Complementary learning fuzzy neural network

Tan, Tuan Zea

2008

Tan, T. Z. (2008). Complementary learning fuzzy neural network. Doctoral thesis, Nanyang Technological University, Singapore.

<https://hdl.handle.net/10356/41276>

<https://doi.org/10.32657/10356/41276>

Complementary Learning Fuzzy Neural Network

Tan Tuan Zea

School of Computer Engineering

A thesis submitted to the Nanyang Technological University
in fulfilment of the requirement for the degree of
Doctor of Philosophy

2008

Abstract

Computational intelligence (CI) is gaining more attention as its applications in various areas grow. Soft computing is one of the popular CI facets because of its ability to handle imprecision and uncertainty. *Artificial Neural Network* (ANN), *neuro-fuzzy system*, *fuzzy expert system*, and *statistical method* are the prominent tools within this discipline. However, these systems infer mainly from the class of interest (positive class). The information about the differences among classes are not fully utilized. Some systems, on the other hand, do not consider the class information at all. Although these systems perform well, their performance could be further enhanced if the contribution from negative class is taken into account. Moreover, constructing knowledge based on single class alone may cause the system to under-perform when the data is imbalanced. Viewing from the other end, most of these systems focus on boosting the accuracy, but disregard the psychological needs of user. They lack the reasoning, inference, and validation processes that user can identify with. CI system based on ANN or statistical method provides no means of understanding the system, while some expert system requires manual construction of knowledge. Furthermore, most of them are independent of biological or psychological principle, which hinder the user acceptance and trust towards the system. The debut of high-dimensional and ultra-huge databases exacerbates the situation.

Thus, it is crucial to take into account the interpretability, the tractability, and the high-dimensionality, on top of the performance. To this end, a novel class of learning paradigm—*Complementary Learning* is proposed. Complementary learning functionally models after the pattern recognition of human being. Since most of the CI task involves pattern recognition, and the fact that human being is effective in learning patterns, mimicking the human pattern recognition could bring forth fruitful outcome. Complementary learning not only has more common ground with user, but also good performance in pattern recognition. Complementary learning embraces learning from positive and negative classes. It also segregates the positive and negative knowledge, and then exploits the lateral inhibiting relationship between the two. These, minimize confusion in inference, in which lesser susceptibility to imbalanced dataset may be offered. In order to improve the tractability and

interpretability of the system, human-like reasoning process is integrated into the system. Criteria are devised to ensure the tractability of the system as well. Since there are data of different natures, different variants of complementary learning is defined and developed: *Genetic Complementary Learning* (GCL) is designed to search for solution in stochastic way. *Hierarchical Complementary Learning* (HCL) is developed to be used in high-dimensional data. Each of the variant drew its inspiration from the complementary learning observed in human being. These complementary learning systems are assessed in medical and bioinformatics applications. Their performances are benchmarked against popular CI tools, and are found to exhibit very competitive performance in classification accuracy as well as system interpretability. In future, complementary decision making framework that constitutes the interplay between different forms of complementary learning will be investigated.

Acknowledgement

I would like to thank my supervisor **Dr. Ng Geok See**, for his valuable guidance, encouragement and help throughout this research. Without his patience and understanding, this research would not be able to reach this stage.

I would like to express my gratitude to **Dr. Quek Hiok Chai** as well for his motivation, encouragement, guidance, and in-time support. Discussion with him is always very inspiring.

I also wish to bring my sincere thanks to **Dr. Tung Whye Loon** for his guidance, his clarification on various difficult concepts, and for his being very helpful and supportive; **Mr. Ang Kai Keng** for his great advices and teaching; **Ms. Bok Lee Fuang** for proof-reading my thesis.

I wish to extend my thanks to **Mr. Tan Swee Huat** and **Mr. Lau Boon Chee** in Center for Computational Intelligence (C2i). They have been very helpful and have kindly provided me with excellent hardware and software support.

Last but not least, my gratitude also goes to my family, my relatives, my friends, my colleagues in C2i, my pastors, my ministry teams (CHCTV), School of Theology Ministry team 15, and my cell group (S26) for their unconditional love, encouragement and support in my pursