, the opposing directions of ions with different strength levels produce an electrical potential across the soma membrane. Figure 2-1 illustrates the information flow in a neuron, while Figure 2-2 shows a slice of the soma membrane which has several channels for ion transportation. As shown in Figure 2-1, at the beginning of the soma, branch-like extensions (called dendrites) are positioned to receive chemicals from other neurons *via* synapses. These chemicals are transformed into electrical impulses and then transmitted to the soma.

The firing state of each neuron is controlled by the axon, which is located at the end of the soma. If the total force of the signals entered into a neuron surpasses the limit of the axon, the neuron fires and triggers an action potential down to the axon terminals. Synapses are embedded to store the neurotransmitter chemicals at the end of the axon terminals. These synapses are attached to the dendrites of the neighbour neurons and allow the transmission of information from one sending neuron (*presynaptic*³ neuron) to other receiving neurons (*postsynaptic*⁴ neurons). The sending information contains action potential (approximately one millisecond in duration), which is created in the presynaptic neuron's axon by regenerative alterations in membrane potential, and acts as an energy that induces neural

³ Presynaptic neuron is delivering the “message” across the synapse to the postsynaptic neuron.

⁴ The postsynaptic neuron is the “receiver” of the neurotransmitter “message” from the presynaptic neuron.

activity (Hodgkin, Huxley, & Katz, 1952; Hall, 2015). The axon can accelerate propagation of electrical signals, if it is covered by myelin sheaths, which perform as insulators and prevent the dissipation of the depolarisation wave caused by an electrical spike triggered in the soma.

When a neuron sends a spike to the postsynaptic neuron through the axon, neurotransmitter chemicals are diffused into the synaptic cleft (a narrow gap between two neurons) and which reacted with receptor proteins of the postsynaptic neurons. The receptor activation allows the transfer of ions from the extra-cellular fluid of the presynaptic neuron to the postsynaptic cell. Different categories of transmission synapses, named excitatory and inhibitory, control the likelihood of the postsynaptic neuron to emit an action potential. Excitatory neurotransmitter increases the potential of the postsynaptic neurons to fire, while inhibitory neurotransmitters suppress the postsynaptic neuron from firing. Therefore, the efficacy of the postsynaptic response is not fixed but adjusted with respect to the released neurotransmitters. This principle is so-called *synaptic plasticity*, which enables learning and memorising in the brain. Comprehensive information about neurons can be found in the standard text book on the matter by Kandel (Kandel, Schwartz, Jessell, Siegelbaum, & Hudspeth, 2000) and (Fuchs, et al., 2012).

Hitherto, this chapter presents how the construction of a neuron enables it to exchange electrical signals among a series of interconnected neurons; thus, propagating the information. Afterwards, two main neuroimaging techniques for recording such neural activities will be discussed.

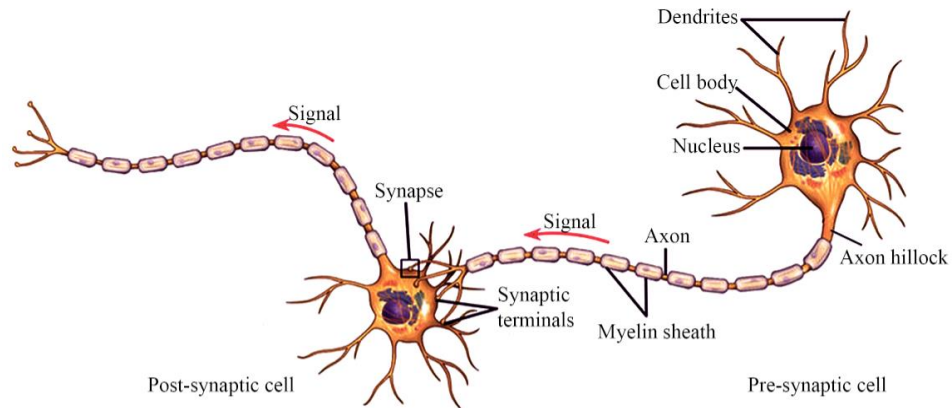


Figure 2-1 Information flows between neurons through the axon, which receives information from a presynaptic neuron and generates an action potential that is sent down to the synapses of the postsynaptic neuron. This figure is modified from (Marcella, 2011).

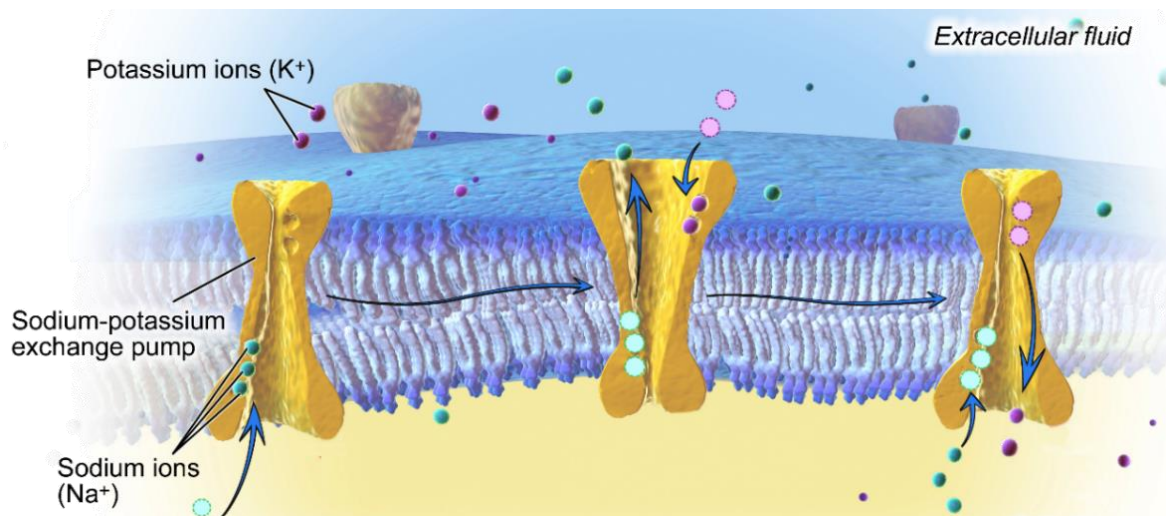


Figure 2-2 Sodium Na^+ and potassium K^+ ions move across the soma membrane through ion channels, this accomplishes electrochemical powers as a result of the ion exchange through specific channels located along the neural membrane (Brady, Siegel, Albers, & Price, 2011). This figure is modified from (Khanacademy, 2017).

A variety of techniques has been devised for recording brain dynamics, such as EEG and fMRI. An EEG is based on the magnetic and electrical activity of the brain that possesses a high temporal resolution (*i.e.* brain signals are recorded in a sequence of milliseconds) but unclear localisation, as the measurement is performed *via* a limited number of electrodes attached to the scalp. In contrast, fMRI data represents brain activity with a high spatial

localisation even though it uses a much lower temporal sampling rate. The principle of the EEG and fMRI data will be explained in the following Section 2.3 and Section 2.4.

2.3 EEG Data

EEG is a method for measuring STBD from cortical activity *via* a number of electrodes attached to the scalp that are connected to a computer interface system (Haas, 2003). The history of EEG measuring goes back to the moment that Richard Caton discovered the electrical nature of neural activity (Caton, 1875). Caton reported that he had used a galvanometer for detecting and measuring the electric impulses from the surface of mammals' brains (a rabbit and a monkey). In 1924, the first human EEG was recorded by Hans Berger (Niedermeyer & da Silva, 2005). By 1938, EEG had expanded as a widespread recognition technique by eminent scientists, leading to practical applications in diagnosis among many countries (Wiedemann, 1994). Figure 2-3 illustrates an EEG recording procedure, which collects brain cortical signals *via* a number of electrodes (also called channels). The EEG signal oscillation is rhythmic; thus, it is typically described in terms of bands of different frequencies as follows:

- Delta band in 0.5-3.5 Hz, at sleep and rest stages.
- Theta band in 3.5-7.5 Hz, at learning, memory and sensory motor processing.
- Alpha band in 7.5-12.5 Hz, at meditation.
- Beta band in 12.5- 30 Hz, at mental calculation, anticipation or tension.
- Gamma band in 30-60 Hz, at attention of sensory perception.

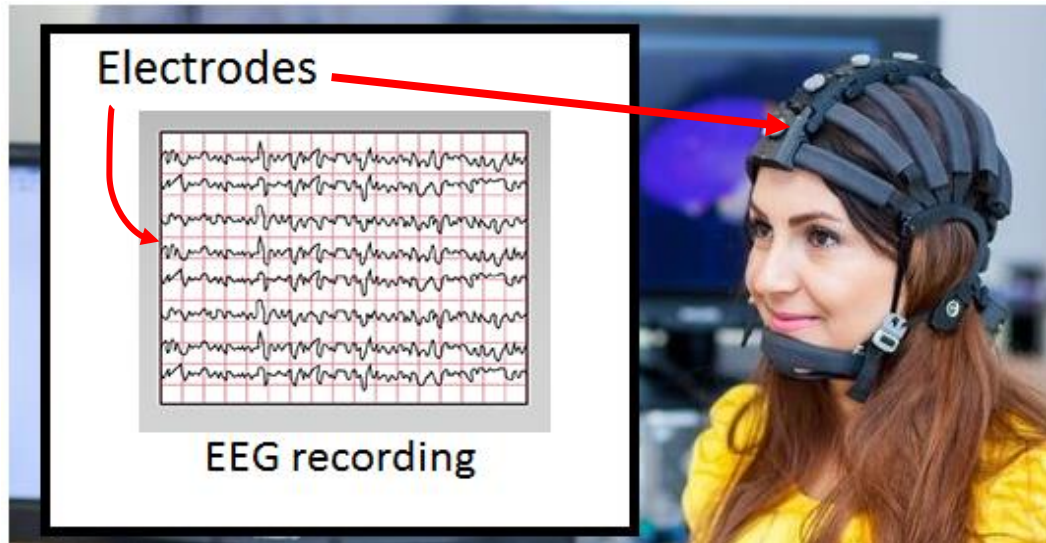


Figure 2-3 EEG recording that shows the electrical charges resulting from the activity of the brain cells.

Figure 2-4 shows the frequency and amplitude of an example of a 14-channel EEG. The signal amplitudes in three different frequencies (6 Hz, 10 Hz and 22 Hz) are shown in colour maps, where red represents a high power while blue is a low power.

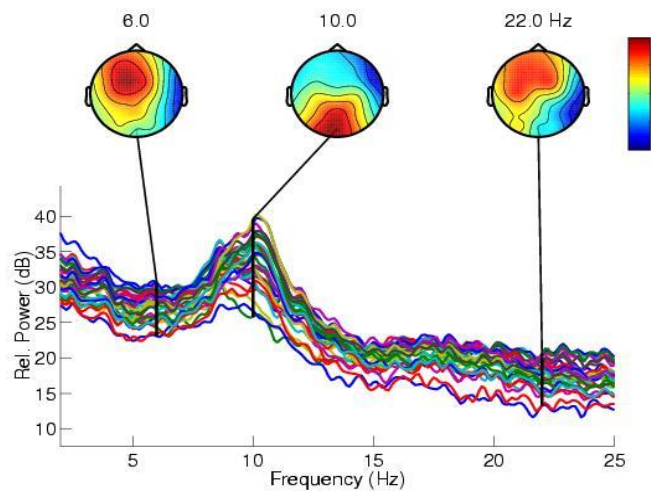


Figure 2-4 EEG band frequency corresponding to signals recorded from 14 electrodes and scalp maps generated by EEGLAB toolbox (Neuroscience, 2016).

2.4 fMRI Data

The spatio-temporal fMRI data is a susceptible indicator of blood flow changes influenced by neural activities. This relies on the statement that neural activity and the cerebral blood flows are highly correlated. Thus, evoked neurons demand a high level of oxygen carried by blood to start processing and firing (Logothetis, Pauls, Augath, Trinath, & Oeltermann, 2001). fMRI uses the Blood Oxygenation Level Dependence (BOLD) contrast method for observing the level of oxygenation in the blood. The BOLD context was first described by Ogawa (Ogawa, et al., 1992). It is a type of specialised brain scan that maps neural activity through imaging the blood flow changes, also called hemodynamic response rate (HDR), in relation to the energy/oxygen consumption within brain cells (Huettel, Song, & McCarthy, 2004). BOLD represents local increases in blood oxygenation as a direct effect of neurotransmitter chemicals that perform local neural signalling.

In the presence of a magnetic field, BOLD contrast is influenced by the paramagnetic nature of deoxyhemoglobin which affects the main magnetic field, leading to a local reduction in main field homogeneity. Paramagnetic deoxyhemoglobin in the blood possesses a stronger magnetic moment whilst oxyhemoglobin is diamagnetic and has little effect. This inhomogeneity of the magnetic field can be measured over time as different illuminations in fMRI data. Therefore, a high level of deoxyhaemoglobin decreases the fMRI intensity, while little deoxyhaemoglobin increases the image intensity.

In the context of neurobiology, HDR refers to the fast distribution of blood to activate neural tissues. If the brain neurons are constantly active with a high level of processing, cerebral

blood flow is vital to retain the neurons, astrocytes, and other cells of the brain. FMRI temporal resolution is limited by the slow rate of HDR time.

FMRI recording occurs over time at many small, three-dimensional areas called “voxels”. Each voxel is a numeric cube which represents the BOLD intensity of thousands of neurons over time. FMRI techniques are non-invasive and have been widely used in cognitive science and neuroscience, providing insight into brain structures and processes for researchers and clinicians (Lindquist, 2008; Liu, et al., 2014; Rodriguez, Anderson, Calhoun, & Adali, 2015; Siegelmann, 2015; Norman, Polyn, Detre, , & Haxby, 2006; Behroozi & Daliri, 2014).

There are numerous common objectives pursued in fMRI data analysis, including: localising the activated brain regions during a mental task, detecting the brain information pathways corresponding to functional activities, diagnosis or prognosis of disease or psychological states, and so forth. The next section will discuss some major analytical methods and their limitations when dealing with STBD.

2.5 Analytical Methods for STBD

Currently there is a huge amount of STBD collected from either healthy subjects or unhealthy subjects during and after treatment. EEG and fMRI have been extensively used for brain study through applying different computational methods. Some familiar tools to process EEG data include: EEGLAB (SCCN, 2017), LORETA (Loreta, 2017), PyEEG (PyEEG Reference Guide, 2010), and so forth. A review (Lotte, Ongedo, Lecuyer, Lamarche, & Arnaldi, 2007) on classification algorithms for STBD in Brain Computer Interface (BCI) has explored different categories, where the most important methods are linear classifiers, non-linear

Bayesian classifiers, neural networks and hybrid models. In the following, I present an overview on current classification methods for STBD.

2.5.1 Overview on Classification Methods

A. Linear Classifiers

Linear classifiers are based on assigning linear decision boundaries between the samples (feature vectors) of different classes. A variety of algorithms has been proposed so far, such as Support Vector Machines (SVMs) (Cortes & Vapnik, 1995; Raghavendra & Deka, 2014), Linear Discriminant Analysis (LDA) (Manly, McDonald, Thomas, McDonald, & Erickson, 2002) and regularised LDA (Guo, Hastie, & Tibshirani, 2007) that is adapted for high-dimensional data space. Both SVM and LDA have been widely used for classification of EEG (Costantini, et al., 2009; Subasi & Gursoy, 2010) and fMRI (Peltier, Lisinski, Noll, & LaConte, 2009). However, when dealing with complex STBD streams, samples cannot be linearly discriminable. This problem is called non-linear classification which cannot be handled by drawing straight discriminative lines in the data space.

The original SVM constructs a hyperplane for linear classification, however, a non-linear classification can be performed by applying a kernel (Cristianini & Shawe-Taylor, 2000) to the hyperplane that allows to transform the feature space to fit the hyperplane. Examples of kernels are polynomial and Gaussian function.

General Linear Method (GLM) (Friston, et al., 1994; Beckmann, Jenkinson, & Smith, 2003) is another type of statistical linear modelling of multivariate data that was used in several neuroimaging analytical tools such as Statistical Parametric Mapping (SPM) (Friston K. , Statistical Parametric Mapping, 1994).

B. Non-linear Classifiers

Non-linear Bayesian classifiers emerged for modelling the probability distributions of each class. Bayesian classifiers are based on probabilities of associated events (dependent) according to a conditional probability principle. An example of such a classifier is the Hidden Markov Model (HMM) (Eddy S. R., 1996; Chakraborty & Talukdar, 2016), which is a probabilistic model for temporal data by assigning probability distributions over sequences of observations. Using HMMs, input data can be classified by passing them through several states together with their transition probabilities as statistical measurements.

Several successful applications of HMMs have been developed in the field of speech recognition and classification (Katagiri & Lee, 1993; Rabiner, 1989). In the field of STBD analysis, research on EEG data classification (Obermaier, Guger, Neuper, & Pfurtscheller, 2001) has shown that the accuracy of BCI-based HMM outperformed the BCI-based LDA. In (Ou, et al., 2015), fMRI data were used for classification of different groups of patients by applying HMM, which generated different sequences of observations (states), based on which the specific test subject was classified. Research (Argunsah & Cetin, 2010) presented that HMMs for EEG classification were improved by using Principle Component Analysis (PCA) (Friston, Frith, Liddle, & Frackowiak, 1993), which is a dimensionality reduction approach that transfers data samples into a new space with a smaller dimension, where different orthogonal principle components perform as linear-subspace representations of the data.

When dealing with samples that are not linearly distributed, PCA transforms result in losing some of the information. Independent Component Analysis (McKeown, et al., 1998),

(Franchin, Tana, Cannata, Cerutti, & Bianchi, 2013) has been proposed to overcome the limitations of the non-linear subspaces in PCA.

HMMs performed well in various STBD classifications, however, when both spatial and temporal information are critical to be preserved and learnt, HMMs do not model the integrated spatial-temporal correlations in data. In addition, HMM is a parametric technique with a fixed number of states and a fixed topology with respect to the observations. The HMM model operates using discrete states and they consider only the previous known state, so deep-learning patterns cannot be captured. They are also lacking from brain-inspired learning to adjust the interconnections. Some limitations of HMMs are reviewed in (Chakraborty & Talukdar, 2016).

C. Artificial Neural Networks (ANNs)

ANNs have been proposed for solving classification or regression tasks in computational data processing. ANNs are constituted of a set of basic cells (called neurons) performing a defined mathematical operation connected and organised in layers, which process input information and assign decision boundaries between samples that belong to different classes. According to their computational units, they were categorised into three generations.

Rosenblatt proposed the first generation of ANNs (Rosenblatt, 1957), called perceptron which was based on McCulloch-Pitts neurons and was inspired by the concept of thresholding in biological neurons. A perceptron neuron is a computational unit with several inputs, each is associated with a weight that resembles synaptic efficiency. A perceptron integrates the inputs and fires if the synaptic weighted sum of inputs reaches a threshold. This

is computed using a step function to perform binary outputs (-1 or 1). This function is time-independent, meaning that the time in which the threshold is exceeded is not considered.

A single layer perceptron consists of one layer of neurons that are fully connected to the input data by weighted connections. An extended version of the perceptron is the Multilayer Perceptron (MLP) (Kruse, et al., 2013) which usually consists of an input layer, one or more hidden layers and an output layer. The step function can be replaced by a linear function, which produces a range of activations, so it is not limited to a binary output.

The second generation of ANNs is related to improving the computational units by including an activation function. In contrast to a fixed threshold value to determine the output, here an activation function (such as non-linear sigmoid) produces outputs which are proportional to the inputs; thus, performing non-linear classifications. Figure 2-5 illustrates a block diagram of an artificial neuron (right) and a simple small network (left) with two input neurons, two hidden layers, and one output neuron.

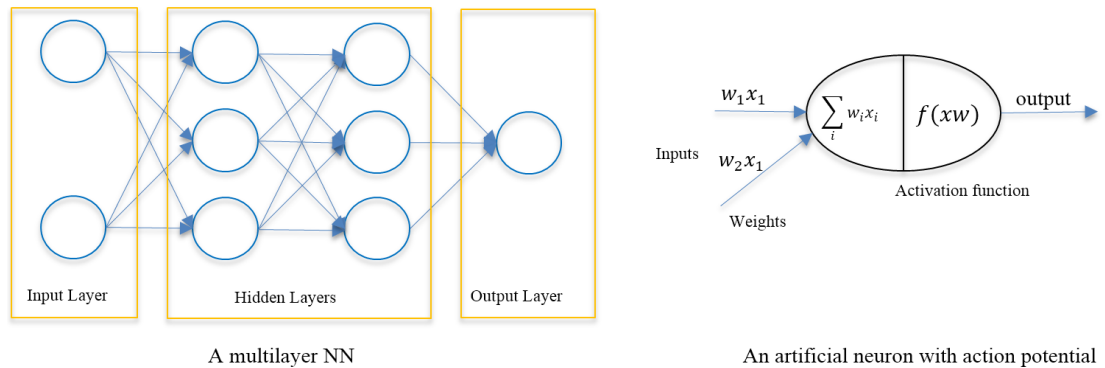


Figure 2-5 A block diagram showing components of an artificial neuron.

Feed-forward ANNs with a back-propagation (BP)-based algorithm can learn time varying inputs (Shinde, Samant, Naik, Ghorpade, & Kale, 2017; Yu, Efe, & Kaynak, 2002). Several

BP learning algorithms have been proposed, such as an on-line neural-network learning algorithm for dealing with time varying inputs (Zhao Y. , 1996; Cilimkovic, 2015) and learning algorithms based on gradient descent (Zhou & Si, 1998; Ranganathan & Natarajan, 2018).

MLPs with non-linear activation functions have been applied for classification of STBD, such as EEG classification related to emotion perception (Yaacob, Abdul, & amaruddin, 2016) and a motor imagery EEG classification (Chatterjee & Bandyopadhyay, 2016). Some other relevant ANN models are Recurrent Neural Networks (RNNs) (Peddinti, Povey, & Khudanpur, 2015; Waibel, Hanazawa, inton, Shikano, & Lang, 1989; Mozer, 1993; Che, Purushotham, Cho, Sontag, & Liu, 2018) that are suitable for sequential data classifications, such as time series. This network is based on back-propagation, meaning that the output of one layer can return back as input to the previous layer for tuning the connections. One problem with this complex network is a vanishing gradient that happens when the activation function cannot make significant change in the output, therefore, the network refuses to learn further.

Although these ANNs are inspired by some properties observed in brain research (Hodgkin, Huxley, & Katz, 1952; Hall, 2015), the neuron's state depends only on the current time of inputs, employing an activation function. To enhance this, the third generation of ANNs, called Spiking Neural Networks (SNNs) emerged in which accumulated inputs over time control an action potential function; thus, it encodes the neuron's firing-time information.

Like the first-generation ANN, a spiking neuron integrates the inputs and fires when the firing threshold is exceeded. In addition to this, a spiking neuron has an inherent dynamic nature

that defines a postsynaptic potential state which changes with time. The postsynaptic potential of a spiking neuron changes with time while streaming inputs. A spiking neuron fires at the time t in which its internal state exceeds the neuron threshold. Therefore, SNNs are considered as brain-inspired computational models that encode properties such as action potential, excitatory postsynaptic potential and inhibitory postsynaptic potential (Izhikevich, 2003).

D. Deep Structured Learning (DSL)

In machine learning, DSL methods refer to learning the data representations in a hierarchical manner, where each layer in the model extracts a different informative level of representation that corresponds to a particular concept in data. The learning procedure can be supervised, unsupervised or semi-supervised. So far, different DSL architectures have been introduced and examples are as follows: Deep Neural Network (DNN) (Liu, et al., 2017), Deep Belief Neural Network (DBNN) (Goodfellow, Bengio, & Courville, 2016), RNN and Convolutional Neural Network (CNN). DNN refers to a network with multiple hidden layers between the input and output layers. In DNNs every layer of neurons extracts informative abstractions and transfers them to the next layer to model complex non-linear relationships. DNNs are usually feed-forward networks, while the data flow can be in any direction in RNNs making them suitable for a wide range of applications (Gers & Schmidhuber, 2001; Sutskever, Vinyals, & Le, 2014; Tomavs, Karafiat, Burget, Černocký, & Khudanpur, 2010).

CNNs (Schmidhuber, 2015) are inspired by the visual cortex, where the firing rate of every sensory neuron is affected by a specific region in the retina, called the neuron's receptive field. Neurons have different specific receptive fields, and they are overlapping. CNNs

consist of three main layers: input layer, feature learning layer and classifier layer. Each of these has several sub-layers. In contrast to MLPs, in CNNs the hidden layers are not fully connected. Through a convolution procedure, every region of neurons (receptive field) from layer i is connected to one neuron in layer $i + 1$, which results in extracting abstractions (i.e. some informative features) from layer i and transfers them to the next layer. Convolution uses a filter (with a specific size) which is sliding over the input values to merge them and generates a feature map. Several convolutions are applied on the input data to develop different feature maps. Training a CNN is similar to MLP training that can be based on backpropagation or gradient descent. CNNs use activation functions, therefore, they can solve non-linear classification tasks.

CNNs supported tremendous achievements in computer vision systems, including image classification (Krizhevsky, Sutskever, & Hinton, 2012), image segmentation (Liang-Chieh, George, Iasonas, Kevin, & Alan L, 2018; Wachinger, Reuter, & Klein, 2018) and object detection (He, Hang, Ren, & Sun, 2016). One of the common issues with CNNs is overfitting that may happen due to the presence of layers which model irrelevant dependencies. CNNs were also used in several applications of STBD, for instance, emotion classification using EEG data (Tripathi, Acharya, Sharma, Sudhanshi, & Bhattacharya, 2017) and the result has shown that CNN outperformed MLP. In research (Rezaei Tabar & Halici, 2016) a motor imagery EEG classification task was performed, and the results suggested that deep learning CNN improved classification performance by at least 9% compared to other conventional approaches. In this section, I reviewed methods that have been applied so far for STBD classifications. The next section reviews data clustering which is an important approach for understanding relationships in STBD.

2.5.2 Overview on Clustering Methods

Clustering aims at objectively organising data samples into homogenous groups where data samples within a group are similar in some sense. So far, many clustering methods have been developed to identify structures in different data types, such as static and temporal data. Data are static when the feature values do not change over time, and they are time series (temporal) if the features comprise values that change over a continuous time interval. With respect to different data types, clustering methods differ significantly in the notion of similarity/distance measures. In the following, I present a review on clustering methods applied to different data domains: (A) static data clustering, (B) time series clustering and (C) dynamic evolving clustering.

A. Clustering Approaches for Static Data

Clustering methods for various static data are classified into five major categories: hierarchical methods (Johnson, 1967), partitioning methods, density-based methods (Ester, Kriegel, Sander, & Xu, 1996), grid-based methods, and model-based methods.

Partitioning clustering divides datasets into k distinct partitions, where samples in each partition share similar characteristics of this cluster. Most of the partitioning clustering algorithms are based on minimising an objective function, which usually refers to the distance between samples and the cluster centre. A typical objective function is as follows:

$$\sum_{i=1}^k \sum_{j=1}^{|C_i|} Dist(x_j, center(i)) \quad (2-1)$$

where $|C_i|$ denotes the number of samples belonging to cluster i and function $Dist$ computes the distance between sample x_j to the centre of cluster i . A partition is hard if each sample belongs to only one cluster, or fuzzy if one sample is allowed to be in more than one cluster up to a different membership degree, e.g. fuzzy c-means (Bezdek, Ehrlich, & Full, 1984; Rai, Bajaj, & Kumar, 2015). Numerous hard partitional clustering methods have been proposed, in which one of the most popular ones is the K-means algorithm (Hartigan & Wong, , 1979; Prabhakar & Rajaguru, 2015). The K-means algorithm classifies a given data set into K centroids, which are pre-defined a priori. The procedure is to assign each sample to the nearest centre and form K clusters. Afterwards, new centroids will be computed with respect to the mean value of each cluster. Over a number of iterations in the algorithm, the K centroids change until they converge to certain locations (which means that no more changes are done) through minimising an objective function, which follows here as a square error function:

$$\sum_{i=1}^K \sum_{j=1}^{|C_i|} \|x_j - center(i)\|^2 \quad (2-2)$$

With respect to different clustering algorithms, the similarity/distance can be measured through different equations as reported in Table 2-1.

Table 2-1 Different techniques for measuring the distance.

Method	Formula
Euclidean distance	$\ a - b\ _2 = \sqrt{\sum_i (a_i - b_i)^2}$
Squared Euclidean distance	$\ a - b\ _2^2 = \sum_i (a_i - b_i)^2$
Manhattan distance	$\ a - b\ _1 = \sum_i a_i - b_i ^2$

Maximum distance	$\ a - b\ _{\infty} = \max_i a_i - b_i $
Mahalanobis distance	$\sqrt{(a - b)^T S^{-1} (a - b)}$ S is a covariance matrix

Hierarchical clustering represents groups of data samples using a tree of clusters, in which the similarity is measured according to a pairwise distance matrix of samples. Every two similar samples can be merged to represent a cluster and then the most similar pair of clusters can merge their members to represent a higher level of clusters (known also as a parent cluster). This procedure will be repeated until all the data samples fall into one cluster, which is the root of the tree. Hierarchical clustering algorithms are either bottom-up (also called agglomerative clustering) or top-down (also called divisive clustering).

Figure 2-6 represents an example of bottom-up hierarchical clustering for 11 alphabetic samples, illustrating how it organises the data space through merging similar pairs of letters. A deficiency of the hierarchical clustering is that the computational complexity is $O(n^2)$ subject to the number of n samples.

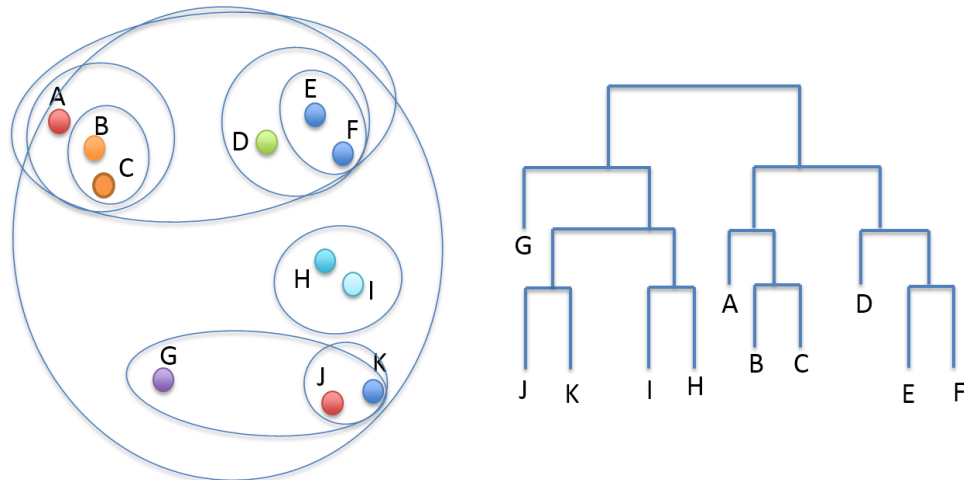


Figure 2-6 A tree diagram of a data space with 11 samples of letters (right) and the corresponding nested clusters (left).

In density-based clustering, unlike the K-means, the number of clusters is not predefined. Clustering is initiated from a set of points, continuously growing as long as the density in the neighbourhood surpasses a threshold.

In grid-based clustering, the sample space is quantised into a finite number of cells, where the operations for clustering are performed. A common example of the grid-based approach is Statistical Information Grid-Based method (STING) (Wang, Yang, & Muntz, 1997).

Model-based clustering undertakes a model for each cluster and aims at creating the best fit of the data to the model. One major method of model-based clustering is the neural network approach. Prominent clustering methods of the neural network field are defined by competitive learning, including Self-organizing Maps (SOM) (Carpenter & Grossberg, 1987; Baig, Ayaz, Gillani, Jamil, & Naveed, 2015; Kohonen, 1998).

B. Clustering Approaches for Time Series

A massive amount of temporal data (time-series data) has been recorded so far in various fields, such as electronic, video/audio, biology, neurology, and so forth. In cases of clustering such data, given a set of time series, the objective is to group similar temporal patterns into the same cluster. This task demands a specific notion of distance measured to estimate the level of similarity between time series distributions. However, the Euclidean distance and other typical measures (used for non-temporal data) are unsuitable measures to evaluate the similarity between time series. In the literature of time series clustering, various methods have been introduced of which some are briefly discussed as follows:

Biclustering (Mirkin, 1998) is a pattern-based clustering technique which simultaneously clusters both rows and columns of a dataset represented by m samples and n -dimensional features as a matrix (m, n) . Every sample vector can represent a pattern with multi features, and biclustering can find a relationship between patterns in such datasets. Biclustering has been used for clustering time series gene expression data (Tanay, Sharan, & Shamir, 2002). Since the biclustering can detect the direction of changes in data variables among samples, it has been considered as a promising technique for clustering time series. However, when dealing with spatio-temporal data, where both space and time need to be integrated and involved in the clustering algorithm, biclustering approaches cannot perform sufficiently as the spatial relationship between features is not incorporated in the analysis.

SOM is a clustering approach, utilising the ANN underpinning unsupervised learning. SOM is performed in two phases: (1) training, which creates a map using input training samples through a competitive procedure and (2) mapping, which classifies a new input sample vector. During the training phase, for every input sample vector, the distance between sample variables and all the nodes in the map will be computed (usually by means of Euclidean distance). Then, with respect to the principle of competitive learning, the node with the smallest distance will win as the best matching unit. The winning neuron and its neighbourhood (within a radius) will be pulled towards the current input sample. The learning procedure will be iteratively performed for all the training samples, and the final network performs a similar response to certain input patterns. Thus, SOM shapes a semantic map where similar sample vectors are mapped close together and dissimilar ones apart.

SOM has been widely used for clustering of EEG data (Hamdoun & Usman, 2016; Joutsiniemi, Kaski, & Larsen, 1995; Orjuela-Caon, et al., 2017), where each EEG spatio-temporal sample was first transformed into one feature vector and then passed on to SOM for learning.

C. Dynamic Evolving Clustering Approaches

The concept of dynamic or evolving clustering differs from the wide spectrum of clustering approaches that attempt to measure the distance/similarity within the whole data space. It rather refers to clustering of a data stream environment which continuously evolves with time. As a result, evolving clustering methods represent the incremental growth of clusters and the creation of new clusters from a stream of vector-based data. So far, several methods for dynamic evolving clustering have been proposed. While SOM in an entire data space assigns similar input vectors into topologically close neurons, Evolving Self-organizing Maps (ESOM) (Deng & Kasabov, 2000) and the DENFIS evolving clustering method (Kasabov & Song, 2002) were introduced for online dynamic clustering of data streams. When dealing with STBD, both ESOM and DENFIS successfully detect the temporal changes in data streams and incrementally assign them into the already generated clusters or develop new clusters for them. However, the temporal components of each data sample are transposed into one feature vector (static vector), where the time is hidden, and no temporal interaction can be extracted anymore. Also, the spatial relationships between the features are not considered in these models.

Aggarwal (Aggarwal, 2003) has proposed a framework for dynamic evolving clustering of spatio-temporal streams. The clustering method considers both spatial and temporal

relationships in data space, however, it creates separately two spatial and temporal models. Therefore, the integrated spatial-temporal similarity in the data is not properly measured and this is a crucial lack for clustering of STBD streams.

It can be concluded from the literature that classification and clustering of STBD have often been done using conventional machine learning methods such as the SVM (Cortes & Vapnik, 1995), MLP, Multiple Linear Regression (MLR), linear regression, or deep learning architectures. Now the question is: *what is missing in the current technologies for STBD analysis?*

Currently, many types of STBD have been collected that capture complex temporal patterns, which need to be modelled and analysed. Various techniques have been developed to analyse the brain's activation, functional connectivity (Tana, Bianchi, Sclocco, Franchin, & Cerutti, 2012; Aggioni, Tana, Arrigoni, Ucca, & Bianchi, 2014) or effective connectivity (Buchel & Friston, 1997). Learning dynamic patterns of spatio-temporal data is a challenging task, as temporal features may manifest complex interactions that may also change dynamically over time. Therefore, the relationship between the spatial and temporal components needs to be considered and learnt. In addition, the spatial information of the temporal sources needs to be learnt both topologically and computationally. To address these needs, developing new analytical methods that can capture interactions among multivariate data is of crucial importance.

Compared to conventional ANNs (first and second generations as explained in Section 2.5.1) SNNs have emerged to integrate space and time components of data into the computation.

SNN models and their *neuromorphic*⁵ highly parallel implementations are advancing quickly (Furber, Galluppi, Temple, & Plana, 2014; Indiveri, et al., 2011). The challenge now for information science and AI is to develop new SNN algorithms and methods for the efficient learning of STBD and for their efficient neuromorphic implementations (Kasabov, 2014).

2.6 Chapter Summary

This chapter reviews two main techniques for recording STBD that have been widely studied in cognitive science and neuroscience research. Then, a historical review on classification and clustering approaches was presented. In the next chapter, SNN principles, models and applications are discussed.

⁵ Neuromorphic refers to a kind of a “dynamical” machine with processors (e.g. neurons and synapses) in which the algorithms simulate complex spatio-temporal dynamics on the computing hardware.

Chapter 3 Spiking Neural Networks

3.1 Introduction

As discussed in the previous chapter, the first and second generation ANNs were developed based on activation functions, which determine the neuron's firing state according to the current inputs at time t . However, neuroscience research indicated that the behaviour of a biological neuron is influenced by the dynamics of the membrane potential over a period of time. This means that the membrane potential fluctuates dynamically while the neuron receives streaming inputs. When the membrane potential suppresses a certain capacity, it generates an action potential (signal) and sends it out. Therefore, the action potential corresponds to the intensity of communication between neurons.

In the third generation ANNs (Maass, 1997), the inherent nature of the spiking neuron is inspired by the principle of action potential to incorporate the previous accumulated inputs. This is much similar to how a biological neuron functions (Izhikevich, 2006; Brette, et al., 2007; Scott, Kasabov, & Indiveri, 2013). The action potential here can be computationally

encoded by binary values (-1 or 1) with a precise timing, called *spikes*⁶. A sequence of spikes represents the times in which a neuron emitted action potentials.

SNNs are computational models that consist of spiking neurons as processing elements, connections between them, and algorithms for learning from data (Thorpe & Gautrais, 1998; Verstraeten, Schrauwen, D’Haene, & Stroobandt, 2007; Masquelier, Guyonneau, & Thorpe, 2009). They transpired as potential means to learn time, space and frequency of complex STBD. In addition to considering the neural synaptic state, SNNs include the timing of spikes in the computation. This means that the current activation level of a spiking neuron depends on the incoming spikes, pushing this value higher and then either firing (if exceeding a threshold) or decaying over time.

So far, numerous methods of SNNs have been schemed such as spatio-temporal pattern recognition (Humble, Denham, & Wennekers, 2012; Kasabov, 2012b), encoding time-series data such as speech data into spike sequences (Van Schaik & Liu, 2005; Delbruck & Lichtsteiner, 2007; Lichtsteiner, Posch, & Delbruck, 2008; Lichtsteiner & Delbruck, 2005; Indiveri, et al., 2011), computational neuro-genetic modelling (Benuskova & Kasabov, 2007), spatio-temporal data learning (Maass, Thomas, & Henry, 2002; Song, Miller, & Abbott, 2000; Dhoble, Nuntalid, Indiveri, & Kasabov, 2012), SNN reservoir computing and liquid state machines (Verstraeten, Schrauwen, D’Haene, & Stroobandt, 2007), classification systems (Kasabov, 2007), neuromorphic design and implementation (Izhikevich, 2006;

⁶ A Spike is a binary value (-1 or 1) at time t , which represents a certain upward or downward change in the signal.

Furber, Galluppi, Temple, & Plana, 2014; Indiveri, et al., 2011) and neuro-computational perspective of brain pathology (Reggia, Ruppin, & Glanzman, 1999).

Many applications of SNNs have been developed, including: multimodal audio-visual information processing (Wysoski, Benuskova, & Kasabov, 2010), STBD modelling (Kasabov, 2014), Brain-Computer Interfaces (BCI) (Anderson, Stolz, & Shamsunder, 1998), moving object recognition (Kasabov, Dhoble, Nuntalid, & Indiveri, 2013), cognitive data modelling (Kasabov & Capecci, 2015), finite automata modelling (Natschlager & Maass, 2002), predictive systems (Tu, et al., 2014). These applications are structured based on the SNN models, which process input stimuli across different evoked cognitive states, acting as an ultimate spatio-temporal data processing machine (Kasabov, 2014; Kasabov, 2010).

3.2 Computational Model of a Spiking Neuron

In a biological neuron structure (as explained in Chapter 2), when the overall power of input signals reaches a certain threshold, an output signal is generated and sent to other neurons connected to it. Therefore, neurons receive and transmit information by means of signals exchanged *via* synapses.

This procedure can be computationally simulated by artificial spiking neurons as information-processing units that accomplish non-linear processing (Anderson, Stolz, & Shamsunder, 1998; Maass, Thomas, & Henry, 2002). A collection of interconnected spiking neurons creates an SNN, where neighbour neurons are influenced by their spiking activities.

Figure 3-1 illustrates a biological neuron and an artificial spiking neuron which resembles the behaviour of a biological neuron's cell. It can be seen that the inputs and outputs in a

spiking neuron are in the form of spike sequences with a precise timing. With respect to different mathematical neuron models introduced in the literature, a spiking neuron can dynamically process the input spikes over time to compute its membrane potential. Incoming spikes make change in a neuron potential and when this exceeds a threshold value, the neuron emits an output spike. Similar to the axons, artificial neurons are interconnected through simulated paths, which are initially established with random weights. Then the connection weights between neurons are modified by transferring spikes across synapses. Numerous computational models of SNNs have been developed so far, some well-known models are listed in the following and discussed afterwards.

- Lapique
- Integrated-and-Fire
- Leaky Integrated-and-Fire
- Izhikevich
- Thorpe's Model
- Probabilistic Spiking Neural Model

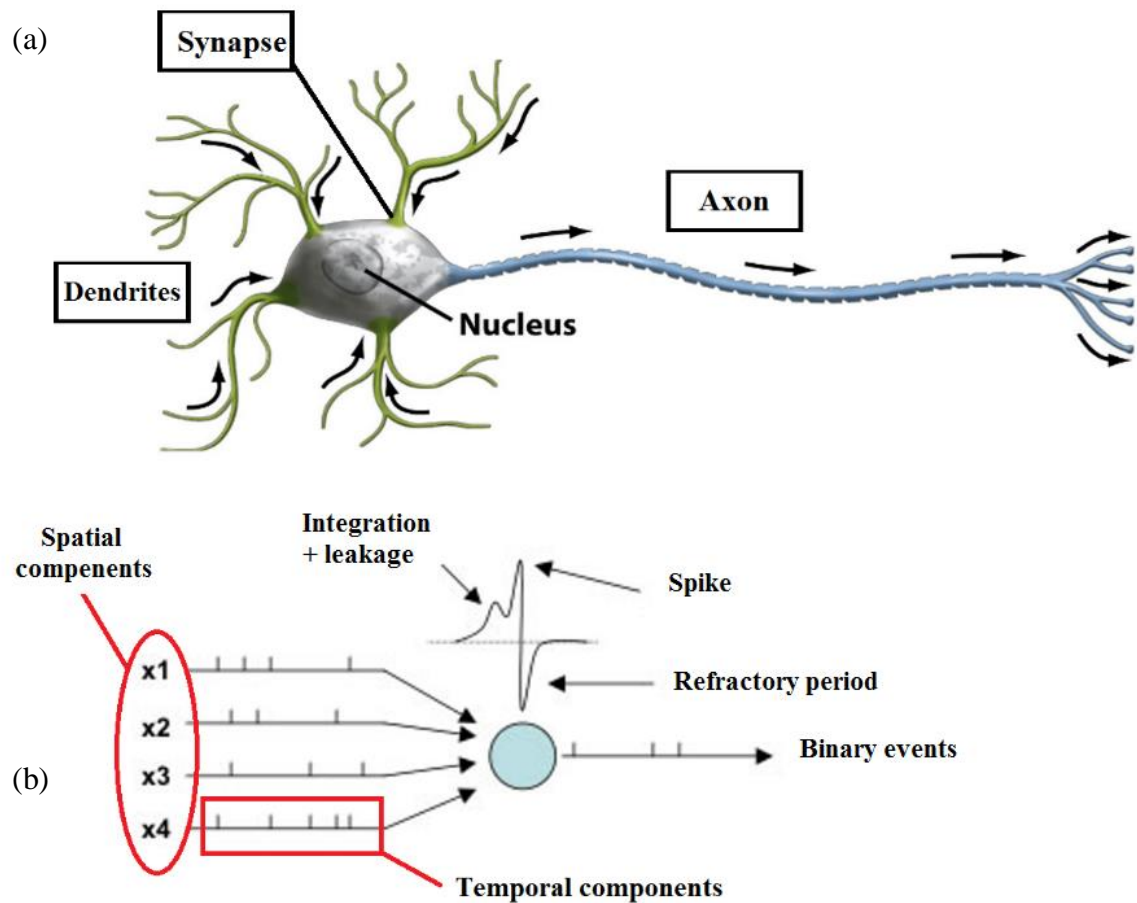


Figure 3-1 (a) Structure of a biological neuron, figure was modified form (Pearson , 2005); (b) Artificial spiking neuron which receives input spike trains, processes them and produces output spikes.

Lapicque Model. One of the earliest models of the neuron was the Lapicque model (Brunel, van, & Mark, 2007), which was constructed according to the mechanism of an electric circuit with a parallel resistor. This model captures the leakage resistance and capacitance of the cell membrane, but the concept of action potentials was not known. In this model, the action potential was set as a constant with different time of occurrence.

Integrated-and-Fire Model. In (Abbott, 1999) the Lapicque has been further developed to incorporate the principle of action potential, in which once the membrane capacitor reaches

to a certain threshold, an action potential is produced and then the membrane potential is reset. This model is called the integrated-and-fire model of a neuron. The integrate-and-fire model of a neuron can be defined by its membrane potential v and capacitance C as follows.

$$C \frac{dv}{dt} = -\frac{1}{R}(V(t) - v_{rest}) + I(t) \quad (3-1)$$

$$v(t^{(f)}) = \theta \quad \text{when} \quad v'(t^{(f)}) > 0 \quad (3-2)$$

where resistor is denoted by R , input current is $I(t)$ and $t^{(f)}$ is the time at which a neuron fires when its membrane potential $v(t^{(f)})$ exceeds the firing threshold θ , while $v'(t^{(f)})$ is its derivative.

Leaky Integrated-and-Fire Model (LIFM). In this model also called “forgetful” (Knight, 1972), the membrane potential $v(t)$ increases with every input spike at a time t , multiplied by the synaptic efficacy (strength), until it reaches a certain threshold θ . After that, an output spike is emitted, and the membrane potential is reset to an initial state. Like a biological neuron performs, when the simulated neuron emits a spike, it does not produce a new spike within a refractory period and its membrane potential $v(t)$ leaks. The membrane potential can have certain leakage between spikes, which is defined by a parameter τ . A schematic representation of the LIFM is illustrated in Figure 3-2 and the neuron action potential is defined as follow:

$$\tau_m \frac{dv}{dt} = v_{rest} - v(t) + RI(t) \quad (3-3)$$

The τ_m is the membrane time constant, v_{rest} is the resting potential, I is the input current and R is the resistance.

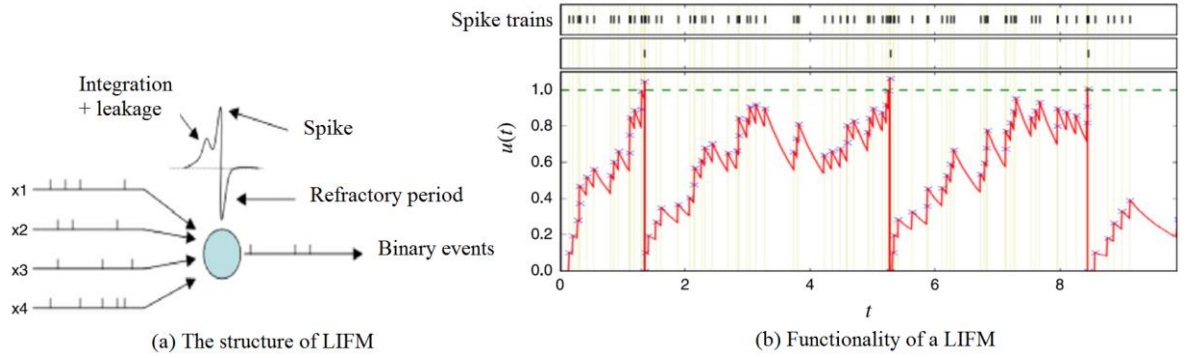


Figure 3-2 The LIFM of a spiking neuron. (a) Schematic representation; (b) Showing an input train of spikes (top row), the emitted output spikes (second row) and the membrane potential changes over time. Figure from (Kasabov, 2014).

Izhikevich Model. The model was introduced by Eugene M. Izhikevich (Izhikevich, 2003), combines the computational efficiency of the integrate-and-fire neurons and the biologically plausible principle of the Hodgkin–Huxley model. The Izhikevich model is implemented by means of a two-dimensional system of ordinary differential equations as follows:

$$\frac{dv}{dt} = 0.04v(t)^2 + 5v(t) + 140 - u(t) + I(t) \quad (3-4)$$

$$\frac{du}{dt} = a(bv(t) - u(t)) \quad (3-5)$$

Once the neuron membrane potential v exceeds a fixed threshold $\theta = 30$ mV , a spike will be emitted and u and v will be reset according to the following rule:

$$\text{if } u \geq \theta \text{ then } \begin{cases} v \leftarrow c \\ u \leftarrow u + d \end{cases} \quad (3-6)$$

where u denotes a membrane recovery variable that models feedback to v and a, b, c and d are dimensionless parameters. The Izhikevich model has been extended and developed in (Izhikevich, 2006), and (Izhikevich & Edelman, 2008).

Thorpe's Model. A variation of LIFM has been proposed by Thorpe (Thorpe S. J., 1990) in which a neuron membrane potential $v(t)$ is only influenced by the order of incoming spikes, when the earlier spikes have higher effects than the next upcoming ones. In this model, when neuron i receives input spikes from the presynaptic neuron j , ($j = 1, 2, \dots, N$) the postsynaptic potential of neuron i , $PSP_i(t)$, is increased according to the order of incoming spikes from N synapses. The earlier the spikes are received from the presynaptic neuron, the greater the impression on the neuron's postsynaptic potential. In this model, connection weights (W_{ji} , $j = 1, 2, \dots, N$) are established as follow:

$$W_{ji} = \sum_{j=1}^N mod^{order(j)} \quad (3-7)$$

where W_{ji} denotes to the connection weight between neurons j and i , mod is a modulation factor within $[0, 1]$ and $order(j)$ is the time order of the following spikes to the presynaptic j from all the connected neurons to j .

Probabilistic Spiking Neural Model (pSNM). In 2010, Kasabov proposed the pSNM (Kasabov, 2010) in which information is stored in the form of connection weights calculated with respect to Thorpe's model plus inclusion of three new probabilistic parameters to the

synaptic connection weight $w_{ji}(t)$. A pSNM is schematically shown in Figure 3-3. The probabilistic parameters are $p_{cj,i}(t)$, $p_{sj,i}(t)$ and $PSP_i(t)$, described as follows:

- $p_{cj,i}(t)$ is a probability parameter that neuron i receives a spike from neuron j at time t via connection w_{ji} between j and i . Here the connections are not created and pruned as in other ANN models (Kasabov, 2007), but a probability parameter is assigned to each connection to represent its structural and functional uncertainty. If $p_{cj,i}(t) = 0$, then this means that there is no connection and no spike propagation between j and i , while $p_{cj,i}(t) = 1$ represents a 100% probability of spike propagation.
- $p_{sj,i}(t)$ represents a probability for the synapse $s_{j,i}$ to be involved in the computation of $PSP_i(t)$ after receiving a spike from neuron j .
- A probability parameter $p_i(t)$ that neuron i emits an output spike at time t when its $PSP_i(t)$ has surpassed a certain threshold.

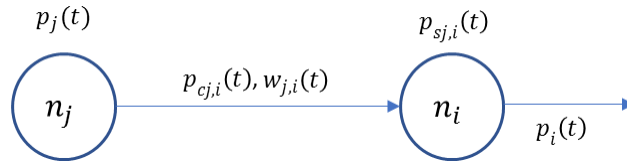


Figure 3-3 Representation of one synaptic connection in a pSNM.

The $PSP_i(t)$ is calculated using the following formula:

$$PSP_i(t) = \sum_{p=t_0, \dots, t} \sum_{j=1, \dots, N} e_j g(p_{cj,i}(t-p)) f(p_{sj,i}(t-p)) w_{j,i}(t) + \eta(t-t_0) \quad (3-8)$$

where e_j equals to 1, if neuron j emitted a spike, and 0 otherwise; $g(p_{cj,i}(t))$ is 1 with a probability $p_{cj,i}(t)$, and 0 otherwise; $f(p_{sj,i}(t))$ equals to 1 with a probability $p_{sj,i}(t)$, and

0 otherwise; t_0 is the time of the last spike emitted by neuron i ; $\eta(t - t_0)$ expresses decay in the PSP. The pSNM is simplified as the LIFM, when all the probability parameters are fixed to be 1.

The parameters $p_{cj,i}(t)$, $p_{sj,i}(t)$ and $PSP_i(t)$ were optimised using a quantum-inspired evolutionary algorithm to maximise the network's accuracy. This algorithm searches for the best parameter values to calculate the neuron's response in the presence of the stimuli. Further information about how to modify these probability parameters is given in (Kasabov, 2010).

3.3 Information Encoding in a Spiking Neuron

As explained in chapter 2, biological neurons send information by sudden and short increases in their electrical energy which generates an action potential or spike as encoded spatio-temporal information. When SNNs are used for learning the spatio-temporal patterns "hidden" in STBD, such data first need to be encoded into sequences of spikes, which are then transferred into the SNN *via* input neurons. Now the question is *what kind of codes can be considered for transmitting information in SNNs?*

Rate codes and pulse codes are two main techniques in neural encoding. In the first two generations of artificial neural networks, rate coding has been used to calculate the neurons' output signals based on only the frequency transmission. However, coding in SNNs is based on precise timing of spikes as computed in pulse coding (Maass & Bishop, 2001; Gerstner & Iistler, 2002).

A simple SNN is composed of an input layer, a hidden layer, and an output layer. Like a biological neuron's dendrites, the input layer receives the original input information and encodes it into a new number of features which are then transferred into the hidden layer's neurons. The input encoding procedure transfers the real value of input information to discrete sequences of spikes as new format of inputs to SNN models.

Different spike encoding algorithms have been proposed so far, some popular ones are: Temporal Contrast (Threshold-based Representation—TBR) (Delbruck T. , 2007; Dhoble, Nuntalid, Indiveri, & Kasabov, 2012), Ben's Spikes algorithm (BSA) (Schrauwen & Van Campenhout, 2003), Population Rank Coding (Bohte S. M., 2004), and Rank Order Coding (Thorpe & Gautrais, 1998). Two important encoding algorithms are explained here:

Temporal Contrast (TBR). This method was originally proposed in 2007 for information encoding in artificial silicon retina (Delbruck & Lichtsteiner, 2007). The method encodes substantial changes in signal amplitude over a given threshold, where the OFF and ON commands of the hardware were dependent to the sign of the changes. However, for dramatic changes in the signal amplitude, it is almost impossible to reconstruct the signal from spike trains generated by TBR. Therefore, TBR has been improved as follows:

For a given signal $S(t)$, where $t = 1, 2, \dots, n$, the signal amplitude variation over time t is denoted by $B(t)$, with a baseline $B(1) = S(1)$. At the next time point t , if the upcoming signal amplitude $S(t)$ is greater than $B(t - 1)$ plus a threshold θ , then a positive spike is generated at time t and $B(t)$ will be replaced by $B(t - 1)$. The encoding of positive and negative spikes is defined as follows:

$$spike(t) = \begin{cases} 1 \text{ and } B(t) \leftarrow B(t-1) + Th & \text{if } S(t) \geq B(t-1) + \theta \\ -1 \text{ and } B(t) \leftarrow B(t-1) - Th & \text{if } S(t) < B(t-1) + \theta \\ 0 & \text{otherwise} \end{cases} \quad (3-9)$$

Rank Order Coding (RO). With respect to the order of neuron firing times, a higher rank will be given to the earlier fired neuron (Thorpe & Gautrais, 1998). RO has been successfully applied in modelling audio visual systems in visual pattern recognition (Wysoski, Benuskova, & Kasabov, 2006) and speech recognition (Loiselle, Rouat, Pressnitzer, & horpe, 2005) applications.

3.4 Learning in SNN Models

Different types of learning rules in SNNs have been explored in the literature. The most popular ones are supervised and unsupervised learning as explained in the following:

Supervised learning. In this learning, the input data and the anticipated outputs are known. For instance, in the classification problem the class labels of samples are known and in a regression problem the real values for regression are given. SNN learns the input data patterns to produce an exact output when a new unknown input is presented. RO learning rule and error-backpropagation (Bohte, Kok, & La Poutre, 2002) are two popular supervised learning examples.

Unsupervised learning. In this learning approach, the desired outputs are not provided, and a training process is performed with unlabelled input data patterns. Hitherto, several unsupervised learning algorithms have been developed in SNN models, the majority of them are constructed to adapt the synaptic weights according to the temporal relation between pre- and postsynaptic action potentials as similarly implemented in Hebbian learning (Song,

Miller, & Abbott, 2000; Hebb, 1949). One example of Hebbian learning is Spike Time Dependent Plasticity (STDP) learning rule which depends on the relative timing of pre- and postsynaptic action potentials (Song, Miller, & Abbott, 2000). The STDP learning rule is defined using the following relation:

$$F(\Delta t) = \begin{cases} A_+ \exp(\Delta t / \tau_+) & \text{if } \Delta t < 0 \\ -A_- \exp(-\Delta t / \tau_-) & \text{if } \Delta t \geq 0 \end{cases} \quad (3-10)$$

where $F(\Delta t)$ defines the synaptic modification elicited from a single pair of pre- and postsynaptic spikes separated by a time interval $\Delta t = t_{pre} - t_{post}$. The parameters A_+ and A_- define the maximum quantities of synaptic modification, which transpire when $\Delta t \approx 0$. The parameters τ_+ and τ_- determine the ranges of pre-to-post-synaptic inter spike intervals over which the synaptic strengthening and weakening occurs.

3.5 SNN Reservoir Computing Systems

Reservoir computing systems are constituted of a group of recurrent connected neurons that form a computational framework, where the input signals are mapped into a higher dimension, called dynamical reservoir. The reservoir's neurons are non-linear information processing units which are typically connected randomly. The neural activities are triggered by the input dynamics and are also influenced by the past. Therefore, for online learning of continues input streams, reservoir computing allows for real-time computation in parallel.

One example of the reservoir methods is Liquid State Machines (LSMs), proposed by Mass (Maass, 2010; Maass, Thomas, & Henry, 2002) that employ LIFM of spiking neurons. The general architecture of LSM includes the following main layers:

- An input layer is randomly connected to the neurons in the reservoir.
- A reservoir of randomly interconnected neurons, each of which fires over time when its action potential value exceeds a certain threshold. Therefore, patterns of spikes are captured after T time-steps.
- An output layer is a simple deterministic readout layer that receives the spike patterns for classification.

The readout for LSMs is typically a multilayer feed-forward NN or linear regression that reads the states of the reservoir and maps them into the desired output. In several application domains, LSMs have shown better performance when compared with conventional ANN models. For instance, in pattern recognition (Verstraeten, Schrauwen, & Stroobandt, 2005), LSM was applied to recognise an isolated word and the results were compared with HMM models. In contrast to the HMM, the LSM has shown to be robust to noise. In (Pape, De Gruijl, & Wiering, 2008; Ju, Jian-Xin, & Antonius, 2010), sound data were transferred into 64 static vectors of frequencies using the Fast Fourier Transformation (FFT) which were further processed through normalisation. Then the pre-processed data were used to demonstrate real-time applicability of the LSM for music recognition. For the classification tasks, readout functions such as Recurrent NN, K-nearest Neighbours (KNN) and perceptron were often used. It can be concluded from these studies that LSMs perform well for the classification of spatio-temporal data. Now the fundamental question is: *what was missing in the current LSMs?*

- The LSMs do not preserve and learn the correlation between the spatial and temporal components in one model.

- LSMs are not developed based on an entirely spike-time constructed computational approach. The readout functions are usually implemented according to conventional machine learning algorithms. However, neurons in LSMs are spiking neurons and they produce patterns of spikes. Therefore, the performance of LSMs can be improved by employing SNN-based classifiers, which results in directly processing the reservoir's spikes in an online- one-pass mode. This suggests a better applicability of SNN models for real-life information processing.
- LSMs benefit from LIFM of neurons which try to model real neurons, however, brain-inspired learning is lacking to capture the spatio-temporal interactions in the form of adaptive connections in the reservoir.
- As a result of random connections in the reservoir, LSMs cannot be spatially interpreted. The spatial information of the temporal variables are not topologically and computationally considered.

In view of both characteristics and limitations of LSMs, SNN features were employed for developing a special type of an LSM, a Spatio-temporal Data Machine (STDM), called NeuCube that is an evolving spiking neural network (eSNN) for better modelling and understanding of STBD (Kasabov, 2007; Kasabov, Dhoble, Nuntalid, & Indiveri, 2013; Kasabov, 2012b). The following Section 3.6 introduces the NeuCube architecture, which will be used in this thesis for development of new methods for STBD analysis.

3.6 NeuCube for STBD Modelling

NeuCube is a generic evolving STDM based on the SNN for modelling, learning, classification/regression, clustering, visualisation and interpretation of spatio-temporal data,

it was first introduced for STBD (Kasabov, 2012b). The NeuCube development system is illustrated as an integrated configuration for spatio-temporal data pattern recognition and it includes the following ten modules:

- Module M1: Generic prototype and testing.
- Module M2: A pySNN simulator for small- and large-scale applications.
- Module M3: A neuromorphic hardware for real-time execution.
- Module M4: A 3-D visualisation and mining.
- Module M5: Exchanging of input/output information.
- Module M6: A neuro-genetic and prototyping testing.
- Module M7: Personalised modelling.
- Module M8: A multi-model brain modelling.
- Module M9: Data encoding and event detection.
- Module M10: Online learning.

In this thesis, Module 1 is explained in the current chapter, while the development of Module 7 is proposed by me as presented in Chapter 7. Figure 3-4 illustrates a diagram of the NeuCube multinodular development architecture for modelling of STBD. The NeuCube Module 1 was initially proposed in (Kasabov, 2012b) for STBD modelling and then further developed in (Kasabov, 2014). Module 1 consists of the following five sub-modules and it is depicted in Figure 3-4 and Figure 3-5:

- Input data mapping and encoding.
- Unsupervised learning in a 3-dimensional (3-D) SNN reservoir, called SNN model.
- Supervised learning and classification in an evolving SNN.

- Parameter optimisation.
- Model visualisation and interpretation.

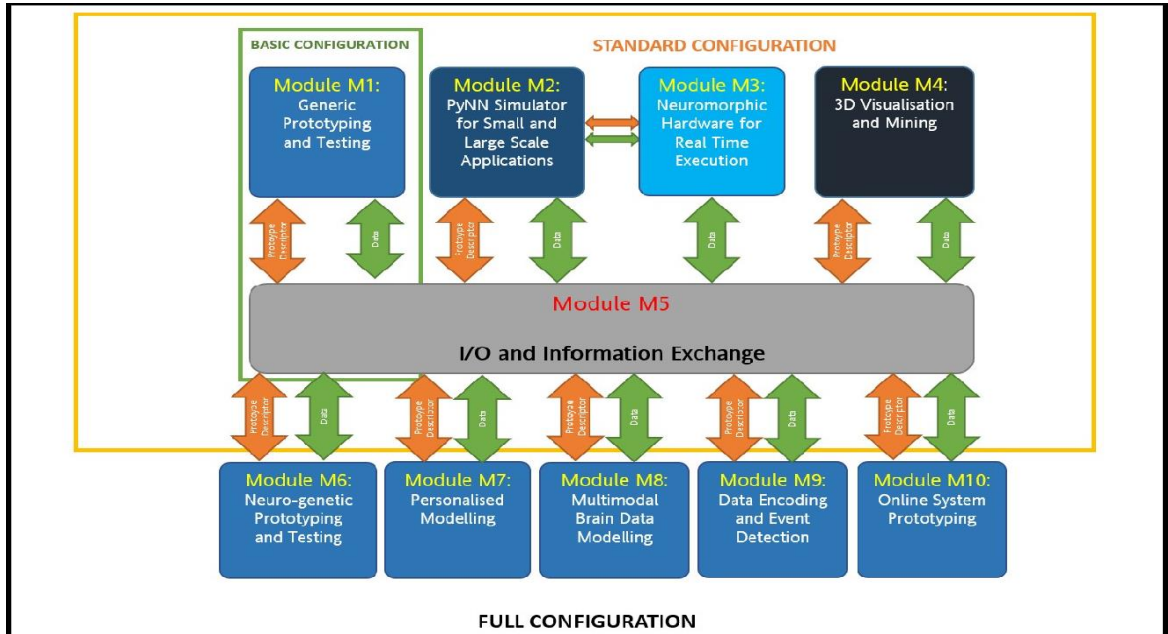


Figure 3-4 The multinodular development architecture of the NeuCube. Figure from (Kasabov, et al., 2016).

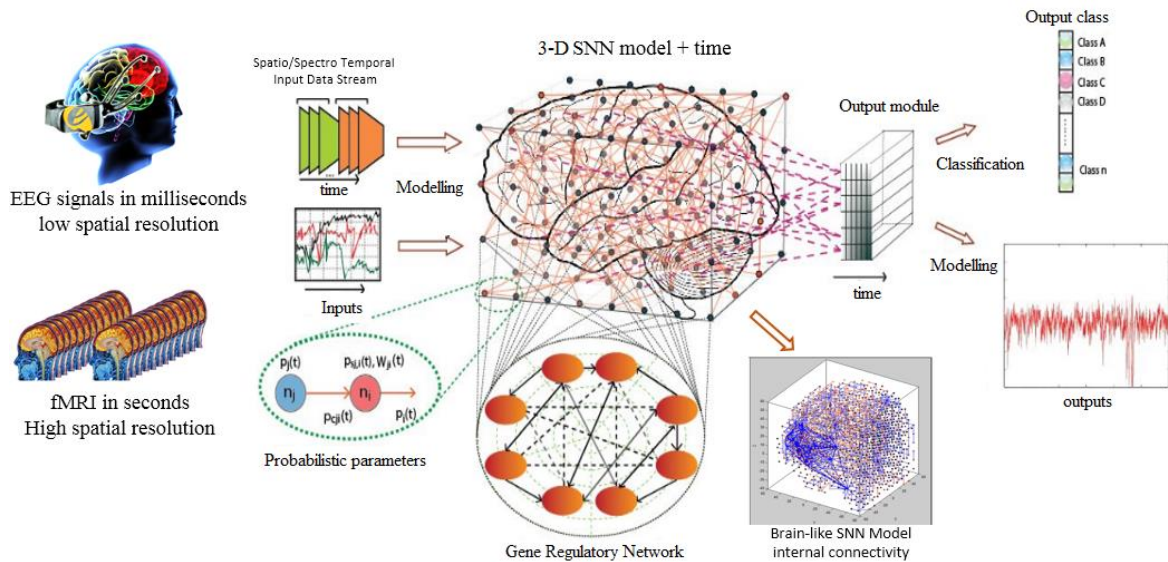


Figure 3-5 A functional diagram of the NeuCube SNN architecture, consisting of: input spatio-temporal data encoding module, 3-D SNN model and the STDP learning, output module for classification/regression, and gene regulatory network (GRN) module. Figure is modified from (Kasabov, 2012b).

3.6.1 Input Data Mapping

The NeuCube mapping sub-module is created as a 3-D SNN structure of a suitable size that maps spatially a brain template, such as the Talairach (Talairach & Tournoux, 1988), the Montreal Neurological Institute (MNI) template (Brett, Christoff, Cusack, & Lancaster, 2001) or coordinates of individual brain data.

If the spatial information of STBD variables is given, I can spatially map these variables into the pre-designed SNN model (the Talairach-based mapping) with respect to the coordinates of these variables as positioned in the Talairach. For instance, the location of each EEG channel (input variable) can be used for localising the input neuron in the SNN model. The size of the SNN model is scalable and controlled by three parameters: n_x, n_y, n_z representing the numbers of neurons along x, y and z directions. This model is used to map the (x, y, z) coordinates of an STBD; so that, the spatial information of the data is preserved. The prominent attitude of spatially mapping the input variables are: (a) construction of accurate spatial models for STBD collected from these variables and (b) better understanding and interpretation of the STBD models, as models are labelled by different functional areas in the Talairach or other templates.

If such spatial information is not available for some datasets, temporal variables can be efficiently mapped to the SNN models with respect to their temporal correlation. The temporal patterns of v variables are first encoded into v spike trains, and then correlation between these spike trains is measured. Variables are then mapped; so that, highly correlated

variables (highly time-dependent) will be mapped to nearby input neurons in an SNN model. Every neuron in an SNN model is a computational unit that can be simulated according to different models of a spiking neuron as described in Section 3.2. As being implemented, the LIFM is used here in NeuCube SNN model. LIFM is commonly used by neuroscientist in the literature (Eugene, 2004; Sterratt, Bruce, Andrew, & David, 2011) and has been also used in this thesis as a brain-inspired and efficient model for a proof-of-concept.

3.6.2 Input Data Encoding

The continuous time series of STBD variables that measure functional activity in the brain were encoded into spike trains. The timing of the spikes corresponds to the time of the changes in the STBD. A spike train, obtained after the encoding process, represents new input information to the SNN model, where the time unit might be different from the real time of the data acquisition (machine computation time *versus* data acquisition time). The spike trains were transferred into the SNN model *via* input neurons which are spatially allocated using the same (x,y,z) coordinates as positioned in a brain template.

The existing implementation of the NeuCube is sustained by four different spike encoding algorithms including:

- Ben's Spiker Algorithm (BSA).
- Temporal Contrast (Threshold-based representation—TBR).
- Step-Forward Spike Encoding Algorithm (SF).
- Moving-Window Spike Encoding Algorithm (MW).

The above spike encoding methods have different features when demonstrating input data. For encoding high frequency signals, it is more suitable to use BSA as it is based on the finite

impulse response technique and the original signal reconstruction is also easy. The BSA method produces only positive (excitatory) spikes, whereas using the other methods mentioned above, negative (inhibitory) spikes can be produced along with the positive spikes. As an example, the TBR encoding method was applied to a time series of one EEG channel data to transfer it into a sequence of spikes as shown in Figure 3-6.

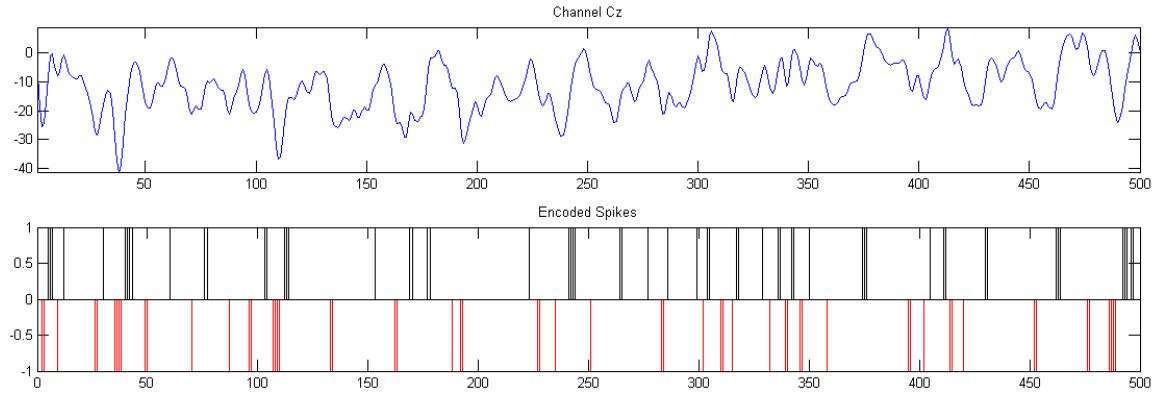


Figure 3-6 An example of encoding EEG data into sequence of positive (black) and negative spikes (red) using the TBR algorithm that is the format of the input data into the NeuCube SNN architecture. The image shows the first 500 data points of one EEG channel (the Cz channel) from (Capecci, et al., 2016).

NeuCube learning is a two-phase process: (1) an unsupervised learning stage in an SNN model, where spatio-temporal relations from the input data are learnt and connection weights are evolved, and (2) a supervised learning stage, where the class information is associated with each training spatio-temporal samples. These learning processes are explained in Section 3.6.3 and Section 3.6.4 respectively.

3.6.3 Unsupervised learning in SNN Models

After mapping the spatial components of STBD to the SNN model, the neuron connectivity is initialised using the small-world connectivity as shown in (Tu, et al., 2014), which is a phenomenon observed in biological systems (Bullmore & Sporns, 2009; Braitenberg &

Schuz, 1998). In this rule, each neuron in the SNN model is connected to its nearby neurons which are within a radius d . These connections are later modified based on the learning of new incoming spikes in time during unsupervised learning which is performed based on the STDP learning rule (Song, Miller, & Abbott, 2000).

STDP learning is performed through transferring spikes (in time) across spatially located synapses and modifying the synapses over time. In this learning, a neuron's postsynaptic potential (PSP) increases by every input spike at time t to reach the firing threshold. Once the PSP exceeds this threshold, the neuron fires and sends a spike to the other neurons that are connected to it. In STDP learning, if neuron i fires before neuron j , the connection weight from i to j will increase, otherwise it will decrease. As a result, STDP adjusts the connection weights between neurons based on the relative timing of a particular neuron's output and input spikes. In this study, the unsupervised learning allows for the SNN model to evolve its connections; so that, they capture spatio-temporal associations between STBD variables representing consecutive brain activities.

3.6.4 Supervised Learning and Classification in Evolving SNN

After the unsupervised learning is completed, for data classification/regression, dynamic evolving SNN (deSNN) (Kasabov, Dhoble, Nuntalid, & Indiveri, 2013) is used to train an output classifier based on an association between class labels and training samples. For each training sample, an output neuron is created and connected to each neuron in the trained SNN model. The initial connection weight between a neuron i from the SNN model and an output neuron j is defined by using the RO learning rule. After establishing the initial connection weights, the same data that have been used for unsupervised learning will be propagated

again through the trained SNN model, sample by sample. The spatio-temporal pattern of activation in the trained SNN model, evoked by each particular sample, will be used as input data to train an output neuron for recognising this pattern. The PSP of neuron j at time t is calculated as follow:

$$PSP(j, t) = \sum mod^{order(i)} W_{ij} \quad (3-11)$$

where mod is a modulation factor (a parameter between 0 and 1) and $order(i)$ is the time order of the following spikes to the connection between neurons i and j . Through this learning rule, the first spike that arrives at the output neuron j will have the highest value. Then, the connection weight W_{ij} will be further modified according to the spike driven synaptic plasticity learning rule using a drift parameter, which is used to modify W_{ij} to take into account the occurrence of the following spikes at neuron j at time t , denoted by $spike_j(t)$, i.e. if there is a spike arriving from neuron i at time t after the first one was emitted, the connection weight increases by a small drift value; otherwise, it decreases by drift as shown in the following:

$$W_{ij}(t) = \begin{cases} W_{ij}(t-1) + drift & \text{if } spike_j(t) = 1 \\ W_{ij}(t-1) - drift & \text{if } spike_j(t) = 0 \end{cases} \quad (3-12)$$

The NeuCube classification accuracy is evaluated through cross-validation for different sets of parameter values and the best accuracy model is saved for recall purposes, for further analysis and adaptation on new data. The use of deSNN allows for a further adaptation of the NeuCube model on new data in an incremental way without re-training the model on old data.

3.6.5 NeuCube Parameter Optimisation

The output classification accuracy depends on the combination of parameters' values. This combination can be optimised using different algorithms, such as: grid-search (exhaustive search), genetic algorithm, and quantum inspired evolutionary algorithm (Schliebs & Kasabov, 2013). Important parameters of a NeuCube model are:

- TBR_{thr} : A self-adaptive bi-directional threshold for STBD encoding to spike trains.
- d : Radius threshold for the initialisation of the SNN connectivity in small-world connectivity rule.
- *STDP learning rate (α)*: A parameter used to modify the connection weights according to repetitively arriving spikes to the synapses. If a neuron i fires before a neuron j , then its connection weight increases, otherwise it decreases with respect to the STDP learning rate (α).
- (Th_o): Threshold of firing for the neurons in the SNN model.
- *deSNN classifier parameters*: These parameters are: *mod* and *drift*.

The trained NeuCube model of STBD can be dynamically visualised in a 3-D virtual reality space for the analysis of brain activities and for a better understanding of spatio-temporal relationships in the data (Kasabov, et al., 2016).

3.7 Chapter Summary

This chapter exposes an overview on computational models of SNNs and introduced the NeuCube SNN architecture for modelling STBD. A NeuCube model supports a meaningful mapping of spatial variables, modelling, learning and model visualising. In the NeuCube

architecture, the random connections in the LSM are replaced by meaningful brain-inspired connections and the learning procedure is based on STDP which captures the spatio-temporal interactions. These features make a NeuCube model meaningful in terms of its interpretation for a better understanding of spatio-temporal characteristics of data. The next chapter will demonstrate a feasibility analysis of the SNN for modelling of real-life STBD.

Chapter 4 SNN Feasibility Study on STBD

4.1 Introduction

In this chapter, I present the feasibility study of the NeuCube SNN architecture using two types of STBD, which are here real-life fMRI and EEG datasets. For each of these studies, I designed an empirical scenario which includes both pattern analysis and classification tasks. I constructed optimal SNN models and trained them with STBD samples. The trained models demonstrate the spatio-temporal interactions between the input data variables in a computational SNN model, rather than an exact structure of the brain's physical neural connectivity. The SNN models are then used for classification of STBD samples with respect to different pre-defined mental activities. The models are also visualised in a 3-D space and statistically analysed to evaluate the level of significance. In addition, I performed a comparative analysis to illustrate how the designed SNN models resulted in an improvement of classification accuracy when compared with conventional methods.

4.2 NeuCube Architecture for STBD Analysis

NeuCube has been used to analyse several STBD such as EEG and fMRI to expose meaningful spatiotemporal patterns while different mental activities are performed. For examples, classifying EEG data for music versus noise perception (Kasabov, Hu, Chen, Scott, & Turkova, 2013); classifying EEG data with respect to different arithmetic tasks

(Kasabov & Capecci, 2015); classifying EEG data in relation to different levels of dementia (Capecci, et al., 2016; Capecci, et al., 2015); and fMRI data mapping and classifying to investigate how a reading task is processed in the brain (Murli, Kasabov, & Handaga, 2014).

Now in this chapter, employing the NeuCube framework, I designed a new empirical study for pattern analysis and classification of STBD. The designed framework allows for measuring the level of confidence in the SNN models and this is accomplished according to the following steps (graphically illustrated in Figure 4-1):

A) Pattern analysis:

1. Design the experimentations.
2. Preparation and sampling of the STBD according to the designed experiments.
3. Encoding of the STBD into spikes.
4. Spatial mapping of STBD into a 3-D SNN model.
5. Unsupervised learning in SNN models, visualisation and interpretation of spiking activity and connectivity in the trained SNN models.

B) Classification:

1. Supervised learning and classification.
2. Parameter optimisation.
3. Statistical analysis of the SNN models.

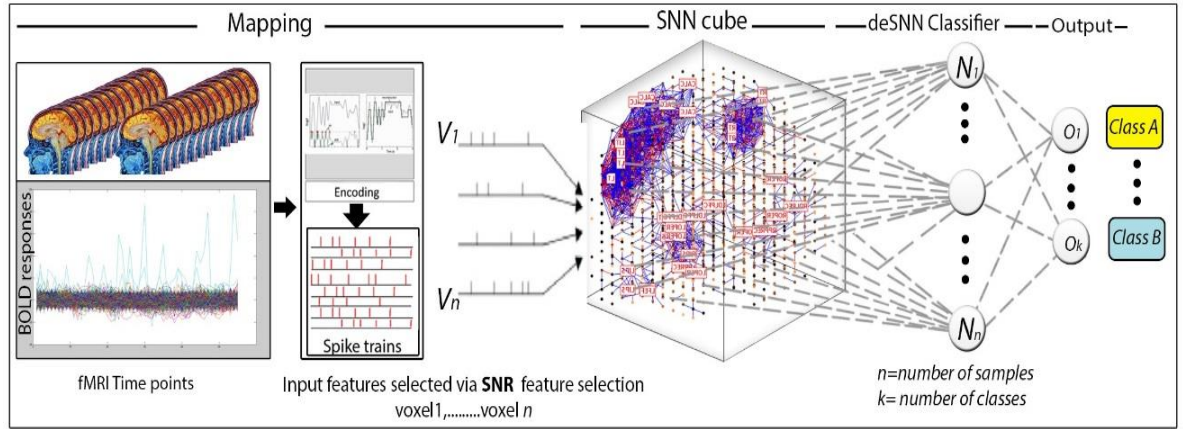


Figure 4-1 A schematic representation of the NeuCube framework for STBD mapping, learning, visualising, and classifying.

The above steps are performed for two feasibility studies on STBD: (1) a new study on fMRI data using SNN models; and (2) an SNN application on EEG data. In both studies, in order to evaluate the level of significance in the trained SNN models, I performed the following:

I created one SNN model per class of data (each class contains n samples) and trained it through an iterative procedure of leave-one-out as follows (shown in Figure 4-2):

1. The SNN model is initialised.
2. The initialised SNN model is trained with $(n - 1)$ samples (one sample is excluded from the training).
3. The average of the spatio-temporal connection weights in the trained SNN model is computed.
4. The hold-out sample is replaced by another sample, then go to step 1 until all the samples are excluded from the training set, one by one (it means that a set of SNN models are initialised and trained with different folds of samples).

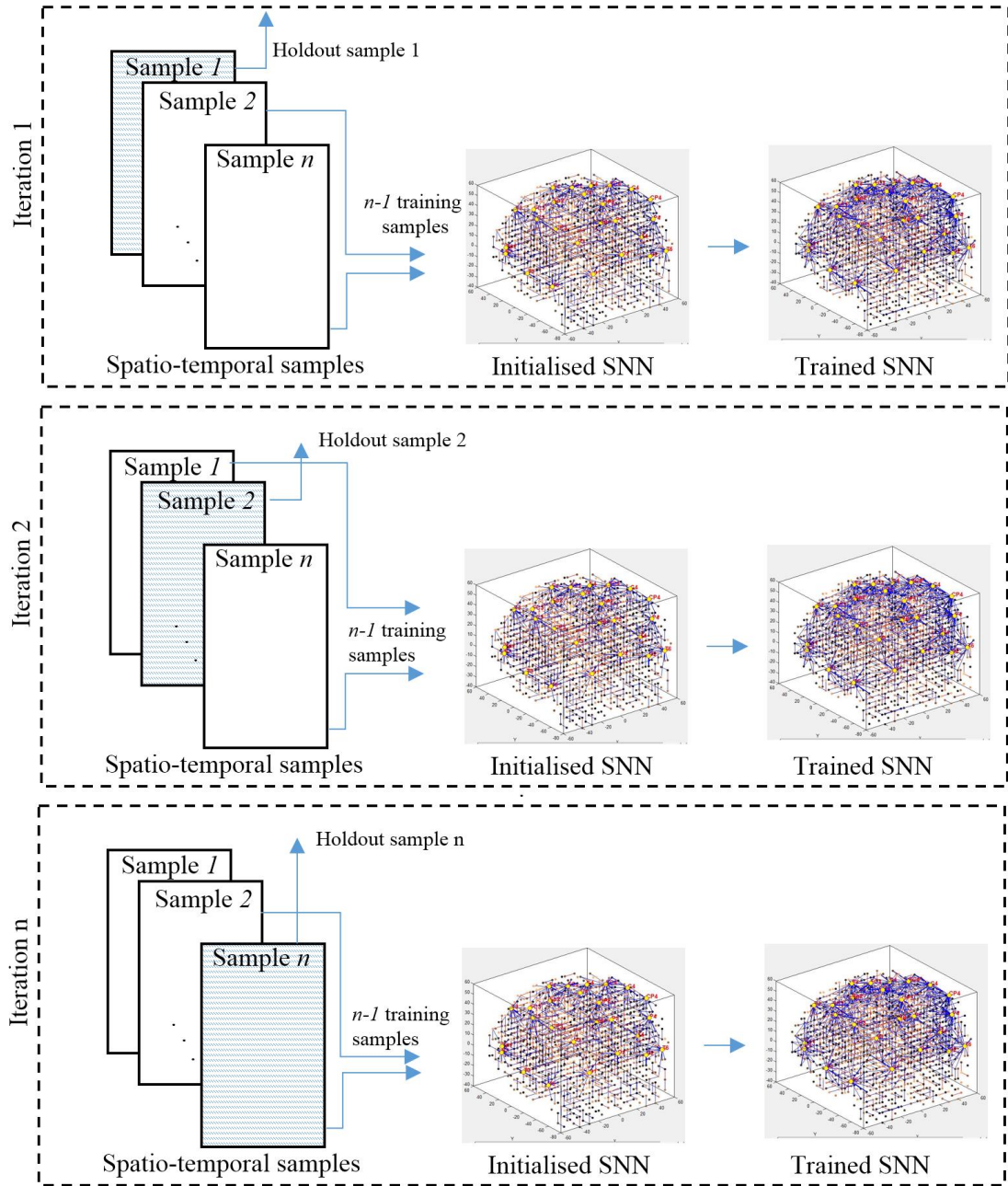


Figure 4-2 Unsupervised training iterations in SNN models by the leave-one-out method. For n samples, the SNN model is initialised n times, and trained by a fold of different $(n - 1)$ samples.

4.3 A New SNN Study on fMRI

In this section, I demonstrate a new study on fMRI data modelling using SNN architecture by designing two experiments related to cognitive processes of reading sentences with different polarities and seeing pictures. These experiments include both classification and pattern analysis.

The case study problem used here belongs to the STAR/PLUS fMRI data set, originally collected by Marcel Just and his colleagues at Carnegie Mellon University’s Center for Cognitive Brain Imaging (CCBI) (Just & Wang, 2001). STAR/PLUS fMRI data sets consist of sequences of images from the whole brain volume captured every 500 milliseconds (two brain images per second) whilst healthy subjects were undertaking a cognitive task. The task consisted of a collection of trials, each started with presenting a stimulus (picture or sentence) that remains on the screen for 4 seconds. Then, a blank screen appeared for another 4 seconds. After that, a second stimulus (picture or sentence) was presented for the next 4 seconds. Subjects were required to press the button ‘Yes’ or ‘No’ to identify whether the sentence described the picture correctly or not. Finally, each trial was followed by a 15-second resting period before the beginning of the next trial. Every trial was 27 seconds in length corresponding to 54 fMRI data points. Further information related to the experiment setting and stimuli, which are not explicitly discussed here, can be found from (Mitchell, et al., 2004). More information about the fMRI scanner, and the data pre-processing is presented in Appendix B. In the following, I present experiments A and B:

- **Experiment A**—fMRI data analysis related to reading affirmative *versus* negative sentences: It relates to the modelling of fMRI data to study how different areas of the

brain were involved in the processing of different sentence polarities. Considering that the hemodynamic response is slow, for each sentence stimulus that displayed on screen for 4 seconds, I used 8 seconds of continuous fMRI data corresponding to 16 images, which involve a 4-second resting gap. There were 20 samples for class 1: reading affirmative sentences and 20 samples for class 2: reading negative sentences. I employed a feature selection method to select informative voxels associated with the highly activated brain regions in response to a reading-related task. With respect to the selected features, I designed here three sessions of classification problems, each was based on using different fMRI voxel features.

- **Experiment B**—fMRI data analysis related to seeing pictures *versus* reading a sentences: It relates to modelling fMRI data to study the voxel activity patterns generated by different stimuli types (picture or sentence). The fMRI data were divided into two partitions (class 1: a subject was seeing pictures, class 2: a subject was reading sentences). I prepared 40 fMRI samples for each class, so in total 80 samples were used in this experiment.

For experiments A and B, the fMRI data of one subject (id: 05680) has been randomly selected from the STAR/PLUS website (Just & Wang, 2001). In this fMRI data, 25 anatomical regions of interest (ROI) were defined as explained in Appendix B.

The fMRI data dimension was defined by the x, y, and z voxel coordinates which compose a volume of $51 \times 56 \times 8$ as mapped in Figure 4-4. In this dimension, 5062 voxels were recorded from the entire brain volume. Experimental results were illustrated here mainly for the visual exploration of the SNN models, but I also performed a quantitative analysis, where numerical information of connection weights can be obtained from the model and statistically analysed.

4.3.1 fMRI Feature Selection

The brain is a complex information processing system. When fMRI data are recorded from the whole brain volume, considering the fact that various areas are involved for processing the input stimuli, this is essential for machine learning to identify the involved and activated voxels for the purpose of bias reduction. Therefore, as I designed my experiments with regards to the analysis of certain stimuli sets (pictures and sentences), I suggested that a feature selection needs to be employed to extract the important voxels (relevant to the cognitive task). In such a way I could ignore the involvement of irrelevant areas.

To this end, I applied the signal-to-noise ratio (SNR) method to select the more vital fMRI voxels with a high power of discrimination between the defined classes. The name ‘signal-to-noise ratio’ is a ‘jargon’ for a well-known statistical method that evaluates how important a variable is to discriminate samples belonging to different classes, one class named as ‘signal’ and the rest as ‘noise’ (being unwanted data). For the case of the two-class problem, an SNR ranking for variable x is calculated as an absolute difference between the mean value μ_{1x} of the variable for class 1 and the mean μ_{2x} of the variable for class 2, divided by the sum of the respective standard deviations as follows.

$$SNR_x = \frac{abs(\mu_{1x} - \mu_{2x})}{Std_{1x} + Std_{2x}} \quad (4-1)$$

Figure 4-3 illustrates the selected voxels from the fMRI data for each of the two experiments A and B, while Table 4-1 shows how many of these voxels belong to which region of interest (ROI). It can be concluded from Table 4-1 (left column) that when the subject was faced with different sentence polarities, the selected fMRI voxels (6 voxels) were located at Left

Dorsolateral Prefrontal Cortex (LDLPFC), which belongs to the middle frontal gyrus (lateral part of Brodmann areas 9 and 46). Table 4-1 (right column) reports about the case when the subject was dealing with pictures/sentences stimuli, and it shows that the more activated fMRI voxels belonged to Calcarine (CALC) in Brodmann area 17.

4.3.2 FMRI Data Mapping into SNN Models

I defined two mapping structures: M_1 relates to personalised mapping of an individual fMRI coordinate and M_2 relates to transferring an individual fMRI coordinate into the Talairach atlas. Figure 4-4 relates to mapping M_1 , which illustrates the spatial mapping of 5062 voxels into an SNN model. Input neurons are allocated and labelled to represent the pre-selected input voxels as per the selection in Table 4-1. Figure 4-5 illustrates that mapping M_2 was used to transfer the coordinates of the pre-selected voxels and map them into an SNN model of 1471 spiking neurons according to the Talairach brain template.

4.3.3 Unsupervised Learning and Pattern Visualisation in SNN Models

In Experiment A, Figure 4-6 (a) shows the initial connections in the SNN model and Figure 4-6(b) shows the modified connections after the STDP unsupervised learning process using the fMRI samples related to both affirmative and negative sentences. Figure 4-6 (b) shows that more and stronger connections were generated in the left hemisphere than in other areas. These connections were established because of more spikes transferred between the neurons located in these areas, reflecting the changes in the corresponding voxels.

These findings suggested that language comprehension, including a reading task, is processed in several areas, and mostly observed in the Left Dorsolateral Prefrontal Cortex,

Broca, and Wernicke as confirmed in (Yuasa, Saito, & Mukawa, 2011). Figure 4-7 (a) shows the SNN model connectivity related to fMRI samples of the affirmative sentences, while Figure 4-7 (b) relates to the negative sentences.

The observed connectivity from Figure 4-7 confirms that the subject performed differently when reading an affirmative (average connection weights=0.61) *versus* negative sentence (average connection weights=1.7). Table 4-2 represents the quantitative information about the average connection weights around each input neuron (input fMRI voxel) that shows the differences between the trained SNN models of affirmative and negative sentences.

It can be seen from Table 4-2 and Figure 4-7 that the connectivity was especially enhanced between the input neurons located in the left hemisphere of the SNN model when reading negative sentences. Distribution of the SNN connection weights around the input voxels located in the left and right hemispheres are illustrated in Appendix B, Figures B-1 and B-2.

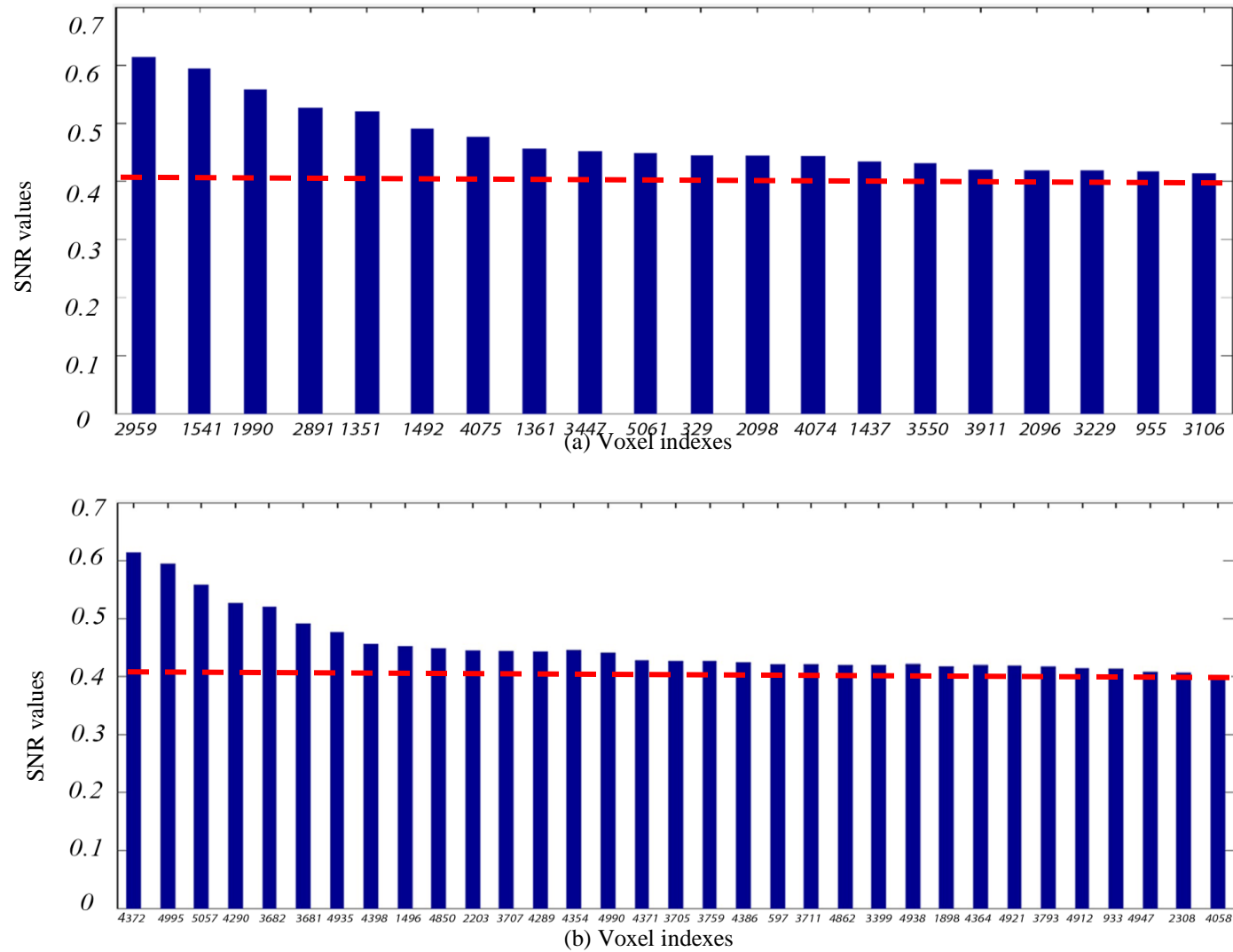


Figure 4-3 The SNR ranking (on the y-axis) of top voxels (on the x-axis) related to (a) the affirmative *versus* negative sentences and (b) the pictures *versus* sentences. The top voxels were selected according to their SNR values that were greater than a threshold= 0.4.

Table 4-1 Informative voxels are selected using an SNR feature selection from two fMRI data sets. The voxels were selected due to their SNR values were higher than a threshold= 0.4.

Activated ROIs in Affirmative <i>vs</i> Negative sentence task (number of voxels selected in Figure 4-3 (a) that belong to each of these regions).	Activated ROIs in Picture <i>vs</i> Sentence task (number of voxels selected in Figure 4-3 (b) that belong to each of these regions).
LT (3), LOPER (3), LIPL (1), LDLPFC (6), RT (2), CALC (1), LSGA (1), RDLPFC (1), RSGA (1), RIT (1)	CALC (5), ROPER (3), LT (4), LOPER (3), LSPL (1), RIPS (3), LPPREC (1), RT (4), LFEF (1), LDLPFC (3), RDLPFC (1), LIPS (2), RPPREC (1), LIT (1)

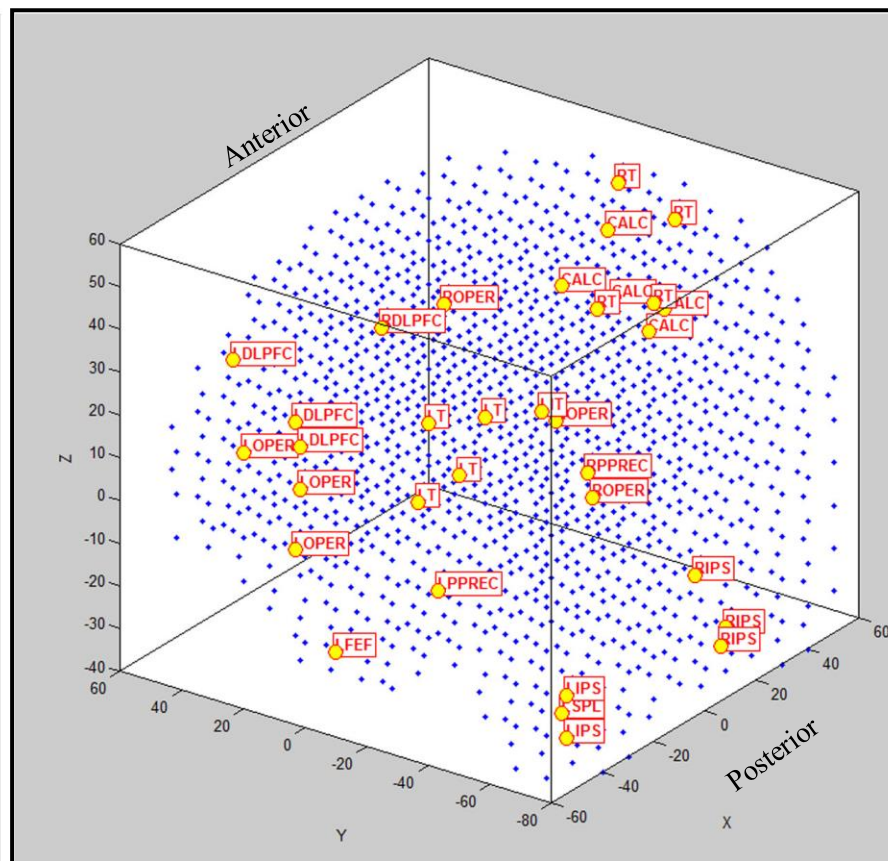
The SNN connections in Figure 4-7 were captured when the models were trained with all of the 20 fMRI samples from affirmative sentences, Figure 4-7 (a), and 20 fMRI samples from the negative sentences, Figure 4-7 (b). Now, to evaluate the confidence of the designed SNN models and to investigate how the models of affirmative and negative sentences are statistically significant, I designed an iterative procedure of unsupervised learning through a leave-one-out method. As shown in Figure 4-2, at each iteration the SNN model is first initialised and then trained with a fold of samples in which one sample is holdout from the training while the rest of samples are involved. Then the average of the connection weights in the trained SNN model is measured. In the next iteration, the SNN model is again initialised and the holdout sample is replaced by another sample to form the training set. This procedure is terminated when all the samples were replaced by the holdout sample, one by one.

A 3D scatter plot illustrating brain activation clusters. The vertical axis is labeled 'Z' and ranges from 1 to 8. The horizontal axis is labeled 'X' and ranges from 10 to 50. The depth axis is labeled 'Y' and ranges from 10 to 50. The plot shows a dense distribution of blue dots representing individual subjects' data points. Yellow spheres highlight specific clusters of activation. Several red boxes with labels are overlaid on the plot, indicating specific regions of interest: LT, LIT, CALC, LOPER, LDLPFC, RT, ROPER, RDLPFC, LPPREC, RPPREC, LIPS, LRS, LSPJ, RIPS, ROPER, and RPPREC. The labels are positioned around the perimeter of the plot, corresponding to the locations of the clusters.

(b)

Figure 4-4 The fMRI data dimension of one person is defined by the maximum value of x, y, and z coordinates of voxels that forms a volume size of 51×56×8. In this dimension, 5062 voxels are captured. These voxel coordinates are mapped into an SNN model. The selected top-informative voxels in Table 4-1 for each experiment are used as input variables and their ROIs are shown in the text boxes. (a) Affirmative *versus* negative sentences; (b) Pictures *versus* sentences.

Figure 4-5 The coordinates of the top-informative voxels in Table 4-1 are transferred to the Talairach and used as input variables, shown as circles along with the ROIs (as text in the boxes) for (a) affirmative *versus* negative sentences and (b) pictures *versus* sentences.



(b)

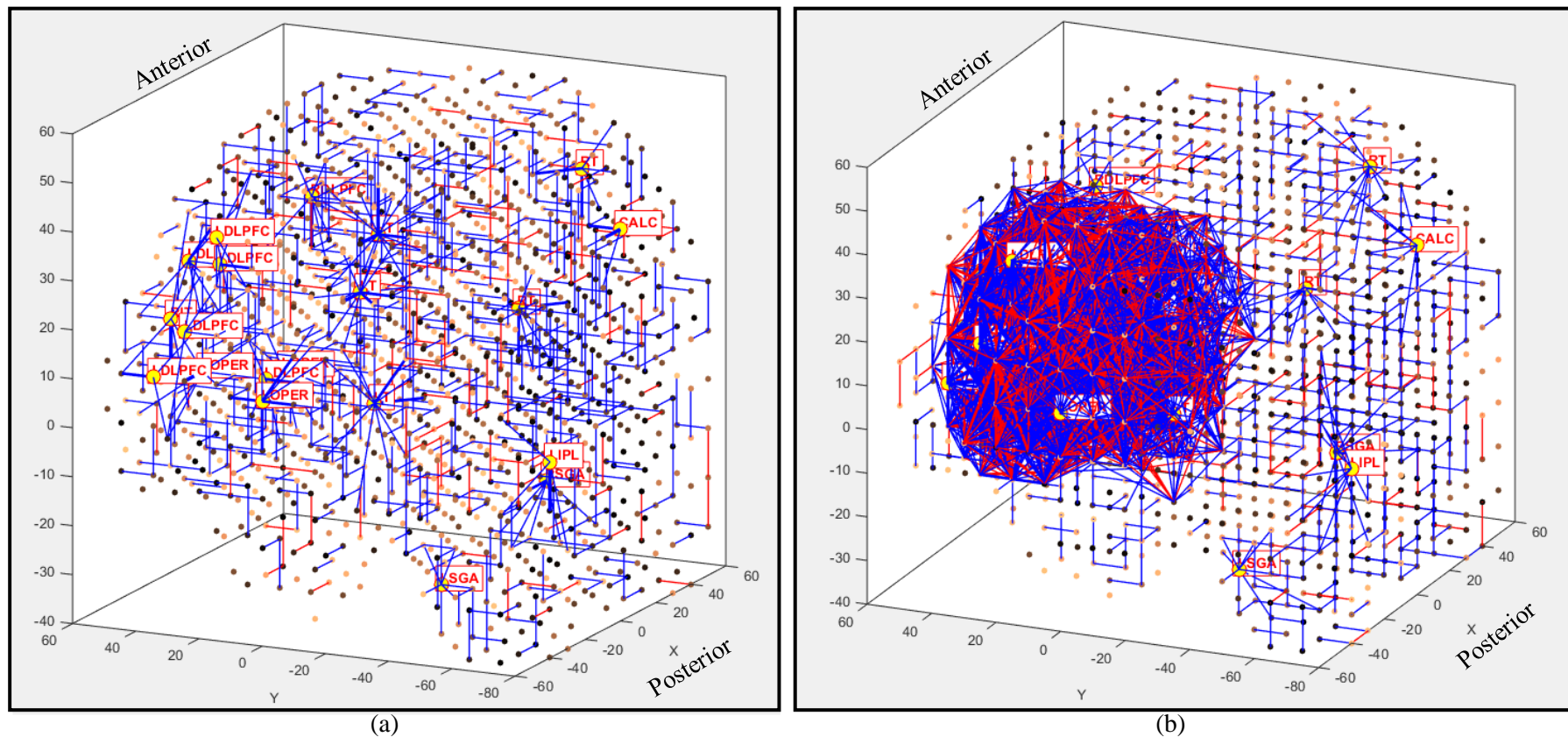


Figure 4-6 (a) The initial connections in the SNN model; (b) The learnt connections (absolute connection weights > 0.08) after STDP unsupervised learning using fMRI data of both affirmative and negative sentences (20 fMRI samples for affirmative sentence and 20 fMRI samples for negative sentences). The SNN models are mapped using the Talairach atlas with an allocation of 20 input voxels (Experiment A).

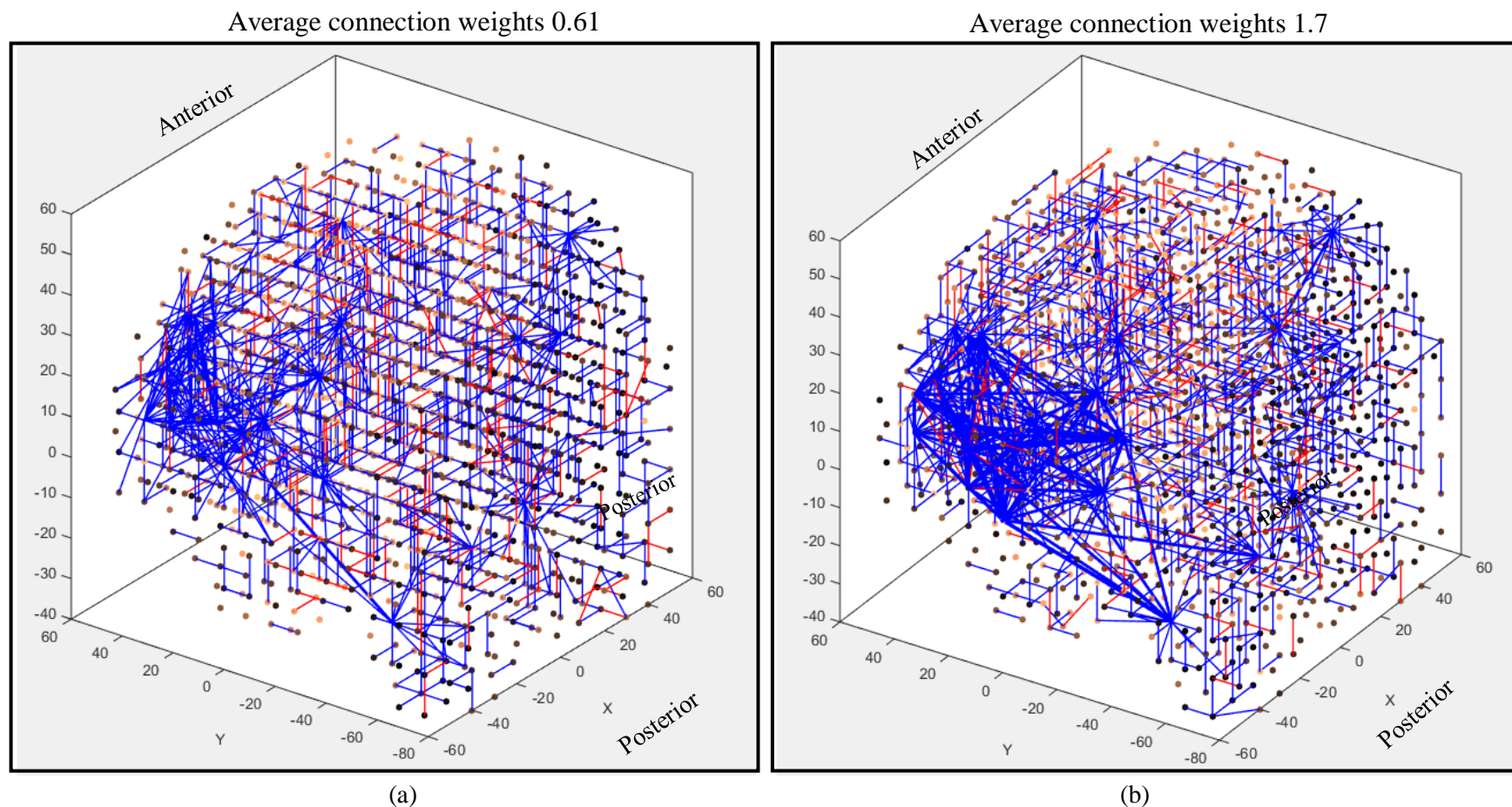


Figure 4-7 (a) The learnt connections (absolute connection weights > 0.08) in an SNN model when only the fMRI samples of affirmative sentences were used (20 fMRI samples); (b) The learnt connections in an SNN model when only the fMRI samples of negative sentences were used (20 fMRI samples).

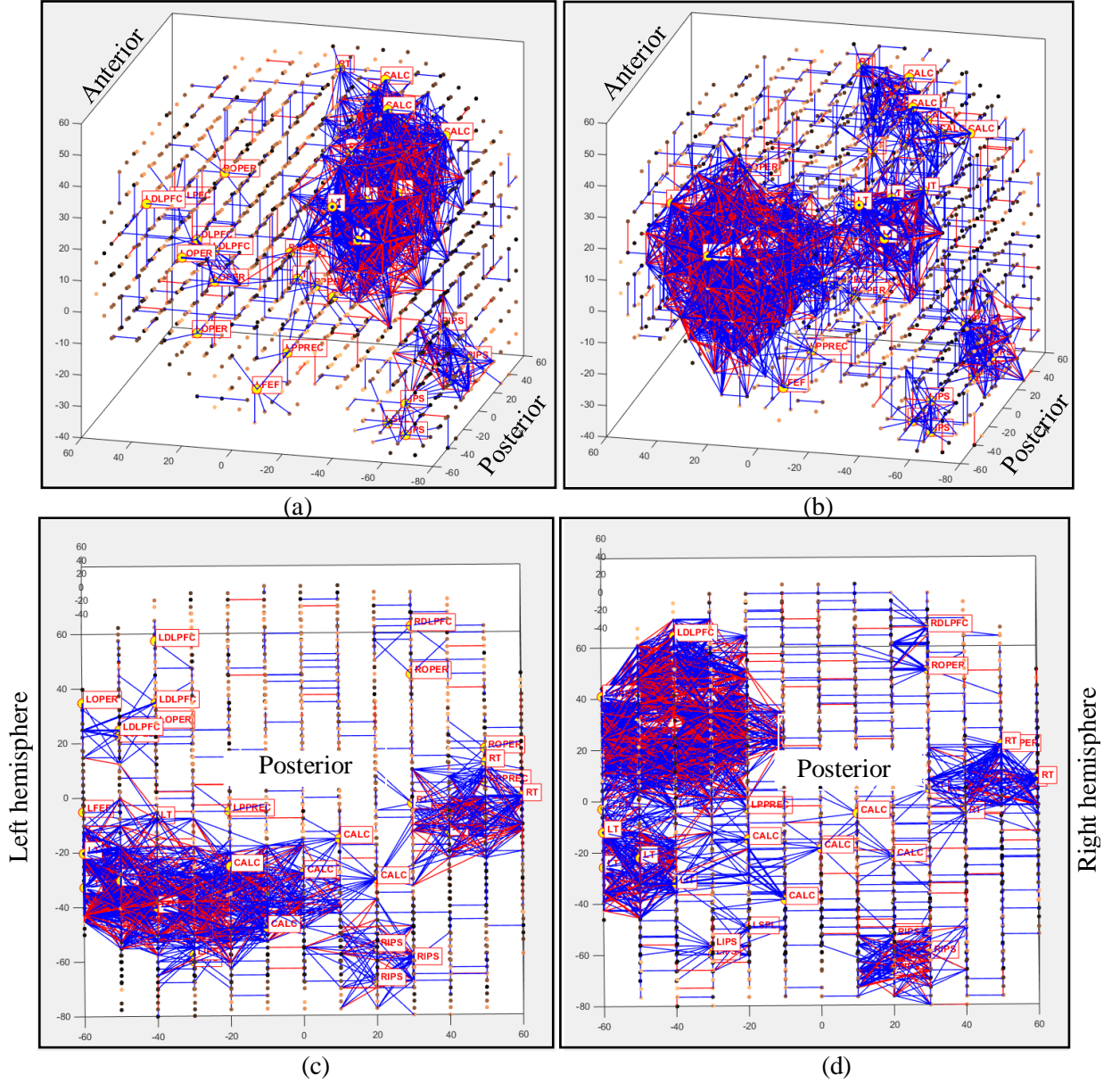


Figure 4-8 (a) Connectivity of an SNN model trained on fMRI data related to seeing pictures; (b) Connectivity of an SNN model trained on fMRI related to reading sentences; (c) 2-D coronal projection of the connectivity of the SNN model from Figure 4-8 (a); (d) 2-D coronal projection of the SNN model from Figure 4-8 (b). Connection weights > 0.08 .

Table 4-3 reports the average of the connection weights in SNN models over 40 training iterations (20 samples of affirmative and 20 samples of negative classes). I applied a t-test to the connection weights of these iterations as reported in Appendix B, Table B-1. The p -

value=1.3E-07 represented that the trained SNN models of affirmative and negative sentences are statistically significant.

In experiment B, which relates to the fMRI data analysis of the picture *versus* sentence observation task, Figure 4-8 (a) shows the learnt connections in an SNN model using samples of seeing pictures and Figure 4-8 (b) relates to reading sentences. Figure 4-8 (c) and Figure 4-8 (d) show the 2-D visualisations of the connectivity from Figure 4-8 (a) and Figure 4-8 (b) correspondingly. After training the SNN model with the fMRI samples of seeing pictures, Figure 4-8 (a) and Figure 4-8 (c) represent stronger spatio-temporal connections between neurons located in the parts of the brain dedicated to vision, such as the Calcarine (CALC) region, which is located in the primary visual cortex in the Occipital Lobe (BA 17).

On the other hand, as shown in Figure 4-8 (b) and Figure 4-8 (d), when the SNN model was trained with spike trains that represent fMRI data related to sentence stimuli, the connections were mostly enhanced in the left hemisphere, particularly, in the Broca and Wernicke areas. This corresponds to the studies about brain areas involved in language comprehension. Figure 4-9 shows the SNN model connectivity after STDP unsupervised training with the use of two different data sets related to affirmative sentences and negative sentences. Figure 4-10 illustrates the SNN models trained on fMRI data of seeing pictures and reading sentences. Figure 4-9 illustrates that stronger connections were generated between the neurons located in the left hemisphere, significantly in the LDLPFC region, when reading negative sentences instead of affirmative sentences.

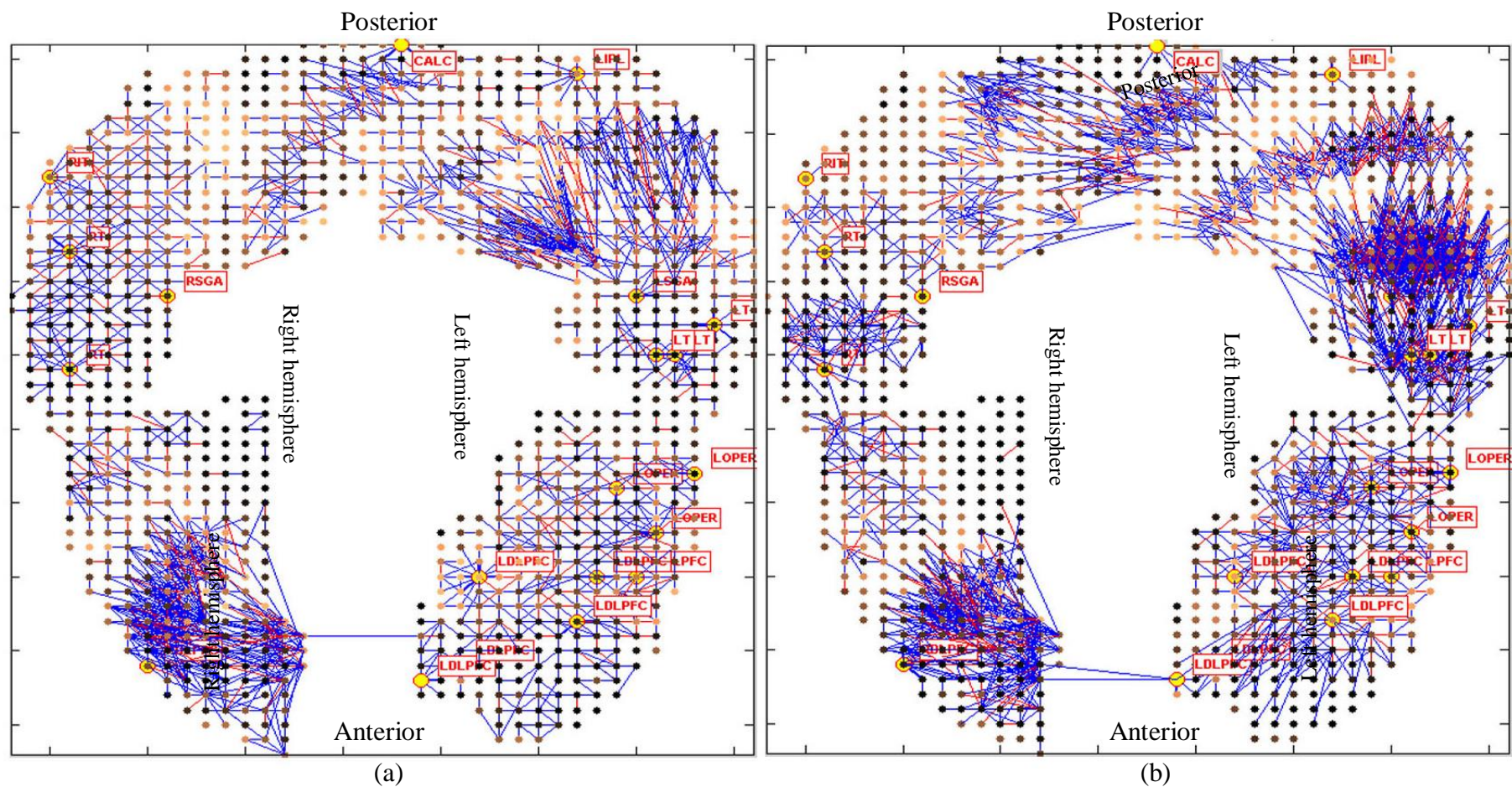


Figure 4-9 2-D coronal projection of the adapted connections in the SNN models after unsupervised learning with two data sets related to affirmative sentences (a) and negative sentences (b). The positive connections are shown in blue and negative ones in red (absolute connection weights > 0.08).

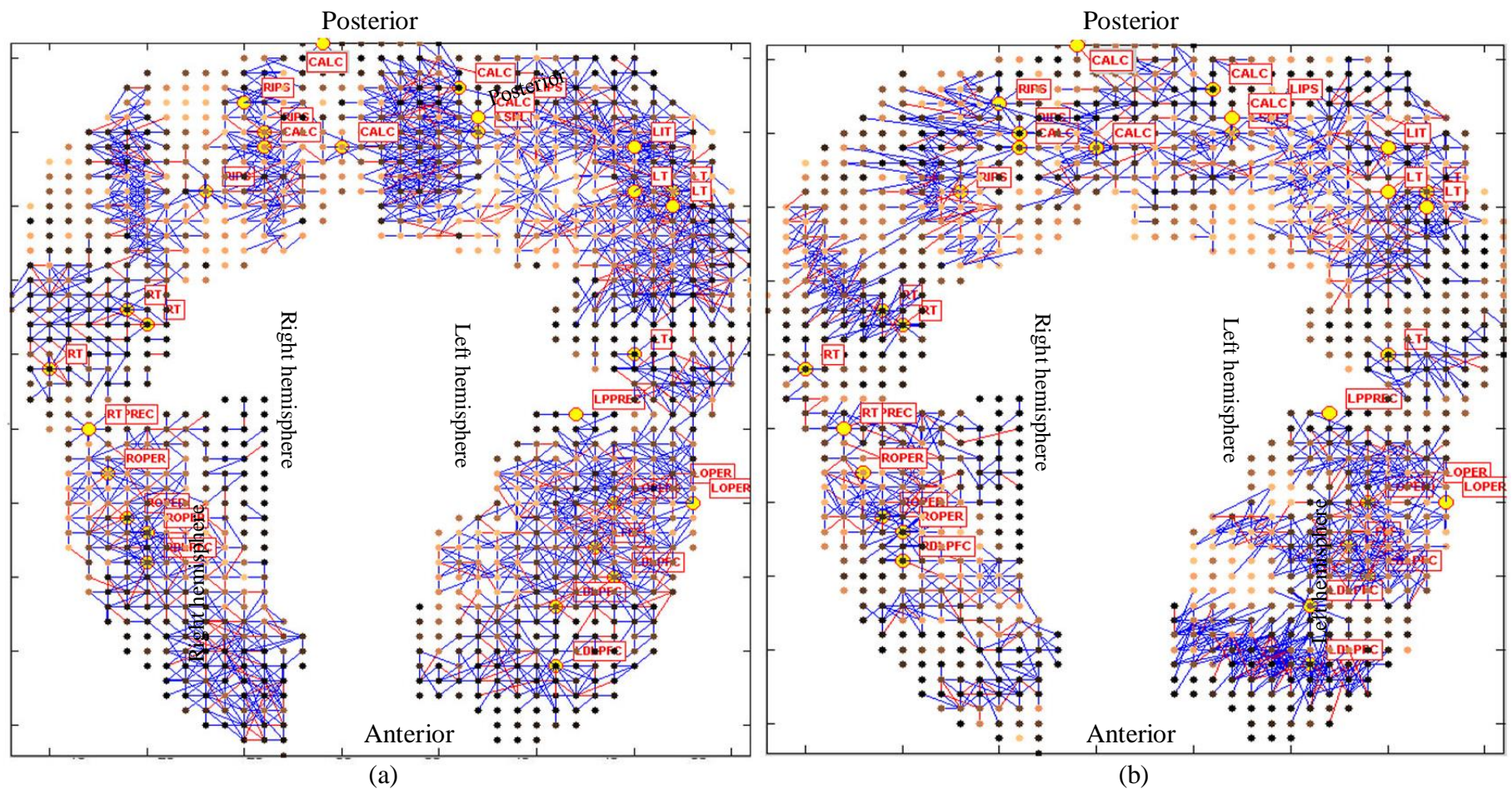


Figure 4-10 2-D coronal projection of the adapted connections in the SNN models after unsupervised learning with two data sets related to (a) seeing pictures and (b) reading sentences. The positive connections are shown in blue and negative ones in red (absolute connection weights > 0.08).

Table 4-2 Average connection weights around each input neuron in the trained SNN models from Figure 4-7 related to affirmative (A) and negative (N) sentences. The sum of the average connection weights across all the input neurons in each SNN model is reported in the last column.

	LT	LOPER	LIPL	LOPER	LDLPFC	LOPER	LT	LDLPFC	RT	CALC	LSGA	LDLPFC	LT	LDLPFC	RT	LDLPFC	LDLPFC	RDLPFC	RSGA	RIT	Avg
N	1.4	0.92	1.87	1.03	2.08	1.12	1.48	0.44	0.2	0.89	1.84	1.03	1.9	0.45	1.1	1.26	0.56	0.19	0.43	1.4	1.7
A	0.9	0.56	1.01	0.87	1.03	0.65	0.89	0.23	0.1	0.43	1.04	0.68	1.1	0.17	0.8	0.24	0.22	0.11	0.32	0.9	0.6

Table 4-3 Average connection weights of the trained SNN models for each iteration. Two SNN models were trained over 40 iterations (20 iterations for Affirmative (A) and 20 for negative (N) sentences) using different folds of samples. At each iteration, one sample was taken out from the training and the model was trained by the remaining samples. This procedure was repeated for all the 20 samples for both affirmative and negative sentences. The p -value represents that the trained SNN models of affirmative and negative sentences are statistically significant

	Number of iterations for training the SNN models of fMRI samples related to affirmative and negative classes																			
N	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
	1.3	0.98	1.3	1.07	0.89	1.23	1.09	1.52	1.21	1.05	1.23	1.01	0.98	1.06	1.6	1.4	1.7	1.8	1.32	1.3
A	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
	0.71	0.6	1.01	1.2	0.71	0.78	0.67	0.98	0.78	1.01	0.98	0.56	0.62	0.49	0.81	0.71	0.99	0.7	0.74	0.83
	Negative sentence class mean: 1.16 Affirmative sentence class mean: 0.73 Negative sentence SD: 0.25 Affirmative sentence SD: 0.18 p-value=1.3E-07																			

Table 4-4 Average connection weights of the trained SNN model for each iteration. Two SNN models were trained over 80 iterations (40 iterations for pictures and 40 for sentences) using different folds of samples. At each iteration, one sample was taken out from the training set and the model was trained by the remaining samples. This procedure was repeated for all the samples in both classes. The p -value represents that the trained SNN models of pictures and sentences are statistically significant.

		Number of iterations for training the SNN models with fMRI samples from picture and sentence classes																			
Picture		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
		1.02	1.98	1.2	1.8	1.7	1.1	1	0.93	0.96	1.3	1.1	0.99	1.05	1.6	1.6	1.5	1.8	1.3	1.8	0.9
		21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
		1.4	1.6	1.8	1.5	1.1	0.98	1.9	1.3	1.3	1.8	1.59	0.8	0.7	0.92	1.9	1.1	1.2	0.8	0.9	1.7
Sentence		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20
		0.8	0.98	1.1	1.1	1.9	1.3	1.4	1.3	1.1	0.98	0.99	1.1	1	1.4	1.2	1.1	1.3	1.1	1.2	0.89
		21	22	23	24	25	26	27	28	29	30	31	32	33	34	35	36	37	38	39	40
		0.78	0.98	0.7	1.2	1.6	1.7	1.3	0.9	0.9	1.1	1.3	0.9	1	1	0.89	0.79	1.1	1.1	1.2	1.3
		Picture class mean: 1.32				Sentence class mean: 1.1				Picture SD: 0.3				Sentence SD:0.2				p -value:0.006			

On the other hand, Figure 4-10 shows that visual areas were more activated than other areas when seeing pictures. The SNN models were trained several times using different folds of the data (leave-one-out method as shown in Figure 4-2). To evaluate how the trained SNN models were statistically significant, I applied a t-test as reported in Table 4-2 , Table 4-3 and Table 4-4.

4.3.4 Classification in SNN Models

The classification task was performed using a Leave-one-out Cross Validation (LOOCV) method. In this method, only one sample is picked up as the test set and a model is built using all the remaining samples (training set). Then the trained model is tested using the single holdout sample. The final classification accuracy is obtained by repeating this procedure for each of the samples and averaging the results. For optimisation, I performed an exhaustive grid-search on combinations of parameters for every sample model, as explained in Appendix B, Table B-2. The performance of the classifier was measured using the F_{score} to evaluate the test's accuracy, as explained in Appendix C, Section C.7.

Table 4-5 summarises the fMRI data classification accuracy of the affirmative sentence class *versus* negative sentence class obtained through three sessions (different voxel features were selected as represented in Table 4-1). The results were also compared with results obtained using conventional machine learning methods. The methods that I used for comparison were: SVM, MLR, MLP and Evolving Clustering Method (ECM). I also compared them with the results obtained by Behroozi in (Behroozi & Daliri, 2014) that employed SVM method for classification of the fMRI samples. It can be seen from Table 4-5 that the designed SNN models achieved better classification accuracy when compared with conventional methods. Table 4-6 represents the classification accuracy of picture *versus* sentence classes obtained

using the designed SNN models and conventional machine learning methods. In both experiments A and B, the SNN models learnt the fMRI patterns from sets of streaming spatio-temporal data over time. In contrast, the same fMRI data were learnt in the conventional machine learning methods as vector-based data, where vectors were formed through the concatenating of temporal frames.

Table 4-5 Classification accuracy of the affirmative sentences (class C1) *versus* negative sentences (class C2) via an SNN model using the LOOCV method. The results of conventional machine learning methods along with the SVM classification results from (Behroozi & Daliri, 2014) are also reported. The experiment is done on a total number of 40 samples (20 samples per class).

Method	Sessions and selected voxels for classification	C1 (Affirm)	C2 (Negat)	Average	F_{Score}
NeuCube	<i>Session I:</i> 20 voxels selected from Table 4-1 (left column)	85.00	95.00	90.00	88.00
	<i>Session II:</i> 20 pre-selected voxels from RDLPFC region	85.00	80.00	82.50	84.00
	<i>Session III:</i> 20 pre-selected voxels from LDLPFC region	90.00	85.00	87.50	84.00
SVM results obtained in (Behroozi & Daliri, 2014)	<i>Session I:</i> classification based on the LDLPFC's voxels	64.00	68.00	66.00	—
	<i>Session II:</i> classification based on the RDLPFC's voxels	65.00	69.00	67.00	—
SVM	SVM Kernal: Polynomial, Degree, Gamma, N/A : 1	70.00	75.00	73.00	71.00
MLP	Number of Hidden Units=180, Number of Training Cycles=600, Output Activation Function- linear.	75.00	65.00	70.00	71.00
ECM	Maximum Field Radius=2; Minimum Field Radius=0.01, M of N=3,	65.00	70.00	70.00	67.00
MLR	Class Performance Variance=0.26	65.00	60.00	63.00	65.00

Table 4-6 Classification accuracy of pictures (class C1) *versus* sentences (class C2) obtained via an SNN model using the LOOCV method. The results of the conventional machine learning methods are also reported. The experiment is done on a total number of 80 samples (40 samples per class).

Classification accuracy results from NeuCube-based model		Accuracy			
Method	Session and selected voxels for classification	C1 (Pic)	C2 (Sen)	Average	F_{Score}
NeuCube	<i>Session I:</i> 33 voxels selected from Table 4-1 (right column)	95.00	90.00	92.00	90.00
Classification results from conventional machine learning methods					
SVM	SVM Kernal: Polynomial, Degree, Gamma, N/A:1	85.00	85.00	85.00	85.00
MLP	Number of Hidden Units: 304, Number of Training Cycles: 300 Output Activation Function: linear	75.00	77.00	76.00	75.00
ECM	Maximum Field R.=1; Minimum Field R.=0.01; M of N= 3	82.00	72.00	77.00	79.00
MLR	Class Performance Variance: 0.13	75.00	62.00	68.00	71.00

4.4 SNN Application on EEG

Employing the NeuCube framework for EEG, I designed three sessions of experiments, each includes both classification and pattern analysis as explained in Section 4.2. The EEG data were recorded *via* 26 EEG channels: Fp1, Fp2, Fz, F3, F4, F7, F8, Cz, C3, C4, CP3, CPz, CP4, FC3, FCz, FC4, T3, T4, T5, T6, Pz, P3, P4, O1, O2, and Oz electrode sites (10-20 International System) whilst participants performed a cognitive GO/ NOGO task. The EEG data were recorded from 70 participants in three groups as follows:

- (a) The Methadone Maintenance Treatment (MMT) group containing 31 subjects (17 male, 14 female) with a mean age of 39.36, standard deviation (SD)=5.14, was recruited by recommendations from the case managers of Auckland Community Alcohol Drug Services (CADS), New Zealand. The current methadone dose was 70.86 (SD=40.61; range 8–180) mg/day.
- (b) The Opiate users (OP) group containing 18 subjects (11 male, 7 female) with a mean age of 37.38 (SD=7.44), was recruited from the Auckland Drug Information Outreach (ADIO) Trust Needle Exchange Services.
- (c) The Healthy control (H) group containing 21 subjects (11 male, 10 female) with a mean age of 36.12 (SD=6.61), was recruited by advertisements (notices posted on notice boards) distributed in a range of local communities, such as the public library, shopping mall, cafes, or by word of mouth. The designed cognitive task was as follows:

During the GO/NOGO task, participants were repeatedly presented with the word ‘PRESS’, which appeared randomly either in red or green. They were instructed to press a button in response to the green ‘PRESS’ (GO) and not respond to the red ‘PRESS’ (NOGO). Further

information about the GO/NOGO task, EEG data recording, participants and demographic and clinical measures are presented in Appendix C. In order to analyse the EEG data, I designed three experimental sessions on the recorded data, each of which investigated different EEG epochs as follows:

- *Session I:* I prepared six EEG sample files, each containing EEG epochs measured from one group (MMT/ OP/ H subjects) per cognitive task (GO *versus* NOGO). Then each sample file was entered separately into the SNN model to capture the differences between the brain activity patterns of different groups of participants performing GO *versus* NOGO responses.
- *Session II:* I only studied the EEG epochs related to the GO trials to compare the brain activity patterns of different groups when performing the same cognitive task.
- *Session III:* I only studied the EEG epochs related to the NOGO trials as these trials are of common interest in studies on response inhibition.

The organisation of the EEG samples, as used for each of these sessions, is presented in Table 4-7.

4.4.1 Pattern Analysis in SNN Models

To evaluate how the SNN models can capture different patterns when they are trained by different classes of data, I visualised them and statistically analysed their spatio-temporal connectivity. Figure 4-11 illustrates the connectivity in SNN models, each was trained by one-fold EEG data (20 samples as the training set when one sample was holdout) from H group in both GO and NOGO. It can be seen from Figure 4-11 that the average connection weights in the trained SNN model of H subjects was smaller in NOGO trials (0.06) when the

response had to be withheld in comparison with GO trials (0.1) when the response was required. Figure 4-12 and Figure 4-13 illustrate the SNN models trained with one-fold EEG data respectively from MMT and OP groups related to GO and NOGO trials. For both MMT and OP subjects, the average connection weights induced by the NOGO trials (0.21 and 0.17 respectively) were greater than those induced by the GO trials (0.1 and 0.08 respectively).

As can be seen from Figure 4-11 to Figure 4-13, the SNN models were trained by only one-fold EEG data as the training set. Now, in order to evaluate the level of confidence for the models, Table 4-8 to Table 4-10 report the average connection weights of the SNN models that were trained over a number of iterations using different folds of the EEG samples (leave-one-out method). Table 4-8 shows that the SNN models of H subjects were trained over 42 iterations (21 iterations per class: GO and NOGO) using different folds of samples. At each iteration, one sample was holdout, then the model was trained using the remaining samples.

Table 4-7 EEG data sets for the three experimental sessions to compare the brain activity patterns of the H, MMT, and OP subjects in a GO/NOGO task. Due to the quality of the data, some participants' data were omitted from the experimental sessions.

Session I: EEG data sample files for GO versus NOGO classification		
classifications	Samples per class	EEG sample file size
GO trials class (68 participants)	21 Healthy subjects	75 EEG time points * 26 channels * 21 samples
	18 OP subjects	75 EEG time points * 26 channels * 18 samples
	29 MMT subjects	75 EEG time points * 26 channels * 29 samples
NOGO trials class (70 participants)	21 Healthy subjects	75 EEG time points * 26 channels * 21 samples
	18 OP subjects	75 EEG time points * 26 channels * 18 samples
	31 MMT subjects	75 EEG time points * 26 channels * 31 samples
Session II: EEG data sample files captured during GO trials		
MMT class vs H class (50 participants)	29 MMT subjects (class 1)	75 EEG time points *26 channels *50 samples
	21 Healthy subjects (class 2)	
OP class vs H class (39 participants)	18 Opiate subjects (class 1)	75 EEG time points *26 channels *39 samples
	21 Healthy subjects (class2)	
MMT class vs OP class (47 participants)	29 MMT subjects (class 1)	75 EEG time points *26 channels *47 samples
	18 Opiate subjects (class 2)	
Session III: EEG data sample files captured during NOGO trials		
MMT class vs H class (52 participants)	31 MMT subjects (class 1)	75 EEG time points *26 channels *52 samples
	21 Healthy subjects (class 2)	
OP class vs H class (39 participants)	18 OP subjects (class 1)	75 EEG time points *26 channels *39 samples
	21 Healthy subjects (class2)	
MMT class vs OP class (49 participants)	31 MMT subjects (class 1)	75 EEG time points *26 channels *49 samples
	18 OP subjects (class 2)	

This procedure was repeated for all the 21 samples in both GO and NOGO (42 training iterations in total). Table 4-9 is related to MMT group and that shows the SNN models were trained over 60 iterations (29 iterations for class GO and 31 iterations for class NOGO) using different folds of samples. Table 4-10 is related to OP group and that shows the SNN models were trained over 36 iterations (18 iterations for each class GO and NOGO) using different folds of samples.

To evaluate how the SNN models of GO and NOGO were statistically significant, I applied a t-test to the average connection weights at each of the training iterations. The corresponding *p*-values are reported in the tables. The *p*-value in all the experimental sessions is less than 0.05, which shows that the results are statistically significant with a high confidence, greater than 99%. Further quantitative information about the interactions between the EEG variables in the SNN models during STDP learning is presented in Appendix C, Section C.6.

4.4.2 Classification in SNN Models

The classification task was performed using the LOOCV method. The performance of the classifier (reported in Table 4-11) was measured using the F_{Score} (explained in Appendix C, Section C.7). For optimisation, I performed an exhaustive grid-search model as explained in Appendix C, Section C.8.

Table 4-12 presents the classification accuracy obtained using other machine learning methods, which are still being actively used in literature for classification of STBD. The methods used for comparison were: SVM and MLP. The classification accuracy results of these methods are reported in Table 4-12. The parameter optimisation procedure and EEG data preparation for the conventional classifiers are presented in Appendix C, Section C.9.

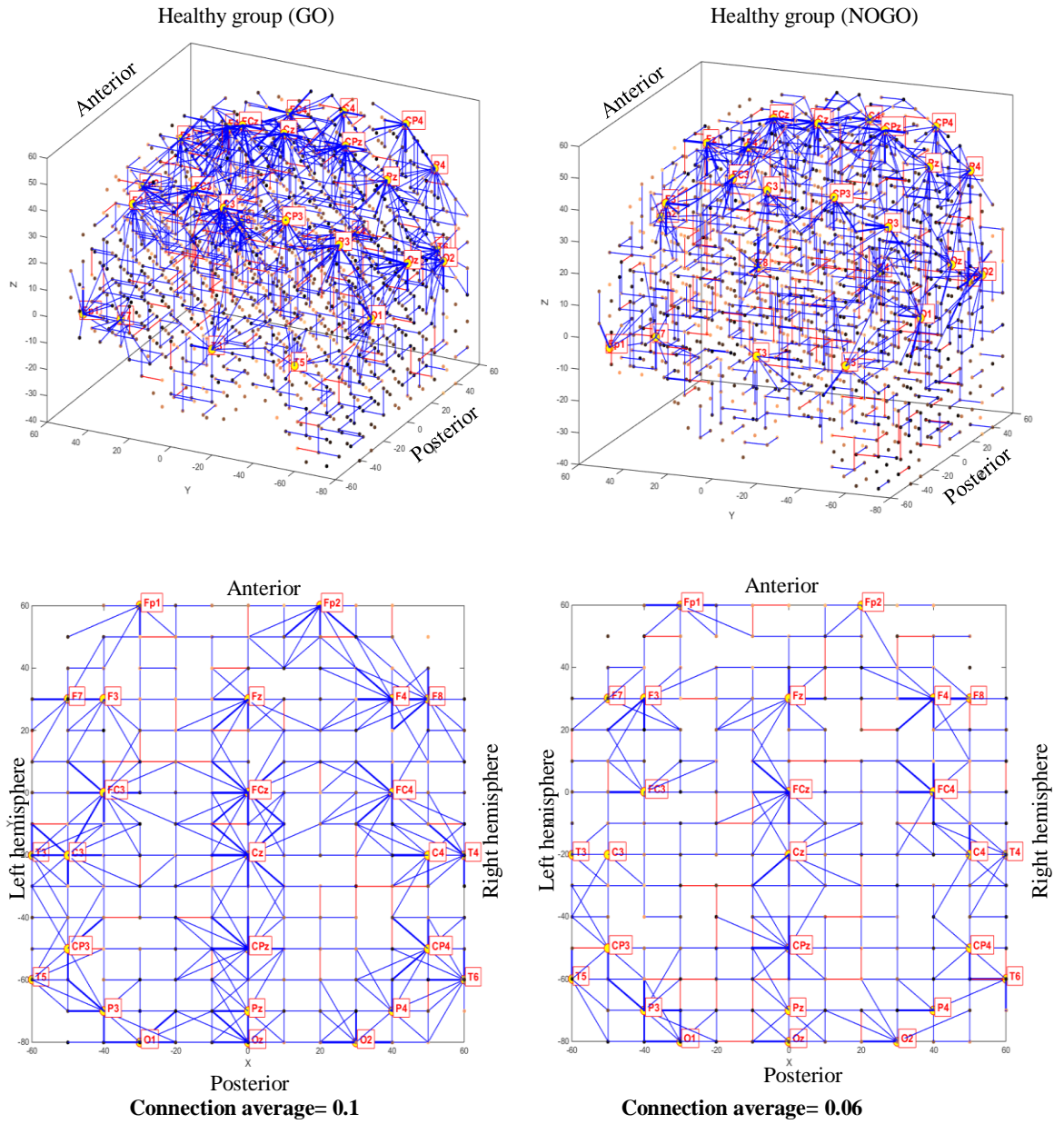


Figure 4-11 Visualisation of the SNN model connectivity of the H group (absolute connection weights > 0.08) after the STDP learning with EEG data of 26 features (channels) for GO and NOGO trials. The average of the connection weights in each trained SNN model is also presented. The blue lines are positive (excitatory) connections, while the red lines are negative (inhibitory) connections. The thickness of the lines identifies the weight of the connection. The 1471 neurons of the SNN model are spatially mapped according to the Talairach atlas (Koessler, et al., 2009). The SNN models are visualised in 3-D (top) and 2-D coronal projection (bottom).

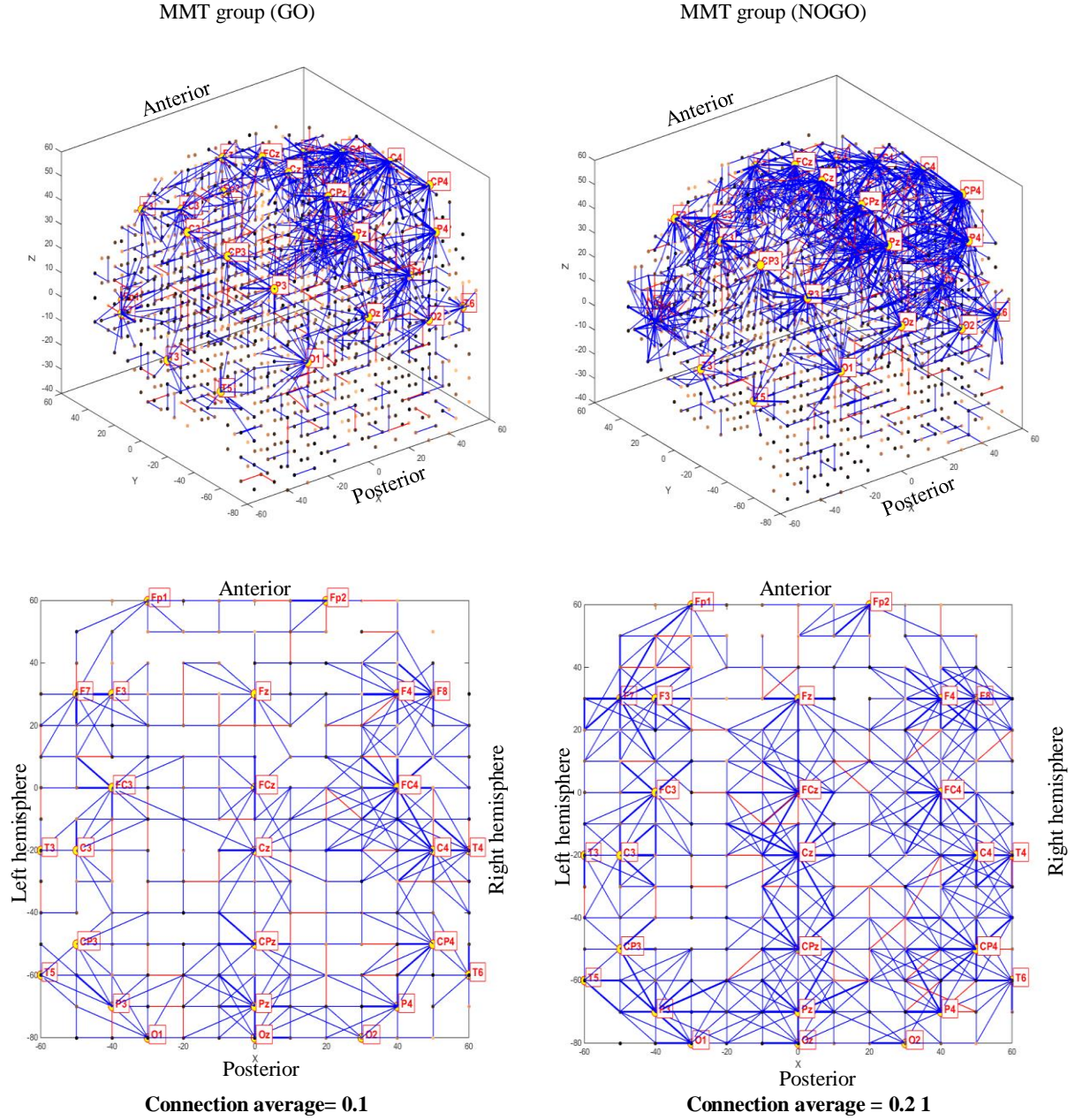


Figure 4-12 Visualisation of the SNN model connectivity of the MMT group (absolute connection weights > 0.08) after the STDP learning with EEG data of 26 features (channels) for GO and NOGO trials. The average of the connection weights in each trained SNN model is also presented. The blue lines are positive (excitatory) connections, while the red lines are negative (inhibitory) connections. The thickness of the lines identifies the weight of the connection. The 1471 neurons of the SNN model are spatially mapped according to the Talairach atlas (Koessler, et al., 2009). The SNN models are visualised in 3-D (top) and 2-D coronal projection (bottom).

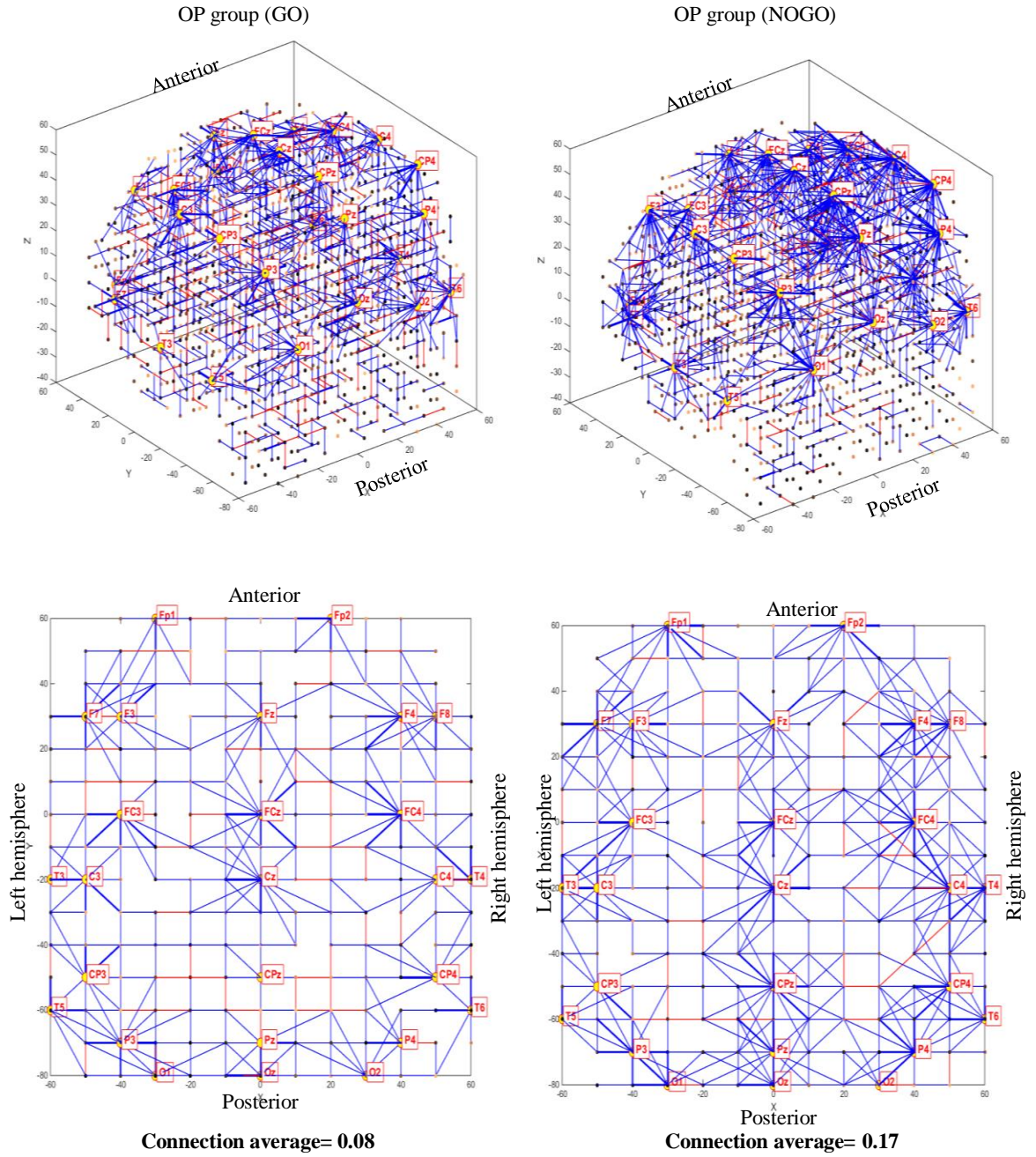


Figure 4-13 Visualisation of the SNN model connectivity of the OP group (absolute connection weights>0.08) after the STDP learning with EEG data of 26 features (channels) for GO and NOGO trials. The average of the connection weights in each trained SNN model is also presented. The blue lines are positive (excitatory) connections, while the red lines are negative (inhibitory) connections. The thickness of the lines identifies the weight of the connection. The 1471 neurons of the SNN model are spatially mapped according to the Talairach atlas (Koessler, et al., 2009). The SNN models are visualised in 3-D (top) and 2-D coronal projection (bottom).

	Number of iterations for training the SNN models of H groups in GO and NOGO classes																				
GO	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
	0.11	0.011	0.11	0.09	0.089	0.098	0.11	0.1	0.11	0.06	0.12	0.11	0.11	0.09	0.08	0.091	0.1	0.1	0.1	0.09	0.08
NG	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	19	20	21
	0.06	0.08	0.07	0.01	0.09	0.07	0.08	0.06	0.08	0.07	0.08	0.07	0.08	0.09	0.08	0.06	0.06	0.06	0.06	0.09	0.08
	GO mean: 0.09 NOGO mean: 0.07 GO SD: 0.02 NOGO SD: 0.01 p -value=0.002																				

Table 4-9 Average connection weights of the SNN models trained over 60 iterations (29 iterations for class GO and 31 iterations for class NOGO) using different folds of samples. At each iteration, one sample was taken out and the model was trained by the remaining samples. This procedure was repeated for all the samples in both GO and NOGO. The p -value represents that the trained SNN models of GO and NOGO are statistically significant with a high confidence, greater than 99%.

	Number of iterations for training the SNN models of MMT groups in GO and NOGO classes															
GO	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	
	0.1	0.19	0.19	0.12	0.18	0.1	0.19	0.18	0.19	0.19	0.107	0.1	0.18	0.1	0.1	
	16	17	18	19	20	21	22	23	24	25	26	27	28	29		
	0.18	0.19	0.1	0.1	0.19	0.17	0.19	0.18	0.18	0.17	0.19	0.19	0.19	0.15		
NG	1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16
	0.21	0.11	0.18	0.1	0.2	0.12	0.27	0.18	0.19	0.19	0.24	0.27	0.12	0.2	0.19	0.21
	17	18	19	20	21	22	23	24	25	26	27	28	29	30	31	
	0.16	0.18	0.2	0.27	0.23	0.24	0.19	0.19	0.18	0.19	0.19	0.24	0.23	0.18	0.19	
	GO mean: 0.15 NOGO mean: 0.19 GO SD: 0.03 NOGO SD: 0.04 p -value=0.001															

Table 4-10 Average connection weights of the SNN models trained over 36 iterations (18 iterations for class GO and 18 for class NOGO) using different folds of samples. At each iteration, one sample was taken out and the model was trained by the remaining samples. This procedure was repeated for all the samples in both GO and NOGO. The p -value represents that the trained SNN models of GO and NOGO are statistically significant with a high confidence, greater than 99%.

	Number of iterations for training the SNN models of OP groups in GO and NOGO classes																			
GO		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	
		0.08	0.07	0.12	0.07	0.15	0.08	0.1	0.09	0.09	0.12	0.1	0.07	0.06	0.09	0.07	0.05	0.05	0.07	
NG		1	2	3	4	5	6	7	8	9	10	11	12	13	14	15	16	17	18	
		0.17	0.2	0.24	0.19	0.12	0.09	0.8	0.12	0.07	0.18	0.17	0.2	0.2	0.15	0.12	0.9	0.9	0.13	
	GO mean: 0.08 NOGO mean: 0.27 GO SD: 0.03 NOGO SD: 0.28 p -value=0.009																			

Table 4-11 The EEG data classification accuracy (in %) using the LOOCV method. Correctly classified samples are shown on the diagonal of the confusion matrix, shown in bold.

Healthy control subjects (H), MMT subjects (M), Opiate subjects (OP)									
	Classes	Accuracy (average)	Confusion table			Testing samples per class	Precision and recall	F_{Score}	
Session I: GO vs. NO classification	H in Go vs NOGO	90.47	Real	Predicted			21 H samples in GO 21 H samples in NOGO	P=0.95 R=0.86	0.90
					GO	NOGO			
				GO	20	1			
		NOGO	3	18					
	MMT in Go vs NOGO	83.20			GO	NOGO	29 MMT samples in GO 31 MMT samples in NOGO	P=0.79 R=0.85	0.82
				GO	23	6			
				NOGO	4	27			
	OP in GO vs NOGO	83.33			GO	NOGO	18 OP samples in GO 18 OP samples in NOGO	P=0.67 R=1.00	0.8
				GO	12	6			
NOGO				0	18				
Session II: OP, MMT, CO classification in GO	MMT vs H subjects	72.57			MMT	H	21 H samples 29 MMT samples	P=0.69 R=0.8	0.74
				MMT	20	9			
				H	5	16			
	OP vs H subjects	83.33			OP	H	18 OP samples 21 H samples	P=0.68 R=1.0	0.8
				OP	12	6			
				H	0	21			
	MMT vs OP subjects	73.27			MMT	OP	29 MMT samples 18 OP samples	P=0.97 R=0.76	0.85
				MMT	28	1			
				OP	9	9			
Session III: OP, MMT, CO classification in NOGO	MMT vs H subjects	84.79			MMT	H	31 MMT samples 21 H samples	P=0.84 R=0.90	0.87
				MMT	26	5			
				H	3	18			
	OP vs H subjects	92.85			OP	H	21 H samples 18 OP samples	P=1.0 R=0.86	0.92
				OP	18	0			
				H	3	18			
	MMT vs OP subjects	83.87			MMT	OP	31 MMT samples 18 OP samples	P=0.87 R=0.89	0.9
				MMT	27	4			
				OP	2	16			

Table 4-12 The EEG data classification in conventional methods using the LOOCV method. The optimal parameter setting is reported in Appendix C, Tables C-12 and C-13.

Healthy control subjects (H), Opiate subjects (OP), Accuracy is reported in %			
Sessions	Classification	SVM	MLP
Session I: GO vs. NOGO classification	H in GO vs. NOGO	65.00	70.00
	MMT in Go vs. NOGO	63.00	69.00
	OP in GO vs. NOGO	67.00	60.00
Session II: OP, MMT, H classification in GO	MMT vs. H subjects	70.00	76.00
	OP vs. H subjects	68.00	78.00
	MMT vs. OP subjects	67.00	63.00

Session III: OP, MMT, H classification in NOGO	MMT vs. H subjects	63.00	77.00
	Op vs. H subjects	73.00	73.00
	MMT vs. OP subjects	63.00	61.00

4.5 Chapter Summary

In this chapter, I presented the feasibility analysis of the NeuCube SNN architecture using two cognitive STBD. This procedure included: spatial mapping of the STBD into a 3-D SNN model, unsupervised STDP learning, SNN model visualisation for a better understanding of the spatio-temporal interactions between the variables, supervised learning in the deSNN classifier, parameter optimisation, and model validation. Some of the key findings of this chapter are as follows:

- a. An average improvement of the classification accuracy by 20% when compared with some other machine learning methods.
- b. Improved understanding and interpretation of the interactions between the STBD variables, in a 3-D brain-inspired model.
- c. In the case of fMRI data, the findings confirm that the trained spatio-temporal connections in the SNN models are compatible with neuroscience literature, which reported that comprehension of negative sentences is cognitively different from affirmative sentences. Containing negative words, such as “not,” in the middle of a sentence can make it more difficult to comprehend, due to its more complex structure. Therefore, this type of sentence may engage more regions of the brain (Christensen, 2009). More detailed analysis on the connectivity related to the task can be performed by neuroscientists to answer different research questions.

- d. In the case of EEG data, the trained SNN model of H subjects was significantly different from people with a history of opiate dependence. The differences appeared less pronounced in people undertaking MMT compared to those who were current opiate users.

It can be seen from the results that the SNN models outperformed the conventional methods in terms of STBD classification accuracy. In addition to this, interpretability in machine learning is of crucial importance. This allows for understanding the relationship between features and the predicted values. Therefore, the model does not act as a black-box information processing system, but as an interpretable model that demonstrates what interactions between the features have triggered the output. Now the question is: *how the SNN models can be further investigated for knowledge discovery in such dynamic learning patterns that evolve over time?* Since clustering is an approach for the detection of relationships and structure in data, in the next chapter I propose a new spatio-temporal clustering in SNN model to study its incremental learning patterns. This will lead to the model interpretability.

4.6 Contribution

In this chapter, I have made the following contributions:

1. I designed the feasibility study of the NeuCube SNN architecture on fMRI and EEG datasets.
 - a. In the case of fMRI data, I selected the informative fMRI voxels.
 - b. I modelled the fMRI data using two spatial mapping structures.
 - c. I improved the classification accuracy.
2. I conducted statistical analysis of the SNN results.
3. I conducted experiments for parameter optimisation.
4. I published parts of the fMRI study in one conference paper as the leading author and in one journal paper as the corresponding author.
5. I published parts of the EEG study as a leading author in an international journal. This paper is a multidisciplinary research which involves researchers from psychology department at AUT and University of Auckland who have conducted the EEG data recording and pre-processing. From a computational point of view, I designed the research protocol, specified for EEG data, performed the empirical study and analysed the results.

Kasabov, N. K., **Doborjeh, M. G.**, & Doborjeh, Z. G. (2017). Mapping, learning, visualization, classification, and understanding of fMRI data in the NeuCube evolving spatiotemporal data machine of spiking neural networks. *IEEE Transactions on Neural Networks and Learning Systems*, 28(4), 887-899.

Doborjeh, M. G., Wang, G. Y., Kasabov, N. K., Kydd, R., & Russell, B. (2016). A spiking neural network methodology and system for learning and comparative analysis of EEG data from healthy *versus* addiction treated *versus* addiction not treated subjects. *IEEE Transactions on Biomedical Engineering*, 63(9), 1830-1841.

Doborjeh, M. G., Capecci, E., & Kasabov, N. (2014). Classification and segmentation of fMRI spatio-temporal brain data with a NeuCube evolving spiking neural network model. *IEEE Symposium on Evolving and Autonomous Learning Systems (EALS)*, 73-80.

Chapter 5 A New Spatio-temporal Clustering of SNN Patterns

5.1 Introduction

In the previous chapter, I have shown that complex spatio-temporal patterns were adapted in the recurrent SNN models while learning from STBD streams. In this chapter, I propose a new clustering method to interpret such spatio-temporal patterns (dynamic learning behaviour), which carried out meaningful information to the classifier. This clustering approach is a technique for knowledge discovery in SNN architecture, which resulted in a better understanding of the relationship between features and output. This also allowed for detecting the SNN abstraction, which resembles deep-learning methods.

Clustering is considered as a main approach in data mining, pattern recognition, and knowledge discovery. This aims to objectively organise data samples into homogeneous groups, where the data samples within a group are similar. The following classes of clustering methods for static vector-based data are distinguished:

- a) Clustering of vector-based data, where the number of clusters is pre-defined (such as C-means and K-means methods as discussed in Chapter 2, Section 2.5.2).
- b) Clustering of time-series data, where the number of the time points are fixed and the time series are represented as vectors, applying the above methods (such as SOM and Biclustering as discussed in Chapter 2, Section 2.5.2).
- c) Dynamic evolving clustering methods, where a stream of vectors is clustered incrementally without pre-defining the number of clusters.

Category (c) from the above refers to adding the time dimension to clustering configuration which devises dynamic clusters. It is in contrast to instant clustering of an entire data space in categories (a) and (b), as dynamic evolving clustering attempts to create incremental clusters of streaming data which continuously evolve with time. In the extant real-life applications, there is a growing demand for models that can dynamically process the time-dependant data streams. To model such dynamics in data, evolving systems (Angelov, Filev, & Kasabov, 2010) have been proposed to reflect the data changes over time through evolving new models and updating them incrementally.

A known sub-field of evolving systems is the Evolving Cluster Model (ECM) that develops unsupervised evolving clusters on data streams (Bifet, Holmes, Kirkby, & Pfahringer, 2010). At each time, ECM receives one static vector-based sample and finds its best fit cluster. As the data stream progresses over time, ECM may manifest a *cluster fusion* problem which refers to merging two distinct clusters due to new samples filling their intervening gap. The overlapping clusters were handled by the concept of fuzzy rules as developed in an evolving fuzzy system (Lughofer, Bouchot, & Shaker, 2011). Lughofer (Lughofer E., 2012) proposed a new split-and-merge method to overcome the fusion problem in conventional ECM. This method detects the temporal changes by an incremental learning and the entropy suggested a perfect estimation of border lines.

Many other dynamic evolving clustering methods have been proposed in the literature including: ESOM (Deng & Kasabov, 2000), DENFIS (Kasabov & Song, 2002), EFuNN (Kasabov, 1998) and evolving framework (Aggarwal, 2003). These methods have successfully discovered the temporal changes in data streams. However, when dealing with STBD where both *time* and *space* information need to be preserved and learnt together, the

current evolving clustering methods do not integrate the spatial and temporal components. They often transpose the temporal information into static vectors, where the time is hidden, and no spatio-temporal interaction can be learnt. In some approaches, such as SOMs, the spatial information is topologically preserved in the map, but not considered in the computation.

Therefore, the current methods need to significantly advance the similarity/distance measures with respect to the spatio-temporal characteristics of STBD. While dynamic evolving clustering of static (vector-based) data has been well explored, dynamic evolving spatio-temporal clustering has been little researched if at all, especially when patterns of changes (events) in the data across *space* and *time* have to be learnt. In principle, spatio-temporal clustering methods can be classified into the following groups:

- 1) Two-tier clustering of the data: according to the spatial information, and separately according to the temporal information (Aggarwal, 2003). In such a way, the integrated spatio-temporal similarity cannot be measured.
- 2) Integrated clustering of both spatial and temporal information (the case in this chapter).
- 3) In case of both static and spatio-temporal data are available, clustering of the integrated static and temporal information first, and then the spatio-temporal information. This can be applied for personalised modelling (will be presented in Chapter 7).

Furthermore, the current evolving clustering methods are proposed for investigating the relationship in the *raw data* space through incremental learning, but without investigating the *model* learning patterns itself. Therefore, they act as black-box information processing systems that solve the problems without discovering the meaningful interactive patterns in

the models that have triggered the outputs. Knowledge discovery in deep-learning patterns generated during the learning time, in an unsupervised mode, from spatio-temporal data streams is of crucial importance for the interpretability. To address this, I suggest dynamic spatio-temporal clustering of the incremental learning patterns is a promising technique to detect meaningful interactions between the features and outputs. Therefore, in this chapter I propose a new model-based clustering method that adds two new features to the current evolving clustering approaches:

- Measuring the *integrated spatio-temporal* similarity in a brain-inspired model, during incremental learning with STBD streams.
- Knowledge discovery in dynamic learning behaviour in recurrent SNN models.

Compared to the current evolving clustering methods that perform a direct clustering of data, the proposed approach in this chapter is for clustering of a model that is being created to learn and capture the essential characteristics of interest from the data. This chapter is structured as follows: Section 5.2 introduces a new spatio-temporal clustering of SNN evolutionary learning patterns. Section 5.3 represents a validity measure for this clustering configuration.

5.2 Clustering of SNN Evolutionary Patterns

This research contributes to the NeuCube with a new dynamic spatio-temporal clustering method in SNN models while learning from streaming data. The proposed method contains procedures for encoding the spatio-temporal data into spikes and for creating dynamic clusters of spiking neurons in a 3-D SNN model, both in space and time. In contrast to conventional clustering techniques, which are based on either spatial or temporal components

(Kohonen, 1998; Kasabov, 1998; Deboeck & Kohonen, 2013; Kasabov, 2007), this method is based on integrated spatial and temporal measures. The main objective is knowledge discovery in the SNN model by detecting similar spatio-temporal patterns of changes (while streaming data), which are dynamically adapted with respect to the interactions between input neurons. This method results in a better interpretation of interactions between features. This is based on unsupervised learning in the SNN architecture as shown graphically in Figure 5-1.

5.2.1 Spatio-temporal Clustering Schema

In this method, the cluster centres are predefined by the spatial locations of the data sources used as input variables (e.g. EEG channels). During unsupervised STDP learning in the SNN model, spikes are transmitted between neurons that cause modifications of the connection weights. The more spikes are transmitted between two neurons i and j , the stronger the connection (w_{ij}) becomes between them, where w_{ij} denotes the weight specifying the connection strength. During the clustering procedure, each neuron can be assigned to different clusters (input variables) with different membership values. This membership is defined with respect to the number of spikes that a neuron receives from each of the inputs. A neuron belongs to a cluster that has received the most spikes from this input when compared with other inputs.

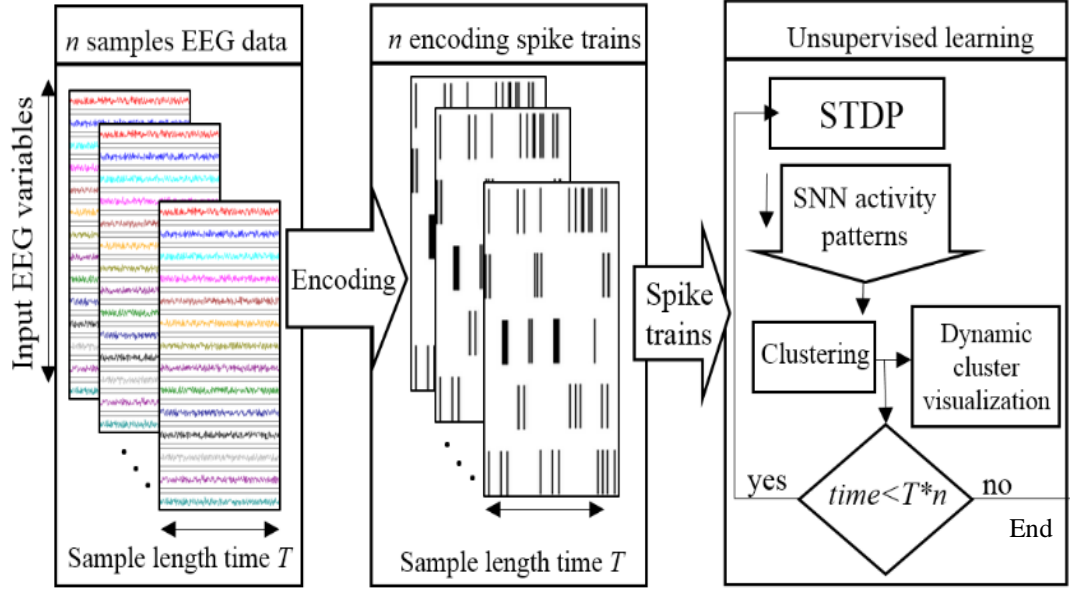


Figure 5-1 A block diagram of the dynamic spatio-temporal clustering method using SNN.

In an SNN model with N neurons, the input neurons are allocated to the cluster centres and labelled by the input variables. The rest of the neuros are unlabelled. The goal is to assign the cluster labels to the unlabelled neurons in the SNN model. To this end, I have used the concept of spreading activation from (Zhou, Bousquet, Lal, & Weston, 2004) and performed as follows:

The neurons in the SNN model are indexed from 1 to N ascendingly according to the order of their (x, y, z) coordinates. The input neurons are marked as the information source and defined using an $N \times v$ matrix F_{src} in which $F_{src}(i, j) = 1$ if neuron i is the input neuron for variable j ; otherwise $F_{src}(i, j) = 0$, where N is the number of neurons in the SNN model and v is the number of input variables (e.g. EEG channels). While streaming sets of spatio-temporal data, each neuron in the SNN model receives a different ratio of information from different input variables. The ratio of the received information can be computed through the following steps:

- 1 An affinity $N \times N$ matrix A is defined on the SNN model that displays the sum of the spikes that are exchanged between neurons i and j ($i = 1, \dots, N$ and $j = 1, \dots, N$) via connection w_{ij} . The amount of information that are exchanged between the neurons is computed as follows:

$$\begin{cases} A'_{ij} = A_{ij} + A_{ji} & i \neq j \\ A'_{ij} = 0 & i = j \end{cases} \quad (5-1)$$

where the element A_{ij} displays the number of spikes transmitted from neuron i to j , while A_{ji} indicates the number of spikes transmitted from neuron j to i . Since a neuron does not send a spike to itself, the entry for A_{ij} is 0 when $i = j$.

$$T_i = \sum_{j=1}^N A'_{ij} \quad i = 1 \text{ to } N \quad (5-2)$$

Thus, T_i is the sum of the elements in the i^{th} row of matrix A' . Then the affinity matrix A is normalised using $S = D A D$, where D is an $N \times N$ diagonal matrix, where its (i, i) -element is defined by $D_{ii} = \left(\frac{1}{\sqrt{T_i}}\right)$ and S is an $N \times N$ normalised matrix that encodes the spike propagation in the SNN model.

- 2 Iterate $F(t + 1) = \alpha S F(t) + (1 - \alpha) F_{src}$ until it converges, where α parameter is in the $(0, 1)$ range. The limit of $F(t)$ is denoted by $F^* = \lim_{t \rightarrow \infty} F(t) = (I - \alpha S)^{-1} F_{src}$, where I is an identity matrix. The output F^* has N rows (representing all neurons in the SNN model) and v columns (representing the input variables).

The element F^*_{ij} represents the relative information amount that a neuron i in the SNN model receives from an input neuron j . By computing the $\arg \max_{j=1,\dots,v} F^*_{ij}$, the neurons in the SNN model are classified into different input variables. This results in clustering the neurons into v inputs. This procedure can be better understood as follows:

In an SNN model, the input information is propagated from input neurons (sources of information) to other neurons. At the beginning of the STDP learning in the SNN model, only the input neurons (centroids of the clusters) have received the information ($F^* = F_{src}$). When the learning procedure increments with sets of spatio-temporal streams over time, the other neurons will also receive a ratio of information from one or more input neurons. Therefore, neurons are being clustered with respect to the amount of information that receive from each of the inputs. In such a way, neural clusters are created and evolved over time in an incremental way during STDP learning.

The dynamic visualisation of the clusters captures the time in which a cluster is created, and it demonstrates how this cluster is changed over time. Such created clusters are 3-dimensional and have different shapes. The size and the formation-time of a cluster represents the importance of the cluster centre in the trained SNN model and therefore, the importance of the corresponding input variable in the STBD. The proposed dynamic spatio-temporal clustering algorithm is given in Table 5-1.

Table 5-1 The dynamic spatio-temporal clustering algorithm, called at each time point t of the STDP learning.

Algorithm 1: Dynamic Spatio-temporal Clustering

Input: Input spike data sp , number of neurons in the SNN model N , number of input variables v , connection weights $w[N, N]$, and parameter α .

Output: A vector of labelled neurons k

- 1: Procedure
 - 2: $[L \ V] = \text{size}(sp)$
 - 3: $F_{src} \in \mathbf{R}^{N \times v}$, $A \in \mathbf{R}^{N \times N}$
-

```

4: for t=1: (L*V) do
5:   Update  $w$  with STDP
6:    $S = D A D$ 
7:    $F^* = (I - \alpha S)^{-1} F_{src}$ 
8:    $k = \arg \max_{j=1,\dots,v} F^*_{ij}$ 
9:   Visualisation of the clusters
10: end for
11: End of procedure

```

5.3 Clustering Validation

Clustering validation has been considered as an essential approach to evaluate the success of clustering configurations (Maulik & Bandyopadhyay, 2002). In general, two main measurement categories are known for clustering validation:

- Internal measures are used to indicate the goodness of a clustering structure without external information, such as class label association (Tan, 2006) (which is the case in this research).
- External measures are used to describe the agreement between two partitions where the first partition is a priori known clustering structure, while the second resulted from the clustering algorithm. A known external measure is the entropy, which assesses the purity of clusters according to given class labels (Wu, Xiong, & Chen, 2009).

In many applications, external information such as class labels are not always accessible; thus, entropy cannot be calculated. Similarly, in the SNN models, when clusters are formed through an unsupervised learning, it is crucial to validate the goodness of clusters. Otherwise, the clusters' contribution would be obscure. The objective in this clustering is to maximize the connection weights between neurons in a cluster and minimise the connection between neurons of the neighbour clusters. To this end, I employed the Silhouette coefficient validity method as an internal measurement technique.

5.3.1 Silhouette Coefficient

The validity measures are usually based on the “cohesion and separation” concept (Tan, 2006; Zhao & Karypis, 2002), which is graphically shown in Figure 5-2. Cohesion measures how similar objects are within a cluster, whereas separation measures how distinct or well-separated a cluster is from other clusters. The objective in valid clustering is maximising the cohesion measure while minimising the separation measure.

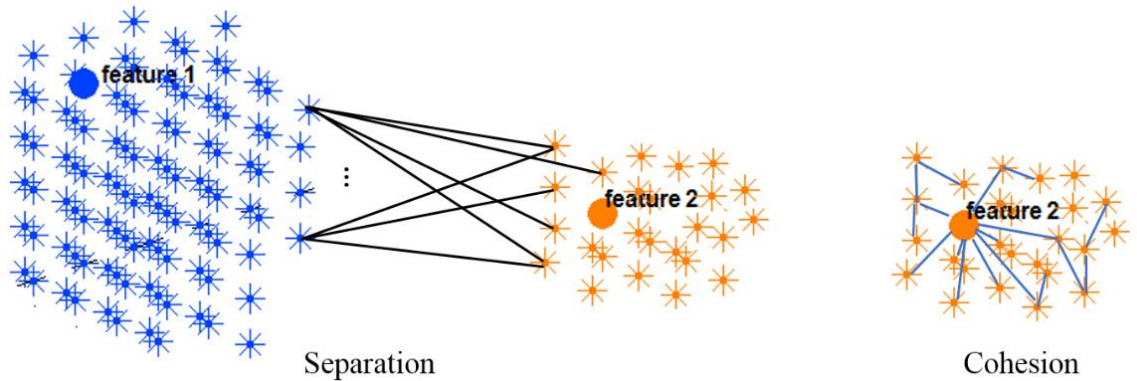


Figure 5-2 Cohesion and separation of two neighbour clusters in an SNN model, where cluster centres are denoted by features 1 and 2.

In an SNN model, for each neuron i within a cluster, $x(i)$ is the cohesion of i to all other neurons in the same cluster. It shows how well i is assigned to its own cluster, so that; a larger value of $x(i)$ refers to a better assignment. In contrast, $y(i)$ is the separation between neuron i in the current cluster and other neurons from a neighbour cluster.

The Silhouette method (Rousseeuw, 1987) validates the homogeneity within clusters through measuring how similar an object is to its own cluster (cohesion) compared to other clusters (separation). The Silhouette value of neuron i is defined as follows:

$$s(i) = \frac{x(i) - y(i)}{\max\{y(i), x(i)\}} \quad (5-3)$$

The Silhouette value is in the range of $-1 \leq s(i) \leq 1$, where a value closer to 1 implies that the object is well-matched to its own cluster. If many objects have a high Silhouette value, then the clustering configuration is suitable. Figure 5-3 illustrates the Silhouette method exemplified using two adjacent clusters in an SNN model.

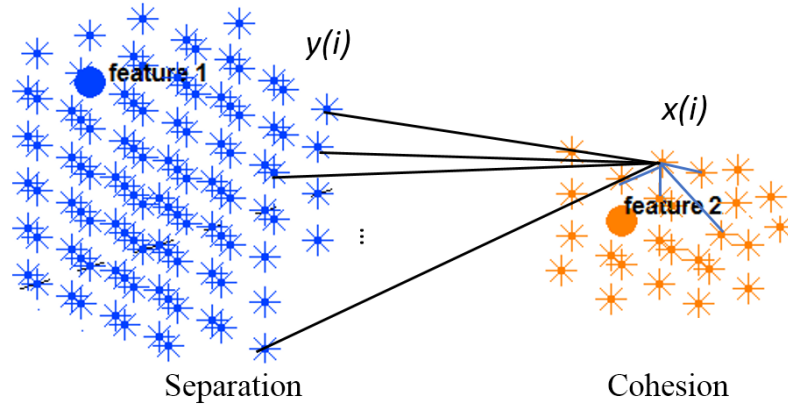


Figure 5-3 Silhouette measure exemplified on two clusters.

Within a cluster, where neuron i is connected to M neurons, the average of the connection weights between i and all the M neurons defines the cohesion of neuron i to its cluster. This cohesion is also weighted by F^*_{iv} , which is the membership value of neuron i to the cluster centre v . Therefore, I defined the cohesion $x(i)$ as follows:

$$x(i) = \frac{\sum_{m=1}^M w_{im}}{M} \times F^*_{iv} \quad (5-4)$$

In contrast, $y(i)$ is the average separation between neuron i and K connected neurons from G neighbour clusters as follows:

$$y(i) = \frac{\sum_{g=1, \dots, G} \frac{\sum_{k=1}^K w_{ik}}{K} \times F_{iv}^*}{G} \quad (5-5)$$

where F_{iv} is the membership value of neuron i to the neighbour cluster centre v . To measure the Silhouette for the SNN clusters, these modified $x(i)$ and $y(i)$ are used in Equation (5-3).

5.4 Chapter Summary

While streaming STBD samples into a 3-D SNN model, spatio-temporal clusters were created and modified in a continuous, incremental way. In such a way, the spatio-temporal relationships of changes in the variables were learnt in the model and the model's spiking activity patterns were incrementally clustered. This method captures significant information about STBD as it records the exact time in which a cluster was formed, and it reveals how this cluster's shape was changed over time. The cluster *size* and the *time* of creation represent the importance of the STBD input variables at different time t of the learning process, providing insights into the input data structures and the SNN learning process. Assessment of the dynamic patterns of the clusters contributes to identify the importance and the involvement of neural clusters in the SNN model. This approach allows for interpreting the hidden learning patterns in the SNN models, that is a significant contribution to machine learning and AI.

The proposed clustering method is in contrast to the current evolving clustering methods (Song & Kasabov, 2001), where the number of clusters are not pre-defined, but evolved with respect to the homogeneity in the *raw data* space when a new sample vector comes. Compared to the extant evolving clustering methods that perform a direct clustering of data,

the proposed approach in this chapter is for clustering of a *model* that is being created to learn and capture the essential characteristics of interest from the data.

In the proposed spatio-temporal clustering method, the centres of clusters are defined in advance (which are the input data variables). During unsupervised STDP learning, the spatio-temporal patterns in the *model* (neural activity in the SNN architecture) are clustered. The method was based on the following scheme: (1) dynamic processes, e.g. brain activity patterns, (2) spatio-temporal data streaming, (3) 3-D SNN models, (4) dynamic spatio-temporal clustering during unsupervised learning, and (5) updating the clusters on new data.

The next chapter demonstrates a feasibility analysis of the proposed clustering method on a real-life case study of EEG data.

5.5 Contribution

In this chapter, I have made the following original contributions:

1. Proposal of an original method for dynamic spatio-temporal clustering of learning patterns in SNN model. This resulted in an interpretation of interactions between STBD variables. It contributes to knowledge discovery in SNN architectures.
2. Visualisation of the dynamic SNN patterns over time.
3. A method for validity measurement of the clustering configuration.
4. I published parts of this research as a leading author in the following journal paper:

Doborjeh, M. G., Kasabov, N., & Doborjeh, Z. G. (2018). Evolving, dynamic clustering of spatio/spectro-temporal data in 3D spiking neural network models and a case study on EEG data. *Evolving Systems*, 9(3), 195-211.

Chapter 6 Feasibility Study of Spatio-temporal Clustering

6.1 Introduction

This chapter applies the proposed clustering method (introduced in Chapter 5) to a case study of EEG data. It illustrates the differences between the patterns of the clusters created in the SNN models across individuals when they performed different cognitive tasks. The proposed clustering is performed through the following steps:

- Step1: STBD preparation.
- Step 2: STBD encoding into spike trains.
- Step 3: Mapping the STBD to 3-D SNN model.
- Step 4: Unsupervised learning using STDP and simultaneously clustering the SNN learning patterns (neural connectivity).

6.2 Clustering Patterns in SNN Models

This section illustrates how the proposed spatio-temporal clustering method can be applied to a case study of EEG data. At step 1, the same EEG data that has been used in Chapter 4, is selected here again for the case study in this chapter. The description of EEG data acquisition was presented in Chapter 4. Six EEG sample files were defined, each containing EEG data captured from one subject group (MMT/ OP/ H) per cognitive task (GO *versus* NOGO). The organisation of the data is presented in Table 6-1. At step 2 and 3, each sample file was separately encoded into spike trains using TBR method and then spatially mapped to an SNN model using the Talairach atlas template as described in (Koessler, et al., 2009). The EEG mapping in the Talairach space is presented in Appendix A. At step 4, the learning process of each EEG sample file was started by entering the first EEG time point to train the SNN model and it was finished after entering the final EEG time point. Simultaneously, spatio-temporal clusters were created with respect to spiking activity in the SNN model.

Table 6-1 EEG data samples used for dynamic clustering to study the activity patterns of H, MMT, and OP subjects in a GO/NOGO task.

EEG data sample files of different subject groups in GO <i>versus</i> NOGO		
Task trials	Samples per class	EEG sample size
GO Trials	21 H Subjects	75 EEG time points * 26 channels * 21 samples
	18 OP subjects	75 EEG time points * 26 channels * 18 samples
	29 MMT subjects	75 EEG time points * 26 channels * 29 samples
NOGO Trials	21 H Subjects	75 EEG time points * 26 channels * 21 samples
	18 OP subjects	75 EEG time points * 26 channels * 18 samples
	31 MMT subjects	75 EEG time points * 26 channels * 31 samples

6.2.1 SNN Clustering during Unsupervised Learning

The spatio-temporal clusters were formed and updated with every new input EEG time point entered, frame by frame. This process can be traced and analysed in terms of:

- The order in which input EEG channels formed the clusters, related to the order of activity of the corresponding areas in an SNN model.
- The evolution of the clusters related to the importance of the clusters positioned in different areas of an SNN model.

Figure 6-1 to Figure 6-6 (see the first frame at top, left) show how the input neurons of the SNN model were allocated to the respective EEG channels for transferring the spike trains into the model. They also demonstrate the evolution of the clusters for 7 selected time points during unsupervised learning in the SNN models of EEG data from different groups of subjects in GO/NOGO task. Cluster creation started from predefined centroids (EEG channels as the source of information) and were adapted after every input EEG time point was entered into the SNN model. The reason that I have chosen different time frames in the visualisation was the time differences in cluster creation across the subject groups with respect to their EEG data. Once new clusters were created during unsupervised STDP learning, a new figure was captured to display the step-wise changes in the cluster evolution.

In Figure 6-1, since there were 21 healthy subjects and 75 EEG time points captured from every subject, the last time point of the training data was $21 \times 75 = 1575$. These results show that when an SNN model was trained with EEG data of H group in GO task, the first created clusters correspond to Fz and FCz channels after entering the 8th EEG time point to the learning process. Those neurons that were clustered by Fz and FCz channels have received the most of the spikes (received a high ratio of information as discussed in 5.2.1) from these corresponding channels (information sources).

Figure 6-2 and Figure 6-3 represent the dynamic clustering of the EEG data related to MMT and OP subjects respectively in GO trials. In the MMT group, the first clusters were created by FP1 and FP2 channels at the 11th time point. In case of OP subjects, the first cluster was created by FP2 at the 2nd EEG time point.

Figure 6-4, Figure 6-5 and Figure 6-6 are related to the NOGO trials. In the H group, the first created cluster was related to the FC3 channel at the 2nd EEG time point. It represents a case where a high ratio of information has been propagated (spikes) into the SNN model *via* the FC3 channel at this time. Therefore, the neurons located around this channel were clustered faster than the other neurons in the SNN model. However, the first clusters for MMT and OP groups were generated later than the clusters in the H group and they belonged to C3 and F8 at the 4th and 9th EEG time points, respectively. These results show that in the MMT and OP groups, slower response was observed from frontal regions and, consequently, a smaller number of spikes entered the SNN model compared to the H group.

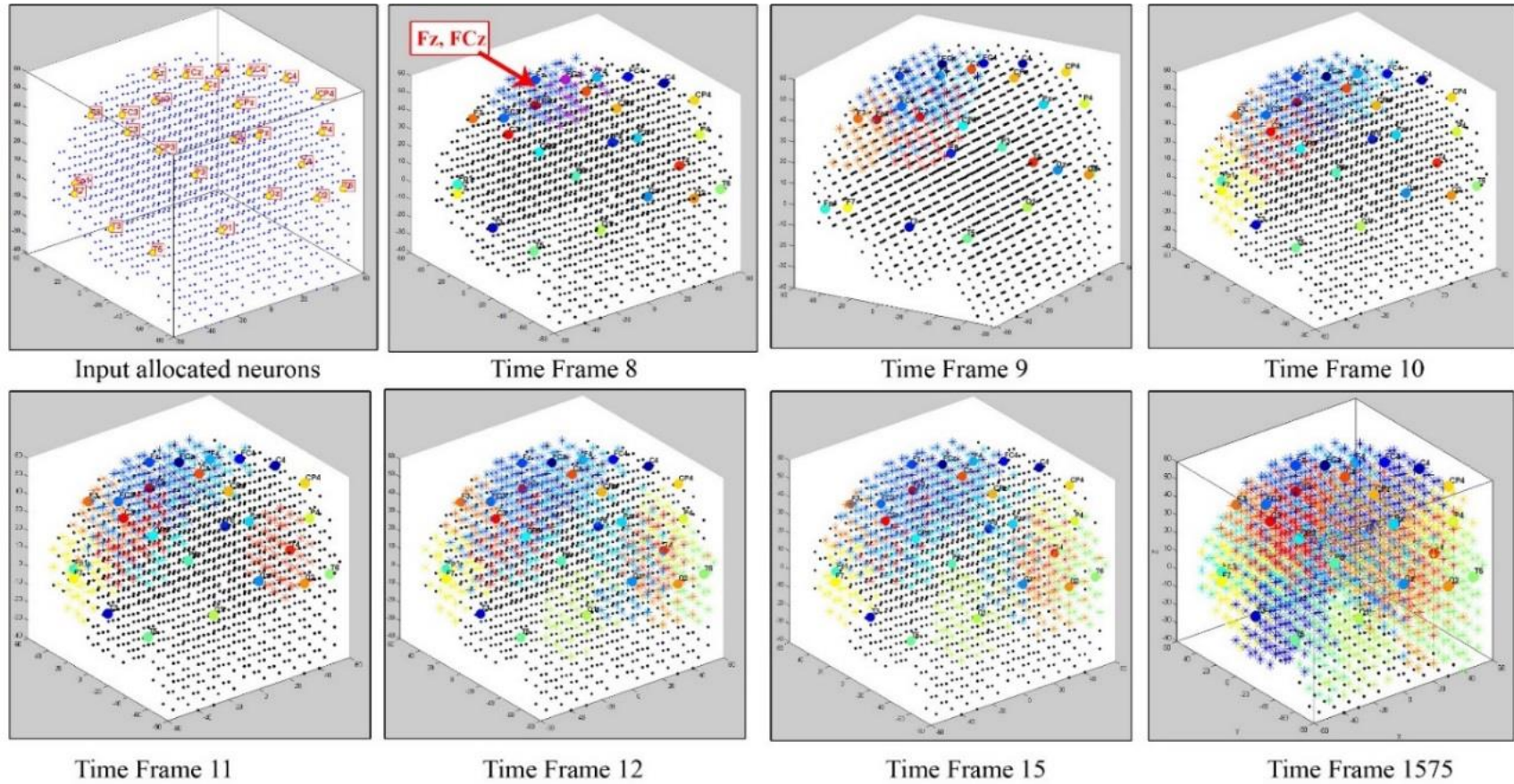


Figure 6-1 A step-wise visualisation of the dynamic cluster evolution corresponding to the 26 EEG channels of 21 healthy subjects in a GO task during unsupervised learning in an SNN model. The total number of time frames is $21 \times 75 = 1575$. The first two clusters are created at the 8th time point of the EEG data that are associated with Fz and FCz channels.

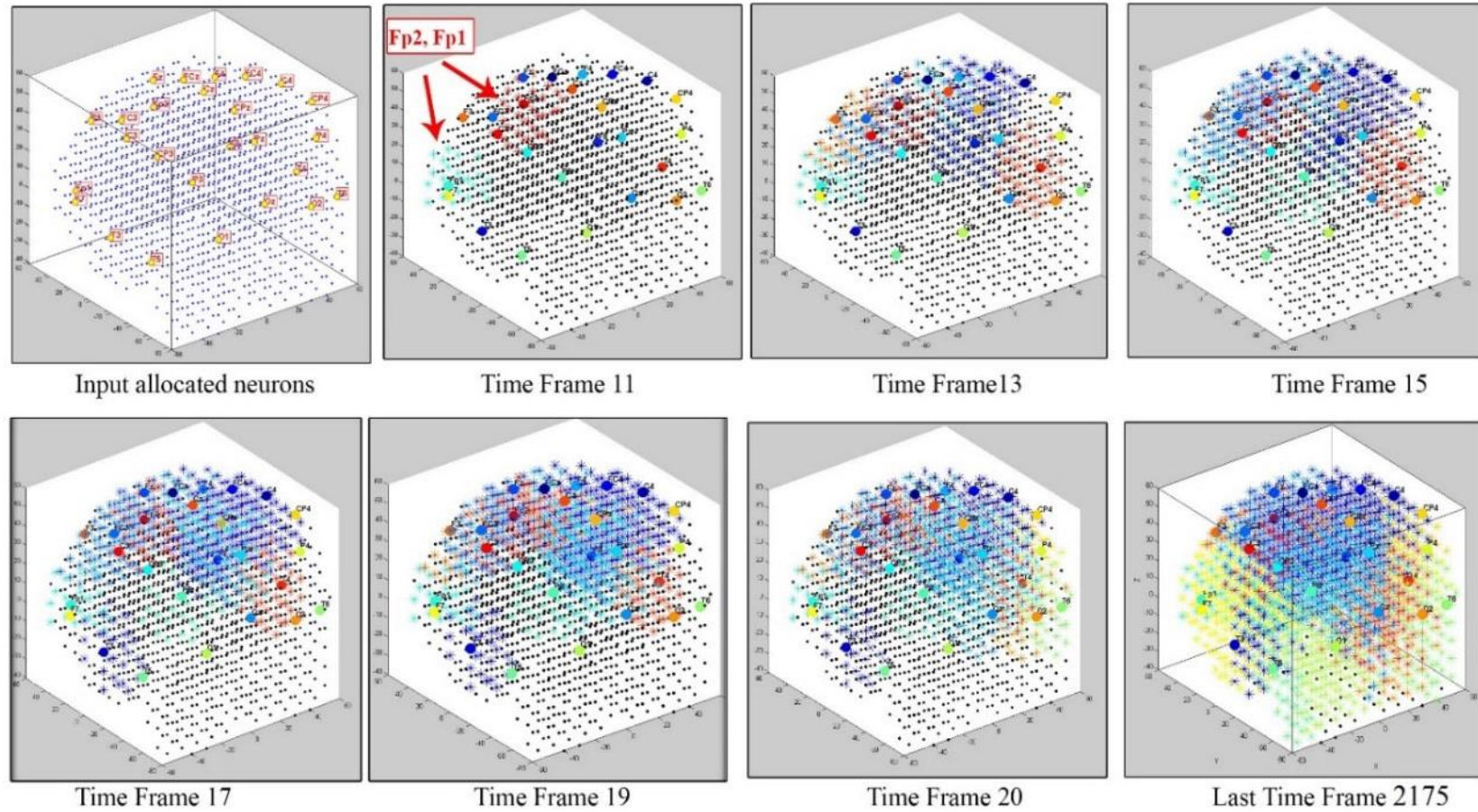


Figure 6-2 A step-wise visualisation of the dynamic cluster evolution corresponding to the 26 EEG channels of 29 MMT subjects in a GO task. The total number of time frames is $29 \times 75 = 2175$. The first two clusters are created at the 11th time point of the EEG data that are associated with Fp2 and Fp1 channels.

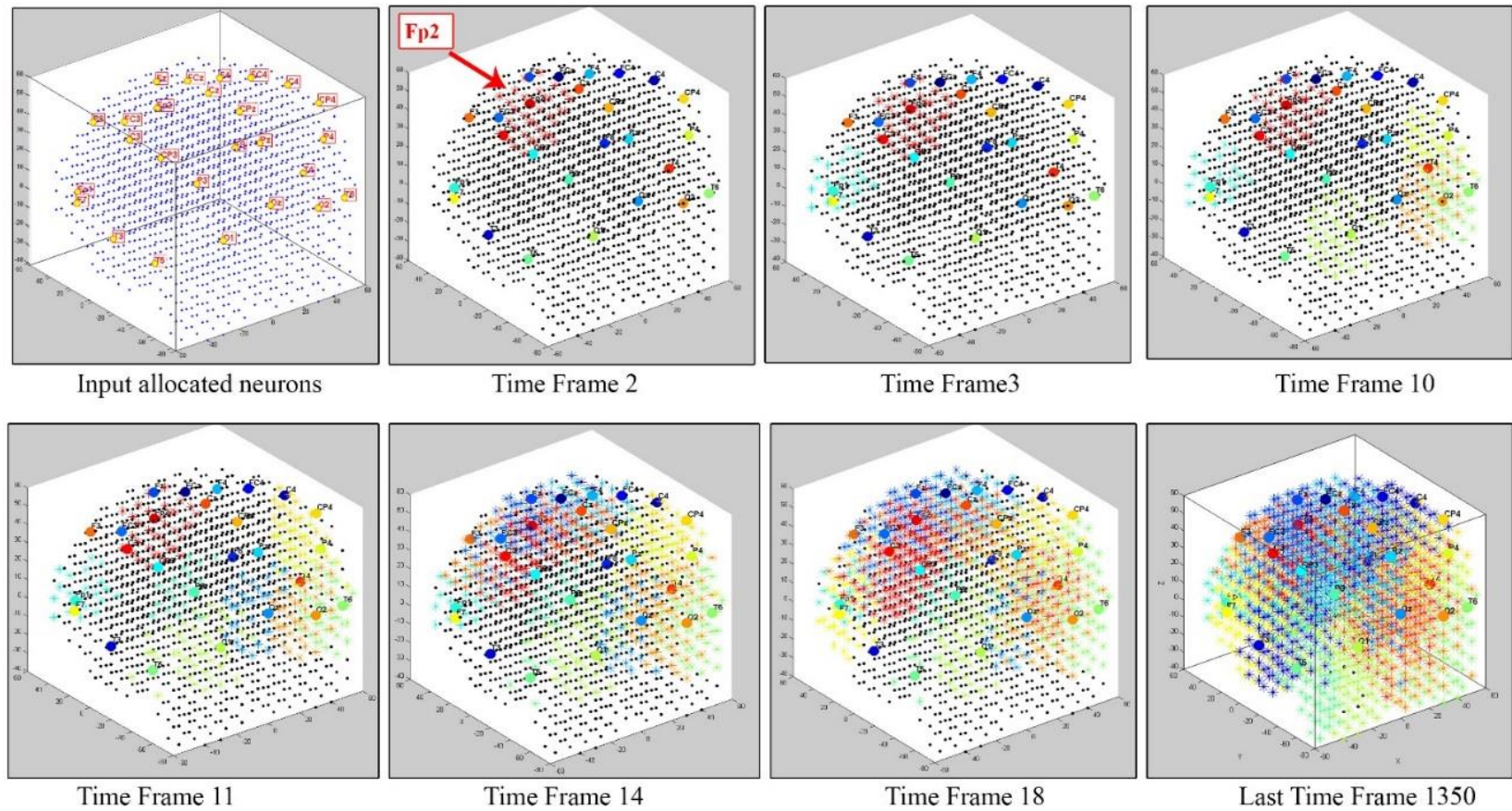


Figure 6-3 A step-wise visualisation of the dynamic cluster evolution corresponding to the 26 EEG channels of 18 OP subjects in a GO task. The total number of time frames is $18 \times 75 = 1350$. The first cluster is created at the 2nd time point of the EEG that is associated with the Fp2 channel.

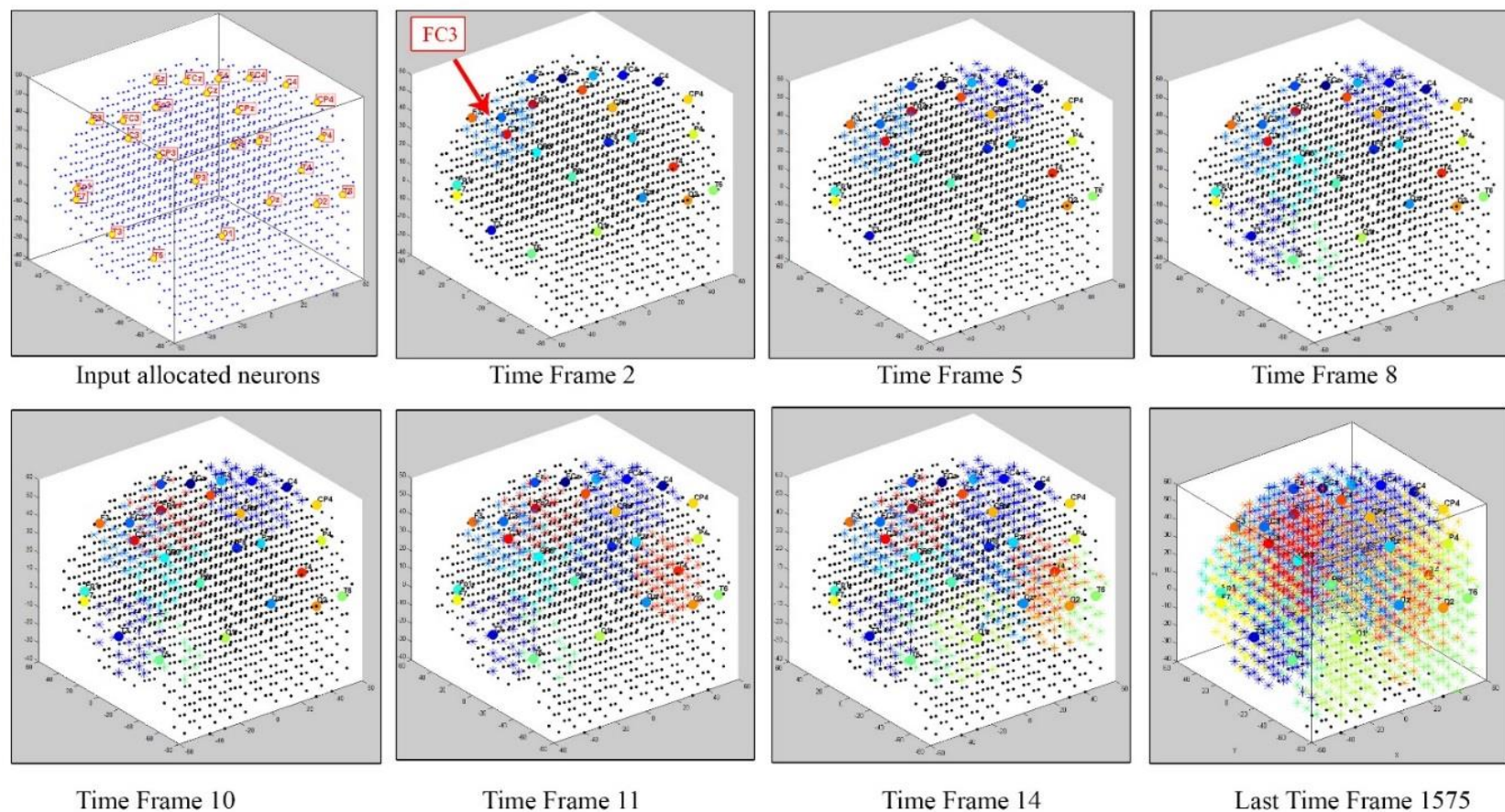


Figure 6-4 A step-wise visualisation of the dynamic cluster evolution corresponding to the 26 EEG channels of 21 healthy subjects in a NOGO task. The total number of time frames is $21 \times 75 = 1575$. The first cluster is created at the 2nd time point of the EEG data that is associated with the FC3 channel.

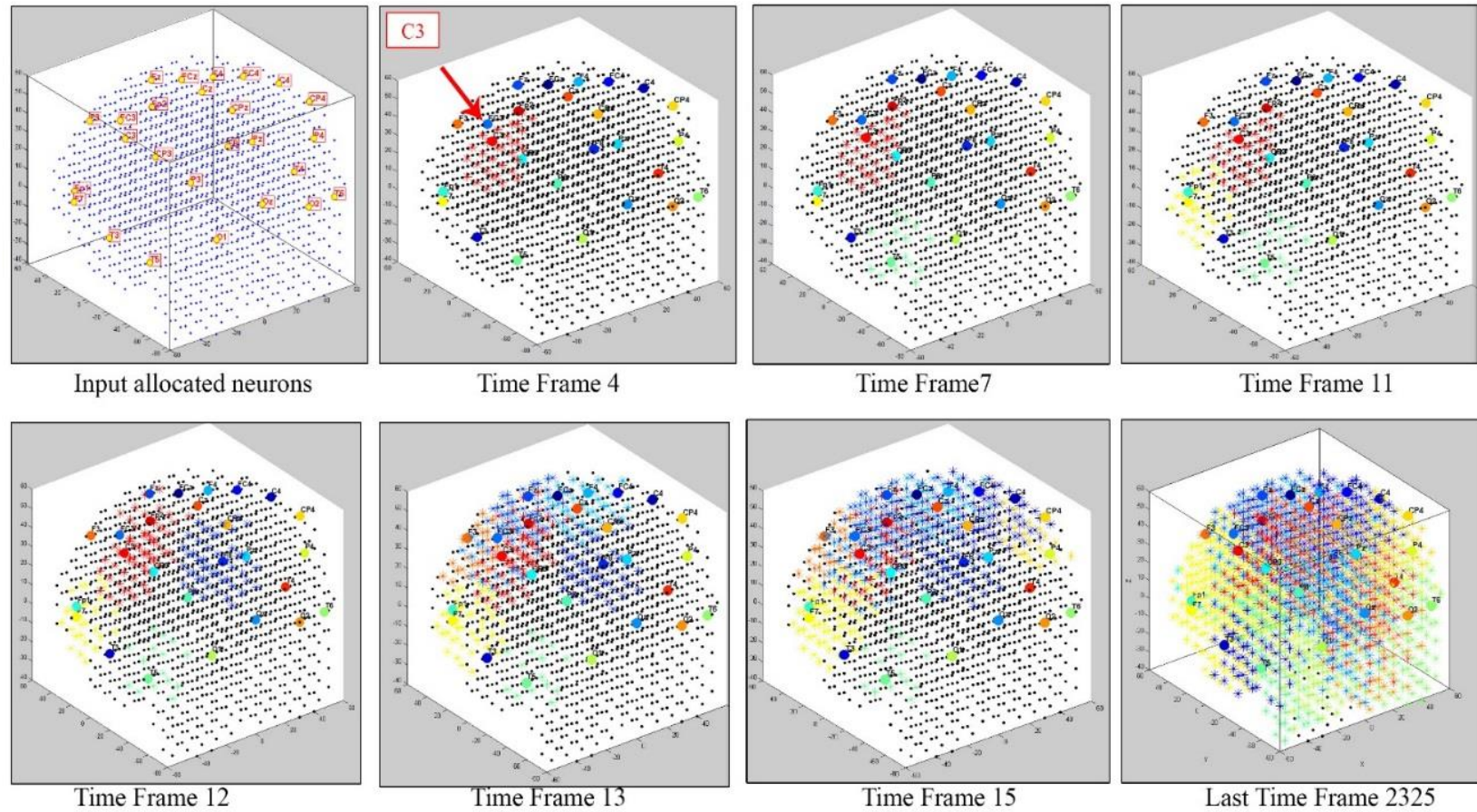


Figure 6-5 A step-wise visualisation of the dynamic cluster evolution corresponding to the 26 EEG channels of 31 MMT subjects in a NOGO task. The total number of time frames is $31 \times 75 = 2325$. The first cluster is created at the 4th time point of the EEG data that is associated with the C3 channel.

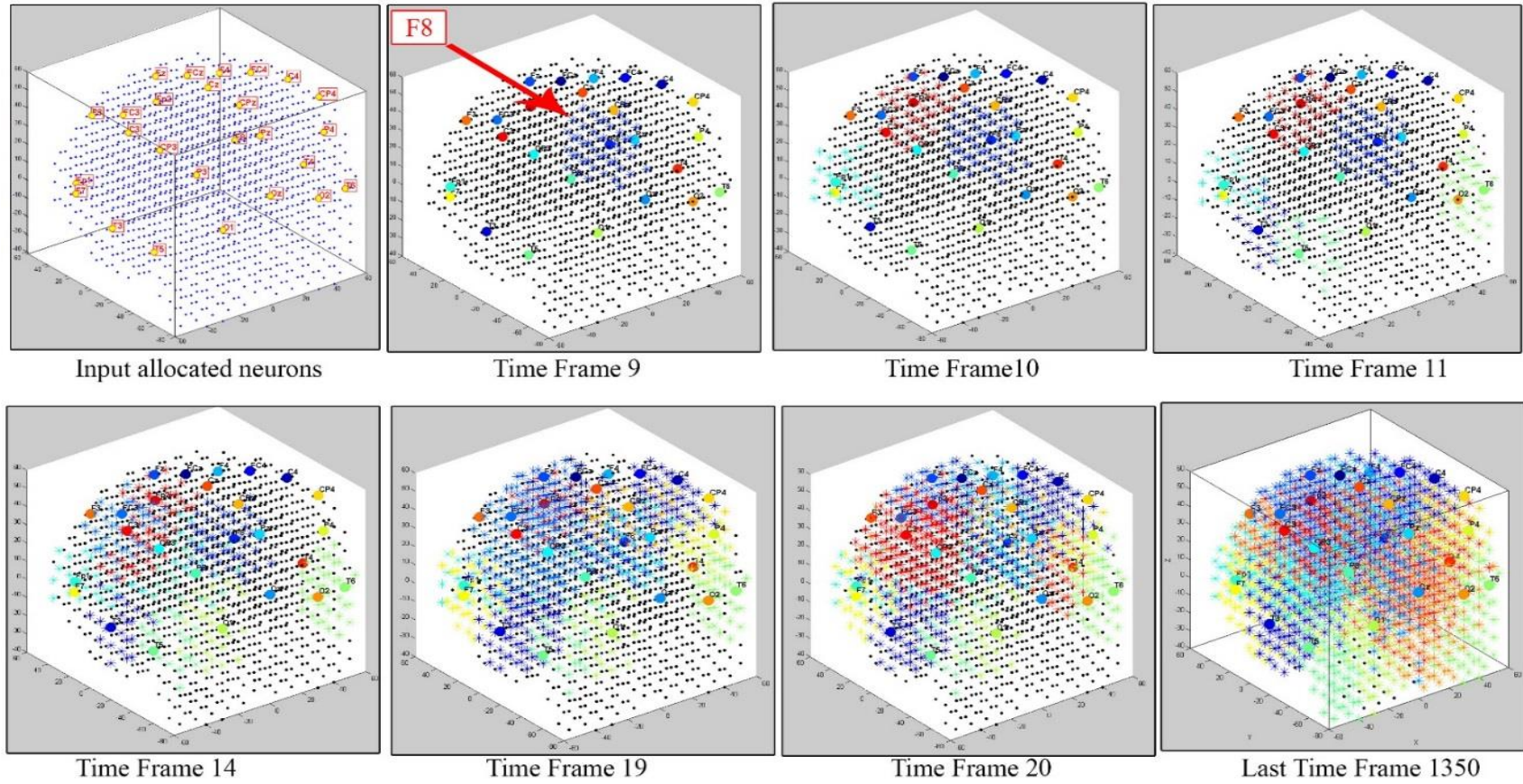


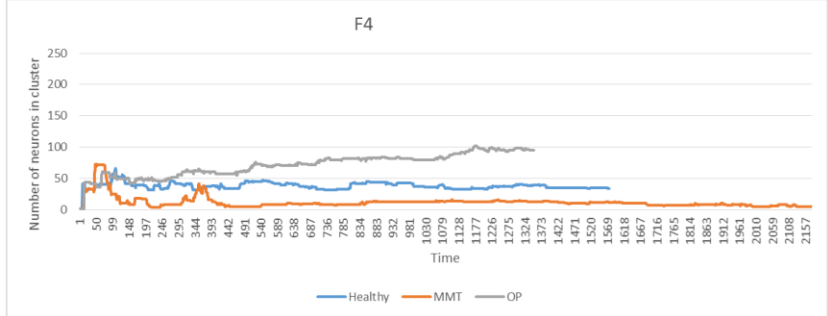
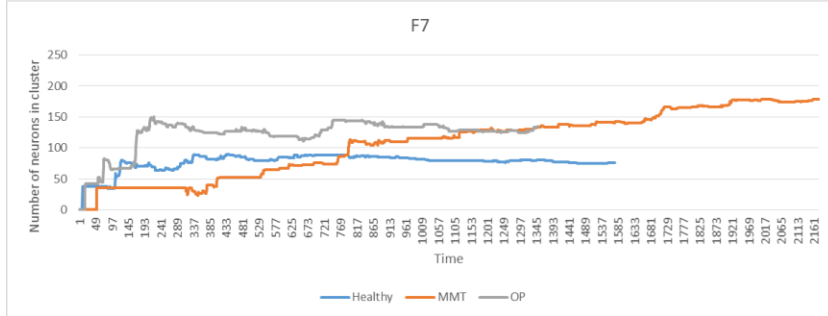
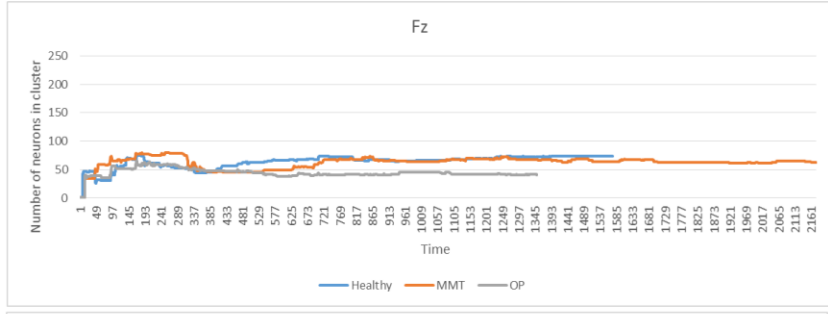
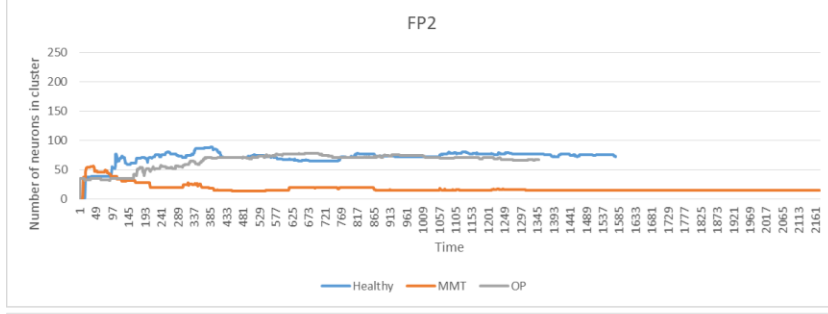
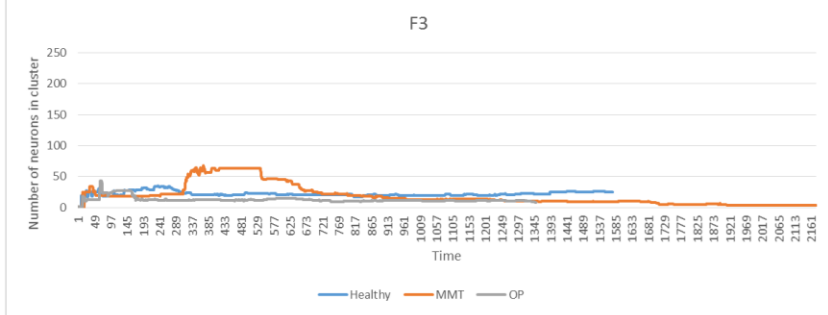
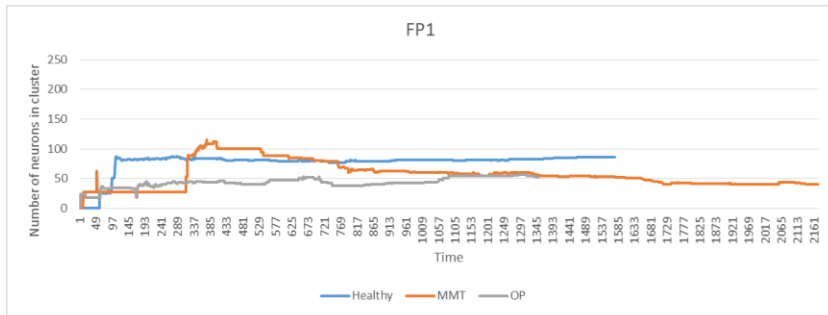
Figure 6-6 A step-wise visualisation of the dynamical cluster evolution corresponding to the 26 EEG channels of 18 OP subjects in a NOGO task. The total number of time frames is $18 \times 75 = 1350$. The first cluster is in the 9th time point of the EEG data that is associated with the F8 channel.

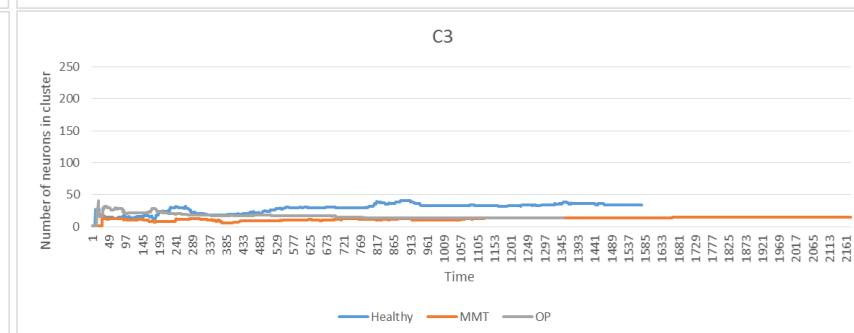
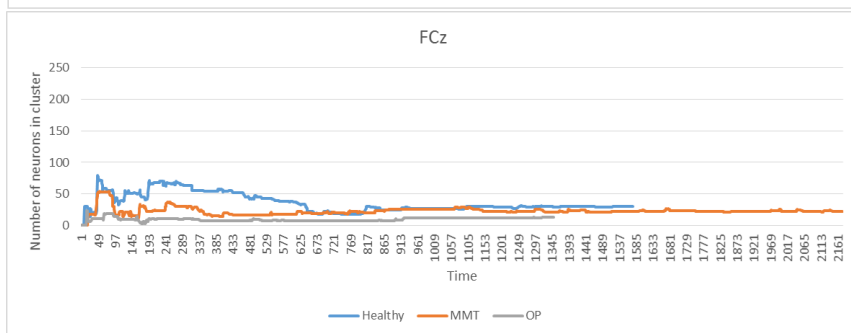
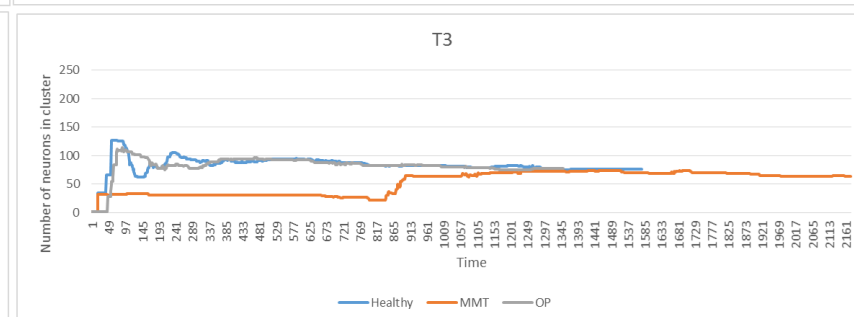
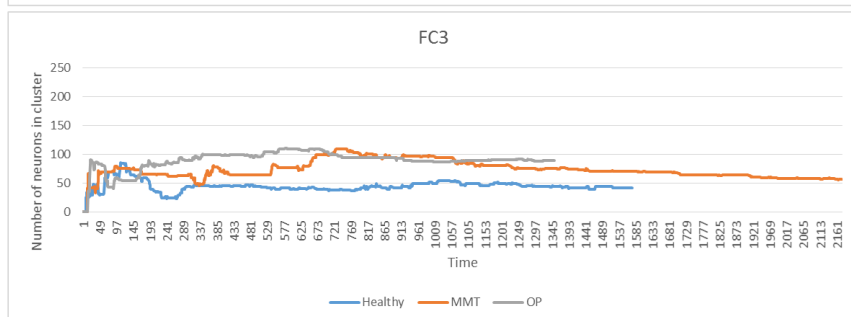
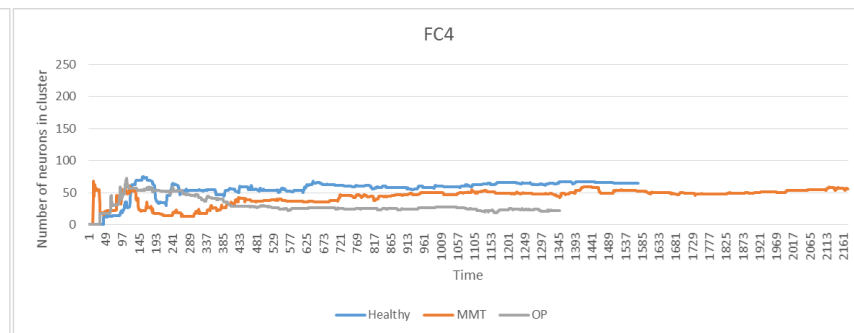
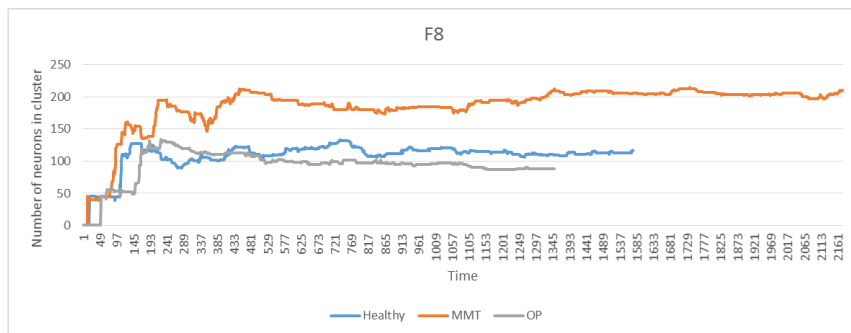
6.2.2 Clustering Evolutionary Patterns

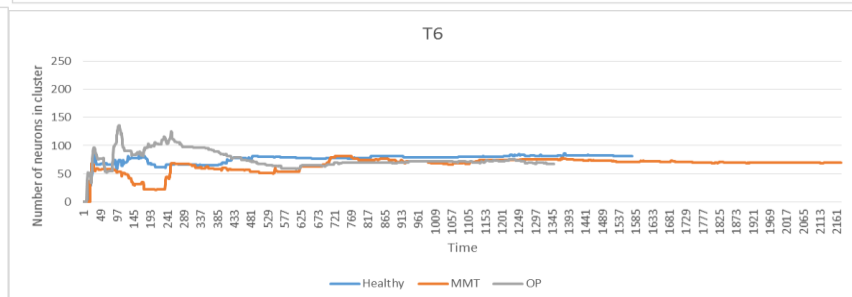
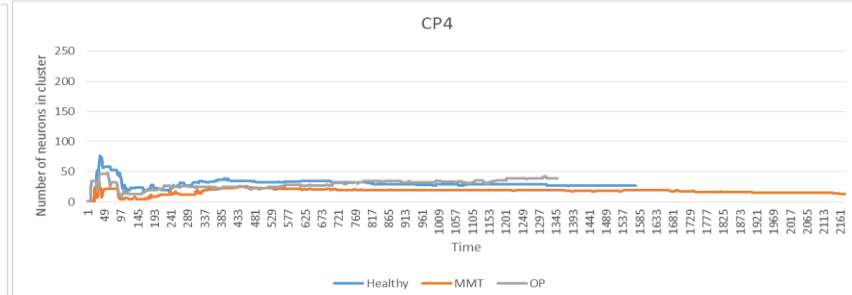
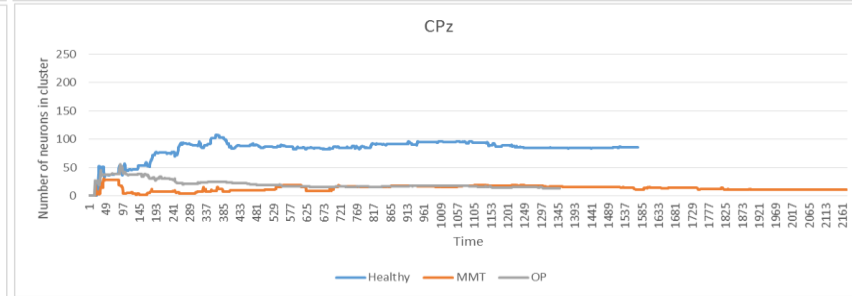
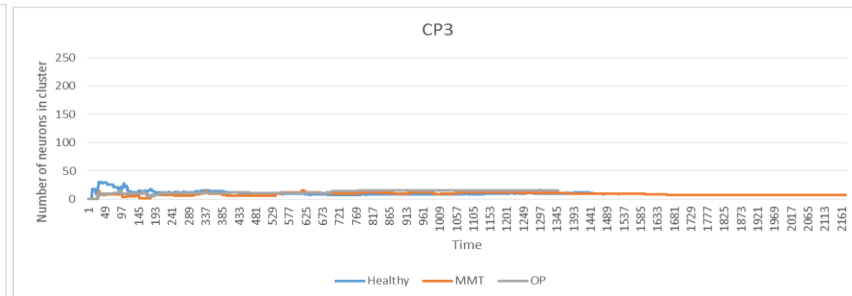
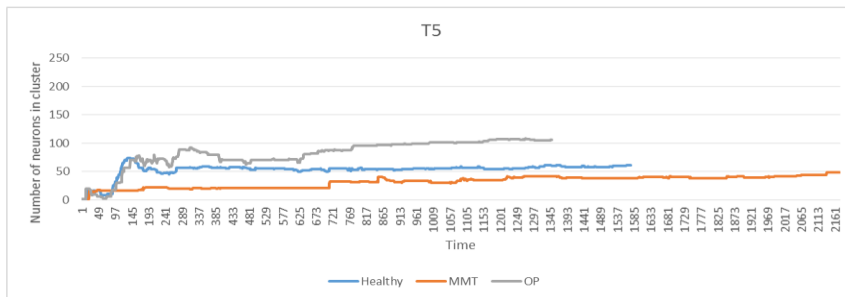
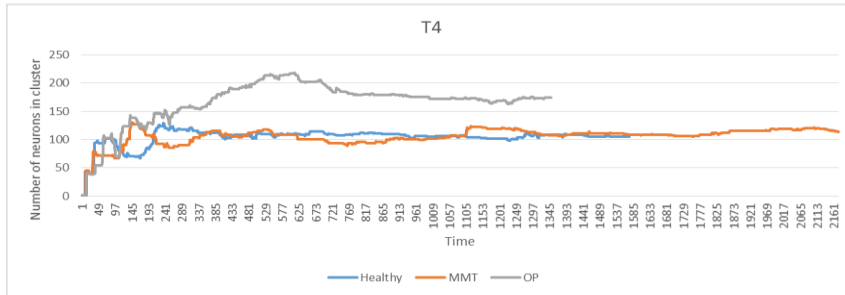
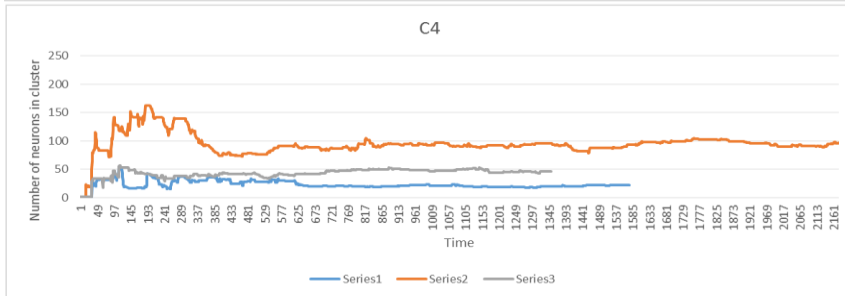
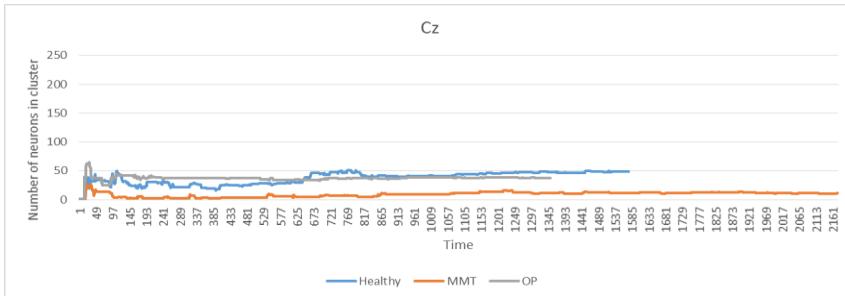
The clusters which evolved in the SNN model during unsupervised learning can also be statistically investigated in terms of the size (number of neurons that belong to each cluster) and also in terms of the cluster creation time. The clusters were scaled up or down with respect to the number of neurons (cluster members) associated with every input EEG variable (cluster centre). A bigger cluster contains a larger number of spiking neurons around the centre, which means that more spikes were propagated *via* this centre to the SNN model. A greater number of input spikes implies that more changes were observed in the brain signals and this reflects stronger brain responses.

By comparing the number of neurons that belong to each EEG channel, I can differentiate the dynamic brain activity patterns captured *via* different EEG channels across the subject groups in GO *versus* NOGO trials. Figure 6-8 is a representation of the cluster size changes, while three SNN models were trained with EEG data of the H, MMT, and OP subjects during the GO trials.

The horizontal axis represents the number of EEG time points entered to the SNN model training *via* input neurons corresponding to the EEG channels. The vertical axis represents the number of neurons that belong to the cluster (cluster size). In this procedure, the clusters are made bigger or scaled down with respect to the number of neurons associated with every input EEG channel through the EEG data training inside the SNN models. The average number of neurons in each cluster for the GO and NOGO tasks are reported, respectively in Table 6-2 and Table 6-3. The quantitative information of the cluster size changes during the whole-time interval of the learning procedure, as illustrated in Appendix D.







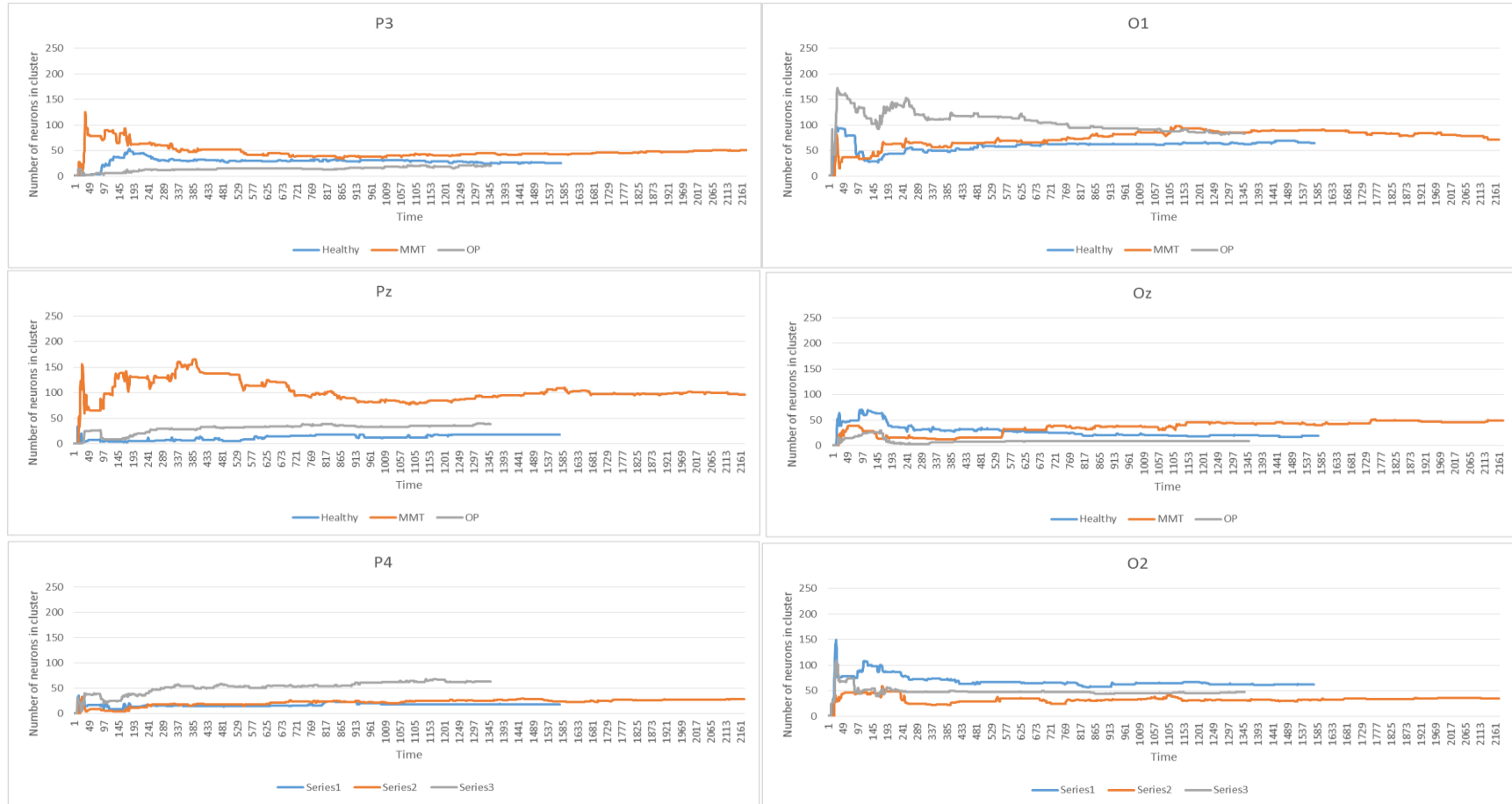
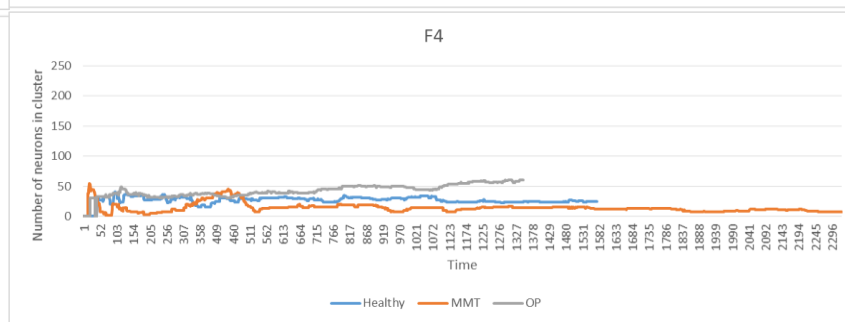
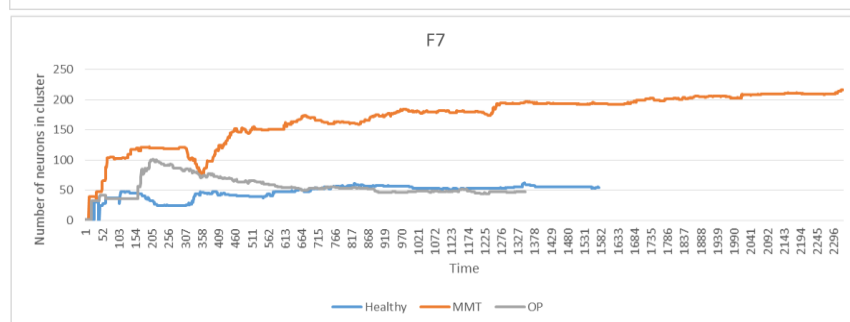
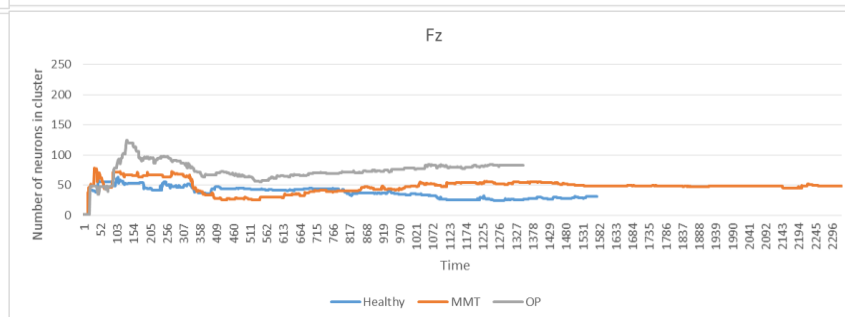
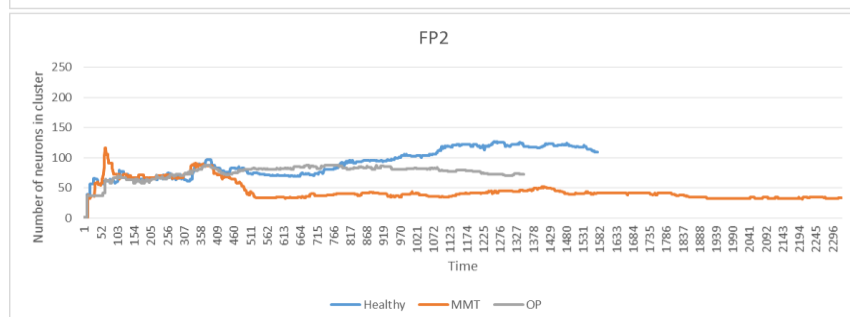
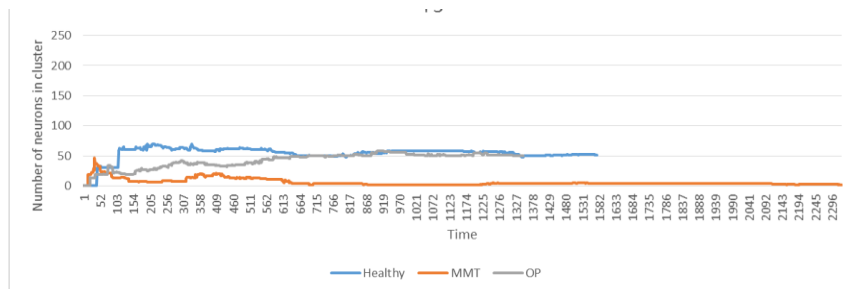
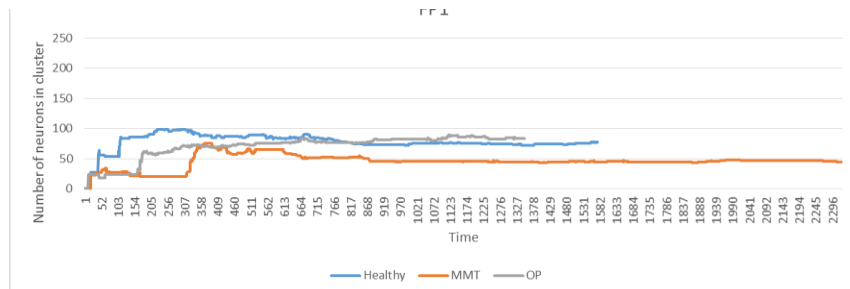
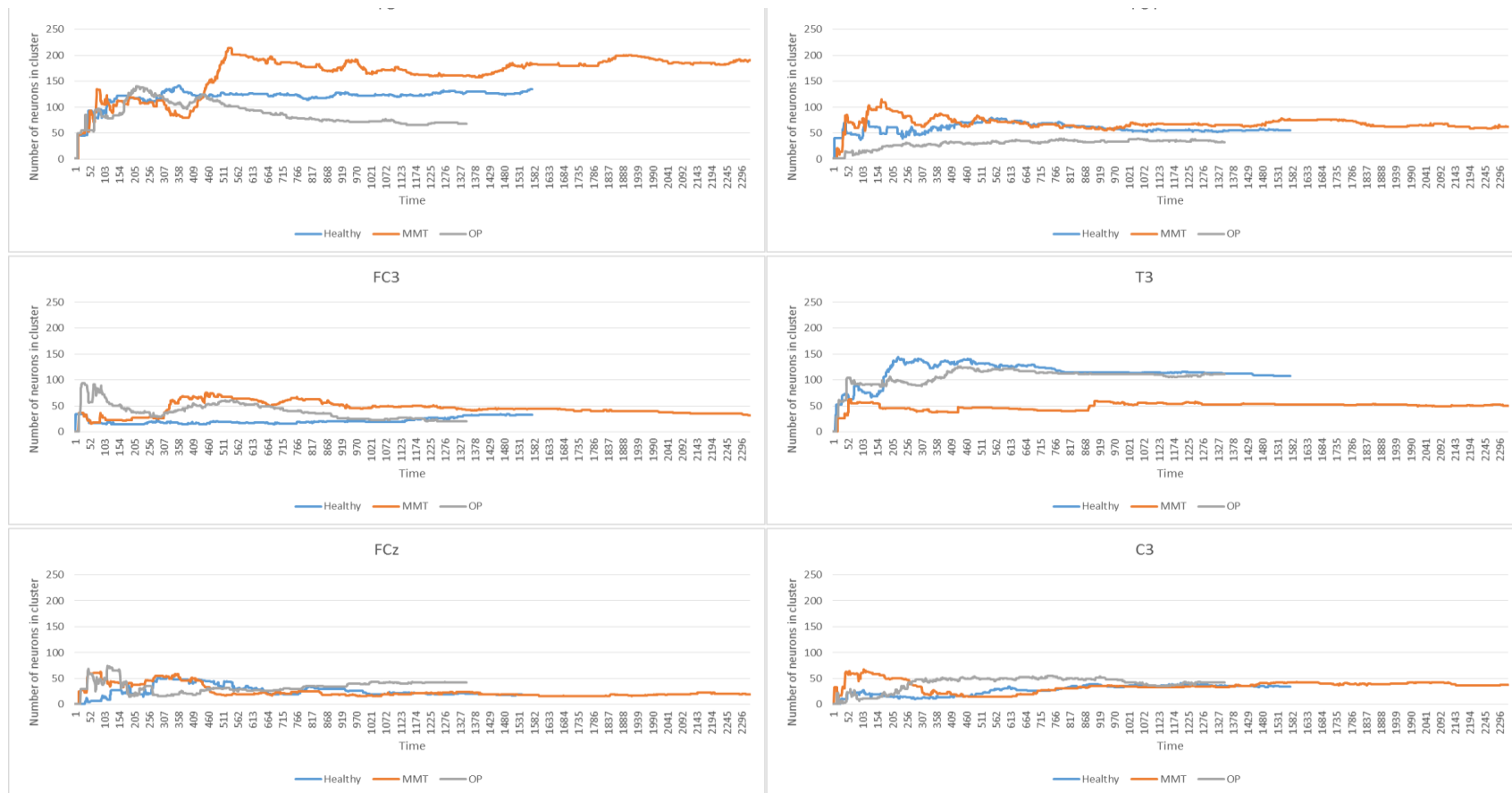


Figure 6-7 There are 26 clusters which are centred at the input neurons corresponding to the EEG channels. The size of the clusters changes while the SNN models are training on EEG data of the H, MMT and OP subjects in GO trials.





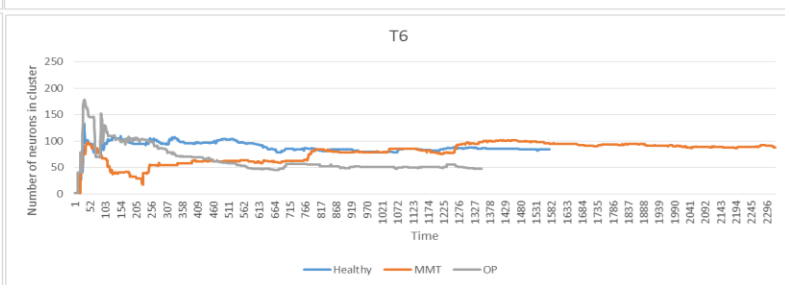
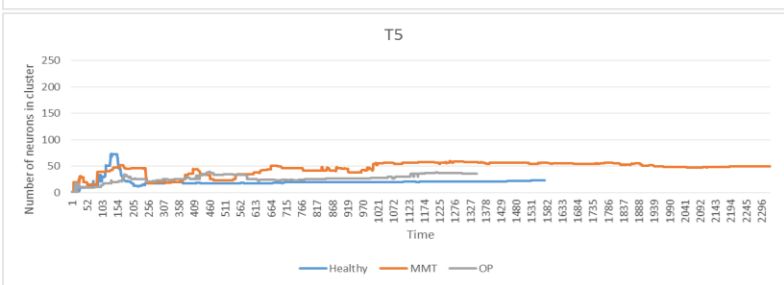
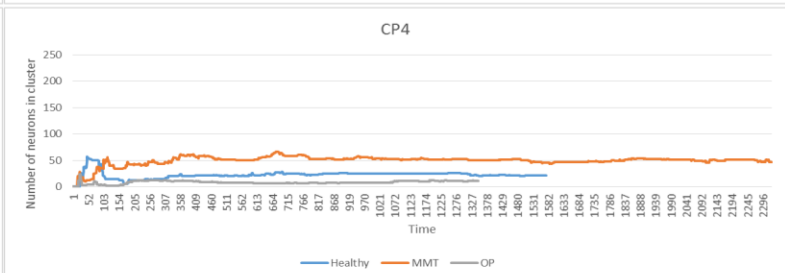
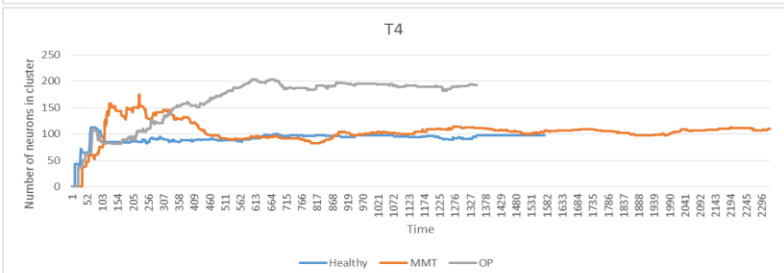
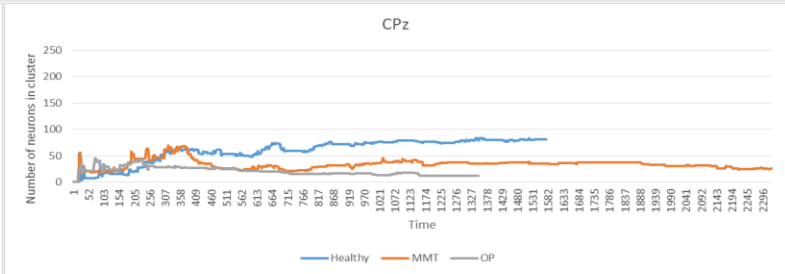
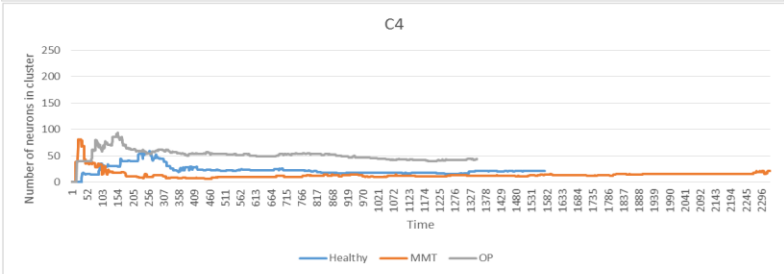
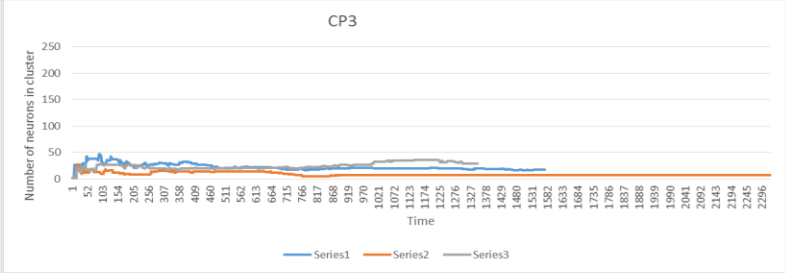
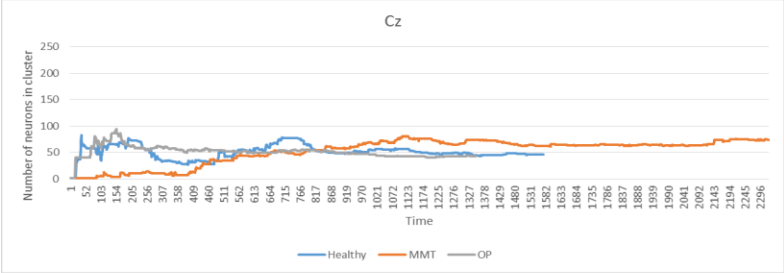




Figure 6-8 There are 26 clusters which are centred at the input neurons corresponding to the EEG channels. The size of the clusters changes while the SNN models are training on EEG data of the H, MMT and OP subjects in NOGO trials.

Table 6-2 The size of all the EEG channel clusters (plotted in Figure 6-7) was averaged over all the time points during the STDP learning in SNN models of H, MMT and OP groups in GO task.

	Fp1	Fp2	F7	F3	Fz	F4	F8	FC3	FCz	FC4	T3	C3	Cz	C4	T4	CP3	CPz	CP4	T5	P3	Pz	P4	T6	O1	Oz	O2
Healthy	77.3	70.7	77	22	64	38	108	45	36	56.1	83	28.9	38	23.1	104	11.1	82.3	30.3	53	29	13	17.1	76.4	59	27	67.7
MMT	56.9	17.8	108	19	63	12	186	74	23	43.6	53	12.2	9.3	94.6	106	9.16	13.3	17.7	32	48	103	22.5	65.9	75	35	33
OP	43.7	64.4	121	13	45	71	93	90	10	30	82	16.8	37	42.7	166	12.6	20.6	29.4	81	14	30	52.8	75.9	106	9.2	47.9

Table 6-3 The size of all the EEG channel clusters (plotted in Figure 6-8) was averaged over all the time points during the STDP learning in SNN models of H, MMT and OP groups in NOGO task.

	Fp1	Fp2	F7	F3	Fz	F4	F8	FC3	FCz	FC4	T3	C3	Cz	C4	T4	CP3	CPz	CP4	T5	P3	Pz	P4	T6	O1	Oz	O2
Healthy	78	91.4	47	54	38	27	120	21	26	59.4	114	27.8	51	22.8	91.7	22.2	60.1	22.7	21	3.1	25	42	88	122	19	50.7
MMT	45.6	44.7	171	6.4	48	14	164	45	25	68.6	49	35.2	52	13.9	104	8.68	33.5	49.8	46	41	35	43	78	73	39	30.6
OP	70.1	75.2	56	43	75	43	87	39	34	30.2	106	40.5	58	51.1	162	24.5	20.8	8.55	27	16	23	32	65	97	36	82.1

As shown in Table 6-2, when participants were dealing with the GO task, the largest average number of neurons belongs to F8 for H and MMT groups with 108 and 186 values respectively, while the largest average for the OP group belongs to T4 with 166 values. On the other hand, Table 6-3 shows that the largest average number of neurons in the NOGO task belongs to F8 (120) for H group and F7 (171), F8 (164) for MMT group and T4 (162) for OP group. Figure 6-8, Table 6-2 and Table 6-3 show that the cluster size changes during the presentation of the EEG data of a particular order of the subject data. Two questions may arise in this regard:

- Would cluster evolution be different within a subject group?
- Would the order of presentation of subject data influence the final clusters?

To address these questions, the clustering experiment was performed 10 times for each subject group using a random order of the subject data presentation. It can be seen from Figure 6-9 that different clusters based on EEG channels have different variability (in terms of the cluster size) across subject groups and within a subject group. Figure 6-10 illustrates the variance in cluster size across the groups. If a cluster has obtained a small standard deviation σ (measured as variation of a cluster size across all samples in one group and across the 10 different experiments), it may indicate that a uniform brain activity was measured from the corresponding EEG channel across all the subjects. For instance, a high value for T4 represents that the T4 cluster was developed differently across the individuals. In order to evaluate the validity of the created clusters, the average of the Silhouette coefficients (presented in Chapter 5) was measured in every cluster as shown in Figure 6-11. The graph shows that all the average Silhouette values are positive and very close to 1 which represents a high goodness value for the clusters.

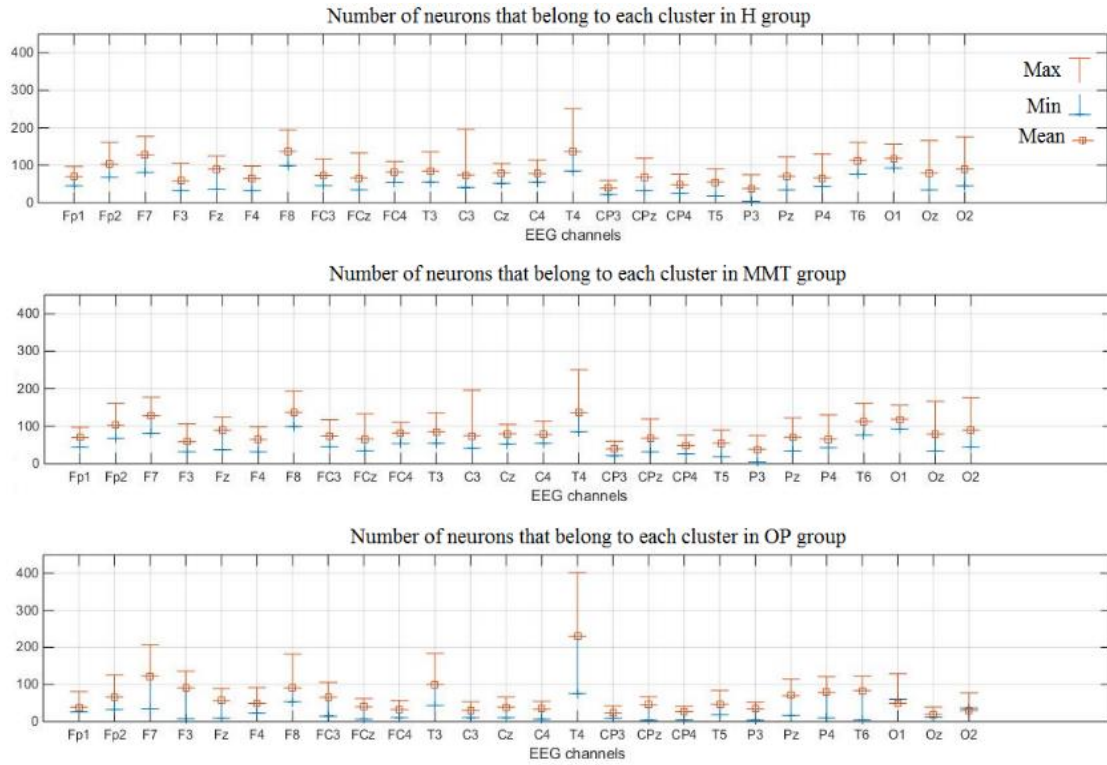


Figure 6-9 The minimum, maximum and the mean of the number of neurons that belong to each cluster. The dynamic clustering was performed 10 times for each group of subjects using random order of the subject data presentations.

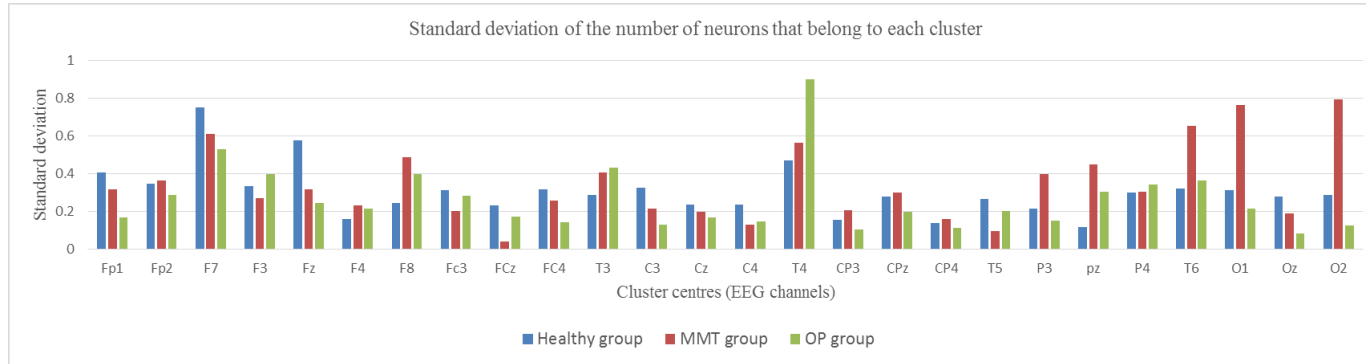


Figure 6-10 The standard deviation σ is reported for healthy (in blue), MMT (in red) and OP (in green). A higher σ value (mostly observed in MMT and OP groups) may represent less stability in cluster size evolved by different order of sample presentation. Blue, red and green colours represent respectively H, MMT and OP groups.

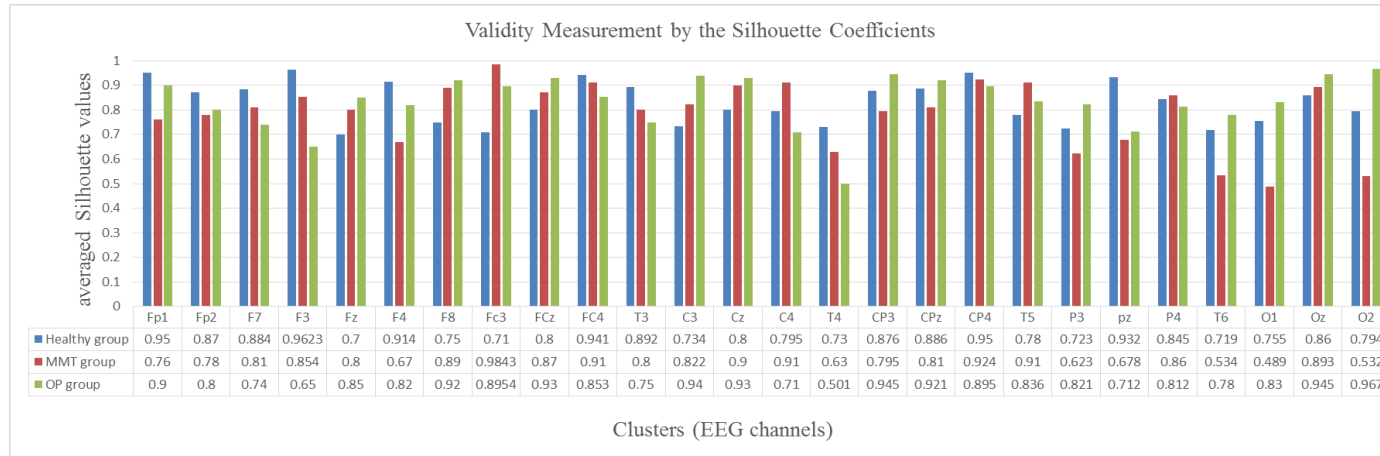


Figure 6-11 Validity measurement of the clusters generated in the case study of EEG data with 26 channels from three groups of subjects (H group in blue, MMT group in red, and OP group in green). The Silhouette value was measured for every neuron in a cluster. Then the Silhouette values were averaged over all the neurons in a cluster and represented as a validity measure for this cluster.

6.2.3 Analysis of Temporal Patterns in Clusters

Every cluster is composed of a number of neurons that received more spikes from the centre of this cluster over the time of the STDP learning. During the STDP, the neuron's postsynaptic potential (PSP) increases by every input spike that arrived to the neuron at time t . Once the PSP reaches a firing threshold, the neuron emits an output spike and sends it to other neurons which are connected to it.

Figure 6-12 exemplifies the PSP rate and the spiking rate in one randomly selected cluster centred by FP1 channel. This result allows us to study the pattern of the cluster creation and adaptation in detail by looking at the number of spikes produced within a cluster and also the changes in the neurons' PSP during the learning.

It can be seen from Figure 6-12 that the PSP rate and the spiking rate resemble the LIFM behaviour as shown in Chapter 3. More analysis of the PSP rate will be presented on the same EEG data case study in the next chapter.

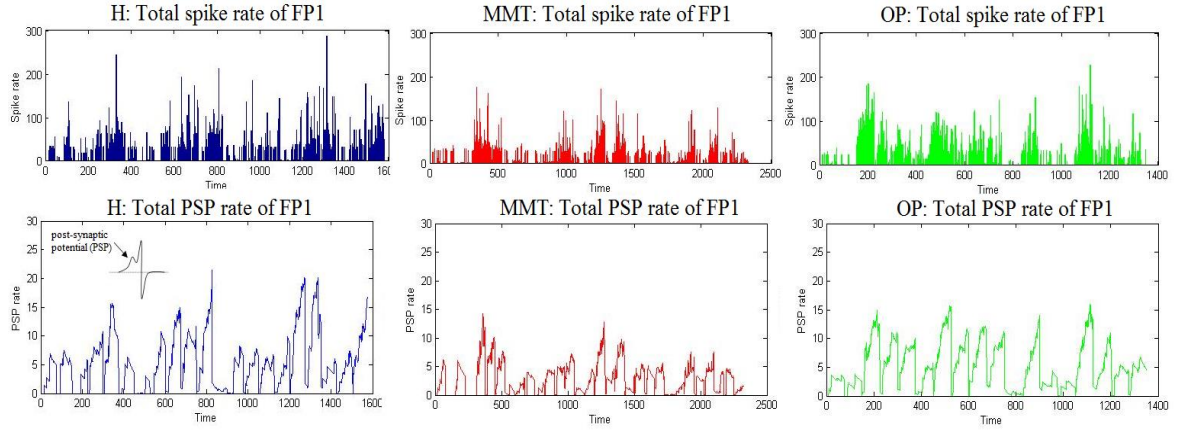


Figure 6-12 The spike rates and PSP rates of the neurons within the FP1 cluster during unsupervised learning of EEG data (blue: H group, red: MMT group, and green: OP).

To translate the experimental results into meaningful interpretations, it is essential to detect the STBD variables that demonstrate a good discrimination between multiple classes. As described in Chapter 5, Section 5.2.1, during the clustering of the SNN model, significant dynamic patterns were associated with each cluster. These are five temporal patterns: input spike train (s_t), the mean of the cluster's postsynaptic potentials ($\mu_{PSP(t)}$), the mean of the cluster's spiking rates (sr_t), the cluster size, and the mean of the neurons memberships (the ratio of information received by neurons as explained in Section 5.2.1). Figure 6-13 and Figure 6-14 show the dynamic clustering patterns of the Fz and T4 channels generated from 5 samples in the H group.

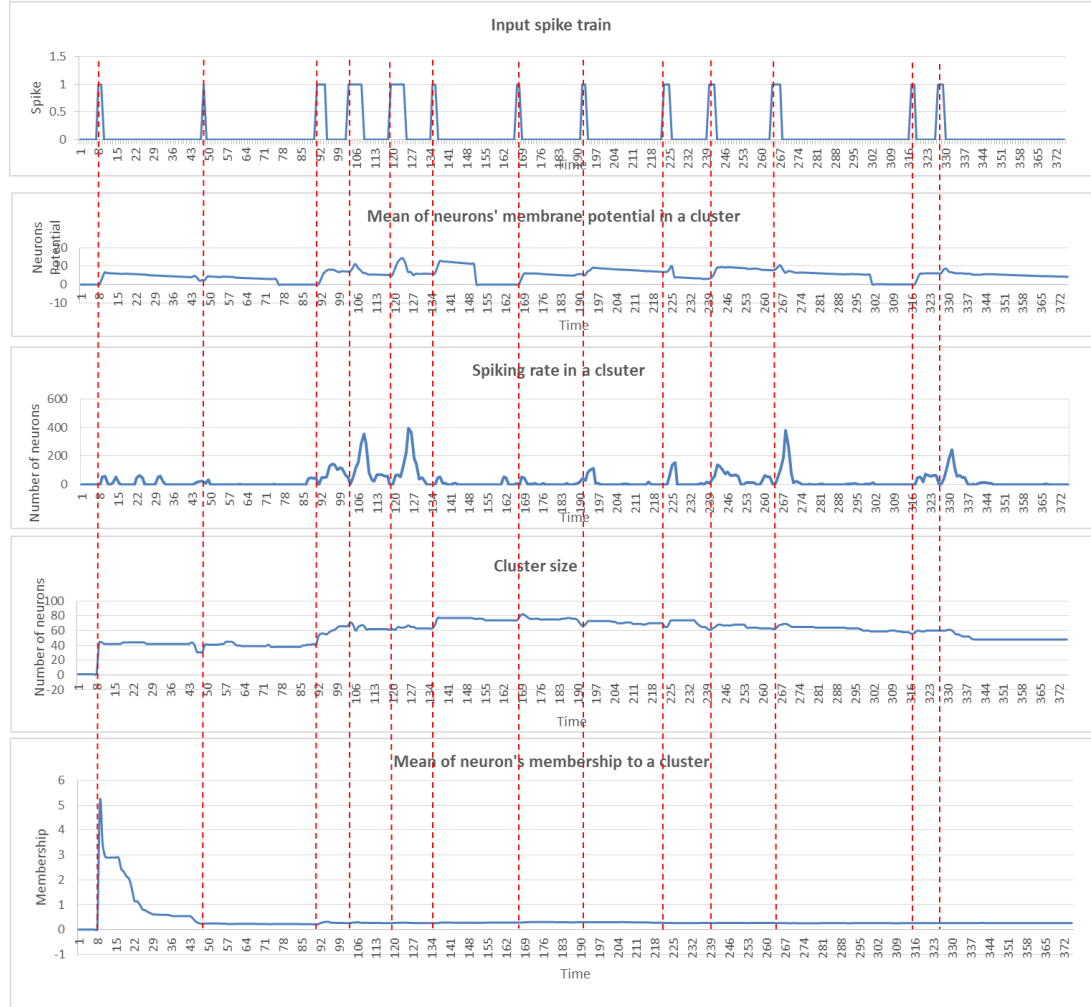


Figure 6-13 Dynamic patterns of one cluster (EEG channel Fz) against an input spike train (s_t) corresponding to 5 samples in class 1 (H subjects). Each sample constitutes of 75 time points.

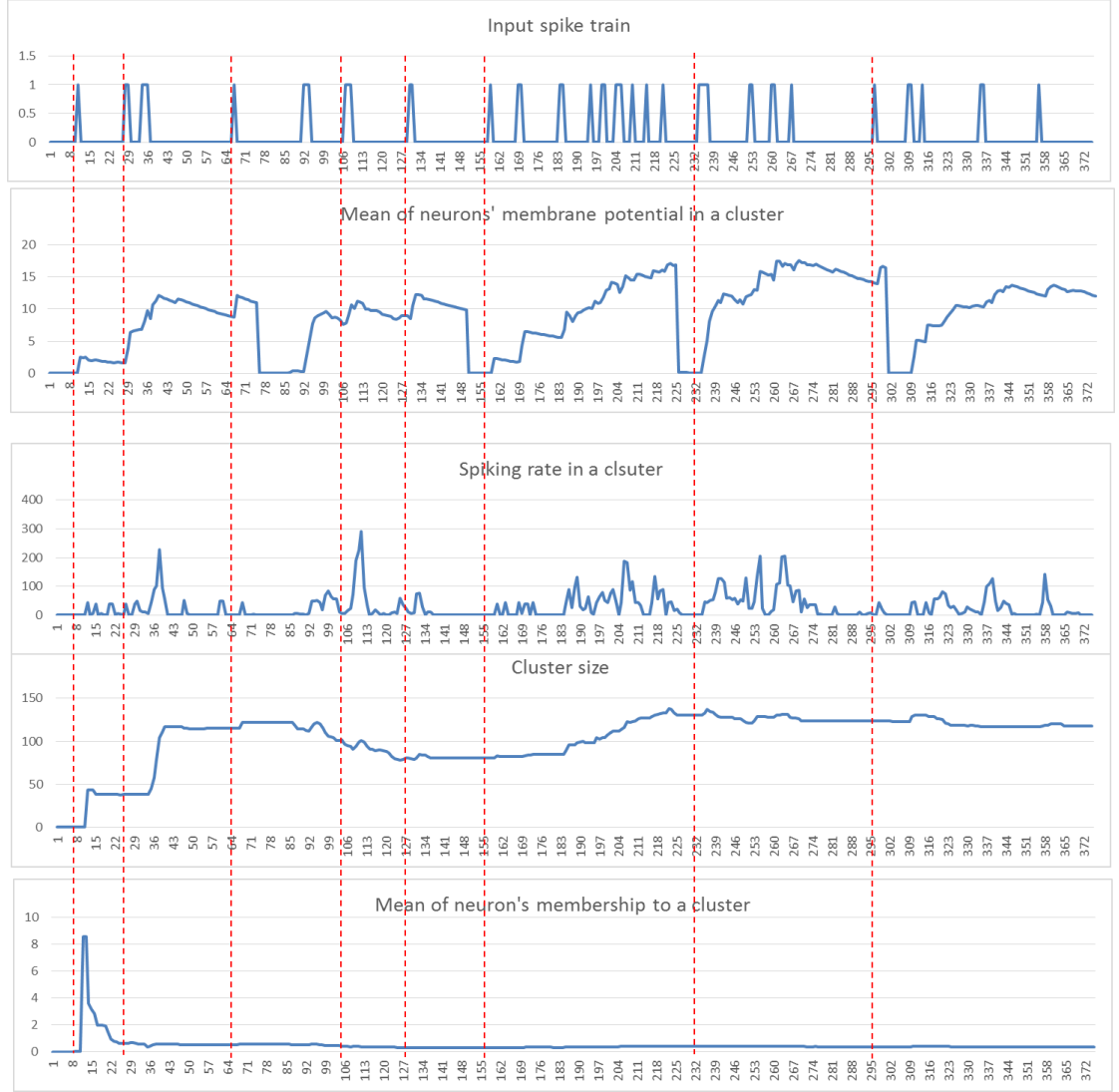


Figure 6-14 Dynamic patterns of one cluster (EEG channel T4) against an input spike train (s_t) corresponding to 5 samples in class 1 (H subjects). Each sample constitutes of 75 time points.

In Figure 6-13, the pattern of the input spike train (s_t) shows the encoded EEG data of channel Fz corresponding to 5 samples from the H group. The s_t is the input data to the NeuCube SNN model *via* the cluster centre (the input neuron). The size of the cluster shows the number of neurons that belong to a cluster at each time t of the learning process. The $\mu_{PSP(t)}$ and sr_t patterns, are respectively, the mean of the PSP and the spike rates measured from the neurons within the Fz cluster.

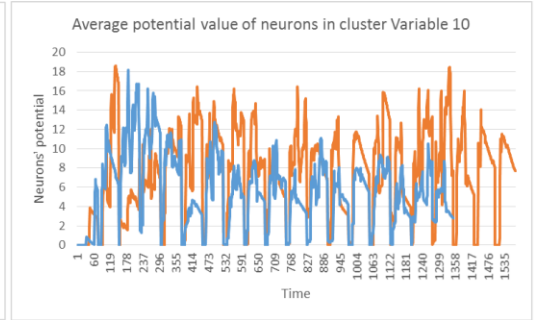
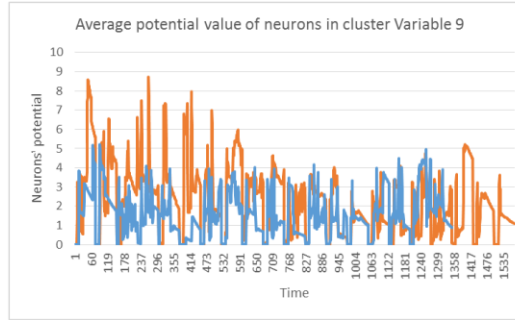
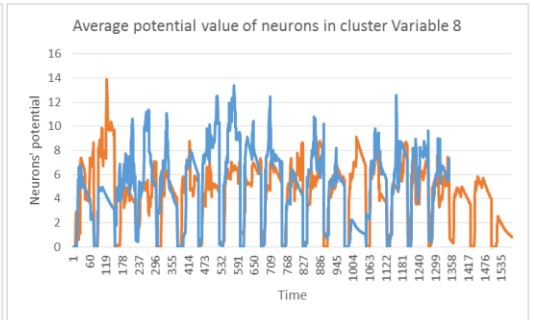
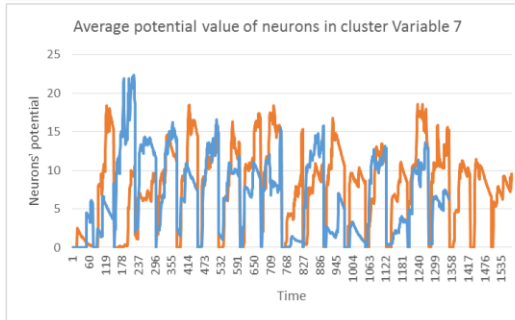
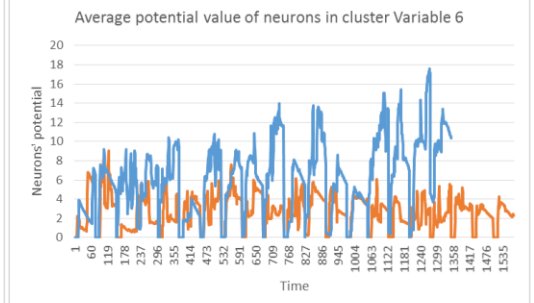
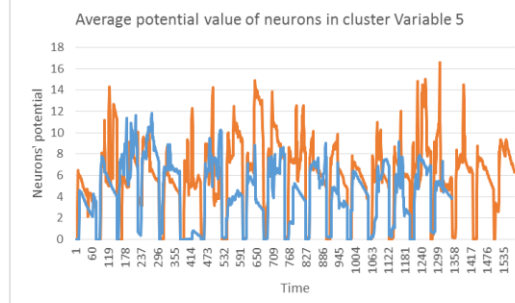
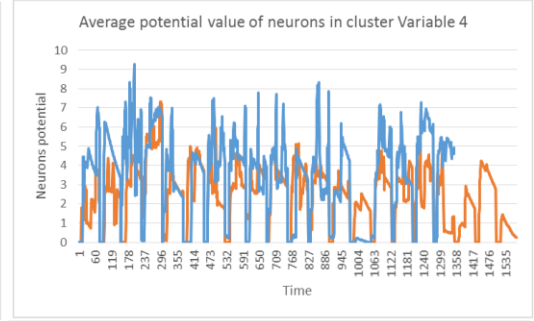
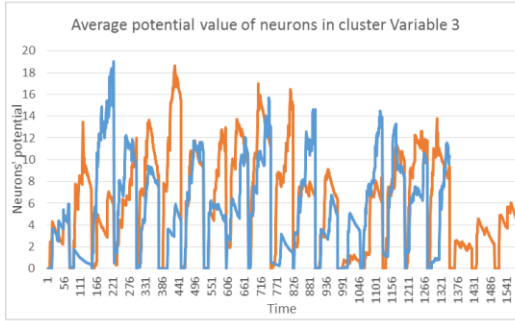
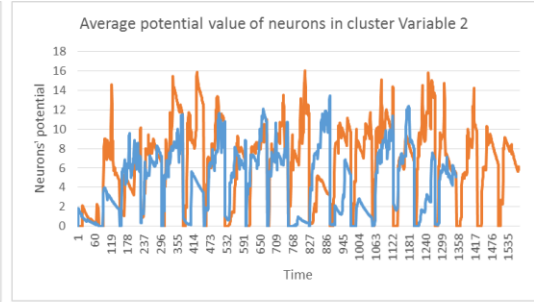
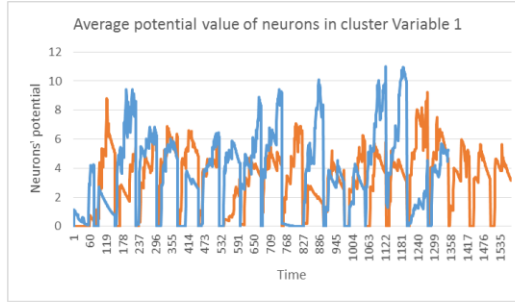
6.2.4 Feature Selection

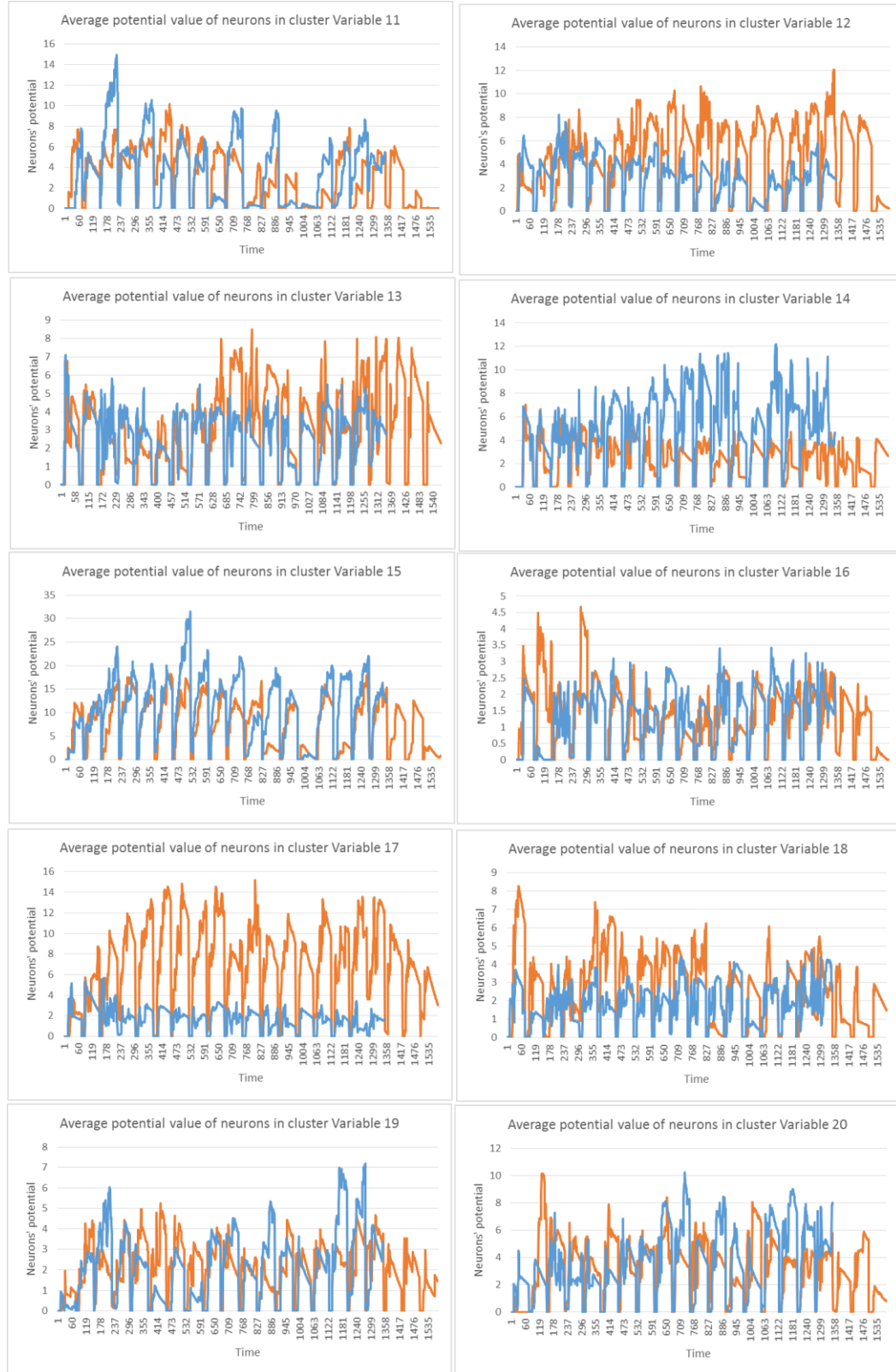
In this section, $\mu_{PSP(t)}$ time series are analysed to reveal how the proposed dynamic clustering approach can be useful to discriminate the EEG data samples across different classes.

Figure 6-15 shows the $PSP(t)$ time series of all the 26 clusters (26 EEG channels: Fp1, Fp2, F7, F3, Fz, F4, F8, FC3, FCz, FC4, T3, C3, Cz, C4, T4, CP3, CPz, CP4, T5, P3, Pz, P4, T6, O1, Oz and O2), which were captured during the learning process with the EEG data of two classes H and OP. The distribution of $\mu_{PSP(t)}$ is illustrated in Appendix D. In order to analyse the $\mu_{PSP(t)}$ time series of 26 clusters of the H and OP groups, the following statistical methods have been used:

- $P_{max}(t)$: Local maximum in the $\mu_{PSP(t)}$.
- Area under the curve.
- Mid of potential.

For each sample, the local maximum of the $\mu_{PSP(t)}$ (the peak of potential— $P_{max}(t)$) is plotted as a dot in time t . This plot in Figure 6-16 can potentially separate the samples across the classes with a different degree of discrimination. The $P_{max}(t)$ plot identifies the most discriminative and informative variables for this classification problem (used as feature selection), consequently it can detect the irrelevant variables which should not be included for this classification task as they may drop the classification accuracy.





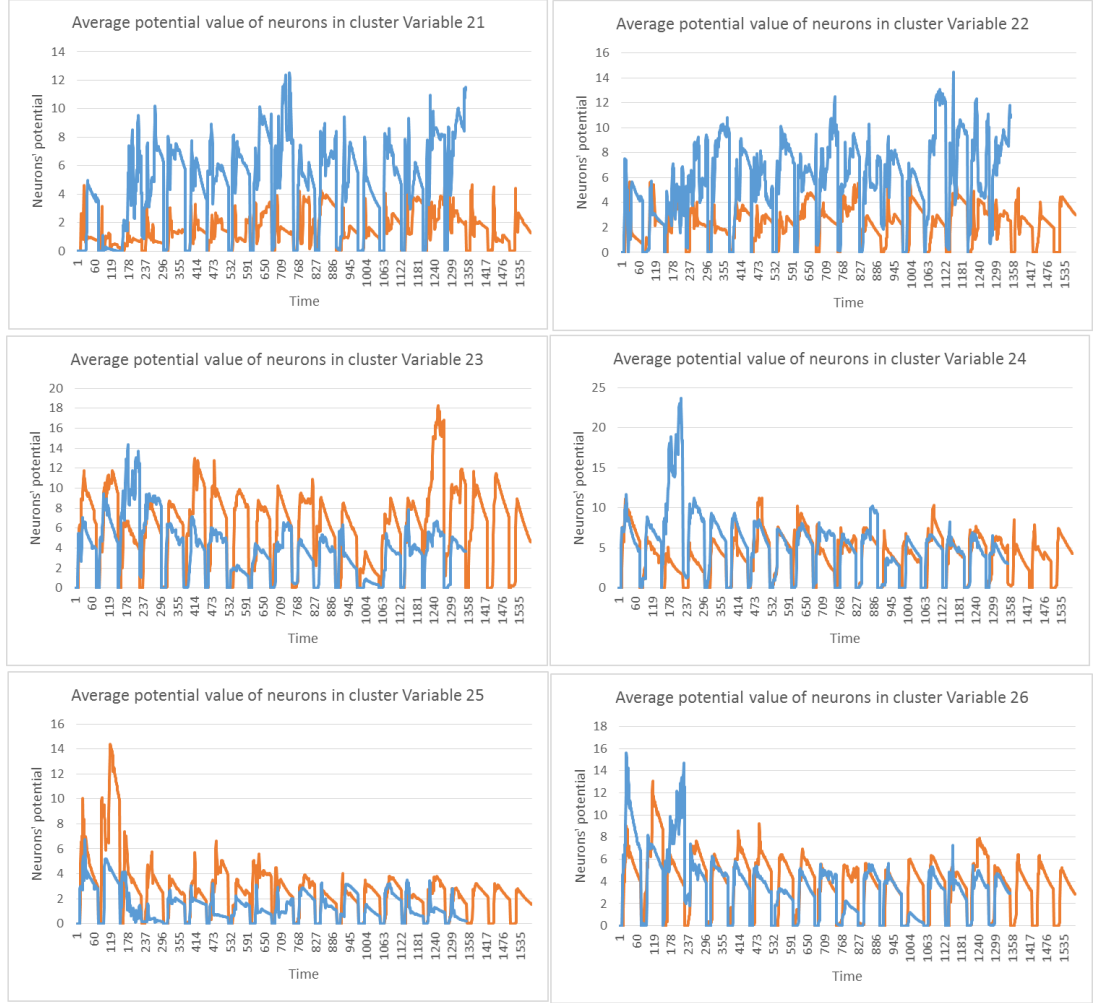


Figure 6-15 The dynamic patterns of the average PSP rates from 26 clusters during the learning process with EEG samples from classes H (in red) and class OP (in blue).

A t-test measure was applied to the P_{max} plots (shown in Table 6-4 and Figure 6-16) to identify how these two classes (H and OP) are statistically significant. Figure 6-17 illustrates the area under the curve of $\mu_{PSP(t)}$ for each sample, computed according to a definite integral, where l is the length of each sample (time points) as follows:

$$\int_1^l P(t) dt \quad (6-1)$$





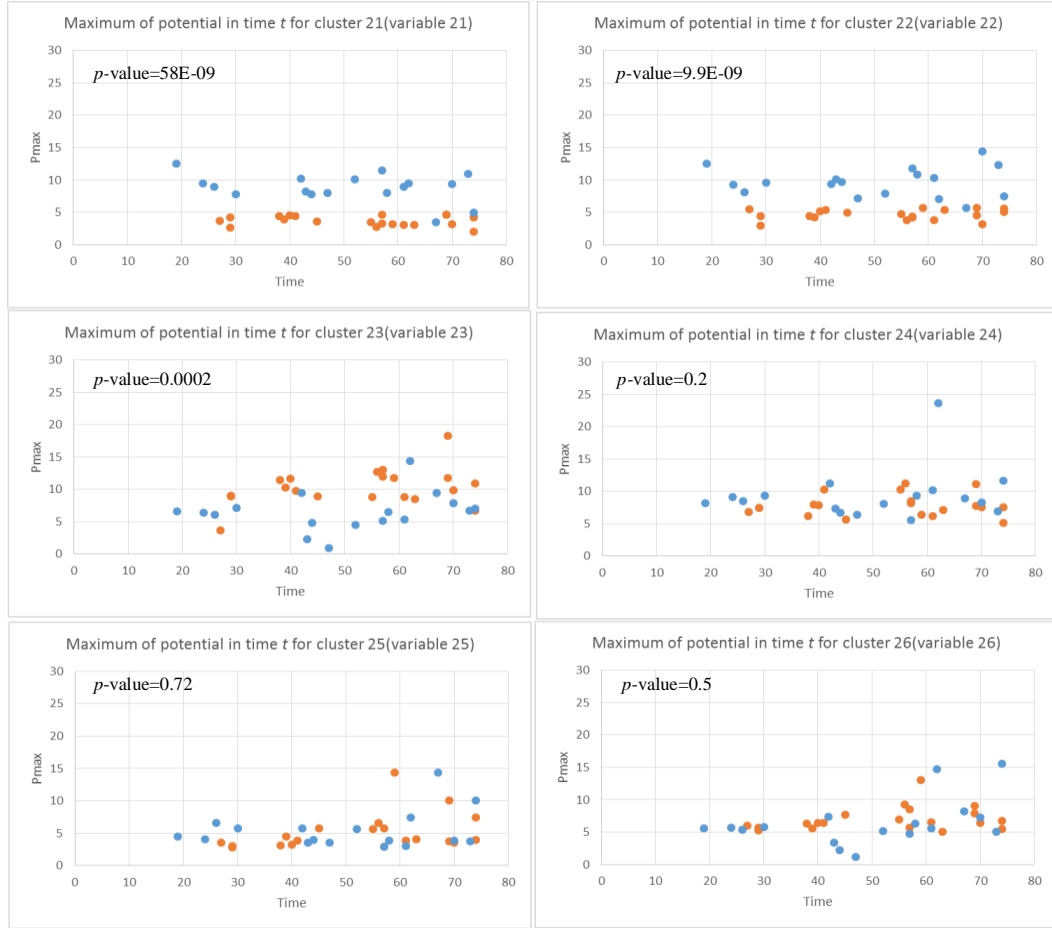


Figure 6-16 For 26 clusters, the local maximum of the potential value, $P_{max}(t)$ are plotted as dots in time t for all samples in class H (red) and class OP (blue). The $P_{max}(t)$ values show discriminative patterns between class H and class OP in EEG variables with small p -value (measured by a t-test) as shown in Table 6-4.

The midrange of a potential (shown in Figure 6-18) is the average within the min value and max value of a curve. The formula to find the midrange is $(\text{high} + \text{low}) / 2$. Table 6-4 represents that the statistical t-test measure was applied to the P_{max} (left), the area under the curve of $\mu_{PSP(t)}$ (middle) and the midrange of the PSP (right) to identify how the clustering patterns of H and OP are statistically significant. The EEG channel CPz (index: 17) has the lowest p -value, representing the highest discriminative power between the samples that belong to different classes.

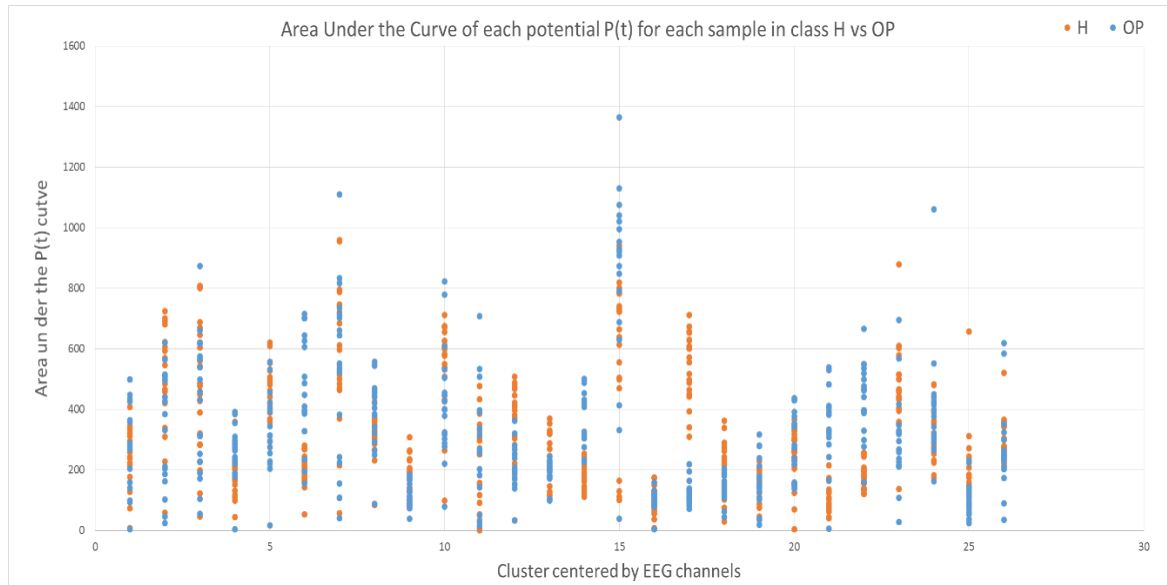


Figure 6-17 The area under the curve of PSP rates for 26 clusters for all the samples in class H (red) and class OP (blue). As shown in Table 6-4, different discriminative power between the samples that belong to class H *versus* OP have been observed.

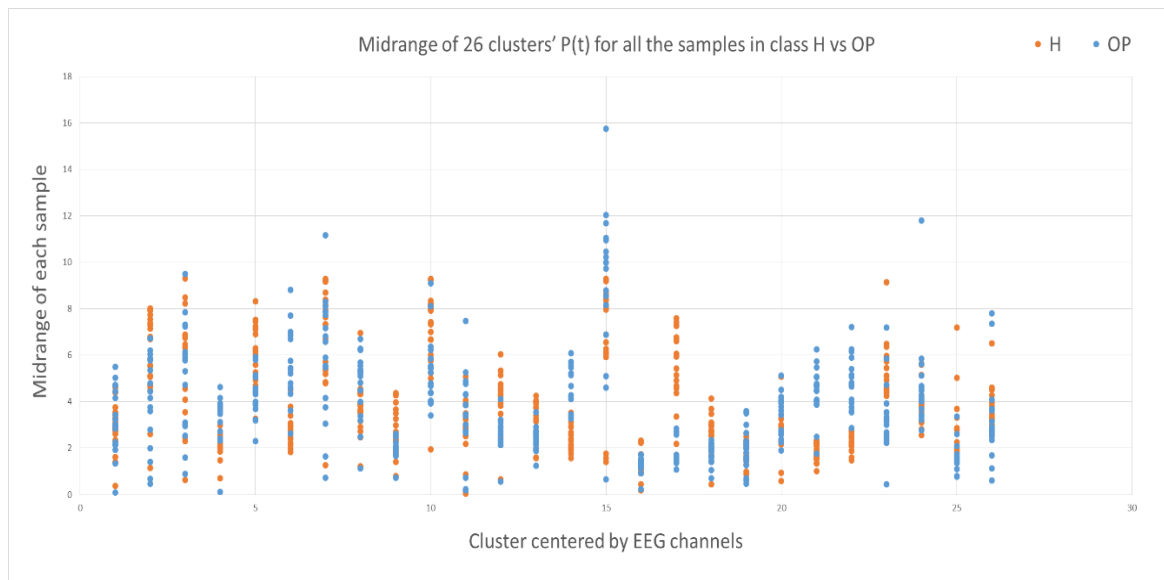


Figure 6-18 The midrange of the PSP rates corresponding to 26 clusters for all samples in H (red) class and OP (blue) class. As shown in Table 6-4, the midrange values have shown different discriminative power between samples belong to class H *versus* OP.

Table 6-4 A t-test measure was applied to the P_{max} (left), the area under the curve of PSP (middle) and the midrange of the PSP (right) to identify how two classes H and OP are statistically significant. EEG channel 17 has the lowest p -value, representing the highest discriminative power between the samples that belong to different classes.

$P_{max}(t)$			Area under curve			Midrange of the PSP		
P-Value	EEG channel	Channel index	P-Value	EEG channel	Channel index	P-Value	EEG channel	Channel index
2.41E-11	CPz	17	1.21E-11	CPz	17	-1E-11	CPz	17
2.38E-09	C4	14	1.37E-08	C4	14	8.4E-09	C4	14
4.78E-09	Pz	21	2.4E-08	P4	22	1.7E-08	Pz	21
9.93E-09	P4	22	1.8E-07	Pz	21	4.9E-08	P4	22
0.00001	F4	6	7.3E-06	F4	6	2.2E-06	F4	6
0.00008	C3	12	3.9E-05	C3	12	8.2E-05	C3	12
0.00008	Fz	5	0.0007	T6	23	0.0001	Fz	5
0.00019	T6	23	0.0025	Fz	5	0.0003	T6	23
0.0004	F3	4	0.002	FCz	9	0.0008	F3	4
0.001	FC4	10	0.002	Oz	25	0.001	CP4	18
0.004	P3	20	0.003	F3	4	0.003	FC4	10
0.004	CP4	18	0.006	CP4	18	0.009	Fp2	2
0.005	T4	15	0.01	T4	15	0.013	Cz	13
0.0059	Cz	13	0.01	Fp2	2	0.013	FCz	9
0.0076	Fp2	2	0.02	FC4	10	0.014	Oz	25
0.0176	FCz	9	0.07	Cz	13	0.02	T4	15
0.0247	FC3	8	0.074	O1	24	0.05	FC3	8
0.21	O1	24	0.09	FC3	8	0.16	O1	24
0.309	F8	7	0.3	F8	7	0.2	P3	20
0.31	T3	11	0.3	P3	20	0.28	F8	7
0.37	CP3	16	0.47	Fp1	1	0.3	T3	11
0.4	Fp1	1	0.48	T3	11	0.49	Fp1	1
0.53	T5	19	0.53	O2	26	0.57	O2	26
0.55	O2	26	0.56	F7	3	0.78	F7	3
0.72	Oz	25	0.9	CP3	16	0.78	T5	19
0.84	F7	3	0.97	t5	19	0.98	CP3	16

As can be seen from Table 6-4, the top 8 channels have shown high discriminative power between class H and class OP. These are variables: 17, 14, 21, 22, 6, 12, 5 and 23 which respectively correspond to CPz, C4, P4, Pz, F4, C3, T6, and Fz channels. These 8 variables were used as top-informative features for classifying the EEG samples into two classes: H and OP. Table 6-5 shows that the classification accuracy results were higher when the SNN model was trained by the EEG data from these 8 important variables than when using all the 26 variables, as reported in Chapter 4.

Table 6-5 The classification accuracy between EEG samples in H and OP in the GO task obtained when using the all the EEG variables *versus* using the 8 top-informative variables.

Methods	NeuCube	SVM	MLP
No feature selection: all 26 variables were used	80.00	68.00	78.00
8 selected variables (feature selection)	90.00	70.00	78.00

6.3 Chapter Summary

In this chapter, the proposed spatio-temporal clustering method was applied to a case study of EEG data recorded from three groups of subjects. Dynamic clustering in an SNN model reflected the activity of input neurons (EEG channels) at each time t of the STDP learning process. This allows for an interpretation of the model, which resulted in knowledge discovery in the SNN evolutionary learning patterns. The assessment of these evolutionary patterns allowed me to identify more important EEG channels (considered as feature selection) that improved the classification accuracy.

Hitherto, the SNN models have demonstrated a successful analysis by considering all the samples (all individuals) in a given data space. However, as the SNN models were trained on the whole data spaces, they could not differentiate across individuals who belong to the same class of data. Therefore, the next chapter presents a new approach for building personalised SNN models for individuals using the most informative subset of samples as the training set.

6.4 Contribution

In this chapter, I have made the following contributions:

1. I designed a feasibility study of the proposed dynamic spatio-temporal clustering on EEG data.
2. Knowledge discovery in SNN learning patterns through the assessment of clustering patterns. This resulted in feature selection and improved the classification accuracy.
3. I published parts of this research as a leading author in the following papers:

Doborjeh, M. G., Kasabov, N., & Doborjeh, Z. G. (2018). Evolving, dynamic clustering of spatio/spectro-temporal data in 3D spiking neural network models and a case study on EEG data. *Evolving Systems*, 9(3), 195-211.

Doborjeh, M. G., & Kasabov, N. (2015, November). Dynamic 3D clustering of spatio-temporal brain data in the NeuCube spiking neural network architecture on a case study of fMRI data. In *International Conference on Neural Information Processing* (pp. 191-198). Springer, Cham.

Chapter 7 A New Personalised Modelling using SNN

7.1 Introduction

Over the last decades, there has been an overwhelming abundance of neurological disorders which revealed a pressing need for neuroscientists to consider various brain imaging techniques for patients. An individual patient may have significant psychologic or behavioural factors. Therefore, neuroscientists have recently suggested personalised treatment to sidestep undesirable influences of conventional treatments on the current medical conditions (e.g., heart disease, diabetes mellitus, and so forth). Numerous personal events and personalised treatments for an individual patient can be achieved if appropriate computational models learn the complex patterns in multivariate data.

Inspired by the idea of personalised treatment, this chapter proposes a new personalised modelling aimed at developing a computational prognostic or diagnostic system. The proposed method is based on integration of different data processing techniques for an appropriate selection of neighbour samples. This has the potential to identify important characteristics of an individual through personalised profiling and improves the classification/prediction of output when compared with global modelling.

In health-related research, massive amounts of static personal clinical data and spatio-temporal data are available for patients that need to be precisely scrutinised. It becomes

apparent that a unifying computational approach is needed for proper analysis, understanding and knowledge discovery of such multifaceted data.

In this chapter, the hypothesis is that personalised modelling with SNN can be successfully used, if the SNN models learn from informative STBD which are selected based on nearest neighbour samples. This chapter addresses the following criteria:

- How to select relevant data samples for a new person for creating a personalised model based on integrated static and spatio-temporal data?
- How does the personalised model improve the performance of the classification/prediction?

To address the above questions, an application of the proposed method will be presented later in the chapter using a case study of EEG data measured along with static clinical information.

7.2 Personalised Modelling for Integrated Static and Dynamic Data

In this section, instead of building a global model and training it with STBD of the whole population, for every person a personalised SNN model (PSNN) will be built and trained only on a subset of STBD which belong to individuals who have similar integrated static clinical factors and dynamic STBD.

The proposed SNN-based personalised modelling system is based on the following steps:

1. Clustering of integrated static-dynamic data is performed using the new algorithm DWWKNN (Dynamic Weighted-Weighted Distance K-nearest Neighbours). For a new

- individual x_i , I rank the contribution of each of the k neighbour samples based on integrated static-dynamic distance to the x_i , giving greater rank to closer neighbours.
2. Select the STBD of those samples that are selected by DWWKNN method.
 3. Using the selected STBD for unsupervised learning in the PSNN model (the STBD of the new person x_i is excluded from training).
 4. Test the PSNN model with STBD of x_i , which is unknown to the model.

The structure of the proposed personalised modelling system is sketched in Figure 7-1.

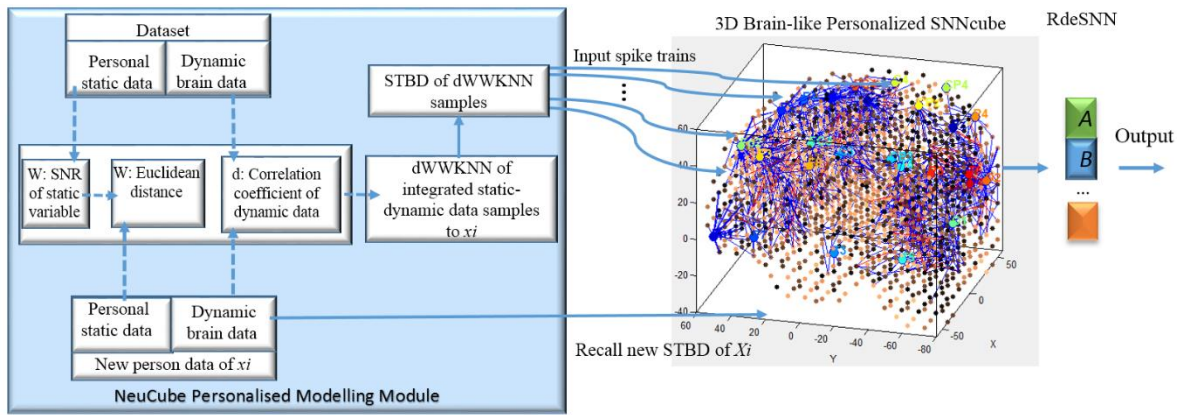


Figure 7-1 A block diagram of the personalised modelling that I proposed for integrated static and dynamic data. An SNN is trained with STBD samples that are found using the proposed DWWKNN method (the new person x_i is excluded from training).

The integrated static-dynamic data clustering procedure is accomplished by the following three subparts:

- a) Forming a cluster C_s in static vector-based data space as nearest neighbours to a new sample vector x_i .
- b) Forming a cluster C_{STBD} in STBD space as nearest neighbours to a new sample data.

- c) Obtaining a cluster of samples that have integrated static-dynamic similarity through intersection of C_{STBD} and C_s . These steps are elucidated in the following:

In the WWKNN approach, the first weight W is defined according to the Euclidean distance between a new sample static vector x_i and other sample static vectors in the global dataset. The main idea behind the WWKNN algorithm is to include one more rank (weight) vector to weigh the importance of the variables. Therefore, the second weight is defined to represent the importance of each variable to the input static vector for which a model is being built.

When calculating the Euclidean distance in a V -dimensional space of input variables, it is usually assumed that all variables have the same impact on the output. However, when the variables are ranked in terms of their discriminative power of class samples over the whole V -dimensional space, different variables have different importance to separate samples across classes. To rank the importance of each input variable, a standard statistical measure, known as Signal-to-Noise Ratio (SNR), was used to evaluate how important a variable is to discriminate samples belonging to different classes, one class named as ‘signal’ and the rest as ‘noise’.

For a two-class problem, SNR values for a variable $\{v = 1, \dots, m\}$ is calculated as an absolute value of the difference between the mean value $M1_v$ of the variable for class 1 and the mean $M2_v$ of the variable for class 2, divided by the sum of the respective standard deviations $\sigma1_v$ and $\sigma2_v$. In a C -class problem, where $C = \{1, 2, \dots, n\}$ for each variable v , the SNR is defined as follows:

$$SNR_v = \frac{\sum_{i=1}^n \frac{abs(\mu i_v - \mu\{C \setminus i\}_v)}{\sigma i_v + \sigma\{C \setminus i\}_v}}{n}, v = 1, \dots, V \quad (7-1)$$

where i denotes which class is named as signal, while $\{C \setminus i\}$ is the rest of the classes as noise. Therefore, σi_v and μi_v are correspondingly the standard deviation and the mean value of the variable v among all the samples in class i . The obtained SNR_v is used to weight the Euclidean distance D_{ij} between a new vector x_i and another sample vector x_j as follows:

$$D_{ij} = \frac{\sqrt{\sum_{v=1}^m SNR_v (x_{i,v} - x_{j,v})^2}}{\sum_{v=1}^m SNR_v} \quad (7-2)$$

D_{ij} is then normalised and assigned in the range of [0 1]. The distance D_{ij} in Equation (7-2) is computed only based on the static data. However, as the dataset is constituted by both static and dynamic information, a new distance needs to be measured with respect to the dynamic STBD. It is performed as follows:

A correlation coefficient is computed between each variable in STBD of sample x_i and the same variable in other samples. The correlation coefficient is a measure that defines the degree to which two time series are associated; it is defined as follows:

$$Corr_{x,y} = \frac{Cov(x,y)}{\sigma_x \sigma_y}, -1 \leq C_{xy} < 1 \quad (7-3)$$

To calculate the correlation coefficient, the covariance of two variables x and y , denoted by $Cov(x,y)$, is divided by the product of the two variables' standard deviations σ_x and σ_y . The correlation $Corr_{xy} = -1$ indicates a great negative correlation, while $Corr_{xy} = 1$ is a

great positive correlation. If $Corr_{xy} = 0$, then means there is no relationship between the two variables. Using the correlation coefficient, I rank samples with respect to their STBD relationships to a new sample, where the higher the rank, the greater a positive correlation between the samples.

Once the distance between static samples and the correlation between the STBD samples have been computed, a number of nearest samples can be selected with respect to two distance thresholds as follows: In static data space, a cluster C_s of static vectors can be extracted in which their computed distance D_{ij} to the new sample vector x_i is less than a distance threshold $t = \mu D_{ij}$. In dynamic STBD, I can extract a cluster C_{STBD} of samples where their computed correlation to the new sample is greater than a threshold $= \mu Corr_{ij}$. Finally, two clusters of samples are extracted, one refers to the samples with similar static information and the other one refers to the samples with similar dynamic STBD patterns to a selected sample. The cluster C_{sd} is then identified using the intersection relation of $C_s \cap C_{STBD}$. The DWWKNN algorithm is represented in Table 7-1.

Table 7-1 The proposed DWWKNN algorithm.

Algorithm 1: DWWKNN	
Input: Static dataset $X(N, V)$ with N samples and V variables, Number of class C , Spatio-temporal data samples of $STBD$	
Output: results	
1:	Procedure
2:	$s = \text{length}(STBD)$
3:	Cor [1: s , 1: s , 1: C] \leftarrow compute pairwise correlation coefficient between STBD samples for each class
4:	for $v=1: V$ do
5:	$SNR_v =$ compute the SNR for variable v in $(X(:, v))$
6:	end for
7:	for $i=1: N$ do
8:	$Center(i, 1: V) \leftarrow \text{vector } x(i, 1: V) \in X$
9:	for $j=1: N-1$ do

```

10: compute pairwise distance  $D_{ij} = \frac{\sqrt{\sum_{v=1}^V SNR_v (Center(i,v) - x(j,v))^2}}{\sum_{v=1}^m SNR_v}$ 
11: end for
12: form a cluster on static data space using  $C_1 \leftarrow find(D_{ij} > \mu D_{ij})$ 
13: for  $c=1:C$  do
14:    $T = \mu Cor[s, s, c]$ 
15:    $C_2 = find\{Cor[1:s, 1:s, c] \geq T\}$ 
16: end for
17: select the common samples in two clusters using  $C_{sd} = C_1 \cap C_2$ 
18: selected-samples ← STBD samples belonging to  $C_{sd}$ 
19: SNN model training with the selected-samples
20: classification(i) = recall STBD of  $x_i$  for testing
21: end for
22: Result =  $\frac{\sum \text{classification}(i)}{\text{length}(\text{selected-samples})}$ 
23: End of Procedure

```

7.3 Personalised Modelling for EEG Data

To exemplify the proposed personalised modelling system, the same EEG data that has been introduced in Chapter 4, is now used here as a case study. This section aims at evaluating how the personalised SNN models, trained on the relevant EEG samples (selected using the proposed DWWKNN algorithm), can perform a better accuracy of results when compared with the SNN models trained on a global space of EEG data (reported in Chapter 4).

This EEG data consists of 67 samples, in which 20 samples are labelled as healthy—H (class 1), 29 samples are labelled as patients undertaking methadone maintenance treatment—MMT (class 2) and 18 samples are labelled as opiate addict patients—OP (class 3). The EEG data were recorded *via* 26 EEG channels. This EEG data has a slightly different number of samples when compared with the data used in Chapter 4. This is because static data were not available for all the participants, so in this chapter I used only those subjects who had both static clinical data and EEG data.

In addition to the spatio-temporal EEG data, personal clinical, static information was also available per subject, such as: gender, age, drug consumption, methadone dose, history of

overdose, and so forth. In total, there were 20 variables for measuring static information from each subject. Table 7-2 shows only 15 randomly selected samples of the static data among all the 67 samples. However, all the 67 samples of the static data space were used in this experiment.

Table 7-2 Five samples are randomly selected from each groups of subjects (in total 15 samples). V1: age; V2: gender (0 is male and 1 is female); V3: level of education; V4: life time nicotine consumption; V5: illness; V6: history of overdose; V7: times of hospitalised; V8: Legal charge; V9: days being in jail; V10: Methadone dose; V11: alcohol consumption in last 30 days; V12: sedative consumption in last 30 days; V13: level of anger; V14: cannabis consumption; V15: hallucinogens consumption; V16: taking ecstasy; V17: amphetamine consumption; V18: barbiturate consumption; V19: heroin; and V20: class label of subject groups (1 is H, 2 is MMT and 3 is OP).

V1	V2	V3	V	V5	V6	V7	V8	V9	V10	V11	V12	V13	V14	V15	V16	V17	V18	V19	V20
37.9	1	12	21	0	0	2	0	0	0	0	0	75	0	0	0	0	0	0	1
36.7	1	13	0	0	0	0	0	0	0	0	0	88	0	0	0	1	0	0	1
45.3	0	16	5	0	0	1	0	0	0	5	0	75	20	0	0	7	0	2	1
41.3	0	12	0	0	0	1	0	0	0	0	0	50	0	0	0	1	0	0	1
34.3	1	13	25	0	0	2	0	0	0	0	0	63	11	0	12	0	0	15	1
41.8	0	13	0	0	0	2	2	0	52	0	30	88	16	0	0	15	3	7	2
39.6	1	12	0	0	3	1	10	48	22.5	0	10	50	20	0	5	13	1	0	2
31	0	13	13	1	0	0	1	0	90	0	0	50	0	2	0	0	0	10	2
39.8	0	11	24	0	4	4	40	64	120	0	30	63	4	15	0	5	1	13	2
40.3	1	11	30	0	0	30	1	0	90	0	0	50	0	4	12	0	0	0	2
38.5	0	11	0	0	0	1	1	24	0	0	0	38	0	0	0	0	0	8	3
45.7	0	13	0	0	1	3	1	0	0	0	0	75	0	3	3	0	0	0	3
28.2	0	16	6	0	1	2	3	0	0	0	10	63	0	5	0	10	0	0	3
39.2	1	13	0	0	2	0	1	0	0	30	0	63	0	0	0	0	0	0	3
38.5	1	13	13	1	1	2	6	0	0	0	0	88	6	0	0	0	0	0	3

The SNR values for these 20 variables of static data (67 instances) were computed and reported in Figure 7-2. It represents that variable 2 (variable gender) has obtained the highest importance for discriminating the samples across the mentioned three classes.

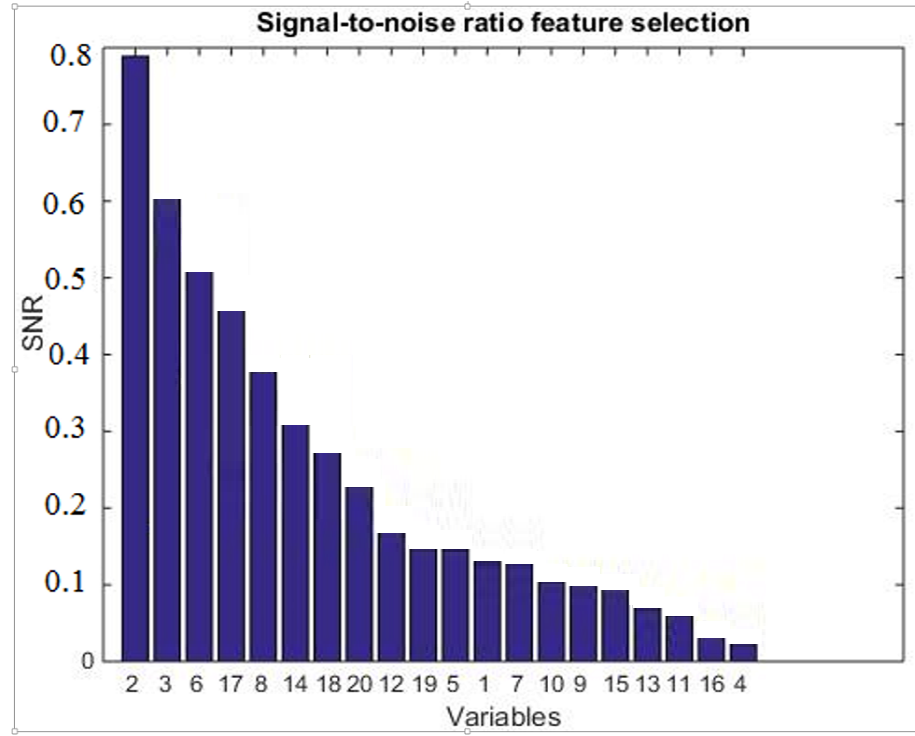


Figure 7-2 The SNR values for the 20 variables of static data.

Figure 7-3 illustrates the user interface of personalised modelling that I developed in MATLAB as part of NeuCube software. In order to make an application of the proposed personalised modelling, the following steps have been performed:

- a) The dynamic EEG data and static vector-based data were loaded into the personalised modelling module. Subjects are labelled by an ID: 1 to 20 are from H group; 21 to 49 are from MMT group; and 50 to 67 are from OP group.
- b) A subject was randomly selected (here it is ID: 4 from H group) for creating a personalised model.
- c) Static data samples of each class were ranked according to Equation (7-1). The distance D in Equation (7-2) was based on only the distance between the static data vector of sample ID: 4 and the other 66 sample vectors. As I am interested to observe how static

data of sample ID: 4 is similar to the static data vectors of the other samples, I computed $D = 1 - D$ as a similarity measure, then it was normalised and plotted in Figure 7-3 (top bar lines). It shows a high similarity between sample ID: 4 and sample ID: 49 (from MMT group).

- d) Employing Equation (7-3), the STBD samples of each class were also ranked with respect to the correlation coefficient between EEG data of sample ID: 4 and the other 66 samples. Figure 7-3 (bottom bar-lines) shows that the EEG data of ID: 4 has a high positive correlation with sample ID: 29 (from MMT group) and a high negative correlation with ID: 49 (from MMT group).
- e) In order to extract a relevant subset of samples to sample ID: 4 with respect to both static data and STBD, two clusters were formed, one on the static data space and the other on the STBD space.

e.1. In the static data, samples with close D values to ID: 4 (greater than threshold $t = \mu D_i, i = 1, \dots, 66$) were grouped as cluster C_s . In this example $t = 0.68$ for subject ID: 4.

e.2. In STBD, a cluster of EEG data samples were formed in which samples were highly correlated to the EEG data of ID: 4. Samples were selected if their correlation values ($Corr$) were greater than a threshold value $T = \mu Corr_i, i = 1, \dots, 66$, meaning a high positive correlation. As there were three classes in this dataset, I have computed three different thresholds per class: $T_1 = 0.39, T_2 = 0.46$, and $T_3 = 0.41$, to select three sub-clusters which each was constituted of samples with correlation above the corresponding T . Then the sub-clusters are merged as one, called *cluster* C_{STBD} . This procedure handles the issue of imbalanced classes by selecting a close number of

samples from each class.

- f) The relevant data samples to sample ID: 4 were selected by finding the common samples (intersection) between two clusters C_s and C_{STBD} . The selected samples were the nearest samples to ID: 4, pertaining to integrated static and dynamic information. In this example, 18 nearest neighbour samples to ID: 4 were selected to build a PSNN model for profiling subject ID: 4.
- g) Finally, the EEG data of these 18 samples were used to train the built PSNN model through STDP unsupervised learning.

The 18 EEG samples were transferred to the NeuCube module 1, where a 3-D PSNN model was spatially mapped using the Talairach template. Then, 26 input neurons were allocated to the 26 EEG channels. The PSNN model was trained using STDP learning. Then the model was tested by EEG data of sample ID: 4, which was excluded from training.

In this experiment, for every subject in dataset, one PSNN model was created and trained with the most informative EEG samples corresponding to subjects who have similar integrated clinical static information and dynamic EEG data to the selected subject. Figure 7-4, Figure 7-5 and Figure 7-6 show the ranking of samples in the data space for six randomly selected subjects belonging to different classes (class 1: H subjects, class 2: MMT subjects, and class 3: OP subjects), where the green highlighted bar lines represent the K nearest neighbour samples to the selected ones (ID: 1 and 15 from H group, 45 and 31 from MMT group, 58, and 61 from OP group) obtained using DWWKNN.

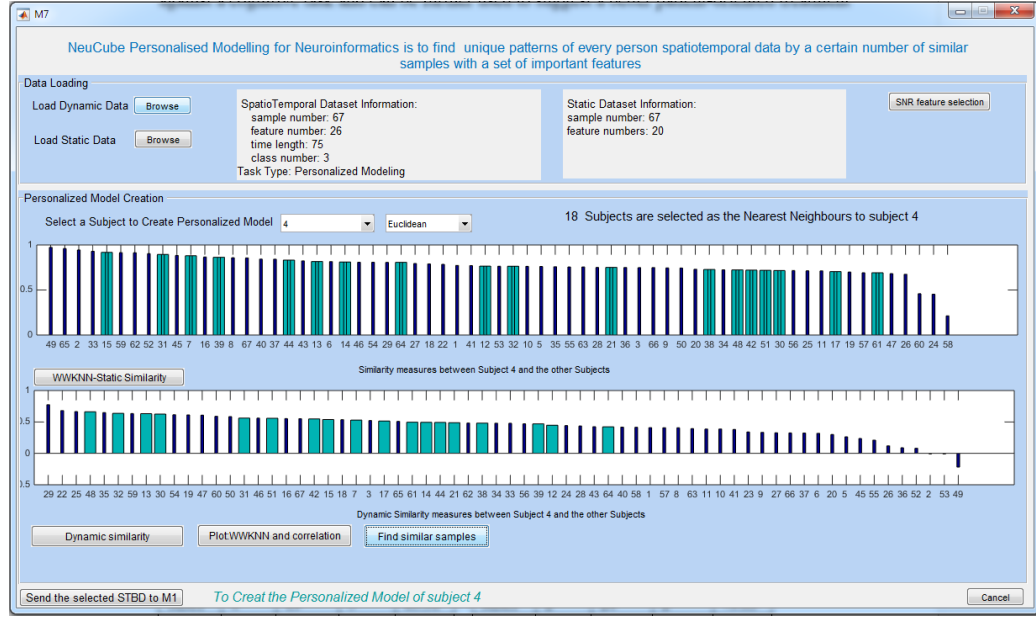


Figure 7-3 The proposed personalised modelling user interface for integrated static-dynamic data, exemplified using a case study of EEG data and static clinical data. For a selected subject id: 4, the relevant samples to it is a cluster of the common samples between C_s and C_{STBD} (green bar lines) defined using the DWKNN.



Figure 7-4 Data samples were ranked according to the integrated static-dynamic similarity to a new data from two H samples id: 1 and id: 15 in (a) and (b) respectively. The EEG data of the neighbour samples (shown in green) are used for training of the PSNN models.



Figure 7-5 Data samples were ranked according to the integrated static-dynamic similarity to the new data from two MMT samples id: 45 and id: 31 in (a) and (b). The EEG data of the neighbour samples (shown in green) are used for training of the PSNN models.



Figure 7-6 Data samples were ranked according to the integrated static-dynamic similarity to the new data from two OP samples id: 58 and id: 61 in (a) and (b). The EEG data of the neighbour samples (shown in green) are used for training of the PSNN models.

The informative EEG data samples were used to create 6 separate PSNN models, each represents the spatio-temporal interaction between EEG channels of an individual person. The EEG samples were first encoded into spike trains and then transferred into a PSNN model for STDP unsupervised learning. During the learning, the connections between neurons of the PSNN model were strengthened or weakened based on the timing of postsynaptic in relation to the presynaptic spikes. Figure 7-7 illustrates six PSNN models, each trained on a subset of informative EEG samples selected through DWWKNN as shown in Figure 7-4 to Figure 7-6.

These results represent that the trained PSNN models captured different spatio-temporal connectivity across subjects, even though they belonged to the same class. For instance, the trained PSNN model of MMT ID: 45 in Figure 7-7 (d) illustrates stronger average connection weights (1.12) than the trained PSNN model of MMT, ID: 31 in Figure 7-7 (c) where the average connection weight is 0.98. These findings can be scrutinised to identify what are the differences in the PSNN models personalised for these two MMT individuals. Subsequently, they can be used to implicate how different individuals in the MMT group have differently responded to the Methadone treatment.

The personalised modelling was performed for all the 67 samples by creating 67 separate PSNN models, each trained on a subset of informative EEG data selected *via* the DWWKNN. After the training process was completed for each PSNN model, the model was tested using EEG data of the sample which the PSNN model was built for.

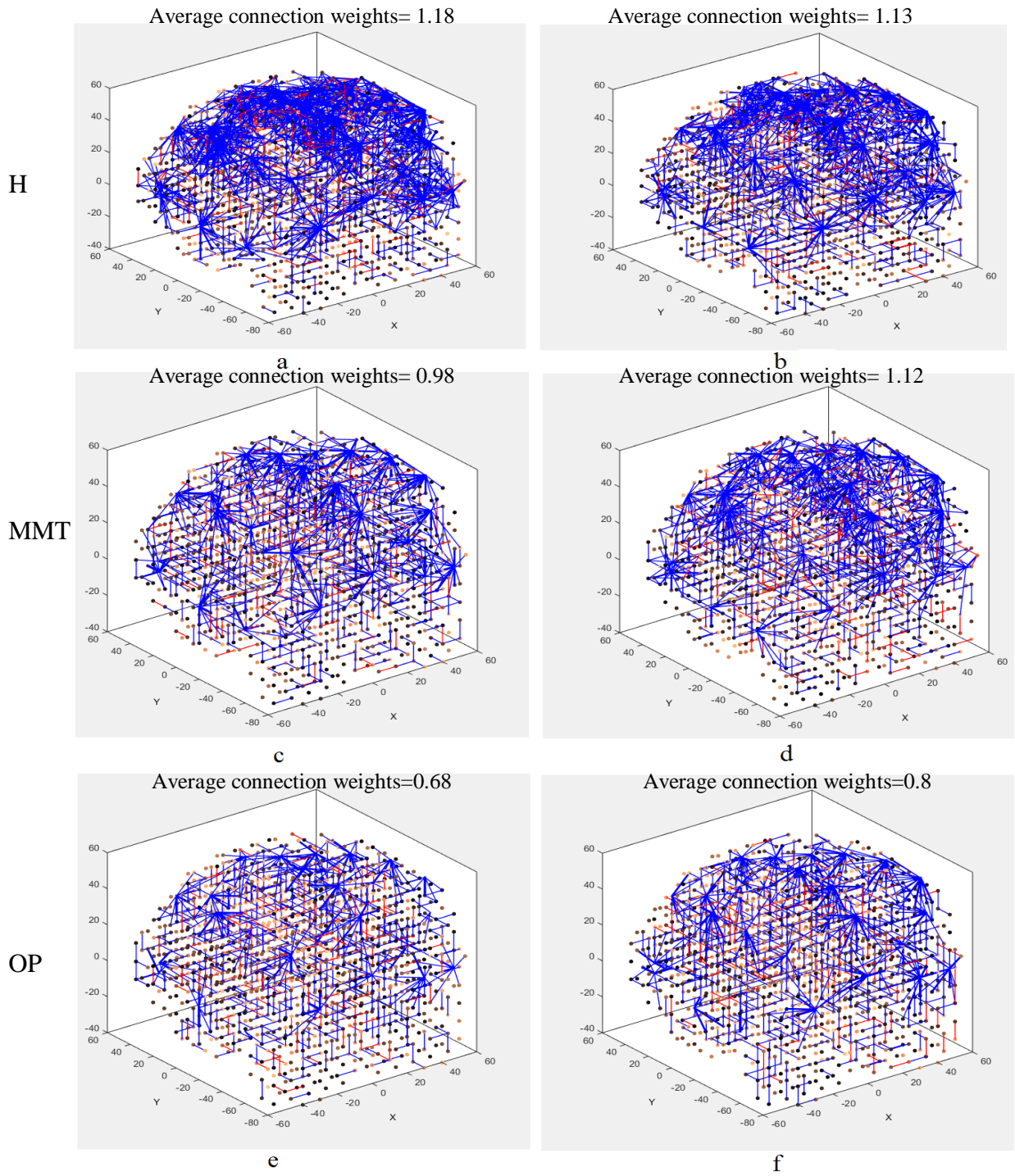


Figure 7-7 The PSNN models were created for 6 randomly selected persons from H (a-b), MMT (c-d), and OP (e-f) groups. Each PSNN model was trained by the closest samples to the corresponding person.

Table 7-3 shows the overall classification accuracy of all 67 PSNN models *versus* a global SNN model, which was obtained using the LOOCV method. The classification accuracy was also compared with two conventional methods SVM and MLP. In Table 2, the proposed DWWKNN clustering approach was compared with different clustering methods (WKNN, WKNN, and KNN) towards building PSNN models for classification. Unlike the proposed DWWKNN, the employed rival methods used here do not include the EEG data correlation into their computation.

Table 7-3 Classification accuracy obtained *via* SNN-based personalised modelling *versus* using a global SNN model. The number of correctly classified samples in each class is shown on the diagonal of the tables. For each person x , one PSNN model is trained by EEG of subjects who have similar integrated static and dynamic data to x and then tested by the EEG data of x , which was unseen during the learning.

One SNN model tested <i>via</i> a leave-one-out method					Personalised modelling of 67 trained SNN models for each subject's data				
Predicted \ Actual	H	M	OP	Accuracy in %	Predicted \ Actual	H	M	OP	Accuracy in %
H	16	3	1	80.00	H	18	2	0	90.00
MMT	4	19	6	65.51	MMT	3	22	4	75.86
OP	2	6	10	55.50	OP	1	4	13	72.22
Overall accuracy (average)				67.00	Overall accuracy (average)				79.36

Table 7-4 Comparison of the classification accuracy (in %) obtained using a global SNN, PSNN and conventional methods including: SVM and MLP. The MLP optimal parameters that resulted the best classification accuracy were found after performing the experiments several times with different parameter setting (learning rate (LR) = [0.01, 0.5], momentum (M) = [0.1, 0.9], training iteration (TI) = [500, 1500], and number of hidden layer (HL) = [2, 6]).

Method	SVM	MLP	NeuCube SNN	NeuCube PSNN
Accuracy	60.00	61.00	67.00	79.36
Parameters	Polynomial kernel degree=3	LR=0.1, M=0.6 TI=1000, HL=3	LR=0.002, Mod= 0.5, Drift=0.02	

Table 7-5 Classification accuracy obtained using the PSNN models with different clustering approaches (DWWKNN, WWKNN, WKNN and KNN) for selecting the nearest neighbour samples to an individual.

PSNN with different clustering methods	dWWKNN	WWKNN	WKNN	KNN
Accuracy	79.36	74.00	72.00	70.00

7.4 Chapter Summary

In this chapter, I proposed a personalised modelling approach based on SNN for integrated static clinical data and dynamic STBD for individuals. The proposed approach has contributed to the NeuCube architecture through introducing new methods for selecting an appropriate size of neighbour samples to build personalised models. A new method, called Dynamic Weighted-Weighted Distance K-nearest Neighbours (DWWKNN) has been introduced as a new method to select the most relevant samples with respect to both static data and spatio-temporal information.

As deduced from PSNN model visualisations, different patterns of connectivity have been formed across individuals belonging to the same class of data. Therefore, PSNN models not only can distinguish samples with respect to their class labels with a high classification accuracy, but they can also be used for a better identification of interactions between spatio-temporal variables in each individual subject's data.

7.5 Contribution

In this chapter, I have made the following original contributions:

1. Proposal of a new personalised modelling system based on the SNN architecture (called PSNN).
2. Proposal of a new clustering approach (named DWWKNN) for integrated static and temporal/spatio-temporal data.
3. Empirical study on the proposed personalised modelling based on EEG data and static clinical health data.
4. I published this study in a conference paper as the leading author.

Doborjeh, M. G., & Kasabov, N. (2016, July). Personalised modelling on integrated clinical and EEG spatio-temporal brain data in the NeuCube spiking neural network system. In *IEEE International Joint Conference on Neural Networks (IJCNN)*, 2016 (pp. 1373-1378).

Chapter 8 Conclusions and Recommendations for Future Work

8.1 Introduction

This chapter discusses the key contributions and achievements of this thesis and further articulates how the research questions, posed in Chapter 1, have been addressed. Some main limitations of this work are then discussed along with an overview of future implications.

8.2 Aims and Methodological Approach

An overview on ANN pointed out that SNNs resemble the activity mechanism in the human brain, due to their ability in encoding *time* information into the computation. Hence, this thesis is based on SNN architecture for modelling, classifying, clustering and a better interpretation of STBD.

The main objectives in this thesis were addressed in two steps: first feasibility analysis and empirical study step, and second, method development step. The first step was to design optimal SNN architectures for modelling different types of STBD (EEG and fMRI). The results obtained in step one implicated that the evolving learning patterns in recurrent SNN models are complex to interpret, due to the composition of spatial information of neurons, which is learnt, in an evolving manner, from temporal data streams over time. Hence, to study such evolving patterns, I proposed a new spatio-temporal clustering approach that resulted in knowledge discovery from SNN architectures.

Additionally, I proposed a new personalised modelling system based on SNN architecture that offers improved personal outcome classification, personalised interpretation and identification of important factors for a person across various types of data, including both static and dynamic (temporal) types.

8.3 Key Findings

The progression of this thesis resulted in the following findings which improved an understanding of SNN patterns triggered by multivariate STBD.

8.3.1 Optimal SNN to Model, Learn and Analyse STBD

When applied to real-life case studies, such as multivariate STBD, the designed SNN models enhanced the classification performance by achieving up to 92% accuracy, which represents an average improvement of 20% when compared with conventional machine learning methods. Compared to conventional classifiers, the designed SNN models of EEG data showed a superior classification accuracy by up to 92% between different cognitive processes performed by different groups of human subjects (opiate addict subjects, a group of addict subjects undertaking treatment, and a group of healthy subjects). In addition to this, the trained SNN models developed significantly different spatio-temporal neural connections illustrated in a 3-D brain-inspired space for different groups of subjects. A t-test measure ($p\text{-value}=0.009<0.05$) confirmed that the trained SNN models were statistically significant with a high confidence, greater than 99%. This allowed for a better interpretation of the spatio-temporal interactions between variables when compared with black-box conventional classifiers and statistical methods such as SVM, MLP and MLR.

I also designed optimal SNN models for analysing benchmark fMRI data. The SNN models have exposed a better discrimination across different mental activities (reading affirmative *versus* negative sentences) than conventional methods.

8.3.2 Knowledge Discovery in SNN Learning Patterns

The trained SNN models resulted in a higher classification accuracy when compared with conventional methods. In addition to this, the model interpretability was also higher than the state-of-the-art machine learning techniques. This has been achieved with proposal of a new spatio-temporal clustering in SNN models while streaming STBD for unsupervised learning. Compared to the current evolving clustering methods, such as ESOM (Deng & Kasabov, 2000), DENFIS (Kasabov & Song, 2002), dynamic evolving clustering (Aggarwal, 2003) that successfully detect the temporal changes in data streams, the proposed (in this thesis) dynamic spatio-temporal clustering method considers both spatial and temporal information together in an SNN model and dynamically clusters the evolving learning patterns over time. An assessment of these spatio-temporal clustering patterns has led to the detection of important discriminative features in the SNN models. Hence, using only the selected informative features for a classification task, an average of 10% increase in accuracy has been achieved. In addition, it revealed a trajectory of sequential activated neural areas in the SNN models, reflecting the importance of the STBD variables with respect to the time at which these clusters were created.

8.3.3 Improved Personalised Modelling using SNN Architecture

I further developed this research towards the proposal of a new personalised modelling system that has contributed to the NeuCube architecture by introducing a new method for selecting an appropriate size of neighbour samples to build personalised models. To this end,

I introduced the DWWKNN as a new method to select the relevant samples with respect to both static data and spatio-temporal information. This approach enhanced the classification accuracy by 12% when compared with global SNN models.

8.4 Empirical and Theoretical Contributions

The findings of this thesis contributed to new knowledge about SNN models of STBD and carried out implications regarding the following aspects including improved extant reservoir computing systems; knowledge discovery in recurrent SNN models; and improved outcome prediction/classification by personalised profiling.

A. Improved Extant Reservoir Computing Systems

The designed SNN models for STBD analysis implicated several advantages when compared with recently developed reservoir computing methods. Compared to LSM, the designed SNN models in this thesis have brain-inspired structure owing to the following criteria:

1. The STBD variables are topologically mapped into a 3-D SNN model which has a brain template (atlas). Spatial mapping of input features (data variables) in the SNN model preserves the spatial information in the brain data variables.
2. Every artificial spiking neuron in an SNN model is a computational unit that resembles a biological neuron model, encoding the timing of spikes.
3. Input STBD are encoded by spikes, emphasising certain changes in the brain data (signals) at a millisecond time scale.
4. Unlike the LSM that initialises random connectivity, in this thesis the SNN connections are initialised with respect to the small-world connectivity.

5. The initialised connections will be adapted using the STDP learning rule, which resembles the brain synaptic plasticity that enables learning and memorising.

B. Knowledge Discovery in Recurrent SNN Models

The proposed spatio-temporal clustering approach allowed to scrutinise the learning evolving patterns in recurrent SNN models. The findings demonstrated that SNN models are not acting as black-box information processing systems which solve a problem without discovering the meaningful interactive patterns that triggered a prediction/classification output. An assessment of these dynamic clustering patterns represented the importance of different areas of neurons that can be used to detect abstractions from SNN models for a further development of deep-learning in SNN architecture. Therefore, the achieved knowledge discovery in SNN models is a significant contribution to machine learning and AI.

C. Improved Personalised Profiling

In this thesis, the proposed personalised modelling system implicated that using an informative subset of samples as training set can result in a better differentiation across individuals, although they may belong to the same class of data. The proposed personalised modelling considers both static and spatio-temporal data to create a personalised SNN model, which resulted in an increased classification accuracy for an individual person. This approach contributed to creating a profile for each person with optimised output classification when compared with a global SNN modelling.

8.5 Limitations of the Thesis

The limitations of this thesis are:

8.5.1 Scope and Parameters of the Research

The EEG data used in the study were small, so the trained models can only be valid on the defined data scope and this is not yet generalised to be tested on any new person's STBD stream. The SNN models of fMRI data belonged to only one human subject and the results were specified to this person rather than being generalised implications for different subjects with different gender. Therefore, the optimised parameters for the designed SNN models were the best parameters for the defined domains. In addition, the data used in this thesis were only STBD, however, the proposed SNN-based methods need to be generalised for other types of spatio-temporal data, including environmental data, seismic data, and so forth.

8.5.2 Methodological Point of View

The encoding procedure to transfer the STBD into spike trains was a threshold contrast method as explained in Chapter 3, Section 3.6.2. However, the encoding method and its parameters should be optimised towards minimising the brain signal reconstruction error. Assessment of different encoding methods and optimising their parameters may lead to choosing another technique to encode EEG or fMRI.

8.6 Future Direction and Implications

Besides the remarks that have been made hitherto, there are a number of avenues that could be explored in the area in the future as follows:

SNN parameter optimisations: The optimisation method used in this thesis was an exhaustive grid-search on a combination of parameters. Each parameter was searched within a specified range. The optimisation procedure needs to be further improved to consider all the possible ranges of the SNN parameters.

Knowledge representation and knowledge transfer: The spatio-temporal connectivity developed in different SNN models need to be transformable to make them communicate with each other. If a trained SNN model can exchange information with another trained SNN model, the concept of “knowledge transfer” between machines, and between machines and humans can be applicable.

Early prediction of evolutionally patterns in SNN models: For further development of the proposed clustering approach presented in Chapter 6, I aim to enhance it towards early prediction of the learning patterns during unsupervised learning in SNN model. To this aim, the dynamics of the SNN clusters need to be mathematically modelled by differential equations. Consequently, using only a temporal chunk of STBD, the next sequential activated areas in the SNN models can be predicted by the clustering patterns. This method needs to be also generalised for other types of spatio-temporal data, including environmental data, seismic data, and so forth.

Deep structured SNN: Using the proposed clustering method, the most informative clusters of neurons were detected. These clusters can be seen as abstractions in a deep structured SNN model that transfer informative trained patterns to the next layer, which is here the deSNN classifier. The pattern of a cluster evolution captures the size of the cluster and the time of the cluster creation as two vital measures that reflect the importance of the cluster

centres (input variables). For detecting the abstractions in SNN models, I will rank the neurons according to the importance of the cluster that each neuron belongs to.

Bio and health informatics: For further development of the proposed personalised modelling system presented in Chapter 7, a generic predictive system for early prediction of health risk factors (neurological events, stroke, seizure, and so forth) will be proposed according to the following system developments:

- Personalised modelling for the prediction of the risk of stroke using static data of patients (that have had a stroke) and temporal environmental data.
- Personalised modelling for the prediction of the progression of Mild Cognitive Impairment (MCI) to Dementia or Alzheimer disease (AD).
- E-Health software development for the prediction of risk factors along with the detection of causal and temporal interactions between reasons, expressed as data variables.

References

- Abbott, L. F. (1999). Lapicque's Introduction of the Integrate-and-Fire Model Neuron (1907). *Brain Research Bulletin*, 50(5), 303-304.
- Aggarwal, C. (2003). A Framework for Diagnosing Changes in Evolving Data. *SIGMOD International Conference on Management of Data* (pp. 575-586). ACM.
- Aggioni, E., Tana, M. G., Arrigoni, F., Ucca, C., & Bianchi, A. M. (2014). Constructing fMRI Connectivity Networks: A Whole Brain Functional Parcellation Method for Node Definition. *Journal of Neuroscience Methods*, 228, 86-99.
- Ahmadi, A., Pearlson, G. D., Meda, S. A., Dager, A., Potenza, M. N., Rosen, R., . . . Tennen, H. (2013). Influence of Alcohol Use on Neural Response to GO/NO-GO Task in College Drinkers. *Neuropsychopharmacology*, 38(11), 2197-2208.
- American Psychiatric Association. (2000). Diagnostic and Statistical Manual of Mental Disorders-IV-TR. *Washington DC: American Psychiatric Association*.
- Anderson, C. W., Stolz, E. A., & Shamsunder, S. (1998). Multivariate Autoregressive Models for Classification of Spontaneous Electroencephalographic Signals During Mental Tasks. *IEEE Transactions on Biomedical Engineering*, 45(3), 277-286.

- Angelov, P., Filev, D., & Kasabov, N. (2010). *Evolving Intelligent Systems: Methodology and Applications* (Vol. 12). John Wiley & Sons.
- Argunsah, A., & Cetin, M. (2010). A Brain–Computer Interface Algorithm based on Hidden Markov Models and Dimensionality Reduction. In *Signal Processing and Communications Applications Conference (SIU)* (pp. 93–96). IEEE.
- Azevedo, F. A., Carvalho, L. R., Grinberg, L. T., & Farfel, J. (2009). Equal Numbers of Neuronal and Nonneuronal Cells Make the Human Brain an Isometrically Scaled-up Primate Brain. *Journal of Comparative Neurology*, 515(5), 532-541.
- Baig, M., Ayaz, Y., Gillani, S., Jamil, M., & Naveed, M. (2015). Motor Imagery based EEG Signal Classification using Self Organizing Maps. *Science International* , 27(2), 1165-1170.
- Beckmann, C. F., Jenkinson, M., & Smith, S. M. (2003). General Multilevel Linear Modeling for Group Analysis in fMRI. *Neuroimage*, 20(2), 1052-1063.
- Behroozi, M., & Daliri, M. (2014). RDLPFC Area of the Brain Encodes Sentence Polarity: A Study Using fMRI. *Brain Imaging and Behavior*, 9(2), 178-189.
- Benuskova, L., & Kasabov, N. (2007). *Computational Neurogenetic Modeling*. Springer.
- Bezdek, J., Ehrlich, R., & Full, W. (1984). FCM: The Fuzzy C-Means Clustering Algorithm. *Computers and Geosciences*, 10(2-3), 191-203.
- Bifet, A., Holmes, G., Kirkby, R., & Pfahringer, B. (2010). Moa: Massive Online Analysis. *Journal of Machine Learning Research*, 11(May), 1601-1604.

- Bohte, S. M. (2004). The Evidence for Neural Information Processing with Precise Spike-Times: A Survey. *Natural Computing*, 3(2), 195-206.
- Bohte, S., Kok, J. N., & La Poutre, H. (2002). Error-Backpropagation in Temporally Encoded Networks of Spiking Neurons. *Neurocomputing*, 48(1), 17-37.
- Brady, S., Siegel, G., Albers, R., & Price, D. (2011). *Basic Neurochemistry: Principles of Molecular, Cellular, and Medical Neurobiology*. Academic Press.
- Braitenberg, V., & Schuz, A. (1998). *Cortex: Statistics and Geometry of Neuronal Connectivity*. Springer Berlin.
- Brett, M., Christoff, K., Cusack, R., & Lancaster, J. (2001). Using the Talairach Atlas with the MNI Template. *Neuroimage*, 13(6), 85-85.
- Brette, R., Rudolph, M., Carnevale, T., Hines, M., Beeman, D., Bower, J. M., . . . Frederick, C. (2007). Simulation of Networks of Spiking Neurons: A Review of Tools and Strategies. *Journal of Computational Neuroscience*, 23(3), 349-398.
- Brunel, N., van, R., & Mark, C. (2007). Quantitative Investigations of Electrical Nerve Excitation Treated as Polarization. *Biological Cybernetics*, 97(5), 341-349.
- Buchel, C., & Friston, K. (1997). Effective Connectivity and Neuroimaging. *SPM-Short Course Notes*, 1-18.
- Bullmore, E., & Sporns, O. (2009). Complex Brain Networks: Graph Theoretical Analysis of Structural and Functional Systems. *Nature Reviews Neuroscience*, 10(3), 186-198.

- Capecchi, E., Doborjeh, Z., Mammone, N., La Foresta, F., Morabito, F., & Kasabov, N. (2016). Longitudinal Study of Alzheimer's Disease Degeneration Through EEG Data Analysis with a NeuCube Spiking Neural Network Model. *International Joint Conference on Neural Networks (IJCNN)* (pp. 1360-1366). IEEE.
- Capecchi, E., Morabito, F., Campolo, M., Mammone, N., Labate, D., & Kasabov, N. (2015). A Feasibility Study of using the Neucube Spiking Neural Network Architecture for Modelling Alzheimer's Disease EEG Data. In *Advances in Neural Networks: Computational and Theoretical Issues* (pp. 159-172). Springer.
- Carpenter, G., & Grossberg, S. (1987). A Massively Parallel Architecture for a Self-Organizing Neural Pattern Recognition Machine. *Computer Vision, Graphics, and Image Processing*, 37(1), 54-115.
- Caton, R. (1875). Electrical Currents of the Brain. *The Journal of Nervous and Mental Disease*, 2(4), 610.
- Caviness, J., Verne, S., Meyer, J., Makris, N., & Kennedy, D. (1996). MRI-Based Topographic Parcellation of Human Neocortex: an Anatomically Specified Method with Estimate of Reliability. *Journal of Cognitive Neuroscience*, 8(6), 566-587.
- Chakraborty, C., & Talukdar, P. (2016). Issues and Limitations of HMM in Speech Processing: A Survey. *International Journal of Computer Applications*, 141(7), 0975-8887.

- Chatterjee, R., & Bandyopadhyay, T. (2016). EEG Based Motor Imagery Classification using SVM and MLP. *International Conference on Computational Intelligence and Networks* (pp. 84-89). IEEE.
- Che, Z., Purushotham, S., Cho, K., Sontag, D., & Liu, Y. (2018). Recurrent Neural Networks for Multivariate Time Series with Missing Values. *Nature Scientific Reports*, 8(1), 6085.
- Christensen, R. K. (2009). Negative and Affirmative Sentences Increase Activation in Different Areas in the Brain. *Journal of Neurolinguistics*, 22(1), 1-17.
- Cilimkovic, M. (2015). Neural Networks and Back Propagation Algorithm. *Institute of Technology Blanchardstown, Blanchardstown Road North Dublin, 15*.
- Cortes, C., & Vapnik, V. (1995). Support-Vector Networks. *Machine Learning*, 20(3), 273-297.
- Costantini, G., Todisco, M., Casali, D., Carota, M., Saggio, G., Bianchi, L., . . . Quitadamo, L. (2009). SVM Classification of EEG Signals for Brain Computer Interface. *Frontiers in Artificial Intelligence and Applications*, 229-233. doi:10.3233/978-1-60750-072-8-229
- Cristianini, N., & Shawe-Taylor, J. (2000). *An Introduction to Support Vector Machines and Other Kernel-Based Learning Methods*. Cambridge University Press.
- Danion, J., eingartner, H., File, S. E., Jaffard, R., Sunderland, T., Tulving, E., & Warburton, D. (1993). Pharmacology of Human Memory and Cognition: Illustrations from the

- Effects of Benzodiazepines and Cholinergic Drugs. *Journal of Psychopharmacology*, 7(4), 371-377.
- Deboeck, G., & Kohonen, T. (2013). *Visual Explorations in Finance: with Self-Organizing Maps*. Springer Science & Business Media.
- Delbruck, T. (2007). *Jaer Open Source Project*. Retrieved 10 15, 2014, from <https://sourceforge.net/p/jaer/wiki/Home/>
- Delbruck, T., & Lichtsteiner, P. (2007). Fast Sensory Motor Control based on Event-Based Hybrid Neuromorphic-Procedural System. In *IEEE International Symposium on Circuits and Systems* (pp. 845-848). IEEE.
- Deng, D., & Kasabov, N. (2000). ESOM: An Algorithm to Evolve Self-Organizing Maps from Online Data Streams. *International Joint Conference on Neural Networks*, 6, pp. 3-8.
- Dhoble, K., Nuntalid, N., Indiveri, G., & Kasabov, N. (2012). Online Spatio-Temporal Pattern Recognition with Evolving Spiking Neural Networks Utilising Address Event Representation, Rank Order, and Temporal Spike Learning. *IEEE World Congress on Computational Intelligence*, (pp. 1-7). Brisbane, Australia.
- Eddy, S. R. (1996). Hidden Markov Models. In *Proceedings in Computational Statistics* (Vol. 6, pp. 361-365). Elsevier.

- Eddy, W., Fitzgerald, M., Genovese, C., Lazar, N., Mockus, A., & Welling, J. (1999). The Challenge of Functional Magnetic Resonance Imaging. *Journal of Computational and Graphical Statistics*, 8(3), 545-558.
- Eddy, W., Fitzgerald, M., Genovese, C., Mockus, A., & Noll, D. C. (1996). Functional Imaging Analysis Software—Computational Olio. (pp. 39-49). Physica-Verlag HD.
- Ester, M., Kriegel, H.-P., Sander, J., & Xu, X. (1996). A Density-Based Algorithm for Discovering Clusters in Large Spatial Databases with Noise. In *Knowledge Discovery and Data Mining* (Vol. 96, pp. 226-231).
- Eugene, I. M. (2004). Which Model to Use for Cortical Spiking Neurons. *IEEE Transactions on Neural Networks*, 15(5), 1063-1070.
- Franchin, T., Tana, M. G., Cannata, V., Cerutti, S., & Bianchi, A. M. (2013). Independent Component Analysis of EEG-fMRI Data for Studying Epilepsy and Epileptic Seizures. *International Conference of the IEEE Engineering in Medicine and Biology Society (EMBC)*, (pp. 6011-6014).
- Friston, K. (1994). Statistical Parametric Mapping. In M. Thatcher, T. A. Hallett, & E. R. Zeffiro, *Functional Neuroimaging: Technical Foundations* (pp. 79-93). San Diego: Academic Press. Retrieved from APA PsycNet: <http://psycnet.apa.org/record/1994-97892-008>
- Friston, K., Frith, C., Liddle, P., & Frackowiak, R. (1993). Functional Connectivity: The Principal-Component Analysis of Large (PET) Data Sets. *Journal of Cerebral Blood Flow and Metabolism*, 13(1), 5-14.

- Friston, K., Holmes, A. P., Worsley, K. J., Poline, J., Frith, C. D., & Frackowiak, R. S. (1994). Statistical Parametric Maps in Functional Imaging: A General Linear Approach. *Human Brain Mapping*, 2(4), 189-210.
- Fuchs, P., Diamond, M., Brown, D., Weisblat, D., Nicholls, J., & Martin, R. (2012). *From Neuron to Brain* (5 ed.). New York: Oxford University Press Inc.
- Furber, S. B., Galluppi, F., Temple, S., & Plana, L. A. (2014). The SpiNNaker Project. *Proceedings of the IEEE*, 102(5), 652-665.
- Gers, F., & Schmidhuber, E. (2001). LSTM Recurrent Networks Learn Simple Context-Free and Context-Sensitive Languages. *IEEE Transactions on Neural Networks*, 12(9), 1333-1340. doi:10.1109/72.963769
- Gerstner, W., & Iistler, W. M. (2002). *Spiking Neuron Models: Single Neurons, Populations, Plasticity* (Vol. 32). Cambridge University Press.
- Gevins, A., Ilan, A. B., Jiang, A., Sam-Vargas, L., Baum, C., & Chan, C. S. (2011). Combined Neuropsychological and Neurophysiological Assessment of Drug Effects on Groups and Individuals. *Journal of Psychopharmacology*, 25(8), 1062-1075.
- Gnana, S. K., & Subramaniam, D. (2013). Review on Methods to Fix Number of Hidden Neurons in Neural Networks. *Mathematical Problems in Engineering*, 2013(425740), 1-11.
- Goodfellow, I., Bengio, Y., & Courville, A. (2016). *Deep Learning*. MIT Press Cambridge.

- Gratton, G., Coles, M. G., & Donchin, E. (1983). A New Method for Off-Line Removal of Ocular Artifact. *Electroencephalography and Clinical Neurophysiology*, 55(4), 468-484.
- Guo, Y., Hastie, T., & Tibshirani, R. (2007). Regularized Linear Discriminant Analysis and its Application in Microarrays. *Biostatistics*, 8(1), 86-100.
- Haas, L. F. (2003). Hans Berger (1873-1941), Richard Caton (1842-1926), and electroencephalography. *Journal of Neurology, Neurosurgery and Psychiatry*, 74(1), 9.
- Hall, J. E. (2015). *Guyton and Hall Textbook of Medical Physiology e-Book*. Elsevier Health Sciences.
- Hamdoun, H., & Usman, A. (2016). EEG-Based Condition Clustering using Self-Organising Neural Network Map. *International Conference on Engineering Applications of Neural Networks* (pp. 147-157). Springer.
- Hartigan, J., & Wong, , M. (1979). Algorithm AS 136: A K-Means Clustering Algorithm. *Journal of the Royal Statistical Society, Series C (Applied Statistics)*, 28(1), 100-108.
- He, K., Hang, X., Ren, S., & Sun, J. (2016). Deep Residual Learning for Image Recognition. *IEEE Conference on Computer Vision and Pattern Recognition* (pp. 770-778). IEEE.
- Hebb, D. (1949). *The Organization of Behavior: A Neuropsychological Approach*. John Wiley & Sons.

- Hodgkin, A. L., Huxley, A. F., & Katz, B. (1952). Measurement of Current-Voltage Relations in the Membrane of the Giant Axon of Loligo. *The Journal of Physiology*, 116(4), 424-448.
- Huettel, S. A., Song, A. W., & McCarthy, G. (2004). *Functional Magnetic Resonance Imaging*. Sinauer Associates Sunderland.
- Humble, J., Denham, S., & Wennekers, T. (2012). Spatio-Temporal Pattern Recognizers Using Spiking Neurons and Spike-Timing-Dependent Plasticity. *Frontiers in Computational Neuroscience*, 6(84), 1-12.
- Indiveri, G., Linares-Barranco, B., Hamilton, T., Van Schaik, A., Etienne-Cummings, R., Delbruck, T., . . . Dudek, P. (2011). Neuromorphic Silicon Neuron Circuits. *Frontiers in Neuroscience*, 5(73), 1-23.
- Izhikevich, E. M. (2003). Simple Model of Spiking Neurons. *IEEE Transactions on Neural Networks*, 14(6), 1569-1572.
- Izhikevich, E. M. (2006). Polychronization: Computation with Spikes. *Neural Computation*, 18(2), 245-282.
- Izhikevich, E. M., & Edelman, G. M. (2008). Large-Scale Model of Mammalian Thalamocortical Systems. *Proceedings of the National Academy of Sciences*, 105(9), 3593-3598.
- Johnson, S. (1967). Hierarchical Clustering Schemes. *Psychometrika*, 32(3), 241-254.

- Joutsiniemi, S.-L., Kaski, S., & Larsen, T. (1995). Self-Organizing Map in Recognition of Topographic Patterns of EEG Spectra. *IEEE Transactions on Biomedical Engineering*, 42(11), 1062-1068.
- Ju, H., Jian-Xin, V., & Antonius, M. (2010). Classification of Musical Styles using Liquid State Machines. *International Joint Conference on Neural Networks (IJCNN)* (pp. 1-7). IEEE.
- Just, M., & Wang, W. (2001). *StarPlus fMRI Data*. Retrieved 05 16, 2018, from StarPlus fMRI Data: <http://www.cs.cmu.edu/afs/cs.cmu.edu/project/theo-81/www/>
- Kandel, E., Schwartz, J., Jessell, T., Siegelbaum, S., & Hudspeth, J. A. (2000). *Principles of Neural Science* (Vol. 4). New York: McGraw-Hill New York.
- Kasabov, N. (1998). Evolving Fuzzy Neural Networks - Algorithms, Applications and Biological Motivation. *World Scientific Methodologies for the Conception, Design and Application of Soft Computing*, 1, 271-274.
- Kasabov, N. (2007). *Evolving Connectionist Systems*. Springer.
- Kasabov, N. (2010). Neural Networks Letter: To Spike or Not to Spike: A Probabilistic Spiking Neuron Model. *Neural Networks*, 23(0893-6080), 16019.
- Kasabov, N. (2012b). Neucube Evospike Architecture for Spatio-Temporal Modelling and Pattern Recognition of Brain Signals. In *IAPR Workshop on Artificial Neural Networks in Pattern Recognition* (Vol. 7477, pp. 225-243). Springer.

- Kasabov, N. (2014). NeuCube: A Spiking Neural Network Architecture for Mapping, Learning and Understanding of Spatio-Temporal Brain Data. *Neural Networks*, 52, 62-76.
- Kasabov, N., & Capecci, E. (2015). Spiking Neural Network Methodology for Modelling, Classification and Understanding of EEG Spatio-Temporal Data Measuring Cognitive Processes. *Information Sciences*, 294 , 565-575.
- Kasabov, N., & Song, Q. (2002). DENFIS: Dynamic Evolving Neural-Fuzzy Inference System and its Application for Time-Series Prediction. *IEEE Transactions on Fuzzy Systems*, 10(2), 144-154.
- Kasabov, N., & Song, Q. (2002). Dynamic Evolving Neural-Fuzzy Inference System and its Application for Time-Series Prediction. *IEEE Transactions on Fuzzy Systems*, 10(2), 144-154.
- Kasabov, N., Dhoble, K., Nuntalid, N., & Indiveri, G. (2013). Dynamic Evolving Spiking Neural Networks for On-Line Spatio-and Spectro-Temporal Pattern Recognition. *Neural Networks*, 41, 188-201.
- Kasabov, N., Hu, J., Chen, Y., Scott, N., & Turkova, Y. (2013). Spatio-temporal EEG data classification in the NeuCube 3D SNN environment: methodology and examples. *International Conference on Neural Information Processing* (pp. 63-69). Springer.
- Kasabov, N., Scott, N., Tu, E., Marks, S., Sengupta, N., Capecci, E., . . . Doborjeh, M. (2016). Evolving Spatio-Temporal Data Machines Based on the NeuCube Neuromorphic

- Framework: Design Methodology and Selected Applications. *Neural Networks*, 78, 1-14.
- Katagiri, S., & Lee, C.-H. (1993). A New Hybrid Algorithm for Speech Recognition Based on HMM Segmentation and Learning Vector Quantization. *IEEE Transactions on Speech and Audio Processing*, 1(4), 421-430.
- Kessler, R. C., & Üstün, B. T. (2004). The World Mental Health (WMH) Survey Initiative Version of the World Health Organization (WHO) Composite International Diagnostic Interview (CIDI). *International Journal of Methods in Psychiatric Research*, 13(2), 93-121.
- Khanacademy. (2017, 06 30). *The Membrane Potential*. Retrieved from Khanacademy: <https://www.khanacademy.org/science/biology/human-biology/neuron-nervous-system/a/the-membrane-potential>
- Knight, B. W. (1972). Dynamics of Encoding in a Population of Neurons. *The Journal of General Physiology*, 59(6), 734-766.
- Knyazev, G. G. (2007). Motivation, Emotion, and their Inhibitory Control Mirrored in Brain Oscillations. *Neuroscience & Biobehavioral Reviews*, 31(3), 377-395.
- Koessler, L., Maillard, L., Benhadid, A., Vignal, J., Felblinger, J., Vespignani, H., & Braun, M. (2009). Automated Cortical Projection of EEG Sensors: Anatomical Correlation via the International 10-10 System. *Neuroimage*, 46(1), 64-72.
- Kohonen, T. (1998). Self-Organizing Maps. *Neurocomputing*, 21(1), 1-6.

- Krizhevsky, A., Sutskever, I., & Hinton, G. (2012). ImageNet Classification with Deep Convolutional Neural Networks. *Neural Information Processing Systems Conference*, (pp. 1097-1105).
- Kruse, R., Borgelt, C., Klawonn, F., Moewes, C., Steinbrecher, M., & Held, P. (2013). Multi-Layer Perceptrons. In *Computational Intelligence* (pp. 47-81). London: Springer.
- Liang-Chieh, C., George, P., Iasonas, K., Kevin, M., & Alan L, Y. (2018). DeepLab: Semantic Image Segmentation with Deep Convolutional Nets, Atrous Convolution, and Fully Connected CRFs. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 40(4), 834-848. doi:10.1109/TPAMI.2017.2699184
- Lichtsteiner, P., & Delbruck, T. (2005). A 64x64 AER Logarithmic Temporal Derivative Silicon Retina. *Research in Microelectronics and Electronics*, 2, 202-205.
- Lichtsteiner, P., Posch, C., & Delbruck, T. (2008). A 128x128 120dB 15 us Latency Asynchronous Temporal Contrast Vision Sensor. *IEEE Journal of Solid-State Circuits*, 43(2), 566-576.
- Lindquist, M. A. (2008). The Statistical Analysis of fMRI Data. *Statistical Science*, 23, 439-464.
- Liu, A., Chen, X., Wang, J. Z., Xu, Q., Appel-Cresswell, S., & McKeown, M. (2014). A Genetically Informed, Group fMRI Connectivity Modeling Approach: Application to Schizophrenia. *IEEE Transactions on Biomedical Engineering*, 61(3), 946-956.

- Liu, W., Wang, Z., Liu, X., Zeng, N., Liu, Y., & Alsaadi, F. (2017). A Survey of Deep Neural Network Architectures and their Applications. *Neurocomputing*, 234, 11-26.
- Logothetis, N., Pauls, J., Augath, M., Trinath, T., & Oeltermann, A. (2001). Neurophysiological Investigation of the Basis of the fMRI Signal. *Nature*, 412(6843), 150-157.
- Loiselle, S., Rouat, J., Pressnitzer, D., & horpe, S. (2005). Exploration of Rank Order Coding with Spiking Neural Networks for Speech Recognition. In *IEEE International Joint Conference on Neural Networks* (Vol. 4, pp. 2076-2080). IEEE.
- Loreta. (2017, 07 11). Retrieved from Loreta: <http://www.uzh.ch/keyinst/loreta>
- Lotte, F., Ongedo, M., Lecuyer, A., Lamarche, F., & Arnaldi, B. (2007). A Review of Classification Algorithms for EEG-Based Brain-Computer Interfaces. *Journal of Neural Engineering*, 4(2), R1.
- Lughofer, E., Bouchot, J., & Shaker, A. (2011). On-line Elimination of Local Redundancies in Evolving Fuzzy Systems. *Evolving Systems*, 2(3), 165-187.
- Lughofer, E. (2012). A Dynamic Split-and-Merge Approach for Evolving Cluster Models. *Evolving Systems*, 3(3), 135-151.
- Maass, W. (1997). Networks of Spiking Neurons: the Third Generation of Neural Network Models. *Neural Networks*, 10(9), 1659-1671.
- Maass, W. (2010). Liquid State Machines: Motivation, Theory, and Applications. *Computability in Context: Computation and Logic in the Real World*, 275-296.

- Maass, W., & Bishop, C. M. (2001). *Pulsed Neural Networks*. MIT press.
- Maass, W., Thomas, N., & Henry, M. (2002). Real-Time Computing without Stable States: A New Framework for Neural Computation based on Perturbations. *Neural Computation*, 14(11), 2531–2560.
- Manly, B. F., McDonald, L. L., Thomas, D. L., McDonald, T., & Erickson, W. (2002). Discriminant Function Analysis. *Resource Selection by Animals: Statistical Design and Analysis for Field Studies*, 171-178.
- Marcella. (2011). *Paul Offit is Right! Autism is About the Synaps and Dendritic Immunity*. Retrieved from VaxTruth: <http://vaxtruth.org/2011/10/offit-synapse/>
- Masquelier, T., Guyonneau, R., & Thorpe, S. (2009). Competitive STDP-based Spike Pattern Learning. *Neural Computation*, 21(5), 1259-1276.
- Maulik, U., & Bandyopadhyay, S. (2002). Performance Evaluation of Some Clustering Algorithms and Validity Indices. *IEEE Transactions on Pattern Analysis and Machine Intelligence*, 24(12), 1650-1654.
- McKeown, M. J., Makeig, S., Brown, G. G., Jung, T.-P., Kindermann, S. S., Bell, A. J., & Sejnowski, T. J. (1998). Analysis of fMRI Data by Blind Separation into Independent Spatial Components. *Human Brain Mapping*, 6(3), 160-188.
- Mirkin, B. (1998). Mathematical classification and clustering. In *Classification, Data analysis, and Data Highways* (pp. 172-181). Springer.

- Mitchell, T., Hutchinson, R., Niculescu, R., Pereira, F., Wang, X., Just, M., & Newman, S. (2004). Learning to Decode Cognitive States from Brain Images. *Machine Learning*, 57(1-2), 145-175.
- Mozer, M. C. (1993). Neural Net Architectures for Temporal Sequence Processing. In *Santa Fe Institute Studies in the Sciences of Complexity-Proceedings Volume* (Vol. 15, pp. 243-243). ADDISON-WESLEY.
- Murli, N., Kasabov, N., & Handaga, B. (2014). Classification of fMRI Data in the NeuCube Evolving Spiking Neural Network Architecture. *International Conference on Neural Information Processing* (pp. 421-428). Springer.
- Natschlager, T., & Maass, W. (2002). Spiking Neurons and the Induction of Finite State Machines. *Theoretical Computer Science*, 287(1), 251-265.
- Neuroscience, S. C. (2016). *Swartz Center for Computational Neuroscience* . Retrieved 09 21, 2016, from <https://scn.ucsd.edu/eeglab/>
- Niedermeyer, E., & da Silva, F. (2005). *Electroencephalography: Basic Principles, Clinical Applications, and Related Fields*. Lippincott Williams & Wilkins.
- Norman, K. A., Polyn, S. M., Detre, , G. J., & Haxby, J. V. (2006). Beyond Mind-Reading: Multi-Voxel Pattern Analysis of fMRI Data. *Trends in Cognitive Sciences*, 10(9), 424-430.

- Obermaier, B., Guger, C., Neuper, C., & Pfurtscheller, G. (2001). Hidden Markov Models for Online Classification of Single Trial EEG Data. *Pattern Recognition Letters*, 22(12), 1299-1309.
- Ogawa, S., Tank, D., Menon, R., Ellermann, J. M., Kim, S. G., Merkle, H., & Ugurbil, K. (1992). Intrinsic Signal Changes Accompanying Sensory Stimulation: Functional Brain Mapping with Magnetic Resonance Imaging. *Proceedings of the National Academy of Sciences*, 89(13), 5951-5955.
- Orjuela-Caon, A., Renteria-Meza, O., Hernandez, L., Ruiz-Olaya, A., Cerquera, A., & Antelis, J. (2017). Self-organizing Maps for Motor Tasks Recognition from Electrical Brain Signals. In *Iberoamerican Congress on Pattern Recognition* (pp. 458-465). Springer.
- Ou, J., Xie, L., Jin, C., Li, X., Zhu, D., Jiang, R., . . . Liu, T. (2015). Characterizing and Differentiating Brain State Dynamics via Hidden Markov Models. *Brain Topography*, 28(5), 666–679.
- Pape, L., De Gruijl, J., & Wiering, M. (2008). Democratic Liquid State Machines for Music Recognition. In *Speech, Audio, Image and Biomedical Signal Processing using Neural Networks* (pp. 191-215). Springer.
- Pearson, P. (2005). *Biological Science* (2 ed.). Retrieved from <https://www.psychologyinaction.org/psychology-in-action-1/2011/04/01/conventional-wisdom-upset-persistent-action-potential-firing-in-distal-axons>

- Peddinti, V., Povey, D., & Khudanpur, S. (2015). A Time Delay Neural Network Architecture for Efficient Modeling of Long Temporal Contexts. *Conference of the International Speech Communication Association*, (pp. 3214-3218). Dresden.
- Peltier, S., Lisinski, J., Noll, D., & LaConte, S. (2009). Support Vector Machine Classification of Complex fMRI Data. *Annual International Conference of the IEEE Engineering in Medicine and Biology Society* (pp. 5381-5384). IEEE. doi:10.1109/IEMBS.2009.5332805
- Prabhakar, S., & Rajaguru, H. (2015). PCA and K-Means Clustering for Classification of Epilepsy Risk Levels from EEG Signals? A Comparative Study Between Them. *International Conference on Intelligent Informatics and Biomedical Sciences (ICIIBMS)* (pp. 83-86). IEEE.
- PyEEG Reference Guide*. (2010, 08 31). Retrieved from PyEEG : <http://pyeeg.sourceforge.net/>
- Rabiner, L. R. (1989). A Tutorial on Hidden Markov Models and Selected Applications in Speech Recognition. *Proceedings of the IEEE*, 77(2), 257-286.
- Rademacher, J., Galaburda, A., Kennedy, D., Filipek, P., & Caviness, J. (1992). Human Cerebral Cortex: Localization, Parcellation, and Morphometry with Magnetic Resonance Imaging. *Journal of Cognitive Neuroscience*, 4(4), 352-374.
- Raghavendra, S., & Deka, P. (2014). Support Vector Machine Applications in the field of Hydrology: A Review. *Applied Soft Computing*, 19, 372-386.

- Rai, K., Bajaj, V., & Kumar, A. (2015). Novel Feature for Identification of Focal EEG Signals with K-Means and Fuzzy C-Means Algorithms. *International Conference on Digital Signal Processing (DSP)* (pp. 412-416). IEEE.
- Ranganathan, V., & Natarajan, S. (2018). A New Backpropagation Algorithm without Gradient Descent. *ArXiv Preprint ArXiv:1802.00027*, 1-15.
- Reggia, J., Ruppin, E., & Glanzman, D. (1999). *Disorders of Brain, Behavior, and Cognition: the Neurocomputational Perspective* (Vol. 121). Elsevier.
- Rezaei Tabar, Y., & Halici, U. (2016). A Novel Deep Learning Approach for Classification of EEG Motor Imagery Signals. *Journal of Neural Engineering*, 14(1), 016003.
- Rodriguez, P., Anderson, M., Calhoun, V. D., & Adali, T. (2015). General Nonunitary Constrained ICA and its Application to Complex-Valued fMRI Data. *IEEE Transactions on Biomedical Engineering*, 62(3), 922-929.
- Rosenblatt, F. (1957). The Perceptron, A Perceiving and Recognizing Automaton Project Para. Report: *Cornell Aeronautical Laboratory*, 85(460-461).
- Rousseeuw, P. J. (1987). Silhouettes: a Graphical Aid to the Interpretation and Validation of Cluster Analysis. *Computational and Applied Mathematics*, 20, 53-65.
doi:doi:10.1016/0377-0427(87)90125-7
- SCCN. (2017, 07 11). Retrieved from What is EEGLAB?:
<https://sccn.ucsd.edu/eeglab/index.php>

- Schliebs, S., & Kasabov, N. (2013). Evolving spiking neural network—a survey. *Evolving Systems*, 4(2), 87-98.
- Schmidhuber, J. (2015). Deep Learning in Neural Networks: An Overview. *Neural Networks*, 61, 85-117.
- Schrauwen, B., & Van Campenhout, J. (2003). BSA, A Fast and Accurate Spike Train Encoding Scheme. In *Proceedings of the International Joint Conference on Neural Networks* (Vol. 4, pp. 2825-2830). IEEE Piscataway, NJ.
- Scott, N., Kasabov, N., & Indiveri, G. (2013). NeuCube Neuromorphic Framework for Spatio-temporal Brain Data and Its Python Implementation. In *Neural Information Processing* (pp. 78-84). Springer.
- Shinde, A., Samant, R., Naik, A., Ghorpade, S., & Kale, S. (2017). Heart Disease Prediction System using Multilayered Feed Forward Neural Network and Back Propagation Neural Network. *International Journal of Computer Applications* (0975-8887), 166(7).
- Siegelmann, H. (2015). *How Brain Architecture Leads to Abstract Thought*. Retrieved 1 8, 2016, from <http://www.umass.edu/newsoffice/article/how-brain-architecture-leads-abstract>
- Song, Q., & Kasabov, N. (2001). ECM- A Novel On-line, Evolving Clustering Method and its Applications. *Foundations of cognitive science*, 631-682.

- Song, S., Miller, K. D., & Abbott, L. F. (2000). Competitive Hebbian Learning Through Spike-timing-dependent Synaptic Plasticity. *Nature Neuroscience*, 3(9), 919-926.
- Stephan, T. (2008). IEEE Transactions on Neural Networks. *Multilayer Perceptrons: Approximation Order and Necessary Number of Hidden Units*, 19(5), 836-844.
- Sterratt, D., Bruce, G., Andrew, G., & David, W. (2011). *Principles of Computational Modelling in Neuroscience*. Cambridge University Press.
- Subasi, A., & Gursoy, M. (2010). EEG Signal Classification Using PCA, ICA, LDA and Support Vector Machines. *Expert Systems with Applications*, 37(12), 8659-8666.
- Sutskever, I., Vinyals, O., & Le, Q. (2014). Sequence to Sequence Learning with Neural Network. In *Advances in Neural Information Processing Systems* (pp. 3104-3112).
- Talairach, J., & Tournoux, P. (1988). Co-planar Stereotaxic Atlas of the Human Brain. 3-Dimensional Proportional System: An Approach to Cerebral Imaging. *Thieme Medical Publishers*. New York.
- Tan, P. N. (2006). *Introduction to Data Mining*. Pearson Education India.
- Tana, M., Bianchi, A., Sclocco, R., Franchin, T., & Cerutti, S. (2012). Parcel-based Connectivity Analysis of fMRI Data for the Study of Epileptic Seizure Propagation. *Brain Topography*, 25(4), 345-361.
- Tanay, A., Sharan, R., & Shamir, R. (2002). Discovering Statistically Significant Biclusters in Gene Expression Data. *Bioinformatics*, 18(suppl_1), S136-S144.

- Thorpe, S. J. (1990). Spike Arrival Times: A Highly Efficient Coding Scheme for Neural Networks. *Parallel Processing in Neural Systems*, 91-94.
- Thorpe, S., & Gautrais, J. (1998). Rank Order Coding. In *Computational Neuroscience* (pp. 113-118). Springer.
- Tomavs, M., Karafiat, M., Burget, L., Černocký, J., & Khudanpur, S. (2010). Recurrent Neural Network Based Language Model. *Conference of the International Speech Communication Association*, (pp. 1054-1048).
- Tripathi, S., Acharya, S., Sharma, R., Sudhanshi, M., & Bhattacharya, S. (2017). Using Deep and Convolutional Neural Networks for Accurate Emotion Classification on DEAP Dataset. *Association for the Advancement of Artificial Intelligence (AAAI)*, (pp. 4746-4752).
- Tu, E., Kasabov, N., Othman, M., Li, Y., Worner, S., Yang, J., & Jia, Z. (2014). NeuCube(ST) for Spatio-Temporal Data Predictive Modelling with a Case Study on Ecological Data. *Neural Networks (IJCNN)*. Beijing, China.
- Van Schaik, A., & Liu, S.-C. (2005). AER EAR: A Matched Silicon Cochlea Pair with Address Event Representation Interface. In *IEEE International Symposium on Circuits and Systems* (Vol. 5, pp. 4213-4216).
- Verstraeten, D., Schrauwen, B., & Stroobandt, D. (2005). Isolated Word Recognition using A Liquid State Machine. In C. I. European Symposium on Artificial Neural Networks. Citeseer.

- Verstraeten, D., Schrauwen, B., D’Haene, M., & Stroobandt, D. (2007). An Experimental Unification of Reservoir Computing Methods. *Neural Networks*, 20(3), 391-403.
- Wachinger, C., Reuter, M., & Klein, T. (2018). DeepNAT: Deep Convolutional Neural Network for Segmenting Neuroanatomy. *NeuroImage*, 170, 434-445.
- Waibel, A., Hanazawa, T., inton, G., Shikano, K., & Lang, K. J. (1989). Phoneme Recognition using Time-delay Neural Networks. *IEEE Transactions on Acoustics, Speech, and Signal processing*, 37(3), 328-339.
- Wang, W., Yang, J., & Muntz, R. (1997). STING: A statistical Information Grid Approach to Spatial Data Mining. In *International Conference on Very Large Data Bases* (Vol. 97, pp. 186-195).
- Wiedemann, H. (1994). The Pioneers of Pediatric Medicine. *European Journal of Pediatrics*, 153(10), 705. doi:10.1007/BF01954482
- Wittchen, H.-U. (1994). Reliability and Validity Studies of the WHO-Composite International Diagnostic Interview (CIDI): A Critical Review. *Journal of Psychiatric Research*, 28(1), 57-84.
- Wu, J., Xiong, H., & Chen, J. (2009). Adapting the Right Measures for k-means Clustering. In *Proceedings of the 15th ACM SIGKDD International Conference on Knowledge Discovery and Data Mining* (pp. 877-886).
- Wysoski, S. G., Benuskova, L., & Kasabov, N. (2010). Evolving Spiking Neural Networks for Audiovisual Information Processing. *Neural Networks*, 23(7), 819-835.

- Wysoski, S., Benuskova, L., & Kasabov, N. (2006). On-line Learning with Structural Adaptation in A Network of Spiking Neurons for Visual Pattern Recognition. In *International Conference on Artificial Neural Networks* (pp. 61-70). Springer.
- Yaacob, H., Abdul, W., & amaruddin, N. (2016). Classification of EEG Signals using MLP Based on Categorical and Dimensional Perceptions of Emotion. *5th International Conference on Information and Communication Technology for the Muslim World (ICT4M)* (pp. 1-6). IEEE.
- Yu, X., Efe, M., & Kaynak, O. (2002). A General Backpropagation Algorithm for Feedforward Neural Networks Learning. *IEEE Transactions on Neural Networks*, 13(1), 51-254.
- Yuasa, M., Saito, K., & Mukawa, N. (2011). Brain Activity when Reading Sentences and Emoticons: An fMRI Study of Verbal and Nonverbal Communication. *Electronics and Communications in Japan*, 94(5), 17-24.
- Zhao, Y. (1996). On-line Neural Network Learning Algorithm with Exponential. *Electronics Letters*, 32(15), 1381-1382.
- Zhao, Y., & Karypis, G. (2002). Evaluation of Hierarchical Clustering Algorithms for Document Datasets. In *Proceedings of the Eleventh international Coerence on Information and Kowledge Mnagement* (pp. 515-524).
- Zhou, D., Bousquet, O., Lal, T., & Weston, J. (2004). Learning with Local and Global Consistency. *Advances in Neural Information Processing Systems*, 16, 321-328.

Zhou, G., & Si, J. (1998). Advanced Neural Network training Algorithm with Reduced Complexity based on Jacobian Deficiency. *IEEE Transactions on Neural Networks*, 9(3), 448-453.

Appendix A Talairach Mapping

The spatial mapping of EEG data variables into the NeuCube-based SNN models was performed with respect to the (x,y,z) coordinates as positioned in the Talairach template (Table A-1).

Table A-1 Anatomical locations of cortical projections from (Koessler, et al., 2009). The BA column represents the id of the corresponding Brodmann areas.

Labels	Talairach coordinates			Gyri		BA
	x avg (mm)	y avg (mm)	z avg (mm)			
FP1	- 21.2 ± 4.7	66.9 ± 3.8	12.1 ± 6.6	L FL	Superior frontal G	10
FPz	1.4 ± 2.9	65.1 ± 5.6	11.3 ± 6.8	M FL	Bilat. medial	10
FP2	24.3 ± 3.2	66.3 ± 3.5	12.5 ± 6.1	R FL	Superior frontal G	10
AF7	- 41.7 ± 4.5	52.8 ± 5.4	11.3 ± 6.8	L FL	Middle frontal G	10
AF3	- 32.7 ± 4.9	48.4 ± 6.7	32.8 ± 6.4	L FL	Superior frontal G	9
AFz	1.8 ± 3.8	54.8 ± 7.3	37.9 ± 8.6	M FL	Bilat. medial	9
AF4	35.1 ± 3.9	50.1 ± 5.3	31.1 ± 7.5	L FL	Superior frontal G	9
AF8	43.9 ± 3.3	52.7 ± 5.0	9.3 ± 6.5	R FL	Middle frontal G	10
F7	- 52.1 ± 3.0	28.6 ± 6.4	3.8 ± 5.6	L FL	Inferior frontal G	45
F5	- 51.4 ± 3.8	26.7 ± 7.2	24.7 ± 9.4	L FL	Middle frontal G	46
F3	- 39.7 ± 5.0	25.3 ± 7.5	44.7 ± 7.9	L FL	Middle frontal G	8
F1	- 22.1 ± 6.1	26.8 ± 7.2	54.9 ± 6.7	L FL	Superior frontal G	6
Fz	0.0 ± 6.4	26.8 ± 7.9	60.6 ± 6.5	M FL	Bilat. medial	6
F2	23.6 ± 5.0	28.2 ± 7.4	55.6 ± 6.2	R FL	Superior frontal G	6
F4	41.9 ± 4.8	27.5 ± 7.3	43.9 ± 7.6	R FL	Middle frontal G	8
F6	52.9 ± 3.6	28.7 ± 7.2	25.2 ± 7.4	R FL	Middle frontal G	46
F8	53.2 ± 2.8	28.4 ± 6.3	3.1 ± 6.9	R FL	Inferior frontal G	45
FT9	- 53.8 ± 3.3	- 2.1 ± 6.0	- 29.1 ± 6.3	L TL	Inferior temporal G	20
FT7	- 59.2 ± 3.1	3.4 ± 5.6	- 2.1 ± 7.5	L TL	Superior temporal G	22
FC5	- 59.1 ± 3.7	3.0 ± 6.1	26.1 ± 5.8	L FL	Precentral G	6
FC3	- 45.5 ± 5.5	2.4 ± 8.3	51.3 ± 6.2	L FL	Middle frontal G	6
FC1	- 24.7 ± 5.7	0.3 ± 8.5	66.4 ± 4.6	L FL	Superior frontal G	6
FCz	1.0 ± 5.1	1.0 ± 8.4	72.8 ± 6.6	M FL	Superior frontal G	6
FC2	26.1 ± 4.9	3.2 ± 9.0	66.0 ± 5.6	R FL	Superior frontal G	6
FC4	47.5 ± 4.4	4.6 ± 7.6	49.7 ± 6.7	R FL	Middle frontal G	6
FC6	60.5 ± 2.8	4.9 ± 7.3	25.5 ± 7.8	R FL	Precentral G	6
FT8	60.2 ± 2.5	4.7 ± 5.1	- 2.8 ± 6.3	L TL	Superior temporal G	22
FT10	55.0 ± 3.2	- 3.6 ± 5.6	- 31.0 ± 7.9	R TL	Inferior temporal G	20
T7	- 65.8 ± 3.3	- 17.8 ± 6.8	- 2.9 ± 6.1	L TL	Middle temporal G	21
C5	- 63.6 ± 3.3	- 18.9 ± 7.8	25.8 ± 5.8	L PL	Postcentral G	123
C3	- 49.1 ± 5.5	- 20.7 ± 9.1	53.2 ± 6.1	L PL	Postcentral G	123
C1	- 25.1 ± 5.6	- 22.5 ± 9.2	70.1 ± 5.3	L FL	Precentral G	4

Cz	0.8 ± 4.9	-21.9 ± 9.4	77.4 ± 6.7	M FL	Precentral G	4
C2	26.7 ± 5.3	-20.9 ± 9.1	69.5 ± 5.2	R FL	Precentral G	4
C4	50.3 ± 4.6	-18.8 ± 8.3	53.0 ± 6.4	R PL	Postcentral G	123
C6	65.2 ± 2.6	-18.0 ± 7.1	26.4 ± 6.4	R PL	Postcentral G	123
T8	67.4 ± 2.3	-18.5 ± 6.9	-3.4 ± 7.0	R TL	Middle temporal G	21
TP7	-63.6 ± 4.5	-44.7 ± 7.2	-4.0 ± 6.6	L TL	Middle temporal G	21
CP5	-61.8 ± 4.7	-46.2 ± 8.0	22.5 ± 7.6	L PL	Supramarginal G	40
CP3	-46.9 ± 5.8	-47.7 ± 9.3	49.7 ± 7.7	L PL	Inferior parietal G	40
CP1	-24.0 ± 6.4	-49.1 ± 9.9	66.1 ± 8.0	L PL	Postcentral G	7
CPz	0.7 ± 4.9	-47.9 ± 9.3	72.6 ± 7.7	M PL	Postcentral G	7
CP2	25.8 ± 6.2	-47.1 ± 9.2	66.0 ± 7.5	R PL	Postcentral G	7
CP4	49.5 ± 5.9	-45.5 ± 7.9	50.7 ± 7.1	R PL	Inferior parietal G	40
CP6	62.9 ± 3.7	-44.6 ± 6.8	24.4 ± 8.4	R PL	Supramarginal G	40
TP8	64.6 ± 3.3	-45.4 ± 6.6	-3.7 ± 7.3	R TL	Middle temporal G	21
P9	-50.8 ± 4.7	-51.3 ± 8.6	-37.7 ± 8.3	L TL	Tonsile	NP
P7	-55.9 ± 4.5	-64.8 ± 5.3	0.0 ± 9.3	L TL	Inferior temporal G	37
P5	-52.7 ± 5.0	-67.1 ± 6.8	19.9 ± 10.4	L TL	Middle temporal G	39
P3	-41.4 ± 5.7	-67.8 ± 8.4	42.4 ± 9.5	L PL	Precuneus	19
P1	-21.6 ± 5.8	-71.3 ± 9.3	52.6 ± 10.1	L PL	Precuneus	7
Pz	0.7 ± 6.3	-69.3 ± 8.4	56.9 ± 9.9	M PL	Superior parietal L	7
P2	24.4 ± 6.3	-69.9 ± 8.5	53.5 ± 9.4	R PL	Precuneus	7
P4	44.2 ± 6.5	-65.8 ± 8.1	42.7 ± 8.5	R PL	Inferior parietal L	7
P6	54.4 ± 4.3	-65.3 ± 6.0	20.2 ± 9.4	R TL	Middle temporal G	39
P8	56.4 ± 3.7	-64.4 ± 5.6	0.1 ± 8.5	R TL	Inferior temporal G	19
P10	51.0 ± 3.5	-53.9 ± 8.7	-36.5 ± 10.0	L OL	Tonsile	NP
PO7	-44.0 ± 4.7	-81.7 ± 4.9	1.6 ± 10.6	R OL	Middle occipital G	18
PO3	-33.3 ± 6.3	-84.3 ± 5.7	26.5 ± 11.4	R OL	Superior occipital G	19
POz	0.0 ± 6.5	-87.9 ± 6.9	33.5 ± 11.9	M OL	Cuneus	19
PO4	35.2 ± 6.5	-82.6 ± 6.4	26.1 ± 9.7	R OL	Superior occipital G	19
PO8	43.3 ± 4.0	-82.0 ± 5.5	0.7 ± 10.7	R OL	Middle occipital G	18
O1	-25.8 ± 6.3	-93.3 ± 4.6	7.7 ± 12.3	L OL	Middle occipital G	18
Oz	0.3 ± 5.9	-97.1 ± 5.2	8.7 ± 11.6	M OL	Cuneus	18
O2	25.0 ± 5.7	-95.2 ± 5.8	6.2 ± 11.4	R OL	Middle occipital G	18

Appendix B Spatio-Temporal fMRI Study

B.1 Spatio-temporal fMRI Data Description

The case study problem used here belongs to the STAR/PLUS fMRI data set, originally collected by Marcel Just and his colleagues at Carnegie Mellon University's Centre for Cognitive Brain Imaging (CCBI) (Just & Wang, 2001). The original dataset was provided in a MATLAB file which contains three major arrays: Data (contains the time-series recorded from voxels), Info (contains information about cognitive task) and Meta (contains the voxels' labels related to their region of interests— ROIs). As stated in (Mitchell, et al., 2004), in order to introduce randomness in the cognitive task, 40 trials were defined. In the first 20 trials, participants were first presented by a picture and then a sentence whereas for remaining 20 trials, they reversed the order of the picture and sentence presentation.

For fMRI acquisition, while participants were performing the cognitive trials, *T2-weighted*⁷ fMRI images were collected using 3T scanner at an interval of 500 milliseconds, and with TE = 18 milliseconds and flip angle of 50°. These settings yield images that have approximately 5000 voxels per subjects in 8 oblique axial slices in two different non-contiguous four-slice volumes. The first volume set captures prefrontal areas and superior parietal regions, while, another volume set covers posterior temporal, inferior frontal and occipital areas. After acquiring T2-weighted fMRI images for each subject, images were pre-

⁷ In T2-weighted both fat and water are hyper-intense and appear bright.

processed using Functional Imaging Analysis Software, Computational Olio (FIASCO) program (Eddy, Fitzgerald, Genovese, Mockus, & Noll, 1996; Eddy, et al., 1999). This pre-processing helps in reducing the artifacts that arise during image acquisition process due to signal drift, head motion, and others. After pre-processing of images, 25 anatomical regions of interest (ROI) were defined. To identify the ROI in fMRI data, two types of brain images were collected for each subject. The first type of image, which has been discussed up to this point, captures brain activation *via* the BOLD response, and is referred to as a functional image. The second type of image, called a structural image, captures the static physical brain structure at higher resolution. For each subject, this structural image was used to identify the anatomical regions of interest, using the parcellation scheme (Caviness, Verne, Meyer, Makris, & Kennedy, 1996) and (Rademacher, Galaburda, Kennedy, Filipek, & Caviness, 1992). For each subject, the mean of their functional images was then co-registered to the structural image; so that, individual voxels in the functional images could be associated with the ROIs identified in the structural image. The achieved ROIs include: left dorsolateral prefrontal cortex (LDLPFC) and right dorsolateral prefrontal cortex (RDLDPFC), calcarine sulcus (CALC), left frontal eye fields (LFEF), right frontal eye fields (RFEF), left inferior parietal lobule (LIPL), right inferior parietal lobule (RIPL), left intraparietal sulcus (LIPS), right intraparietal sulcus (RIPS), left inferior frontal gyrus (LIFG), left opercularis (LOPER), right opercularis (ROPER), supplementary motor areas (SMA), left and right inferior temporal lobule (LIT, RIT), left and right posterior precentral sulcus (LPPREC, RPPREC), left and right supramarginal gyrus (LSGA, RSGA), left temporal lobe (LT), right temporal lobe (RT), left and right triangularis (LTRIA, RTRIA), left superior parietal lobule (LSPL) and right superior parietal lobule (RSPL).

B.2 Statistical Analysis of the SNN Models of fMRI

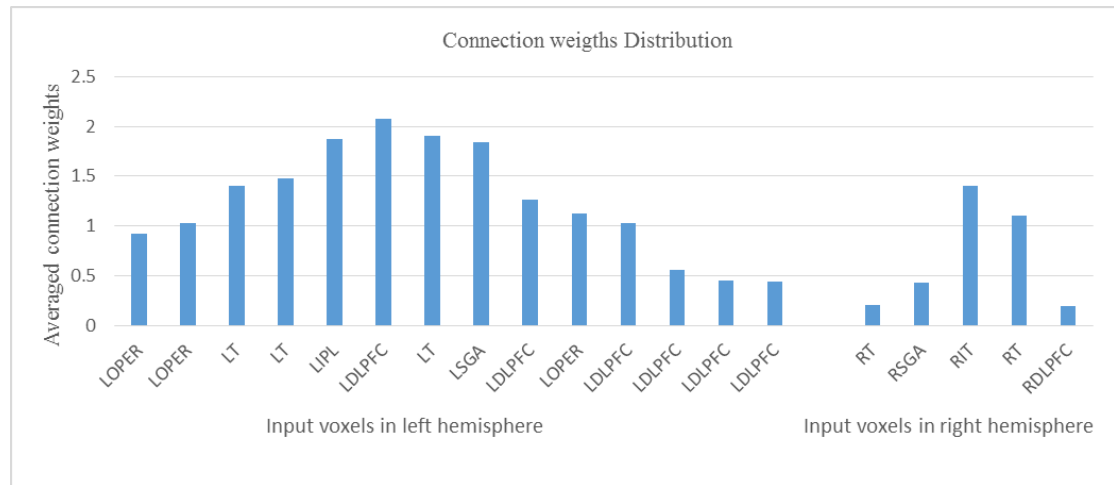


Figure B-1 Distribution of the average connection weights around the input voxels located in the left and right hemispheres of the trained SNN models of related respectively to negative sentences.

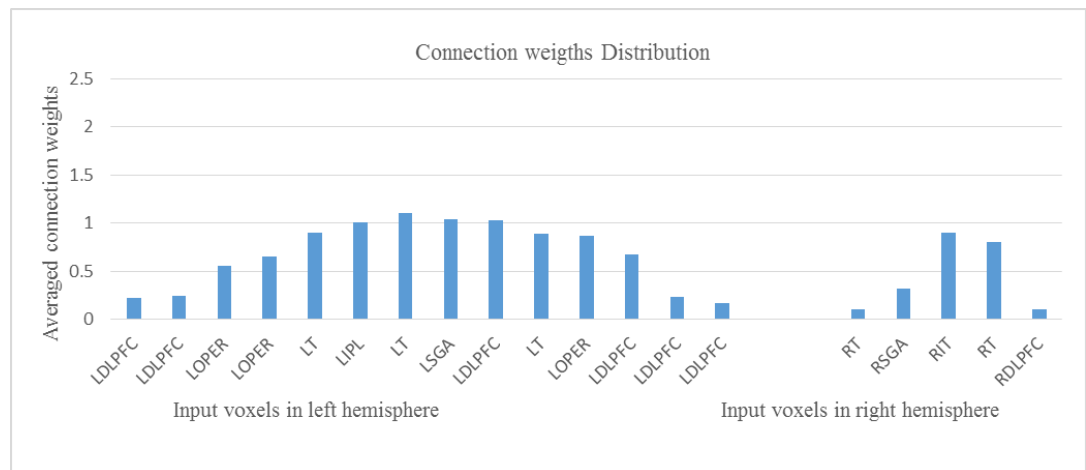


Figure B-2 Distribution of the average connection weights around the input voxels located in the left and right hemispheres of the trained SNN models of related respectively to affirmative sentences.

Table B-1 T-Test results of the training iterations in the SNN models of fMRI data related to affirmative sentences and negative sentences (right) and seeing pictures and reading sentences (left). The t-test was measured in Excel using “Two-Sample Assuming Unequal Variances” method.

	Picture	Sentence		Affirmative	Negative
Mean	1.3	1.12	Mean	1.2	0.7
Variance	0.13	0.06	Variance	0.06	0.03
Observations	40	40	Observations	20	20
P(T<=t) one-tail	0.003		P(T<=t) one-tail	6.50E-08	
t Critical one-tail	1.6		t Critical one-tail	1.6	
P(T<=t) two-tail	0.006		P(T<=t) two-tail	1.30E-07	
t Critical two-tail	2		t Critical two-tail	2.03	

B.3 SNN Parameter Optimisation

For optimisation, I performed an exhaustive grid-search on combination of parameters for every sample’s model. Each parameter was searched within a range, specified by the minimum and maximum, through several iterations related to the number of steps for moving from minimum to maximum. For every model creation, I chose three main parameters (STDP learning rate, *mod* and *drift* parameters) to be optimised. The parameters were selected by assigning 10 steps between the minimum and maximum values of each parameter range. Therefore, for every model creation, 1000 iterations of training (using all samples except the holdout sample) and testing (using the single holdout sample) were performed with different combination of these three parameters. Then the parameters that resulted in the best accuracy in most of the iterations have been reported as the optimal parameters, shown in Table B-2. The TBR threshold, neuron firing threshold and small-world radius parameters were fixed to 0.5, 0.5, and 2.5 respectively.

Table B-2 Optimal parameter setting of the SNN models for different experiments and sessions with the benchmark fMRI data.

Classification tasks	Experiment Sessions	<i>mod</i>	<i>drift</i>	STDP rate
Experiment A: Affirmative vs negative sentences	Session I	0.56	0.03	0.006
	Session II	0.45	0.03	0.005
	Session III	0.51	0.02	0.005
Experiment B: Pictures vs sentences	Session I	0.56	0.03	0.006
Parameter range		0.4-0.95	0.001-0.5	0.001-0.01

Appendix C EEG Study

To validate the proposed methods in this thesis, I have used a case study of EEG data which was recorded under an ethical approval granted by the *Northern Regional X Ethics Committee* of New Zealand. The data were recorded at the School of Pharmacy, University of Auckland by Dr Grace Wang, Senior Lecturer /Addictions Programme Leader from Faculty of Health and Environmental Science at Auckland University of Technology. All participants signed informed consent to certify their voluntary participation.

C.1 Participants

Inclusion criteria. All participants recruited were between 18– 45 years of age, had a basic English literacy skill and were able to provide written informed consent. Participants in the opiate user group were required to meet the Diagnostic and Statistical Manual of Mental Disorders (DSM-IV) (American Psychiatric Association, 2000) criteria for opiate dependence which was diagnosed using the Composite International Diagnostic Interview (CIDI) (Kessler & Üstün, 2004). This requires participants to be physically dependent on opiates as evidenced by a history of withdrawal symptoms and to have been actively using opiates for a minimum of one year prior to the study date. Participants in the opiate user group were not allowed to be currently undertaking MMT. The inclusion criteria for healthy control subjects were no current or lifetime history of drug or alcohol abuse other than nicotine dependence.

Exclusion criteria. Exclusion criteria were based on the results of the CIDI (Kessler & Üstün, 2004), which was administered to all participants, and included a history of psychotic disorder, depression, cardiac disease, endocrine disorder, head trauma, neurological disease and self-reported current pregnancy or breastfeeding. The CIDI is a well-validated clinical assessment tool that has been used extensively for clinical and research purposes (Wittchen, 1994). It provides detailed information based on the DSM-IV (American Psychiatric Association, 2000) and includes 22 diagnostic sections that assess mood disorders, anxiety disorders, substance use, disorder, childhood disorder and other disorders. There are additional sections screening for the presence of cognitive impairments, such as memory, speech and learning, and current physical status (i.e. headache, speech and sensor perception). At the time of testing, none of the participants were experiencing symptoms that could be attributed to acute drug intoxication or withdrawal. To minimise potential confounds between groups, the age, gender, educational levels and ethnicity were matched and controlled by recruitment. The quantity of other drug use by MMT and opiate user groups was not controlled for during recruitment because the quantity of a drug consumed can be influenced by its overall quality; self-report of the amounts consumed is also known to lead to underestimates (Danion, et al., 1993). Table C-1 presents the demographic data and history of Methadone use in H, MMT, H and OP groups.

Table C-1 Demographic data and history of Methadone use in the healthy control (H), methadone maintenance treatment (MMT) and opiate (OP) groups. Standard deviation (SD).

Group	Number of participants	Mean age (SD)	Male % (n)	Female % (n)	Mean duration years of education (SD)	Mean duration years of opiate/Methadone use (SD)
MMT	31	39.36 (5.14)	54.83 (17)	45.16 (14)	12.06 (SD=2.00)	Opiate use: 10.03 (6.08) MMT use: 7.29 (SD=6.39)
OP	18	37.38 (7.44)	61.11 (11)	38.88 (7)	12.47 (SD=1.46)	Opiate use: 11.41 (8.60)
H	21	36.12 (6.61)	52.38 (11)	47.61 (10)	13.71 (SD=1.73)	—

C.2 Cognitive GO/NOGO Task

GO/NOGO task is a psychological test to measure a participant's capacity for response control and sustained attention. During the task, participants were repeatedly presented by the word 'PRESS' (for 500 ms). The colour of the word 'PRESS' was presented randomly in either red or green. Participants were instructed to respond by pressing a button with the index finger of both hands in response to the word that appeared in green (GO) and not respond to the word that appeared in red (NOGO). Based on the most literature about GO/NO-GO task, the brain response inhibition, as a core executive function, is expected to be observed in prefrontal, frontal, dorsal, ventral, and parietal regions, which are related to human response inhibition. On the other hand, psychological reports showed there is a direct relation between the response reduction in prefrontal cortex magnitude and addictive behaviour, due to the drug effects on these brain functions.

Participants were asked to complete the practice trial prior to the real test to ensure that they understood the task. At this stage, the word 'PRESS' was presented in the same colour 6 times in a row. There were 28 sequences, 21 of which were presented in green and 7 in red, presented in a pseudo-random order, with an inter-stimulus interval of 1 second. The task duration was approximately 5 minutes. Speed and accuracy of response were stressed equally in the task instructions.

During a GO/NOGO task, a participant is required to perform an action given certain stimuli (e.g. press a button-GO) and inhibit that action under a different set of stimuli (e.g. not press

that same button- NOGO). Typically, NOGO stimuli are rare and task instructions are to execute a fast GO response. Therefore, there is increased conflict when NOGO stimuli are presented. Although the GO/NOGO task appears simple, requiring a response according to a conditional rule reflects high level cognitive functions including decision making, response selection, and inhibition. Evidence shows that groups characterised by clinically relevant impulsivity, e.g., drug users, tend to show diminished inhibition of responses to NOGO stimuli; thus, making more errors of commission (Ahmadi, et al., 2013).

C.3 EEG Data Acquisition

EEG has been extensively used for brain studies including addiction research. It is recognised as a sensitive measure of drug effects on the brain, which often manifest as changes in the size and time course of the postsynaptic potentials (Gevins, et al., 2011) as reflected in alterations in EEG signals. It has been shown that reinforcing effects of many drugs mediated by the mesolimbic dopamine pathway modify EEG recordings (Knyazev, 2007).

For EEG recordings, a QuickCap (Neuroscan 4.3) 40 sensor shielded cap was used to acquire EEG data from the cephalic sites. The 26 Ag/AgCl sintered electrodes included Fp1, Fp2, Fz, F3, F4, F7, F8, Cz, C3, C4, CP3, CPz, CP4, FC3, FCz, FC4, T7, T8, P7, P8, Pz, P3, P4, O1, O2, and Oz electrode sites (10–20 International System). EEG was recorded relative to the average of A1 and A2 (mastoid) electrode sites. Horizontal eye movements were recorded with electrodes placed 1.5 cm laterally to the outer canthus of each eye. Vertical eye movements were recorded with electrodes placed 3 mm above the middle of the left eyebrow and 1.5 cm below the middle of the left bottom eye-lid. Skin resistance was kept at <5 kOhms. Scalp and EOG potentials were amplified and digitised continuously by a system (NuAmps,

SCAN 4.3) having frequency response from direct current (DC) to 100 Hz (above which attenuating by 40 dB per decade), and a sampling rate of 500 Hz. EEG data were screened visually for artifacts, normal variants and changes in alertness (the technician screening these data was blinded to group status). To reduce muscle artefacts in the EEG signal, the participants were instructed to assume a comfortable position and avoid movement during recording. Electrical impedance was always $<5\text{ K}\Omega$.

C.4 Event-Related Potentials (ERPs) Processing

Artefact correction was conducted by the Brain Resource Company (BRC) of Australia. EEG data were corrected using the method of (Gratton, Coles, & Donchin, 1983), which computes separate propagation factors for blinks and eye movements on the basis of the residuals in the EEG channels after subtraction of event-related activity on either channel. Then, for each channel, the individual single-trial epochs were filtered with a low-pass filter function that attenuates frequencies above 25 Hz. Scalp EEG recording was segmented into epochs, centred on each single event corresponding to the stimulus. Event-related potentials (ERP) was acquired by averaging EEG signals from several stimulus presentations during the performance of a task. The averaging extracts the spontaneous ‘noisy’ background fluctuations from scalp recordings leaving only stimulus-related electrical activity in a time-locked windows. ERP response to all GO stimuli with a correct button response (only) were included in the GO average file. Similarly, ERP response to NOGO stimuli with a correct button response (only) were included in the NOGO average file.

C.5 T-Test Results

A t-test was applied to evaluate how the trained SNN models of H, MMT and OP subjects were statistically significant as shown in Tables C-2 and C-3.

Table C-2 T-Test results of the training iterations in the SNN models of H, MMT and OP groups in GO and NOGO tasks. The T-Test was computed in Excel using “Two-Sample Assuming Unequal Variances” method.

H			MMT			OP		
	GO	NOGO		GO	NOGO		GO	NOGO
Mean	0.09	0.07	Mean	0.1	0.2	Mean	0.08	0.2
Variance	5E-04	0.0003	Variance	0.001	0.001	Variance	6E-04	0.07
Observations	21	21	Observations	29	31	Observations	18	18
df	38		df	58		df	17	
t Stat	3.2		t Stat	-3.4		t Stat	-2.9	
P(T<=t) one-tail	0.001		P(T<=t) one-tail	0		P(T<=t) one-tail	0.004	
t Critical one-tail	1.6		t Critical one-tail	1.6		t Critical one-tail	1.8	
P(T<=t) two-tail	0.002		P(T<=t) two-tail	0.001		P(T<=t) two-tail	0.009	
t Critical two-tail	2.02		t Critical two-tail	2		t Critical two-tail	2.1	

Table C-3 T-Test results of the FIN graphs of the H, MMT and OP groups in GO and NOGO trials. The T-Test was computed in Excel using “Paired Two Sample for Means” method.

GO		H	MMT		H	OP		MMT	OP
	Mean	38.5	19.2	Mean	38.5	22.9	Mean	19.2	22.9
	Variance	705	352	Variance	705	315	Variance	352	315
	Observations	26	26	Observations	26	26	Observations	26	26
	Pearson Correlation	0.03		Pearson Correlation	0.16		Pearson Correlation	0.25	
	df	25		df	25		df	25	
	t Stat	3.07		t Stat	2.69		t Stat	-0.9	
	P(T<=t) one-tail	0		P(T<=t) one-tail	0.01		P(T<=t) one-tail	0.2	
	t Critical one-tail	1.71		t Critical one-tail	1.71		t Critical one-tail	1.71	
	P(T<=t) two-tail	0.01		P(T<=t) two-tail	0.01		P(T<=t) two-tail	0.4	
	t Critical two-tail	2.06		t Critical two-tail	2.06		t Critical two-tail	2.06	
NOGO		H	MMT		H	OP		MMT	OP
	Mean	29.3	37.6	Mean	29.3	48.6	Mean	37.6	48.6
	Variance	283	473	Variance	283	1024	Variance	473	1024
	Observations	26	26	Observations	26	26	Observations	26	26
	Pearson Correlation	0.54		Pearson Correlation	0.33		Pearson Correlation	0.5	
	df	25		df	25		df	25	
	t Stat	-2.2		t Stat	-3.18		t Stat	-2	
	P(T<=t) one-tail	0.02		P(T<=t) one-tail	0		P(T<=t) one-tail	0.03	
	t Critical one-tail	1.71		t Critical one-tail	1.71		t Critical one-tail	1.71	
	P(T<=t) two-tail	0.03		P(T<=t) two-tail	0.003		P(T<=t) two-tail	0.06	
	t Critical two-tail	2.06		t Critical two-tail	2.06		t Critical two-tail	2.06	

C.6 Feature Interaction in SNN Models

During the STDP learning procedure, input neurons accumulate spikes to the SNN model and, if the neurons that received the spikes cross an activation threshold, they also emit output spikes. That spikes are sent out to all the connected neurons, and then the neurons will likewise accumulate activity as a function of receiving spikes and, after crossing some threshold, fire. In such a way, spikes are transferred between neurons and propagated to the SNN model.

To have a better understanding of the spike propagation in SNN models, I developed a graph-based representation of the number of spikes exchanged between the neurons. Figures C-1 to C-6 illustrate the six Feature Interaction Networks (FINs), each of which captures the number of spikes transmitted between 26 EEG features (channels) during unsupervised STDP learning. Each node in FIN represents a group of neurons connected to an input neuron (EEG channel) and the arcs represent relative spike amounts transmitted between these groups. The amount of spike interaction between any 2 adjacent groups of neurons (each connected to one input neuron) was computed with respect to the number of spikes exchanged between them divided by the total STDP learning time. The wider the arc between nodes, the more spikes were transmitted between the corresponding groups of neurons. Tables C-4 to C-9 represent the quantitative information of FINs that demonstrate the level of spatio-temporal interaction between the EEG channels with respect to the number of spikes transmitted between them. By comparing FIN graphs developed for H subjects in GO *versus* NOGO, it can be seen that the spike communication was larger between the most of inputs while the subjects were performing GO trials than NOGO trials. On the other hand, both MMT and OP subjects demonstrated increased spike communication in a wide range of areas during the

NOGO trials than GO trials. These findings represent different SNN models were developed for people with a history of opiate and healthy subjects. For instance, for OP subjects in NOGO trials wide lines were created between F4 and C4 (line weight= 16.15) and F4 and P4 (line weight= 25.2). These links represent a great number of spikes was transmitted between groups of neurons around these channels while the subjects were undertaking NOGO trials. In contrast, different level of interaction between these channels was observed for H subjects during NOGO trials. These findings were evaluated by a t-test to represent whether the FIN graphs were statistically significant across the groups. Table C-10 represents the p -values for FIN graphs (applied to the last column) of H, MMT and OP subjects in GO and NOGO tasks. It represents that in both GO and NOGO trials the FIN graphs of the H and OP subjects were significantly different (p -value=0.012<0.05 in GO and =0.003 <0.05 in NOGO). Also, the FIN graphs of H and MMT were also significantly different as the p -value=0.005<0.05 in GO and =0.03<0.05 in NOGO. However, the statistical analysis on the FIN graphs of MMT and OP subjects in GO has not shown a meaningful significance as p -value=0.4>0.05.

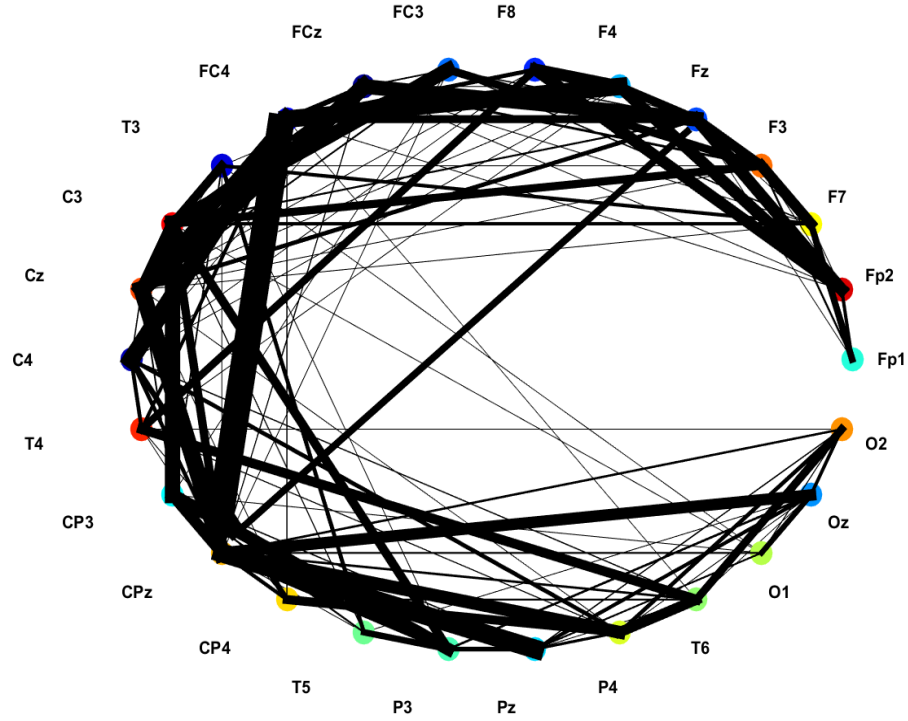


Figure C-1 The spike interaction between 26 neural groups representing 26 EEG channels in H group in GO task. The thicker the line that connects two nodes that represent the corresponding EEG channels, the more spikes were transmitted between corresponding groups.

Table C-4 The amount of spike transformation between 26 neural groups representing 26 EEG channels in H group in GO task, computed as the number of spikes exchanged between 2 groups divided by the STDP time.

	Fp1	Fp2	F7	F3	Fz	F4	F8	FC3	FCz	FC4	T3	C3	Cz	C4	T4	CP3	CPz	CP4	T5	P3	Pz	P4	T6	O1	Oz	O2	Σ	
Fp1	0	3	6	1.8	0.4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	11.3	
Fp2	3	0	0	0.4	10.6	11	10	0	0	0.2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	35.3	
F7	6	0	0	9.7	0.4	0	0	1.2	0	0	3.1	3.8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	24.5	
F3	1.8	0.4	9.7	0	8.7	0	0	6.9	2.1	0	0	7	0.1	0	0	0	0	0	0	0	0	0	0	0	0	0	37.2	
Fz	0.4	10.6	0.4	8.7	0	7.4	0.2	1.1	12.7	6.8	0	1.6	3.7	0	0	0	6	0	0	0	0	0	0	0	0	0	60.1	
F4	0	10.5	0	0	7.4	0	10	0	0.3	15	0	0	0.1	0	0.1	0	0	0	0	0	0	0	0	0	0	0	44.2	
F8	0	10.2	0	0	0.2	10	0	0	0	3.6	0	0	0	0.3	6.6	0	0	0	0	0	0	0	0	0	0	0	31.7	
FC3	0	0	1.2	6.9	1.1	0	0	1.3	0	0.2	12.5	0.9	0	0	0	0	0	0	0	0	0	0	0	0	0	0	24.6	
FCz	0	0	0	2.1	12.7	0.3	0	1.3	0	4.5	0	4	14.5	0.1	0	0	1	0	0	0	0	0	0	0	0	0	41.1	
FC4	0	0.2	0	0	6.8	15	3.6	0	4.5	0	0	0	9.8	15.3	3.2	0	26.2	0.7	0	0	0	0	0	0	0	0	85.8	
T3	0	0	3.1	0	0	0	0	0.2	0	0	0	7.9	0	0	1.4	0.1	0	3.6	0.4	0	0	0	0	0	0	0	17	
C3	0	0	3.8	7	1.6	0	0	12.5	4	0	7.9	0	10.7	0	0	14.1	7.7	0	0	7.9	0	0	0	0	0	0	77.5	
Cz	0	0	0	0.1	3.7	0.1	0	0.9	14.5	9.8	0	10.7	0	1.2	0	0.9	18	0.4	0	0	0	0.1	0	0	0	0	61	
C4	0	0	0	0	0	0	0.3	0	0.1	15.3	0	0	1.2	0	3.5	0	3.5	5	0	0	0	3.6	0.7	0	0	0	33.5	
T4	0	0	0	0	0	0.1	6.6	0	0	3.2	0	0	0	3.5	0	0	0.3	0.5	0	0	0	0.6	8.3	0	0	1	24.5	
CP3	0	0	0	0	0	0	0	0	0	0	1.4	14.1	0.9	0	0	10	0	1.6	11.1	0.7	0	0	0.1	0	0	0	40.3	
CPz	0	0	0	0	6	0	0	1	26.2	0.1	7.7	18	3.5	0.3	10	0	3.3	0	6.6	16.3	11.1	2	1.6	12.9	2	0	129.5	
CP4	0	0	0	0	0	0	0	0	0	0.7	0	0	0.4	5	0.5	0	3.3	0	0	0	0	7.7	1.6	0	0	0	19.3	
T5	0	0	0	0	0	0	0	0	0	0	3.6	0	0	0	0	1.6	0	0	0	5.5	0	0	0	0	0	0	10.9	
P3	0	0	0	0	0	0	0	0	0	0	0.4	7.9	0	0	0	11.1	6.6	0	5.5	0	4	0	0	0	2	0	37.7	
Pz	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.7	16.3	0	0	4	0	2.4	0	1.5	3.2	1.1	29.7	
P4	0	0	0	0	0	0	0	0	0	0	0	0	0	0.1	3.6	0.6	0	11.1	7.7	0	0	2.4	0	9	0	0.4	3.6	38.9
T6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.7	8.3	0	2	1.6	0	0	9	0	0	0.1	7.1	29.1	
O1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.1	1.6	0	0	0	1.5	0	0	0	4.9	0.5	8.9	
Oz	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	12.9	0	0	2	3.2	0.4	0.1	4.9	0	3	26.9	
O2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	2	0	0	0	1.1	3.6	7.1	0.5	3	0	18.5	

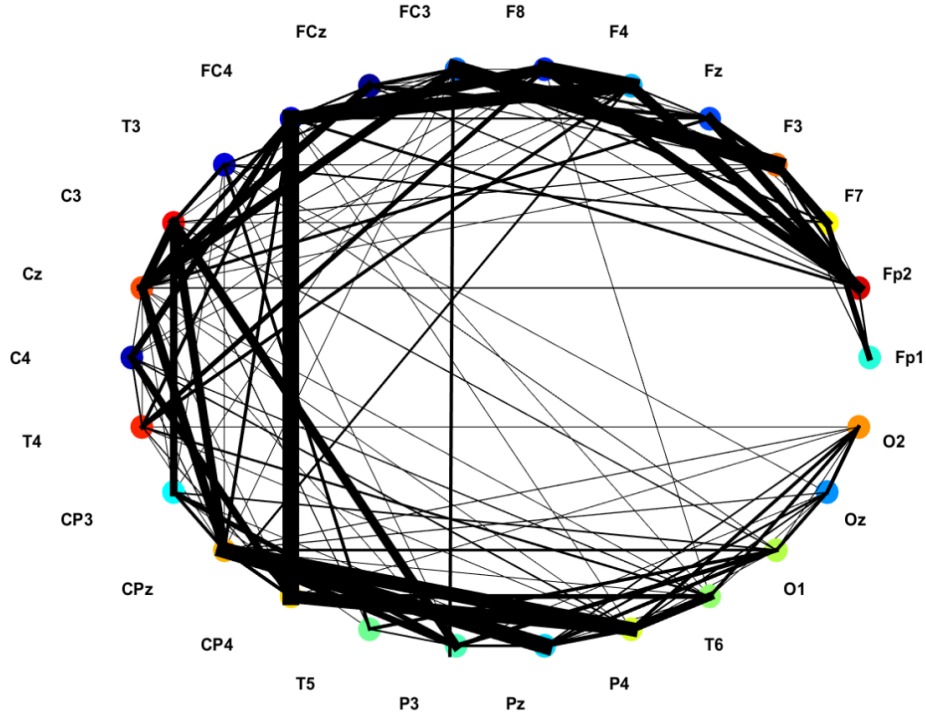


Figure C-2 The spike interaction between 26 neural groups representing 26 EEG channels in H group in NOGO task. The thicker the line that connects two nodes that represent the corresponding EEG channels, the more spikes were transmitted between corresponding groups.

Table C-5 The amount of spike transformation between 26 neural groups representing 26 EEG channels in H group in NOGO task, computed as the number of spikes exchanged between 2 groups divided by the STDP time.

	Fp1	Fp2	F7	F3	Fz	F4	F8	FC3	FCz	FC4	T3	C3	Cz	C4	T4	CP3	CPz	CP4	T5	P3	Pz	P4	T6	O1	Oz	O2	Σ
Fp1	0	1.7	5.3	1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	8.1
Fp2	1.7	0	0.2	2.9	8.6	9.6	10	0.3	0.8	2.4	0	0	1.1	0	0	0	0	0	0	0	0	0	0	0	0	0	38.1
F7	5.3	0.2	0	8	0	0	0	2	0	0	1.7	0.3	0	0	0	0	0	0	0	0	0	0	0	0	0	0	17.7
F3	1	2.9	8	0	5.1	0	0	19	1	0	0.2	0.6	0.4	0	0	0	0	0	0	0	0	0	0	0	0	0	38.5
Fz	0	8.6	0	5.1	0	0.9	0.4	2.2	3.5	1.7	0	0	2.6	0	0	0	0	0	0	0	0	0	0	0	0	0	25.4
F4	0	9.6	0	0	0.9	0	14	0	0.1	11.9	0	0	0.2	0.1	3.3	0	3	0	0	0	0	0	0	0	0	0	43.8
F8	0	10	0	0	0.4	14	0	0.1	0.1	10.1	0	0	0.5	0.2	4.2	0	0	0	0	0	0	0	0	0	0	0	40
FC3	0	0.3	2	19	2.2	0	0.1	0	3.1	0.1	1.7	0	10.8	0	0	0	0.4	0	0	3	0	0	0	0	0	0	43.2
FCz	0	0.8	0	1	3.5	0.1	0.1	3.1	0	1.9	0	0.3	7.5	0	0	0	0	0	0	0	0	0	0	0	0	0	18.7
FC4	0	2.4	0	0	1.7	12	10	0.1	1.9	0	0	0	2.8	5.3	2.9	0	2.9	20.6	0	0	0	0.4	0.1	0	0	0	63.6
T3	0	0	1.7	0.2	0	0	0	1.7	0	0	0	3.5	0	0	0	0.2	0	0	2.9	0.1	0	0	0	0.5	0	0	11.1
C3	0	0	0.3	0.6	0	0	0	0	0.3	0	3.5	0	7.7	0	0	6.8	7.7	0	0.6	9.3	0	0	0	1.1	0	0	38.5
Cz	0	1.1	0	0.4	2.6	0.2	0.5	10.8	7.5	2.8	0	7.7	0	0.6	0	0.3	9.9	0.2	0	0	0	0.2	0	0	0	0	45.4
C4	0	0	0	0	0	0.1	0.2	0	0	5.3	0	0	0.6	0	1.5	0	1	6.4	0	0	0	0.9	0.7	0	0	0	17.1
T4	0	0	0	0	0	3.3	4.2	0	0	2.9	0	0	0	1.5	0	0	0	1.1	0	0	0	0.1	2.1	0	0	0	15.7
CP3	0	0	0	0	0	0	0	0	0	0	0.2	6.8	0.3	0	0	0	2.9	0	0.3	4.5	0.1	0	0	1.4	0	0	16.9
CPz	0	0	0	0	0	3	0	0.4	0	2.9	0	7.7	9.9	1	0	2.9	0	3.1	0	5.8	14	12.2	0.1	2.2	1.3	0.5	67.9
CP4	0	0	0	0	0	0	0	0	0	20.6	0	0	0.2	6.4	1.1	0	3.1	0	0	0	11.4	4.1	0	0	0	0	47.1
T5	0	0	0	0	0	0	0	0	0	0	2.9	0.6	0	0	0	0.3	0	0	0	0.9	0	0	0	2.6	0	0	7.4
P3	0	0	0	0	0	0	0	3	0	0	0.1	9.3	0	0	0	4.5	5.8	0	0.9	0	2.2	0	0	3.7	0.5	0	30.4
Pz	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.1	14	0	0	2.2	0	4	0	3.2	1.9	1.5	27.2
P4	0	0	0	0	0	0	0	0	0	0.4	0	0	0.2	0.9	0.1	0	12.2	11.4	0	0	4	0	7.2	0	1.4	2.8	41.1
T6	0	0	0	0	0	0	0	0	0	0.1	0	0	0	0.7	2.1	0	0.1	4.1	0	0	0	7.2	0	0	0.1	2.8	17.6
O1	0	0	0	0	0	0	0	0	0	0	0.5	1.1	0	0	0	1.4	2.2	0	2.6	3.7	3.2	0	0	0	2.2	0.5	17.8
Oz	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1.3	0	0	0.5	1.9	1.4	0.1	2.2	0	3.4	11
O2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.5	0	0	0	1.5	2.8	2.8	0.5	3.4	0	11.8

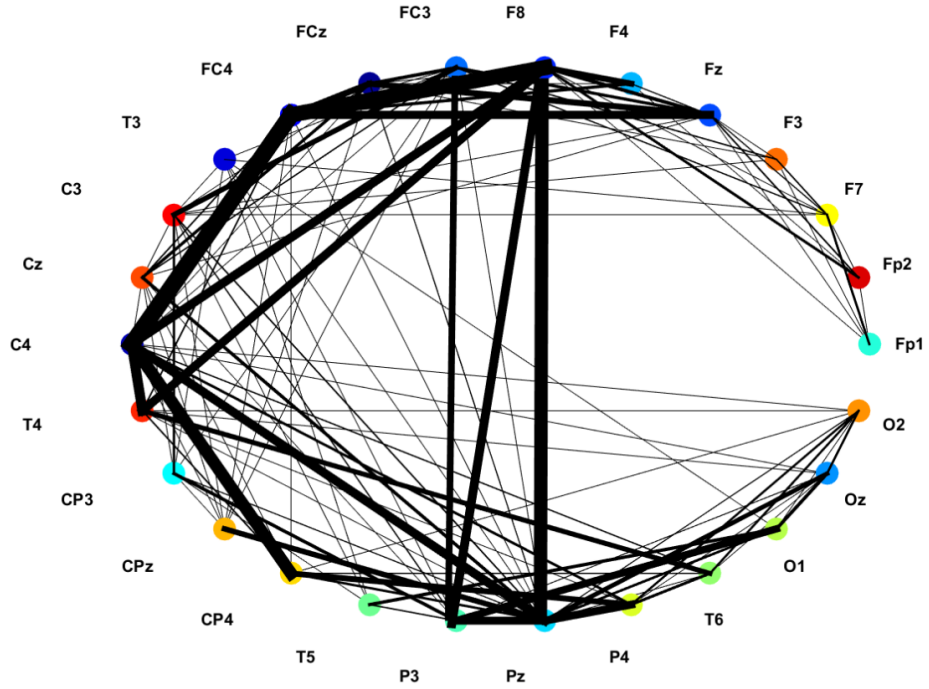


Figure C-3 The spike interaction between 26 neural groups representing 26 EEG channels in MMT group in GO task. The thicker the line that connects two nodes that represent the corresponding EEG channels, the more spikes were transmitted between corresponding groups

Table C-6 The amount of spike transformation between 26 neural groups representing 26 EEG channels in MMT group in GO task, computed as the number of spikes exchanged between 2 groups divided by the STDP time.

	Fp1	Fp2	F7	F3	Fz	F4	F8	FC3	FCz	FC4	T3	C3	Cz	C4	T4	CP3	CPz	CP4	T5	P3	Pz	P4	T6	O1	Oz	O2	Σ	
Fp1	0	0	1.6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1.7	
Fp2	0	0	0	0	0.5	0	1.7	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2.3	
F7	1.6	0	0	1.3	1	0	0	1.4	0	0	0.6	0.1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	6.4	
F3	0	0	1.3	0	0.3	0	0	0.4	0	0	0	0.1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	2.3	
Fz	0	0.5	1	0.3	0	0.8	2.9	3	6.1	6.4	0	0.1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	21.4	
F4	0	0	0	0	0.8	0	5.5	0	0	1.6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	8.1	
F8	0	1.7	0	0	2.9	5.5	0	0	0	12.3	0	0	0	9.2	9.1	0	0	0	0	9.4	10.1	0	0	0	0	0	60.4	
FC3	0	0	1.4	0.4	3	0	0	0	3.3	0	0.2	3.1	1.9	0.1	0	0.8	0	0	8.4	0.4	0	0	0	0	0	0	23.5	
FCz	0	0	0	0	6.1	0	0	3.3	0	3.7	0	0.1	3.5	1.9	0	0	0.1	0	0	0	0	0	0	0	0	0	19.1	
FC4	0	0	0	0	6.4	1.6	12	0	3.7	0	0	0	1.5	19	0.5	0	0.1	0.1	0	0	0.5	0	0	0	0	0	45.9	
T3	0	0	0.6	0	0	0	0	0	0.2	0	0	0.2	0	0	0	0.1	0	0	0.5	0.2	0	0	0	0	0	0	2.1	
C3	0	0	0.1	0.1	0.1	0	0	3.1	0.1	0	0.2	0	0.2	0	0	1.4	0	0	0.1	1	0.1	0	0	0	0	0	6.9	
Cz	0	0	0	0	0	0	0	1.9	3.5	1.5	0	0.2	0	0.8	0	0.1	0.6	0.1	0	0.2	2	0	0	0	0	0	11.3	
C4	0	0	0	0	0	0	0	9.2	0.1	1.9	19	0	0	0.8	0	11	0	0.1	12.1	0	0	8.3	2.1	2.7	0	0.1	0.5	68.6
T4	0	0	0	0	0	0	9.1	0	0	0.5	0	0	0	11	0	0	0	0	0.1	0	0	0.2	4.4	0	0	0.4	26	
CP3	0	0	0	0	0	0	0	0.8	0	0	0.1	1.4	0.1	0	0	0	0	0	0.1	3	1.2	0	0	0	0	0	6.9	
CPz	0	0	0	0	0	0	0	0	0.1	0.1	0	0	0.6	0.1	0	0	0	0	0	0	4.8	0	0	0	0	0	5.9	
CP4	0	0	0	0	0	0	0	0	0	0.1	0	0	0.1	12.1	0	0	0	0	0	0	0.5	3	0.1	0	0	0	16	
T5	0	0	0	0	0	0	0	0	0	0	0.5	0.1	0	0	0.1	0.1	0	0	0	1.2	0	0	0	3.6	0	0	5.7	
P3	0	0	0	0	0	0	9.4	8.4	0	0	0.2	1	0.2	0	0	3	0	0	1.2	0	8.1	0	0	6.2	1	0	39	
Pz	0	0	0	0	0	0	10	0.4	0	0.5	0	0.1	2	8.3	0	1.2	4.8	0.5	0	8.1	0	7	0.1	2.6	4.8	0.3	51.5	
P4	0	0	0	0	0	0	0	0	0	0	0	0	0	2.1	0.2	0	0	3	0	0	7	0	2.9	0	0.8	1.7	18	
T6	0	0	0	0	0	0	0	0	0	0	0	0	0	2.7	4.4	0	0	0	0.1	0	0	0.1	2.9	0	0	1	11.5	
O1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3.6	6.2	2.6	0	0	0	3.9	0	16.6	
Oz	0	0	0	0	0	0	0	0	0	0	0	0	0	0.1	0	0	0	0	1	4.8	0.8	0	3.9	0	2.5	13.5		
O2	0	0	0	0	0	0	0	0	0	0	0	0	0	0.5	0.4	0	0	0	0	0	0.3	1.7	1	0	2.5	0	6.9	

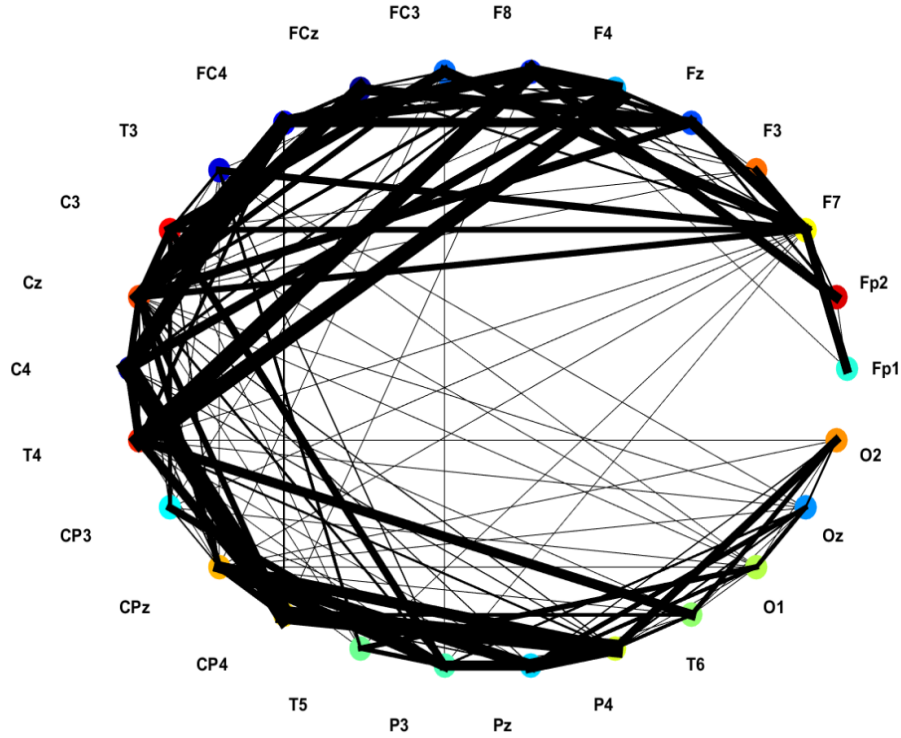


Figure C-4 The spike interaction between 26 neural groups representing 26 EEG channels in MMT group in NOGO task. The thicker the line that connects two nodes that represent the corresponding EEG channels, the more spikes were transmitted between corresponding groups.

Table C-7 The amount of spike transformation between 26 neural groups representing 26 EEG channels in MMT group in NOGO task, computed as the number of spikes exchanged between 2 groups divided by the STDP time.

	Fp1	Fp2	F7	F3	Fz	F4	F8	FC3	FCz	FC4	T3	C3	Cz	C4	T4	CP3	CPz	CP4	T5	P3	Pz	P4	T6	O1	Oz	O2	Σ
Fp1	0	0.2	8.1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	8.4
Fp2	0.2	0	1.5	0	2.5	0.7	8.5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	13.5
F7	8.1	1.5	0	6.2	9.5	0	0.8	9.5	2.1	0	6.1	6.9	5.5	0	0	0	0	0	0	0.1	0	0	0	0	0	0	56.9
F3	0	0	6.2	0	0.4	0	0	1.2	0.1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	8.1
Fz	0	2.5	9.5	0.4	0	3.5	4.4	1.4	8.9	6.8	0	0	7.7	0	0	0	0	0	0	0	0	0	0	0	0	0	45.5
F4	0	0.7	0	0	3.5	0	13	0	0.7	7.2	0	0	0.2	7.1	16	0	0	0	0	0	0	0	0	0	0	0	48.9
F8	0	8.5	0.8	0	4.4	13	0	0	0.5	11.2	0	0	3.1	2.4	15	0	0	0	0	0	0	0	0	0	0	0	59.4
FC3	0	0	9.5	1.2	1.4	0	0	0	1.8	0	0.2	9.2	5	0	0	0.1	0.1	0	0	0.1	0	0	0	0	0	0	29
FCz	0	0	2.1	0.1	8.9	0.7	0.5	1.8	0	3.2	0	0.3	13.5	0	0	0	0	0	0	0	0	0	0	0	0	0	31.4
FC4	0	0	0	0	6.8	7.2	11	0	3.2	0	0	0	13.2	14.7	3.4	0	0.1	1.7	0	0	0	0	0	0	0	0	61.7
T3	0	0	6.1	0	0	0	0	0.2	0	0	0	2.4	0	0	0	0.3	0	0	2.9	1.2	0	0	0	0.1	0	0	13.6
C3	0	0	6.9	0	0	0	0	9.2	0.3	0	2.4	0	6.8	0	0	3.9	1.5	0	0.3	6.8	0	0	0	0	0	0	38.6
Cz	0	0	5.5	0	7.7	0.2	3.1	5	13.5	13.2	0	6.8	0	6.2	1.5	0.7	5	5.7	0	2.2	1	1.1	0	0	0	0	79.2
C4	0	0	0	0	0	7.1	2.4	0	0	14.7	0	0	6.2	0	8.1	0	1.2	17.4	0	0	0	0.6	0	0	0	0	58.2
T4	0	0	0	0	0	16	15	0	0	3.4	0	0	1.5	8.1	0	0	0	8.3	0	0	0	1.9	8.3	0	0	0.1	63
CP3	0	0	0	0	0	0	0	0.1	0	0	0.3	3.9	0.7	0	0	0	1.3	0	0.3	6.6	0.2	0	0	0	0	0	13.8
CPz	0	0	0	0	0	0	0	0.1	0	0.1	0	1.5	5	1.2	0	1.3	0	6.1	0	9.1	18.9	9.5	0	0.2	0.1	0	53.8
CP4	0	0	0	0	0	0	0	0	0	1.7	0	0	5.7	17.4	8.3	0	6.1	0	0	0.2	18.9	0.8	0	3.3	4.4	0	62.1
T5	0	0	0	0	0	0	0	0	0	0	2.9	0.3	0	0	0	0.3	0	0	0	4	0	0	0	4.8	0	0	12.5
P3	0	0	0.1	0	0	0	0	0.1	0	0	1.2	6.8	2.2	0	0	6.6	9.1	0	4	0	8.5	0	0	4.7	2	0	45.9
Pz	0	0	0	0	0	0	0	0	0	0	0	0	1	0	0	0.2	18.9	0.8	0	8.5	0	11.5	0	3.3	4.4	0.7	49.7
P4	0	0	0	0	0	0	0	0	0	0	0	0	1.1	0.6	1.9	0	9.5	18.8	0	0	11.5	0	3.4	0.1	1.6	7.1	56.2
T6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	8.3	0	3.1	0	0	0	0	3.4	0	0	0	4.4	19.4
O1	0	0	0	0	0	0	0	0	0	0	0.1	0	0	0	0	0	0.2	0	4.8	4.7	3.3	0.1	0	0	3.8	0.3	17.8
Oz	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.1	0	0	2	4.4	1.6	0	3.8	0	1.9	14.1
O2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.1	0	0	0	0	0	0.7	7.1	4.4	0.3	1.9	0	14.6

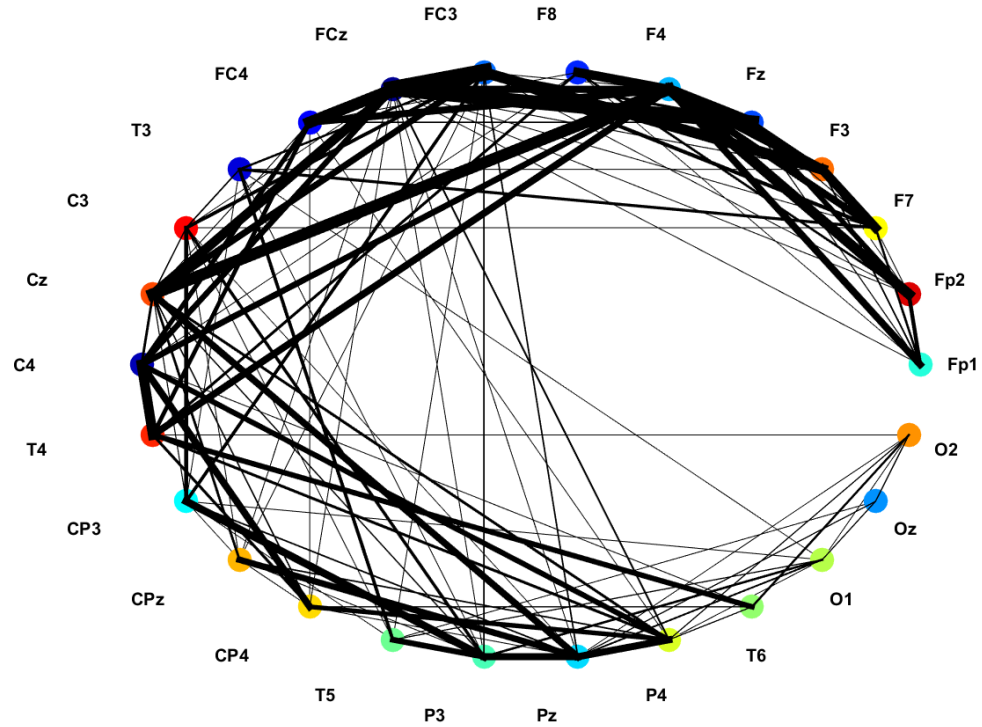


Figure C-5 The spike interaction between 26 neural groups representing 26 EEG channels in OP group in GO task. The thicker the line that connects two nodes that represent the corresponding EEG channels, the more spikes were transmitted between corresponding groups.

Table C-8 The amount of spike transformation between 26 neural groups representing 26 EEG channels in OP group in GO task, computed as the number of spikes exchanged between 2 groups divided by the STDP time.

	Fp1	Fp2	F7	F3	Fz	F4	F8	FC3	FCz	FC4	T3	C3	Cz	C4	T4	CP3	CPz	CP4	T5	P3	Pz	P4	T6	O1	Oz	O2	Σ
Fp1	0	3.3	1.4	1.4	0.5	4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	10.9
Fp2	3.3	0	0	0.2	4.3	5.8	2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	15.9
F7	1.4	0	0	5.9	0.5	4	0	3.2	0.1	0	2.5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	18
F3	1.4	0.2	5.9	0	4	0	0	9.1	0.5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	21.3
Fz	0.5	4.3	0.5	4	0	9.1	0	2.4	11.8	1.2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	34.2
F4	4	5.8	4	0	9.1	0	7.9	0	10.1	9.6	0	0	12	7.2	7.9	0	0	0	0	0	0	0	0	0	0	0	77.9
F8	0	2	0	0	0	7.9	0	0	0	0.4	0	0	0	0	2.3	0	0	0	0	0	0	0	0	0	0	0	12.8
FC3	0	0	3.2	9.1	2.4	0	0	0	9.4	0	1.8	2.3	5.4	0	0	2.8	0.2	0	0	0.8	0.8	0	0	0	0	0	38.9
FCz	0	0	0.1	0.5	11.8	10	0	9.4	0	9.9	0	0.5	6.5	4.8	0	0	0.2	0.4	0	0	0.4	0.9	0	0	0	0	56.1
FC4	0	0	0	0	1.2	9.6	0.4	0	9.9	0	0	0	0.5	3.2	3.6	0	0	0.1	0	0	0	0.1	0	0	0	0	28.8
T3	0	0	2.5	0	0	0	0	1.8	0	0	0	1.2	0	0	0	0.3	0	0	2.8	0.4	0	0	0	0	0	0	9.2
C3	0	0	0	0	0	0	0	2.3	0.5	0	1.2	0	0.7	0	0	3.5	0.1	0	0.3	2.6	0.3	0	0	0	0	0	11.8
Cz	0	0	0	0	0	12	0	5.4	6.5	0.5	0	0.7	0	2.1	0	0.5	2.1	0.7	0	0.7	6.8	2.3	0	0	0	0	40.8
C4	0	0	0	0	0	7.2	0	0	4.8	3.2	0	0	2.1	0	8.4	0	0	7.6	0	0	0	5.6	0	0	0	0	39.3
T4	0	0	0	0	0	7.9	2.3	0	0	3.6	0	0	0	8.4	0	0	0	2.4	0	0	0	2.3	5.6	0	0	0	32.8
CP3	0	0	0	0	0	0	0	2.8	0	0	0.3	3.5	0.5	0	0	0	0	0	0.3	6.4	0.8	0	0	0	0	0	15
CPz	0	0	0	0	0	0	0	0.2	0.2	0	0	0.1	2.1	0	0	0	0	0	0	0.3	5.4	1.3	0	0	0	0	24.2
CP4	0	0	0	0	0	0	0	0	0.4	0.1	0	0	0.7	7.6	2.4	0	0	0	0	0	0.3	2.5	0.6	0	0	0	15.1
T5	0	0	0	0	0	0	0	0	0	0	2.8	0.3	0	0	0	0.3	0	0	0	4.9	0	0	0	0	0	0	8.5
P3	0	0	0	0	0	0	0	0.8	0	0	0.4	2.6	0.7	0	0	6.4	0.3	0	4.9	0	6.6	0	0	1.1	0	0	24.2
Pz	0	0	0	0	0	0	0	0.8	0.4	0	0	0.3	6.8	0	0	0.8	5.4	0.3	0	6.6	0	8.3	0	0.9	0.9	0	32
P4	0	0	0	0	0	0	0	0	0.9	0.1	0	0	2.3	5.6	2.3	0	1.3	2.5	0	0	8.3	0	0	0.1	0	0	23.9
T6	0	0	0	0	0	0	0	0	0	0	0	0	0	5.6	0	0	0.6	0	0	0	0	0	0	0	0	1.8	8.2
O1	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1.1	0.9	0.1	0	0	1	0.1	3.5
Oz	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1	0	1	2.1
O2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	1.8	0.1	1	0	0	3.2

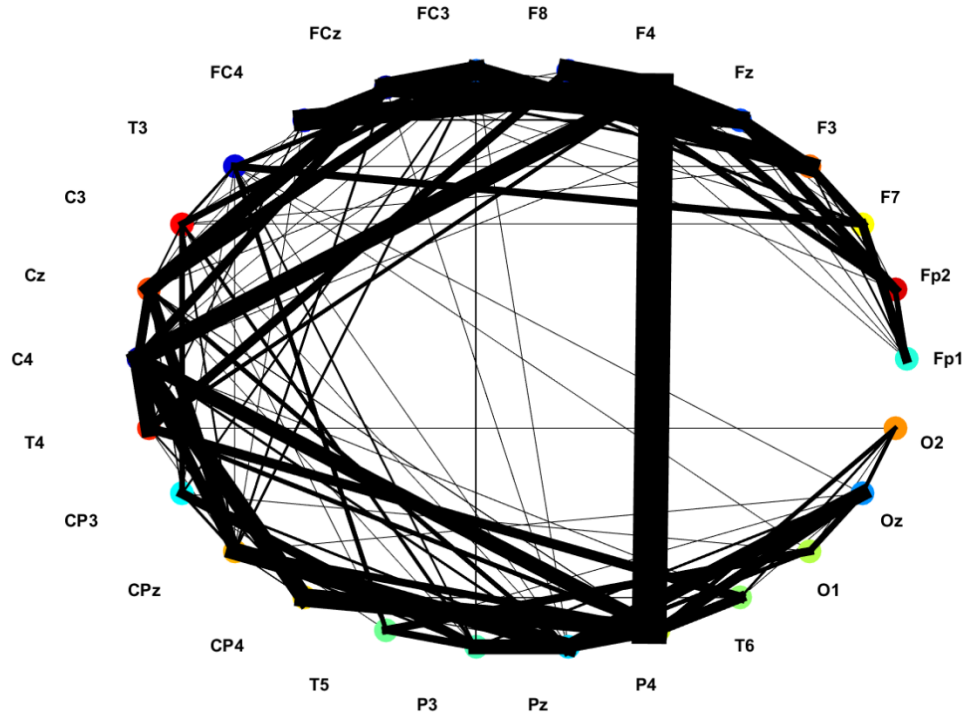


Figure C-6 The spike interaction between 26 neural groups representing 26 EEG channels in OP group in NOGO task. The thicker the line that connects two nodes that represent the corresponding EEG channels, the more spikes were transmitted between corresponding groups.

Table C-9 The amount of spike transformation between 26 neural groups representing 26 EEG channels in OP group in NOGO task, computed as the number of spikes exchanged between 2 groups divided by the STDP time.

	Fp1	Fp2	F7	F3	Fz	F4	F8	FC3	FCz	FC4	T3	C3	Cz	C4	T4	CP3	CPz	CP4	T5	P3	Pz	P4	T6	O1	Oz	O2	Σ
Fp1	0	4.8	9.6	1.7	0.5	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	16.8
Fp2	4.8	0	0.1	1.5	8.3	9.4	8.8	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	33.2
F7	9.6	0.1	0	10.3	0.3	0	0	3	0	0	6.6	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	30.2
F3	1.7	1.5	10	0	11.4	0.1	0	16.2	2.4	0	0.4	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	44.5
Fz	0.5	8.3	0.3	11.4	0	14	8.1	4	15.5	0.9	0	0	0	0.3	0	0	0	0	0	0	0	0	0	0	0	0	64
F4	0	9.4	0	0.1	14.3	0	18	0.1	9.7	18.4	0	0	0.1	16.1	3.5	0	0	0	0	0	0	25.2	0	0	0	0	115.8
F8	0	8.8	0	0	8.1	18	0	0	0	0.2	0	0	0	0.4	8.2	0	0	0	0	0	0	0	0	0	0	0	44.3
FC3	0	0	3	16.2	4	0.1	0	17.1	0	4.4	7.3	9.8	0.3	0	1.8	2.2	0	0	0.8	0.4	0	0	0	0	0	0	68.1
FCz	0	0	0	2.4	15.5	9.7	0	17.1	0	6.7	0	0	13.8	12.8	0	2.1	0	0	0	0.1	0	0	0	0	0	0	80.8
FC4	0	0	0	0	0.9	18	0.2	0	6.7	0	0	0	0.2	0.1	1.4	0	0	0	0	0	0	0	0	0	0	0	28.2
T3	0	0	6.6	0.4	0	0	0	4.4	0	0	2.3	0	0	0	0.5	0	0	4.1	2.7	0.1	0	0	0.1	0	0	0	21.7
C3	0	0	0	0	0	0	0	7.3	0	0	2.3	0	1.5	0	0	4.8	1.8	0	0.1	3	1.2	0	0	0	0	0	22.7
Cz	0	0	0	0	0	0.1	0	9.8	13.8	0.2	0	1.5	0	8	0	0.7	14.7	0	0	0.4	4.2	0.6	0	0	0	0	54.4
C4	0	0	0	0	0.3	16	0.4	0.3	12.8	0.1	0	0	8	0	20	0	9.5	21.6	0	0	2.2	19.4	0.5	0	0	0	111.5
T4	0	0	0	0	0	3.5	8.2	0	0	1.4	0	0	0	19.6	0	0	1.7	0	0	0	4.1	8.6	0	0	0	0	47.5
CP3	0	0	0	0	0	0	0	1.8	0	0	0.5	4.8	0.7	0	0	0	2.4	0	0.3	8.3	2.3	0	0	0	0	0	21.4
CPz	0	0	0	0	0	0	0	2.2	2.1	0	0	1.8	14.7	9.5	0	2.4	0	1.1	0	2.1	21.6	6.6	0	0	0	0	64.6
CP4	0	0	0	0	0	0	0	0	0	0	0	0	0	21.6	1.7	0	1.1	0	0	0	0.6	21.2	0.6	0	0	0	47
T5	0	0	0	0	0	0	0	0	0	0	4.1	0.1	0	0	0	0.3	0	0	0	6.4	0	0	0	7.2	0.2	0	18.5
P3	0	0	0	0	0	0	0	0.8	0	0	2.7	3	0.4	0	0	8.3	2.1	0	6.4	0	9	0	0	2.5	2.2	0	38
Pz	0	0	0	0	0	0	0	0.4	0.1	0	0.1	1.2	4.2	2.2	0	2.3	21.6	0.6	0	9	0	23.3	0	3.8	13	0	82.4
P4	0	0	0	0	0	25	0	0	0	0	0	0	0.6	19.4	4.1	0	6.6	21.2	0	0	23.3	0	9.2	0	9	3.9	122.9
T6	0	0	0	0	0	0	0	0	0	0	0	0	0	0.5	8.6	0	0.6	0	0	0	0	9.2	0	0	0.1	2.1	21.3
O1	0	0	0	0	0	0	0	0	0	0	0.1	0	0	0	0	0	0	0	7.2	2.5	3.8	0	0	0	5.6	0	19.5
Oz	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0.2	2.2	13	9	0.1	5.6	0	3.5	33.9
O2	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	0	3.9	2.1	0	3.5	0	9.7

Table C-10 T-test was applied to the last column of the FNI graphs for H, MMT and OP groups to evaluate the models' significance in GO and NOGO. *P*-values are reported in the last row.

GO			NOGO		
H	MMT	OP	H	MMT	OP
11.3	1.7	10.9	8.1	8.4	16.8
35.3	2.3	15.9	38.1	13.5	33.2
24.5	6.4	18	17.7	56.9	30.2
37.2	2.3	21.3	38.5	8.1	44.5
60.1	21.4	34.2	25.4	45.5	64
44.2	8.1	77.9	43.8	48.9	115.8
31.7	60.4	12.8	40	59.4	44.3
24.6	23.5	38.9	43.2	29	68.1
41.1	19.1	56.1	18.7	31.4	80.8
85.8	45.9	28.8	63.6	61.7	28.2
17	2.1	9.2	11.1	13.6	21.7
77.5	6.9	11.8	38.5	38.6	22.7
61	11.3	40.8	45.4	79.2	54.4
33.5	68.6	39.3	17.1	58.2	111.5
24.5	26	32.8	15.7	63	47.5
40.3	6.9	15	16.9	13.8	21.4
129.5	5.9	9.9	67.9	53.8	64.6
19.3	16	15.1	47.1	62.1	47
10.9	5.7	8.5	7.4	12.5	18.5
37.7	39	24.2	30.4	45.9	38
29.7	51.5	32	27.2	49.7	82.4
38.9	18	23.9	41.1	56.2	122.9
29.1	11.5	8.2	17.6	19.4	21.3
8.9	16.6	3.5	17.8	17.8	19.5
26.9	13.5	2.1	11	14.1	33.9
18.5	6.9	3.2	11.8	14.6	9.7
H - M	H - OP	M - OP	H - M	H - OP	M - OP
0.005	0.012	0.4	0.03	0.003	0.05

C.7 Test Accuracy Evaluation using F-Score Measurement

In a binary classifier of two classes (positive and negative), if positive samples are classified into positive class, they are called true positive (*TP*). However, if they are misclassified to negative class, they are called false positive (*FP*). On the other hand, for the negative class, if negative samples are classified into negative class, they are called true negative (*TN*) and

if they are misclassified to positive class, they are called false negative (FN). To compute the score, both the precision p and the recall r of the test results need to be considered. Precision p (shown in Relation C-1) is measured with respect to the number TP results divided by the number of all positive results returned by the classifier. Recall r is measured according to the number of TP results divided by the number of all relevant samples (shown in Relation C-2). The F_{Score} (shown in Relation C-3) is the harmonic average of the precision and recall ($0 < F_{Score} < 1$), where the best value is 1 (perfect precision and recall) and worst is 0.

$$P = \frac{TP}{TP + FP} \quad (C-1)$$

$$r = \frac{TP}{TP + FN} \quad (C-2)$$

$$F_{Score} = 2 \cdot \frac{precision \cdot recall}{precision + recall} \quad (C-3)$$

C.8 SNN Parameter Optimisation

For optimisation, I performed an exhaustive grid-search on combination of parameters for every sample's model. Each parameter was searched within a range, specified by the minimum and maximum, through several iterations related to the number of steps for moving from minimum to maximum. For every model creation, I chose three main parameters (STDP learning rate, *mod* and *drift* parameters) to be optimised. The parameters were selected by assigning 10 steps between the minimum and maximum values of each parameter range. Therefore, for every model creation, 1000 iterations of training (using all samples except the holdout sample) and testing (using the single holdout sample) were performed with different combination of these three parameters. Then the parameters that resulted in the best accuracy

in most of the iterations have been reported as the optimal parameters, shown in Table C-11.

The TBR threshold, neuron firing threshold and small-world radius parameters were fixed to 0.5, 0.5, and 2.5 respectively.

Table C-11 The optimal NeuCube parameters that resulted from a grid-search to optimise the classification accuracy as an objective function.

Session	EEG sample files used in NeuCube classification	<i>mod</i>	<i>drift</i>	STDP rate
I	H subjects in Go <i>vs</i> H subjects in NOGO	0.56	0.03	0.003
	MMT subject in GO <i>vs</i> MMT subject in NOGO	0.56	0.02	0.0019
	OP subjects in GO <i>vs</i> OP subjects in NOGO	0.45	0.025	0.0019
II	MMT subject <i>vs</i> H subjects (GO task)	0.56	0.01	0.002
	Opiate subjects <i>vs</i> H subjects (GO task)	0.45	0.01	0.006
	MMT subject <i>vs</i> Opiate subjects (GO task)	0.51	0.01	0.005
III	MMT subjects <i>vs</i> H subjects (NOGO task)	0.51	0.02	0.002
	Opiate subjects <i>vs</i> H subjects (NOGO task)	0.45	0.03	0.002
	MMT subjects <i>vs</i> OP subjects (NOGO task)	0.45	0.025	0.0019
	Parameter range	0.4-0.95	0.001-0.5	0.001-0.01

C.9 Parameter Setting for Conventional Methods

To perform the classification problem using conventional classifiers, first the spatio-temporal sample representation needed to be transferred into vector-based representation. The information of every input sample in SNN is presented in a 2-D matrix, where columns are the STBD variables and rows are the continuous temporal information. To transfer one sample data into one feature vector, I aggregated every l time-points (a time-window) of a variable to obtain one value. Figure C-7 graphically shows the representation of STBD samples for NeuCube (in a) *versus* conventional methods (in b).

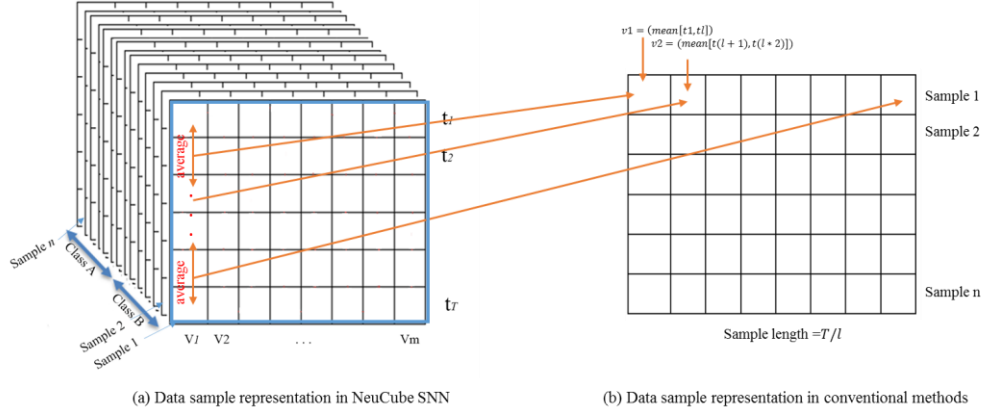


Figure C-7 (a) The 3-dimensional input data format for NeuCube SNN architecture. Every sample is presented as a 2-D information, where columns represented the data variables (v) and rows are the temporal information (t). This example contains 2 classes of samples; (b) transformation of the spatio-temporal samples into vector-based samples by taking average over the temporal data with a time-window= l .

In the case of EEG data used in Chapter 4, each sample size was (75, 26), where 26 EEG channels recorded 75 time points of temporal information. To transfer this 2-D data into one vector, I defined $l = 15$; so that, I took average over every 15 time-points which resulted to obtain 5 values ($75/15 = 5$) for each EEG channel. Since, there were 26 channels, the dimension of the final obtained sample vector is ($26 \times 5 = 130$). To optimise the conventional classifiers, I performed as follows:

In MLP, the optimal number of hidden neurons was calculated using the formula suggested by Trenn (Stephan, 2008) and also mentioned in (Gnana & Subramaniam, 2013). The number of hidden neurons Nh is defined as $(n + no - 1)/2$, where n is the number of input neurons (input EEG channels) and no is the number of output neurons for the classifier. Therefore, In Chapter 4 the number of hidden neurons in MLP was $(26 + 2)/2 = 14$. For the rest of the main parameters (learning rate, momentum, training iteration, number of hidden layers), I performed the classification experiment several times with different values to find the optimal parameters which resulted in the best classification accuracy. Table C-12 represents

that for each experimental session, the classification was performed with different parameter settings (learning rate range= [0.01, 0.5], momentum range= [0.1, 0.9], training iteration range= [500, 1500], and number of hidden layer range= [2, 6]). In SVM, the classification accuracy was performed based on two kernel functions: polynomial and radial basis function (RBF). I considered the polynomial degree from 2 to 5 and the RBF width from 0.2 to 1 with 0.2 intervals. The best SVM's parameters that resulted in the best classification accuracy are reported in Table C-13.

Table C-12 The MLP optimal parameters were found after performing the experiments several times using different parameter settings (learning rate (LR) = [0.01, 0.5], momentum (M) = [0.1, 0.9], training iteration (TI) = [500, 1500], and number of hidden layer (HL) = [2, 6]).

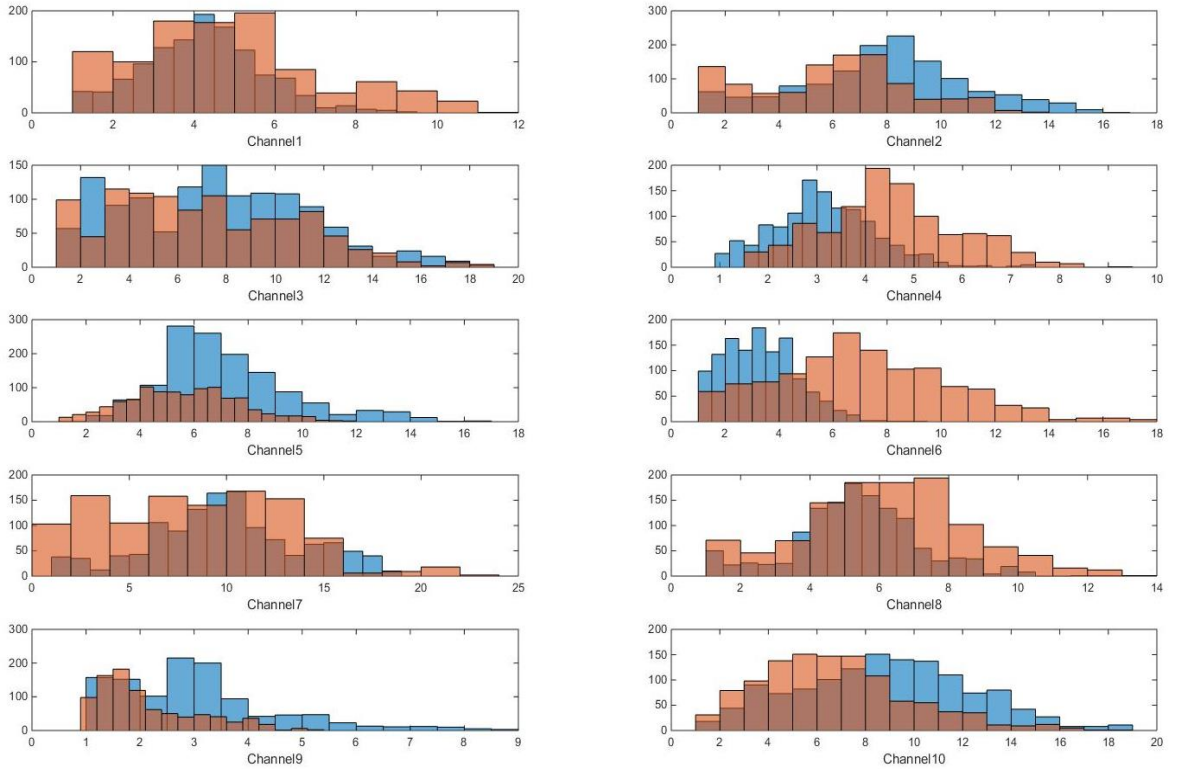
MLP optimised parameters that resulted the best classification accuracy in experimental <i>session I</i>				
Runs	5	10	20	20
Parameter [range]	HL [2,6]	IT[500,1500]	LR[0.01,0.5]	M [0.1,0.9]
H in GO vs. NOGO	5	800	0.1	0.8
MMT in Go vs. NOGO	5	1000	0.1	0.7
MMT in Go vs. NOGO	4	800	0.2	0.7
Optimised parameters that resulted the best classification accuracy in experimental <i>session II</i>				
Parameter [range]	HL [2,6]	IT[500,1500]	LR[0.01,0.5]	M [0.1,0.9]
MMT vs. H subjects	5	1100	0.1	0.6
OP vs. H subjects	4	1200	0.3	0.8
MMT vs. OP subjects	4	1200	0.2	0.6
Optimised parameters that resulted the best classification accuracy in experimental <i>session III</i>				
Parameter [range]	HL [2,6]	IT[500,1500]	LR[0.01,0.5]	M [0.1,0.9]
MMT vs. H subjects	6	1000	0.3	0.6
Op vs. H subjects	5	1200	0.4	0.8
MMT vs. OP subjects	5	1200	0.3	0.8

Table C-13 The SVM optimal parameters that resulted the best classification accuracy were found after performing the experiments several times with different parameter setting (polynomial degree within [2, 5] and (RBF) kernel degree within [0.2, 1]).

SVM optimised parameters that resulted the best classification accuracy in experimental <i>session I</i>									
Kernel	polynomial [2,5]				RBF [0.2,1]				
Kenner degree	2	3	4	5	0.2	0.4	0.6	0.8	1
H in GO vs. NOGO	55.0	65.0	63.0	52.0	47.0	48.0	55.0	54.0	48.0
MMT in Go vs. NOGO	53.0	63.0	59.0	50.0	55.0	38.0	55.5	52.0	55.0
MMT in Go vs. NOGO	67.0	64.0	66.0	58.0	55.0	60.0	50.0	50.0	50.0
SVM optimised parameters that resulted the best classification accuracy in experimental <i>session II</i>									
MMT vs. H subjects	70.0	62.0	60.0	50.0	65.0	68.0	54.0	44.0	45.0
OP vs. H subjects	51.0	68.0	63.0	50.0	55.0	38.0	55.0	52.0	55.0
MMT vs. OP subjects	67.0	64.0	66.0	66.0	62.0	54.0	54.0	40.0	40.0
SVM optimised parameters that resulted the best classification accuracy in experimental <i>session III</i>									
MMT vs. H subjects	50.0	53.0	63.0	53.0	60.0	62.0	64.0	51.0	40.0
Op vs. H subjects	68.0	73.0	71.0	50.0	51.0	48.0	50.0	48.0	44.0
MMT vs. OP subjects	56.0	63.0	60.0	55.0	56.0	62.0	62.0	55.0	55.0

Appendix D Dynamic Clustering Patterns

The discriminative patterns in some EEG channels was also captured through visualisation of the samples' distribution across the classes, as shown in Figure D-1. Quantitative information about the changes in the size of the clusters are presented in Tables D-1 to D-6. I have applied a t-test to the $P_{max}(t)$, the area under the curve, and the mid of potential as shown in Tables D-7 to D-9.



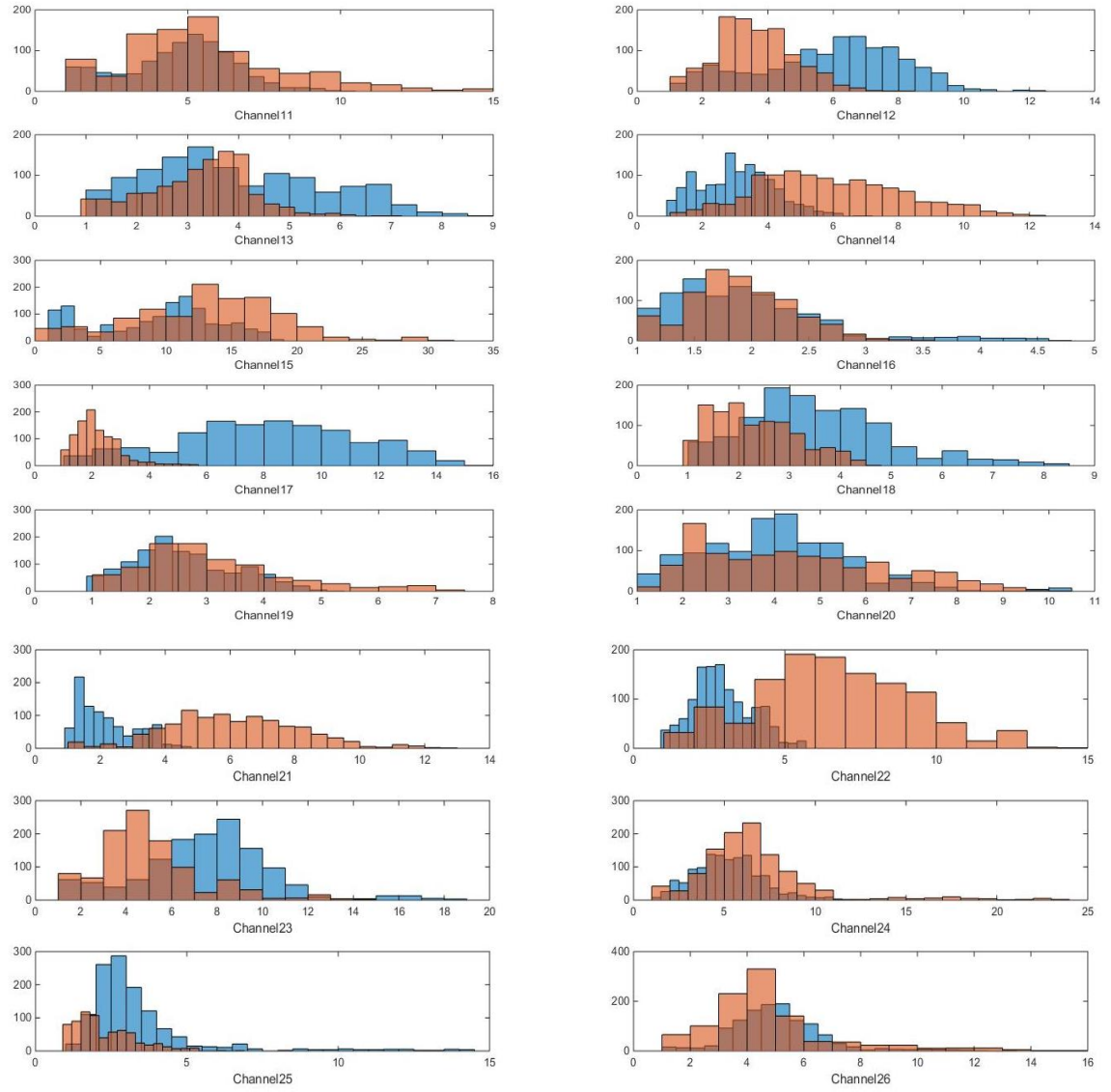


Figure D-1 Distribution of the samples with respect to the PSP rates in class H (red) and class OP (blue) for 26 EEG channels.

Table D-1 Clusters sizes are changing over time during the STDP learning in SNN model of 21 H subjects in GO (in total 1575 time points were entered and learnt).

	Fp1	Fp2	F7	F3	Fz	F4	F8	FC3	FCz	FC4	T3	C3	Cz	C4	T4	CP3	CPz	CP4	T5	P3	Pz	P4	T6	O1	Oz	O2
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
3	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
4	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
5	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
6	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
7	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
8	1	1	1	1	43	1	1	1	30	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
9	1	1	1	19	43	41	1	26	30	1	1	27	1	1	1	1	1	1	1	1	1	1	1	1	1	1
10	1	1	38	17	47	41	1	26	30	1	1	27	1	1	1	1	1	1	1	1	1	1	1	1	1	1
11	1	1	38	17	47	41	1	35	30	1	1	18	1	1	44	18	1	1	1	1	1	1	1	1	1	1
12	1	1	38	17	47	41	1	35	30	1	1	18	1	1	44	18	1	1	1	1	1	1	1	1	1	1
13	1	1	38	17	47	41	1	35	30	1	1	18	1	1	44	18	1	1	1	1	34	35	30	36	15	26
14	1	1	38	17	47	41	1	35	30	1	1	18	1	1	39	18	1	1	1	1	21	36	35	36	31	22
15	1	1	38	17	47	41	1	35	30	1	1	18	1	1	39	18	1	1	1	1	0	26	37	36	52	30
16	1	1	38	17	47	41	1	35	30	1	1	18	1	1	39	18	1	1	1	1	0	26	37	41	51	34
17	1	37	38	17	47	37	1	26	30	1	35	27	1	1	39	18	1	1	15	1	16	26	37	41	41	34
18	1	37	38	17	47	37	1	26	30	1	35	27	1	1	39	18	1	1	15	1	16	26	37	41	41	34
1574	86	73	76	25	74	34	116	42	30	65	76	34	49	22	105	10	86	27	61	26	18	18	82	65	19	63
1575	86	72	76	25	74	34	117	42	30	65	76	34	49	22	105	10	86	27	61	26	18	18	82	65	19	63

Table D-2 Clusters sizes are changing over time during the STDP learning in SNN model of 29 MMT subjects in GO (in total 2175 time points were entered and learnt).

	Fp1	Fp2	F7	F3	Fz	F4	F8	FC3	FCz	FC4	T3	C3	Cz	C4	T4	CP3	CPz	CP4	T5	P3	Pz	P4	T6	O1	Oz	O2
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
3	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
4	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
5	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
6	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
7	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
8	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
9	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
10	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
11	28	37	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
12	28	37	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
13	28	37	1	25	1	28	45	42	1	45	1	1	1	1	45	1	1	1	1	1	1	1	1	1	1	1
14	28	37	1	25	34	31	42	42	1	68	1	1	1	0	45	1	1	1	1	1	1	1	1	1	1	1
15	28	34	1	0	34	34	42	67	1	45	1	1	1	23	45	1	1	1	1	28	1	1	1	1	1	1
16	28	34	1	0	34	34	42	67	1	63	1	1	1	18	45	1	1	1	1	28	30	1	1	1	1	1
17	28	37	1	0	34	31	42	67	1	63	32	1	1	18	45	1	0	1	1	28	52	1	1	1	1	1
18	28	47	1	0	34	31	39	67	1	63	32	1	25	18	45	1	0	1	1	25	55	1	1	1	1	1
2174	40	14	178	4	63	5	210	57	22	56	64	15	11	96	114	8	11	13	48	51	96	29	70	72	49	35
2175	40	14	178	4	63	5	210	57	22	56	64	15	11	97	113	8	11	13	48	51	96	29	70	72	49	35

Table D-3 Clusters sizes are changing over time during the STDP learning in SNN model of 18 OP subjects in GO (in total 1350 time points were entered and learnt).

	Fp1	Fp2	F7	F3	Fz	F4	F8	FC3	FCz	FC4	T3	C3	Cz	C4	T4	CP3	CPz	CP4	T5	P3	Pz	P4	T6	O1	Oz	O2	
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
2	1	35	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
3	24	35	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
4	24	35	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
5	24	35	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
6	24	35	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
7	24	35	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
8	24	35	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
9	24	35	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
10	24	35	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	32	39	40	1	24	
11	24	35	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	27	0	0	1	31	39	92	0	25
12	24	35	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	27	20	0	1	31	39	72	0	25
13	24	35	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	35	20	14	1	22	53	58	0	26
14	24	35	1	18	39	41	1	36	12	1	1	27	40	1	1	1	1	1	35	20	14	1	22	53	58	0	26
15	24	31	1	18	40	44	1	37	24	1	1	27	27	1	40	1	1	1	35	20	14	1	23	47	58	0	31
16	24	31	1	18	40	44	1	37	24	1	1	27	27	1	40	1	1	1	35	20	14	1	23	47	58	0	31
17	24	32	36	13	39	44	1	36	12	1	1	27	40	1	40	1	26	35	20	14	1	22	51	58	0	28	
18	18	32	42	13	39	44	1	36	12	1	1	41	40	1	40	1	27	35	20	13	1	21	51	42	16	29	
1349	52	67	135	8	42	95	88	89	13	22	77	14	37	46	174	15	13	39	106	21	39	64	68	83	9	47	
1350	52	67	135	8	41	95	88	90	13	22	77	14	37	46	174	15	13	39	106	21	39	64	68	83	9	47	

Table D-4 Clusters sizes are changing over time during the STDP learning in SNN model of 21 H subjects in NOGO (in total 1575 time points were entered and learnt).

	Fp1	Fp2	F7	F3	Fz	F4	F8	FC3	FCz	FC4	T3	C3	Cz	C4	T4	CP3	CPz	CP4	T5	P3	Pz	P4	T6	O1	Oz	O2
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	1	1	1	1	1	1	1	34	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
3	1	1	1	1	1	1	1	34	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
4	1	1	1	1	1	1	1	34	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
5	1	1	1	1	1	1	1	34	1	41	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
6	1	1	1	1	1	1	1	34	1	41	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
7	1	1	1	1	1	1	1	34	1	41	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
8	1	1	1	1	1	1	1	34	1	41	33	1	1	1	1	27	1	1	17	1	1	1	1	1	1	1
9	1	1	1	1	1	1	1	34	1	41	33	1	1	1	1	27	1	1	17	1	1	1	1	1	1	1
10	1	39	1	1	1	1	1	34	1	41	33	1	1	1	1	27	1	1	17	1	1	1	1	1	1	1
11	1	39	1	1	1	1	46	34	1	41	52	1	1	1	44	27	1	1	17	1	1	1	1	1	1	1
12	1	37	1	1	1	1	48	34	1	41	52	1	1	1	44	27	1	1	17	1	1	1	33	1	1	1
13	1	37	1	1	1	1	48	34	1	41	52	1	1	1	44	27	1	1	17	1	1	1	33	1	1	30
14	1	37	1	1	1	1	48	34	1	41	53	1	1	1	44	27	1	1	16	1	1	1	33	33	26	30
15	1	37	1	1	1	1	48	34	1	41	53	1	1	1	44	27	1	1	16	1	1	1	33	33	26	30
16	1	37	1	1	1	1	48	34	1	41	53	1	1	1	44	27	1	1	16	1	1	1	33	33	26	30
17	27	39	1	1	42	1	46	34	1	41	53	1	1	1	44	27	1	1	16	1	1	1	33	33	26	30
18	27	39	1	1	42	1	46	34	1	41	53	1	1	1	44	27	1	1	16	1	1	1	33	33	26	30
1574	78	109	54	52	31	25	135	33	18	55	108	34	46	21	97	17	81	22	23	7	21	45	84	121	21	49
1575	78	109	54	52	31	25	135	33	18	55	108	34	46	21	97	17	81	22	23	7	21	45	84	121	21	49

Table D-5 Clusters sizes are changing over time during the STDP learning in SNN model of 26 MMT subjects in NOGO (in total 2325 time points were entered and learnt).

	Fp1	Fp2	F7	F3	Fz	F4	F8	FC3	FCz	FC4	T3	C3	Cz	C4	T4	CP3	CPz	CP4	T5	P3	Pz	P4	T6	O1	Oz	O2
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
3	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1
4	1	1	1	1	1	1	1	1	1	1	1	33	1	1	1	1	1	1	1	1	1	1	1	1	1	1
5	1	1	1	1	1	1	1	1	1	1	1	33	1	1	1	1	1	1	1	1	1	1	1	1	1	1
6	1	1	1	1	1	1	1	1	1	1	1	33	1	1	1	1	1	1	1	1	1	1	1	1	1	1
7	1	1	1	1	1	1	1	1	1	1	1	33	1	1	1	1	1	1	20	1	1	1	1	1	1	1
8	1	1	1	1	1	1	1	1	1	1	1	33	1	1	1	1	1	1	20	1	1	1	1	1	1	1
9	1	1	1	1	1	1	1	1	1	1	1	33	1	1	1	1	1	1	20	1	1	1	1	1	1	1
10	1	1	1	1	1	1	1	1	1	1	1	33	1	1	1	1	1	1	20	1	1	1	1	1	1	1
11	1	1	39	1	1	1	1	1	1	1	1	33	1	1	1	1	1	1	20	1	1	1	1	1	1	1
12	1	38	39	1	1	1	47	1	1	1	1	33	1	1	1	1	1	1	20	1	1	1	1	1	1	1
13	1	38	39	19	36	35	47	27	1	1	1	33	1	1	1	1	1	1	20	1	1	1	1	1	1	1
14	1	32	39	19	38	39	47	36	25	21	1	24	1	39	1	1	1	1	20	1	1	1	1	1	1	1
15	1	32	39	19	38	39	47	36	25	21	1	24	1	39	1	1	1	21	20	1	1	1	1	1	1	1
16	1	32	39	19	46	44	47	36	25	21	24	17	1	39	1	15	1	21	20	21	31	1	1	1	1	1
17	1	32	39	19	46	55	47	36	25	19	26	17	1	39	1	15	0	21	18	21	52	1	1	1	1	1
18	1	32	39	19	46	55	47	36	25	19	26	17	1	39	1	15	0	21	18	21	52	1	1	1	1	1
2324	45	33	216	2	48	8	190	32	20	62	50	38	74	21	110	7	26	47	50	40	25	42	88	92	51	32
2325	45	33	216	2	48	7	191	32	20	62	50	38	74	21	110	7	26	47	50	40	25	42	88	92	51	32

Table D-6 Clusters sizes are changing over time during the STDP learning in SNN model of 18 OP subjects in NOGO (in total 1350 time points were entered and learnt).

	Fp1	Fp2	F7	F3	Fz	F4	F8	FC3	FCz	FC4	T3	C3	Cz	C4	T4	CP3	CPz	CP4	T5	P3	Pz	P4	T6	O1	Oz	O2	
1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
2	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
3	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
4	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
5	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
6	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
7	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
8	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
9	1	1	1	1	1	1	50	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
10	24	37	1	1	1	1	50	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	1	
11	24	37	1	1	1	1	50	1	1	1	29	1	1	1	1	1	1	1	1	18	1	1	1	40	1	1	1
12	24	37	1	1	1	1	50	1	1	1	29	1	1	1	1	1	1	1	1	18	1	1	1	40	31	1	1
13	24	37	1	1	1	1	50	1	1	1	29	1	1	1	1	1	1	1	1	18	18	1	1	40	31	1	1
14	24	37	1	1	1	1	50	35	1	1	29	22	1	1	1	15	1	1	1	18	12	1	1	40	31	1	1
15	24	37	1	1	1	1	50	35	1	1	29	22	1	1	1	15	1	1	1	18	12	1	1	40	31	1	1
16	24	37	1	1	1	1	50	35	1	1	29	22	1	1	1	15	1	1	1	18	8	34	37	40	31	1	1
17	24	37	1	1	1	1	50	35	1	1	29	22	1	40	1	15	24	0	18	8	34	49	40	31	1	1	
18	24	37	1	1	1	1	50	35	1	1	47	22	1	40	1	15	24	12	0	8	34	37	40	31	1	1	
1349	84	73	48	51	83	60	68	20	43	32	111	42	67	44	193	29	12	11	36	35	19	38	48	78	46	87	
1350	84	73	48	51	83	60	68	20	43	32	111	42	67	44	193	29	12	11	36	35	19	38	48	78	46	87	

Table D-7 T-test measure was applied to the PSP's area under the curve for 26 clusters which belong to 21 healthy subjects and 18 OP subjects.

	Fp1		Fp2		F7		F3		Fz		F4		F8		FC3		FCz		FC4			
1		8.6	95.9	59.9	46.5	226.6	228.0	131.4	271.6	292.7	203.7	219.4	107.3	58.0	264.9	324.4	166.6	307.5	166.6	98.8	79.0	
2		311.5	93.7	420.8	163.2	477.5	55.6	152.0	287.5	482.5	344.5	281.1	243.3	720.9	249.9	545.6	173.0	231.9	173.0	611.5	503.8	
3		177.3	448.6	338.7	441.6	282.1	874.0	202.3	391.1	391.7	531.7	53.8	1109.0	215.9	421.6	232.3	101.3	178.9	101.3	264.4	821.5	
4		258.7	358.8	483.6	425.5	428.5	616.6	358.2	385.0	482.3	556.2	198.5	834.0	469.4	446.1	314.4	126.6	263.8	126.6	507.3	778.6	
5		336.3	287.2	691.0	515.4	645.8	431.0	171.9	233.5	313.7	422.5	175.6	735.2	746.9	384.4	286.8	98.7	206.0	98.7	626.1	608.4	
6		324.9	141.5	701.1	211.4	808.7	189.6	230.7	190.5	367.1	17.3	192.1	535.5	794.5	323.1	341.5	39.7	234.9	39.7	671.4	220.1	
7		271.4	281.6	605.4	495.0	559.8	567.0	165.1	297.3	397.6	454.4	230.3	713.9	684.9	543.4	307.6	114.4	140.2	114.4	654.3	531.5	
8		72.9	282.4	309.0	384.2	487.0	312.0	186.6	286.6	527.3	224.0	272.3	551.6	596.2	556.6	348.2	152.8	260.5	152.8	674.7	324.2	
9		222.1	364.8	507.8	563.9	687.8	498.6	219.1	233.4	620.1	274.6	244.7	521.9	742.5	464.2	369.6	85.5	198.2	85.5	598.4	379.0	
10		295.0	433.0	592.6	498.7	801.4	659.7	230.1	276.5	554.0	402.0	173.1	643.4	959.0	469.1	357.3	85.7	192.0	85.7	432.1	425.6	
11		314.8	4.2	622.3	24.0	603.9	104.8	218.4	176.5	506.8	220.0	194.2	41.3	369.6	291.8	358.4	78.0	111.8	78.0	626.2	278.0	
12		128.5	425.6	228.6	621.5	448.9	621.2	210.6	265.2	406.7	393.0	267.6	816.1	514.7	443.6	374.1	102.2	168.2	102.2	581.8	445.8	
13		215.5	204.3	570.0	205.0	390.2	253.2	167.7	220.3	423.2	227.9	208.2	225.1	788.6	342.2	327.6	73.4	99.3	73.4	430.0	302.7	
14		221.9	159.6	545.6	102.9	46.5	172.4	116.3	5.1	358.1	314.2	228.1	108.1	527.8	87.4	423.7	83.2	91.0	83.2	576.5	398.8	
15		238.6	499.4	680.9	512.1	321.6	575.3	204.6	309.8	412.6	312.9	175.0	701.7	611.2	457.7	332.8	134.3	88.1	134.3	673.3	455.9	
16		242.1	499.6	424.9	513.5	542.2	539.1	186.8	242.5	389.8	255.1	143.2	155.1	463.5	403.6	359.0	100.1	101.1	100.1	446.7	287.2	
17		407.2	99.2	724.0	185.9	669.0	458.0	218.1	355.8	610.1	393.9	185.0	660.5	955.4	440.7	362.5	183.7	141.1	183.7	711.2	400.4	
18		345.0	267.5	466.1	331.8	621.3	315.7	109.5	296.9	440.7	295.7	166.0	383.1	746.2	381.4	328.0	128.6	99.6	128.6	673.6	317.0	
19		267.3		464.4		121.8		99.6		463.0		213.3		483.5		266.3		192.3		539.7		
20		238.6		440.9		198.8		212.3		390.5		161.6		548.5		298.4		127.5		602.8		
21		240.6		458.5		284.1		44.7		495.4		161.0		499.8		85.2		85.9		549.0		
P-value		0.4725		0.017		0.5309		0.003		0.003		5E-04		0.0007		1E-11		0.003		0.022		
		T3		C3		Cz		C4		T4		CP3		CPz		CP4		T5		P3		
1		298.5	142.0	173.6	268.1	216.5	218.1	171.3	226.9	498.1	332.4	120.9	88.9	133.0	125.4	339.6	155.9	46.1	19.5	4.3	136.5	
2		252.6	257.1	220.8	207.4	247.1	243.5	173.7	233.6	556.1	630.5	172.7	5.1	341.5	219.1	185.6	61.8	171.9	140.8	328.0	153.0	
3		397.4	708.0	283.3	361.7	119.0	213.4	217.5	325.5	638.8	1075.7	77.9	87.6	441.8	194.8	120.6	121.5	148.9	278.6	335.9	239.0	
4		349.9	335.0	381.0	281.6	146.8	214.5	254.4	275.2	941.3	907.0	175.7	107.3	606.4	110.9	138.5	114.0	158.8	169.2	260.8	159.2	
5		385.6	533.2	220.3	320.2	116.1	197.2	243.8	316.3	722.8	874.0	149.2	116.3	600.1	139.7	289.5	149.6	207.4	150.1	239.9	153.8	
6		477.2	182.6	305.3	203.3	127.9	99.1	190.8	432.4	819.1	930.2	116.9	107.2	652.6	136.6	361.9	108.6	204.8	39.1	296.6	144.6	
7		432.8	338.0	397.5	252.0	103.7	211.3	231.9	319.4	799.8	1363.7	64.2	99.7	656.6	133.3	218.1	153.8	197.2	127.6	230.2	283.4	
8		297.8	261.3	479.8	253.4	204.3	208.5	135.1	418.1	727.4	1040.2	99.3	78.4	556.0	104.2	243.3	130.1	150.9	35.8	301.6	274.4	
9		302.8	50.7	468.7	204.8	269.9	231.2	169.0	452.6	637.1	847.2	87.2	131.1	712.1	164.1	262.5	117.2	187.9	168.4	317.5	376.2	
10		306.2	508.2	409.0	196.1	370.4	195.3	162.5	498.1	664.0	1130.3	37.6	89.0	465.1	113.2	270.6	181.4	153.8	238.8	242.8	430.5	
11		91.3	15.7	469.4	185.8	288.7	171.5	206.5	501.1	741.5	413.3	74.3	81.0	630.5	128.8	276.4	152.0	138.9	109.4	306.6	229.0	
12		151.3	391.2	371.4	139.5	354.1	210.6	146.2	488.0	164.9	921.1	132.2	106.0	496.4	90.4	30.9	141.7	87.5	213.7	258.5	345.4	
13		116.8	29.0	313.7	170.6	192.8	104.8	127.9	314.4	613.0	687.6	55.9	92.3	572.2	82.1	199.9	193.1	167.7	145.6	124.1	218.6	
14		5.9	16.0	445.4	33.1	230.4	175.3	184.3	303.9	101.6	39.1	115.8	123.4	450.3	122.1	136.5	45.3	147.4	102.5	366.8	150.1	
15		53.7	268.8	423.0	154.2	204.5	205.0	191.2	495.1	782.8	995.9	104.8	158.5	608.0	106.2	163.0	159.4	172.7	145.2	256.8	392.3	
16		326.4	202.2	366.1	151.0	224.8	181.5	120.6	428.7	129.3	953.7	106.4	120.3	487.5	95.9	186.2	111.3	158.0	317.6	213.5	436.2	
17		156.7	396.9	487.5	238.9	322.3	228.9	190.7	407.4	784.6	1020.8	121.5	122.6	627.1	72.4	232.6	162.9	208.1	281.6	203.7	353.8	
18		251.7	316.0	508.3	194.4	319.6	200.8	187.1	419.6	735.4	790.6	126.8	122.9	672.9	104.8	224.4	205.2	230.6	179.9	260.1	342.2	
19		272.2		408.8		327.1		140.8		503.6		97.1		517.5		102.3		124.3		149.2		
20		34.0		413.9		329.0		112.0		468.9		91.6		393.5		75.6		148.4		268.2		
21		1.0		32.9		180.1		190.2		112.2		7.8		310.6		121.1		76.0		70.2		
P-value		0.6236		8E-07		0.0636		1E-08		0.017		0.696		1E-11		0.005		0.9468		0.537		
		Pz	P4		T6		O1		Oz		O2											
1		68.3	166.8	120.3	277.9	514.6	318.9	482.3	449.5	311.2	230.1	342.9	584.0									
2		40.2	7.2	184.5	159.9	578.0	416.5	275.0	394.8	656.8	224.0	521.1	352.2									
3		44.5	307.8	205.3	298.6	329.8	694.8	174.1	1061.0	235.1	118.6	304.1	618.0									
4		61.8	393.3	162.2	440.6	436.4	568.1	182.2	551.2	207.6	24.0	356.8	324.1									
5		89.0	408.0	139.3	535.3	404.0	321.1	264.6	425.3	159.3	109.6	278.8	299.1									
6		89.8	285.1	195.7	397.1	609.0	267.6	253.8	398.7	152.2	101.8	356.7	252.1									
7		108.2	412.0	157.0	388.8	500.3	327.4	438.6	422.6	244.0	58.6	346.6	262.9									
8		79.0	383.2	121.0	498.7	442.8	108.4	323.4	307.0	149.0	92.3	272.1	172.7									
9		165.0	482.5	244.6	397.7	395.9	215.3	480.5	377.5	271.4	60.0	301.3	209.4									
10		101.8	529.0	173.5	518.7	464.8	295.6	256.8	416.1	184.7	85.3	249.4	265.1									
11		135.8	318.6	257.6	463.3	465.9	214.4	353.5	302.2	178.2	132.0	260.8	89.6									
12		215.6	404.6	135.9	395.7	457.8	237.9	359.4	442.3	118.9	86.0	239.7	262.3									
13		80.2	333.6	155.5	467.3	359.5	212.6	232.1	163.0	132.2	142.6	222.										

Table D-8 T-test measure was applied to the PSP's midrange values of 26 clusters which belong to 21 healthy subjects and 18 OP subjects

	Fp1		Fp2		F7		F3		Fz		F4		F8		FC3		FCz		FC4	
1	0.78	4.26	2.3	1.39	4.95	5.95	3.95	7.03	6.53	4.61	6.82	7.25	2.54	6.12	8.26	6.76	8.58	5.2	3.89	6.82
2	8.8	2.71	14.59	4.02	13.48	1.81	4.39	6.27	14.31	7.8	9.02	9.2	18.4	8.31	13.9	4.98	6.56	5.22	18.59	12.48
3	5.64	9.43	10.18	9.61	7.08	19	4.68	9.28	11.19	11.62	3.94	9.18	10.74	22.35	9.66	10.63	5.41	3.5	16.67	18.18
4	6.54	6.85	9.47	8.32	12.04	12.3	7.33	7.55	10.56	11.84	5.89	8.71	9.7	14.35	7.13	11.38	8.73	4.08	12.05	16.25
5	6.89	6.05	15.48	11.6	13.69	9.46	4.8	6.98	8.69	8.89	5.83	10.45	14.65	16.22	6.34	11.1	7.36	3.95	13.36	11.69
6	6.61	3.85	15.89	5.62	18.64	5.94	4.98	4.68	12.31	6.43	4.81	9.6	18.45	10.96	8.8	8	7.97	1.44	16.41	10.98
7	5.68	6.48	13.39	11.7	11.74	11.8	5.96	7.52	14.26	10.28	5.95	10.76	14.66	16.6	7.81	12.56	7.01	3.96	14.88	12.77
8	3.24	5.91	9.34	8.89	12.95	6.23	4.67	7.1	12.53	10.12	7.59	9.16	16.33	11.82	7.07	13.41	5.98	3.8	16.2	9.59
9	4.72	8.9	11.06	12.1	13.7	12.1	4.24	7.81	14.92	8.83	6.02	10.89	17.38	11.12	7.36	10.44	4.94	4.02	14.67	8.81
10	5.44	9.45	13.57	10.7	16.98	15.7	4.91	7.72	13.84	8.64	5.23	13.99	18.35	15.46	7.2	12.44	4.63	3.95	10.04	10.87
11	7.08	0.22	16.05	0.97	16.48	3.18	5.15	4.85	12.59	8.01	6.18	11.52	15.29	1.46	8.68	7.04	3.57	3.5	16.41	7.84
12	2.83	10.09	5.22	13.5	8.18	14.6	4.14	8.34	10.12	9.02	5.78	13.76	10.92	15.74	8.78	10.78	3.68	4.17	10.51	11.09
13	5.24	8.32	11.13	13.5	9.11	14.6	3.73	7.87	9.89	7.75	5.31	8.7	16.79	15.74	7.95	10.23	4.03	3.43	13.34	8.1
14	6.26	4.28	11.71	2.83	1.27	5.1	2.97	0.24	7.89	6.41	4.41	5.25	10.4	3.31	9.12	2.28	2.81	3.33	11.78	8.03
15	5.78	11.03	15.11	12.4	11.64	14.5	4.78	7.16	12.05	9.19	6.09	15.41	13.43	13.15	9.05	12.57	4.1	4.49	15.87	9.37
16	9.23	4.54	15.85	7.55	12.65	10.6	5.21	7.27	15.04	7.94	4.89	17.63	18.58	13.64	7.83	9.62	4.76	4.99	16.25	10.51
17	7.52	5.68	14.76	7.18	13.78	11.6	5.03	5.47	16.64	7.37	5.54	13.4	15.9	7.51	7.62	8.99	3.86	3.93	18.47	8.75
18	5.98	7.23	14.29	7.1	4.6	8.82	4.25	6.2	14.51	7.61	5.23	4	11.45	11	5.46	9.2	5.21	4.2	15.99	7.2
19	5.64		10.28		4.95		4.18		9.33		4.28		10.84		5.82	0	3.63		14.02	
20	5.1		8.97		6.05		1.42		9.39		3.69		9.59		2.52	0	1.63		11.52	
21	5.65		8.97		6.12		4.2		10.1		4.4		9.2		2.2	0	3.4		10.2	
P-value	0.407		0.008		0.845		5E-04		7E-05		1E-05		0.31		0.02		0.02		0.001	
	T3		C3		Cz		C4		T4		CP3		CPz		CP4		T5		P3	
1	7.7	7.77	4.82	6.46	6.79	7.12	7.03	6.89	12.1	9.22	3.47	2.5	4.38	5.15	8.27	3.7	1.96	0.93	12.1	9.22
2	5.07	5.46	5.75	4.46	5.49	5.08	6.62	6.54	12.27	13.78	4.48	1.67	8.73	5.39	4.4	1.4	4.41	2.85	12.27	13.78
3	7.68	14.96	6.96	8.23	3.27	5.82	5.88	6.75	17.14	24.08	3.55	2.38	10.3	5.66	4.04	3.97	2.99	6.03	17.14	24.08
4	6.62	6.01	8.67	5.77	3.89	4.39	5.37	8.3	17.54	20.94	4.67	2.4	11.97	2.95	4.4	3.2	4.44	4.33	17.54	20.94
5	6.94	10.54	4.49	6.25	3.19	5.3	6.24	8.55	13.77	20.43	2.72	2.66	13.34	3.15	7.39	3.82	4.97	3.25	13.77	20.43
6	10.17	5.3	7.87	5.18	3.81	2.46	5.81	8.23	18.37	19.96	2.55	3.09	14.56	2.95	6.61	2.48	5.25	1.23	18.37	19.96
7	8.14	7.69	9.49	4.72	3.98	4.46	5.18	8.5	17.3	30.85	2.9	2.97	14.85	2.8	5.75	3.34	4.63	3.08	17.3	30.85
8	6.98	6.78	9.5	5.86	5.27	5.48	5.18	9.35	16.35	31.51	2.17	2.67	13.19	2.85	5.53	2.82	3.33	1.43	16.35	31.51
9	6.48	1.45	10.31	5.46	8	4.26	4.06	10.42	16.87	22.85	1.83	2.83	14.53	3.33	5.41	2.93	4.05	3.97	16.87	22.85
10	5.8	9.77	9.02	5.08	7.49	4.91	3.52	10.29	13.13	21.89	1.23	2.3	12.14	3.31	5.04	4.39	4.22	4.51	13.13	21.89
11	4.37	0.36	10.65	4.26	8.51	4.64	4.29	11.38	16.75	19.25	2.26	1.85	15.16	2.89	6.23	3.26	3.04	2.52	16.75	19.25
12	5.87	9.52	7.79	4.44	6.55	4.84	3.14	11.41	3.51	19.47	2.74	3.41	9.81	3.02	0.9	3.35	2.58	5.33	3.51	19.47
13	5.61	8.6	7.66	4.47	6.28	3.81	4.78	11.41	12.23	17.6	2.52	2.37	11.89	3.08	4.11	4.12	4.43	4.58	12.23	17.6
14	0.35	0.49	8.99	1.14	5.33	3.93	3.95	6.73	10.62	1.29	2.7	2.72	9.17	2.73	3.42	2.13	3.05	3.63	10.62	1.29
15	7.84	6.89	8.55	4.28	7.86	5.49	4.7	12.16	15.93	19.99	2.41	3.42	13.34	2.99	6.07	4.13	3.97	6.99	15.93	19.99
16	5.01	8.64	9.2	5.78	7.97	5.28	3.8	10.97	18.54	22.09	2.94	2.98	13.56	3.44	5.51	4.72	4.66	7.18	18.54	22.09
17	5.67	5.52	12.09	4.41	8.08	3.75	4.65	11.1	16.14	16.32	2.75	2.52	13.51	2.14	4.97	4.37	4.66	4.19	16.14	16.32
18	6.04	2.1	8.54	4.02	8.04	4.31	4.27	11.36	11.84	17.22	2.32	2.21	10.84	2.14	4.03	4.37	3.55	4.96	11.84	16.27
19	1.73		8.18		7.51		4.06		12.53		1.94		9.36		3.87		2.98		12.53	
20	0.09		1.31		3.84		4.12		2.81		0.38		6.71		2.82		1.73		2.81	
21	0.04		8.2		7.2		6.2		11.2		11.1		6.27		1.82		1.39		2.14	
P-value	0.31		3E-05		0.006		2E-09		0.0051		0.376		2E-11		0		0.53		0.004	
	Pz		P4		T6		O1		Oz		O2									
1	4.62	4.98	5.66	7.49	11.75	7.05	11.17	11.68	10.04	10.04	9.01	15.59								
2	3.14	3.5	5.68	5.73	11.76	9.43	6.39	8.91	14.37	14.37	13.04	8.17								
3	2.01	9.52	5.05	7.1	6.7	14.4	5.14	23.62	7.38	7.38	6.73	14.72								
4	3.59	10.17	4.94	9.42	8.92	9.47	5.65	11.22	5.74	5.74	7.63	7.38								
5	3.06	8.05	3.82	10.8	8.76	6.5	6.21	9.3	3.85	3.85	6.48	6.31								
6	3.27	7.74	4.29	9.55	12.99	7.12	8.24	9.37	5.71	5.71	8.55	5.82								
7	2.71	8.94	3.76	8.15	12.74	6.03	11.26	8.5	6.61	6.61	9.23	5.41								
8	3.17	8.16	3.21	10.2	9.89	2.29	7.59	7.37	3.58	3.58	6.45	3.36								
9	3.43	10.12	4.79	7.9	8.77	4.47	10.27	8.06	5.61	5.61	6.94	5.18								
10	3.88	12.53	4.16	12.5	10.25	6.57	7.98	8.16	4.5	4.5	5.55	5.55								
11	4.18	7.76	5.58	9.66	10.91	4.86	7.53	6.74	3.93	3.93	5.49	2.25								
12	4.25	8.97	2.97	10.3	9.06	5.36	7.5	10.23	2.97	2.97	5.63	5.55								
13	3.05	9.42	5.39	9.31	8.52	6.35	7.14	9.11	4.01	4.01	5.04	5.67								
14	3.7	8.03	5.44	7.18	3.68	0.9	6.79	6.42	3.48	3.48	6.01	1.21								
15	4.39	9.32	5.39	14.4	9.79	7.84	10.3	8.26	3.8	3.8	6.37	7.25								
16	4.66	10.97	4.53	12.3	18.27	6.68	7.73	6.91	3.78	3.78	7.89	5.02								
17	4.66	11.48	4.22	11.8	11.92	5.13	8.53	5.55	2.85	2.85	5.67	4.71								
18	4.51	11.9	5.14	11.2	11.7	5.32	7.9	5.13	3.24	2.54	6.42	4.82								
19	4.43		4.46		11.46		6.23		3.12	0	6.36									
20	2.67		4.47		8.95		7.46		2.79	0	5.22									
21	2.71		4.13		8.55		7.46		2.37	0	5.22									
P-value	5E-09		1E-08		2E-04		0.212		0.7293		0.5532									

Table D-9 T-test measure was applied to $P_{max}(t)$ of 26 clusters which belong to 21 healthy subjects and 18 OP subjects.

	Fp1		Fp2		F7		F3		Fz		F4		F8		FC3		FCz		FC4		
1	0.776	4.26	2.291	1.38	4.941	5.94	3.955	7.0287	6.5219	4.6009	6.8195	7.241	2.533	6.12	8.2538	6.754	8.571	5.2	3.888	6.817	
2	8.797	2.7	14.58	4.012	13.48	1.809	4.397	6.2687	14.305	7.7965	9.0191	9.196	18.4	8.309	13.906	4.977	6.551	5.21	18.583	12.48	
3	5.636	9.42	10.17	9.609	7.078	19	4.684	9.2783	11.184	11.615	3.935	9.174	10.737	22.35	9.6505	10.63	5.406	3.5	16.668	18.18	
4	6.531	6.85	9.469	8.312	12.03	12.27	7.32	7.5418	10.553	11.84	5.8867	8.706	9.6925	14.34	7.123	11.37	8.726	4.08	12.05	16.24	
5	6.89	6.05	15.48	11.55	13.69	9.459	4.807	6.9795	8.6841	8.8826	5.8275	10.45	14.647	16.22	6.3341	11.1	7.353	3.95	13.356	11.68	
6	6.61	3.84	15.89	5.616	18.63	5.933	4.985	4.6717	12.304	6.4245	4.8082	9.592	18.448	10.96	8.7936	7.998	7.963	1.44	16.407	10.98	
7	5.677	6.48	13.38	11.69	11.74	11.77	5.954	7.5167	14.258	10.274	5.9487	10.76	14.658	16.6	7.8006	12.55	7.003	3.96	14.88	12.76	
8	3.24	5.91	9.331	8.886	12.95	6.226	4.663	7.0926	12.526	10.117	7.5859	9.157	16.328	11.81	7.0665	13.41	5.975	3.8	16.195	9.587	
9	4.714	8.89	11.05	12.09	13.7	12.06	4.24	7.8061	14.915	8.8269	6.0185	10.89	17.377	11.11	7.3585	10.43	4.94	4.01	14.665	8.809	
10	5.431	9.45	13.57	10.72	16.98	15.69	4.903	7.7197	13.838	8.6358	5.23	13.99	18.341	15.46	7.191	12.44	4.63	3.94	10.036	10.87	
11	7.077	0.21	16.04	0.962	16.48	3.176	5.142	4.8499	12.588	8.0069	6.172	11.51	15.29	1.454	8.6715	7.039	3.569	3.49	16.409	7.832	
12	2.83	10.1	5.21	13.45	8.172	14.64	4.132	8.3305	10.117	9.018	5.7719	13.75	10.92	15.73	8.7713	10.77	3.678	4.16	10.502	11.08	
13	5.235	8.32	11.13	13.45	9.106	14.64	3.724	7.8692	9.8817	7.7476	5.3065	8.693	16.785	15.73	7.948	10.22	4.03	3.42	13.339	8.098	
14	6.256	4.28	11.71	2.821	1.268	5.092	2.961	0.2305	7.8895	6.4034	4.4095	5.244	10.392	3.305	9.1187	2.276	2.805	3.33	11.771	8.029	
15	5.771	11	15.1	12.4	11.64	14.49	4.779	7.1562	12.047	9.1866	6.0892	15.4	13.423	13.15	9.0482	12.57	4.1	4.49	15.868	9.363	
16	9.224	4.54	15.85	7.545	12.65	10.61	5.209	7.2639	15.034	7.9346	4.8818	17.62	18.579	13.64	7.8277	9.618	4.751	4.98	16.248	10.51	
17	7.513	5.67	14.75	7.176	13.78	11.58	5.027	5.4656	16.633	7.3661	5.5393	13.39	15.894	7.51	7.6194	8.988	3.858	3.92	18.467	8.745	
18	5.976	7.23	14.28	7.1	4.596	8.82	4.248	6.2	14.501	7.61	5.2246	4	11.441	11	5.4543	9.2	5.202	4.2	15.982	7.2	
19	5.639		10.27		4.944		4.172		9.3249		4.2757		10.838		5.8179		3.628		14.017		
20	5.1		8.963		6.046		1.412		9.3848		3.6818		9.5847		2.5192		1.62		11.52		
21	5.646		8.963		6.12		4.2		10.1		4.4		9.2		2.2		3.4		12.1		
P-value	0.407		0.008		0.845		5E-04		7E-05		1E-05		0.3097		0.0248		0.018		0.0018		
1	7.704	7.77	4.825	6.462	6.797	7.122	7.036	6.8927	12.105	9.2205	3.4741	2.501	4.3803	5.151	8.2741	3.705	1.966	0.94	12.105	9.221	
2	5.072	5.46	5.752	4.463	5.494	5.087	6.625	6.5421	12.273	13.788	4.4886	1.672	8.7308	5.392	4.4016	1.403	4.419	2.85	12.273	13.79	
3	7.688	15	6.961	8.233	3.274	5.829	5.889	6.7572	17.149	24.08	3.5537	2.389	10.301	5.667	4.0482	3.978	2.992	6.04	17.149	24.08	
4	6.626	6.01	8.679	5.778	3.899	4.397	5.378	8.3069	17.548	20.943	4.6753	2.401	11.979	2.955	4.4093	3.202	4.447	4.33	17.548	20.94	
5	6.946	10.5	4.494	6.253	3.2	5.308	6.248	8.5545	13.772	20.433	2.7288	2.66	13.346	3.156	7.3998	3.826	4.973	3.26	13.772	20.43	
6	10.18	5.3	7.871	5.188	3.812	2.47	5.819	8.2321	18.373	19.966	2.5559	3.092	14.568	2.96	6.6137	2.488	5.251	1.23	18.373	19.97	
7	8.146	7.69	9.493	4.721	3.982	4.463	5.184	8.5056	17.305	30.851	2.907	2.974	14.858	2.802	5.757	3.342	4.639	3.08	17.305	30.85	
8	6.988	6.79	9.502	5.869	5.277	5.488	5.184	9.3503	16.356	31.517	2.1798	2.677	13.194	2.858	5.5397	2.824	3.338	1.43	16.356	31.52	
9	6.482	14.6	10.32	5.468	8.006	4.267	4.066	10.425	16.879	22.852	1.8359	2.838	14.534	3.339	5.4152	2.934	4.059	3.97	16.879	22.85	
10	5.801	9.77	9.023	5.085	7.491	4.916	3.521	10.291	13.137	21.896	1.2353	2.308	12.148	3.312	5.0431	4.397	4.224	4.52	13.137	21.9	
11	4.38	0.36	10.65	4.264	8.511	4.641	4.298	11.387	16.75	19.259	2.2628	1.855	15.167	2.891	6.2341	3.263	3.047	2.52	16.75	19.26	
12	5.87	9.52	7.793	4.449	6.557	4.842	3.142	11.415	3.5137	19.476	2.747	3.412	9.818	3.021	0.9059	3.354	2.582	5.34	3.5137	19.48	
13	5.615	8.6	7.664	4.477	6.284	3.813	4.781	11.415	12.23	17.605	2.5223	2.372	11.894	3.08	4.1138	4.128	4.438	4.59	12.23	17.61	
14	0.352	0.49	9	1.148	5.332	3.935	3.951	6.7395	10.62	1.2967	2.7041	2.724	9.1756	2.734	3.4259	2.139	3.059	3.63	10.62	1.297	
15	7.85	6.9	8.556	4.288	7.86	5.499	4.704	12.164	15.938	19.993	2.4128	3.427	13.348	2.999	6.0746	4.138	3.98	6.99	15.938	19.99	
16	5.017	8.64	9.201	5.783	7.979	5.288	3.809	10.977	18.549	22.09	2.9471	2.985	13.564	3.446	5.5168	4.722	4.664	7.19	18.549	22.09	
17	5.674	5.52	12.09	4.415	8.088	3.758	4.652	11.106	16.145	16.327	2.7542	2.525	13.511	2.142	4.9727	4.375	4.663	4.2	16.145	16.33	
18	6.046	2.1	8.546	4.021	8.042	4.31	4.271	11.36	11.84	17.221	2.3268	2.212	10.848	2.142	4.0378	4.375	3.559	4.97	11.84	16.27	
19	1.735		8.188		7.517		4.061		12.536		1.9435		9.3634		3.8703		2.983		12.536		
20	0.099		1.314		3.841		4.127		2.8143		0.38197		6.7171		2.8285		1.74		2.8143		
21	0.05		8.2		7.2		6.2		11.2		11.1		6.271		1.8285		1.395		2.143		
P-value	0.311		3E-05		0.006		2E-09		0.0051		0.37464		2E-11		0.0048		0.531		0.0041		
1	4.62	4.99	5.661	7.5	11.76	7.05	11.17	11.689	10.046	10.046	9.0145	15.6									
2	3.147	3.51	5.683	5.736	11.77	9.434	6.398	8.9136	14.374	14.374	13.043	8.176									
3	2.01	9.52	5.057	7.103	6.702	14.4	5.142	23.625	7.3859	7.3859	6.7391	14.72									
4	3.596	10.2	4.941	9.422	8.922	9.477	5.66	11.227	5.7491	5.7491	7.6367	7.382									
5	3.06	8.05	3.822	10.8	8.765	6.506	6.218	9.3057	3.8518	3.8518	6.4834	6.312									
6	3.279	7.75	4.298	9.554	12.99	7.123	8.244	9.3737	5.7101	5.7101	8.551	5.828									
7	2.711	8.95	3.767	8.158	12.75	6.039	11.27	8.5024	6.6128	6.6128	9.238	5.413									
8	3.172	8.16	3.215	10.16	9.895	2.292	7.594	7.3792	3.581	3.581	6.4521	3.368									
9	3.431	10.1	4.795	7.9	8.774	4.474	10.28	8.062	5.6157	5.6157	6.9484	5.187									
10	3.883	12.5	4.168	12.49	10.25	6.579	7.983	8.168	4.508	4.508	5.5544	5.554									
11	4.189	7.76	5.58	9.661	10.92	4.864	7.533	6.7473	3.9315	3.9315	5.4961	2.253									
12	4.251	8.97	2.978	10.28	9.063	5.365	7.509	10.233	2.974	2.974	5.6346	5.559									
13	3.058	9.42	5.394	9.317	8.527	6.359	7.146	9.1103	4.0186	4.0186	5.0499	5.67									
14	3.701	8.03	5.441	7.187	3.686	0.901	6.792	6.4295	3.4886	3.4886	6.019	1.213									
15	4.392	9.33	5.397	14.45	9.791	7.845	10.31	8.2604	3.8085	3.8085	6.3703	7.255									
16	4.665	11	4.539	12.33	18.27	6.681	7.74	6.9157	3.7828	3.7828	7.8952	5.025									
17	4.664	11.5	4.229	11.81	11.93	5.133	8.54	5.5513	2.8554	2.8554	5.6798	4.718									
18	4.515	11.9	5.145	11.2	11.7	5.329</															

Appendix E Co-authored Publications

Z. Gholami, **M. G. Doborjeh**, N. Kasabov, “Attentional Bias Pattern Recognition in Neuromarketing using EEG Data and the NeuCube Evolving Spatio-Temporal Data Machine Based on Spiking Neural Networks”, *Cognitive Computation Journal*, 2017.

N. Kasabov, and **M. G. Doborjeh**, Z. Gholami, “ Mapping, Learning, Visualisation, and Classification of fMRI Data in a Spatio-Temporal Machine of Evolving Spiking Neural Networks”, *IEEE Transaction on Neural Networks and Learning Systems*, vol. 28, no. 4, pp. 887-899, 2016.

M. G. Doborjeh, G. Y. Wang and N. Kasabov, “A Spiking Neural Network Methodology and System for Learning and Comparative Analysis of EEG Data From Healthy Versus Addiction Treated Versus Addiction Not Treated Subjects.”, *IEEE Transactions on Biomedical Engineering*, vol. 63, no. 9, pp. 1830-1841, 2016.

N. Kasabov, N. Scott, E. Tu, M. Ottman, **M. G. Doborjeh**, E. Capecci, et al, “Evolving Spatio-Temporal Data Machines Based on Neuromorphic Principles: Design Methodology and Selected Applications”, *Neural Networks*, vol. 78, pp. 1-14, 2015.

M. G. Doborjeh, N. Kasabov, Z. Gholami, “Evolving, dynamic clustering of spatio/spectro-temporal data in 3-D spiking neural network models and a case study on EEG data” *Evolving Systems*, pp.1-17, 2017.

N. Kasabov, L. Zhou, **M. G. Doborjeh**, Z. Gholami, J. Yang, “New algorithms for encoding, learning and classification of fmri data in a spiking neural network architecture: a case on modelling and understanding of dynamic cognitive processes”, *IEEE Transactions on Cognitive and Developmental Systems*, no.99, 2016.

M. G. Doborjeh, N. Kasabov, “Personalised Modelling on Integrated Clinical and EEG Spatio-Temporal Brain Data in the NeuCube Spiking Neural Network Architecture”, *IJCNN*, pp. 1373-1378, Vancouver, 2016.

H. Kawano, A. Seo, Z. Gholami, N. Kasabov, **M. G. Doborjeh**, “Analysis of Similarity and Differences in Brain Activities between Perception and Production of Facial

Expressions Using EEG Data and the NeuCube Spiking Neural Network Architecture”, *ICONIP in Kyoto*, vol. 9950, pp. 221-227, 2016.

Z. Gholami, **M. G. Doborjeh**, N. Kasabov, “Efficient Recognition of Attentional Bias using EEG data and the NeuCube Evolving Spatio-Temporal Data Machine”, *ICONIP in Kyoto*, vol. 9950, pp. 645-653, 2016.

M. G. Doborjeh, E. Capecci, and N. Kasabov, “Classification and Segmentation of fMRI Spatio-Temporal Brain Data with NeuCube Evolving Spiking Neural Network Model,” *IEEE Symposium on Evolving and Autonomous Learning Systems (EALS)*, Florida, U.S.A. pp. 73-80, 2014.

M. G. Doborjeh, N. Kasabov, “Dynamic 3-D Clustering of Spatio-Temporal Brain Data in the NeuCube Spiking Neural Network Architecture on a Case Study of fMRI Data”, *ICONIP in Turkey*, pp. 191-198, 2015.

M. G. Doborjeh, J. Israel Espinosa Ramos, et al, “From von Neumann architecture and Atanasoff’s ABC to Neuromorphic Computation and Kasabov’s NeuCube. Part II: Applications”, Springer book, (2017).

Z. Gholami., N. Kasabov., **M.G. Doborjeh**, A. Sumich “Modelling peri-perceptual brain processes in a deep learning spiking neural network architecture”, *Nature Scientific reports*, vol. 8, no. 1, p.8912 (2018).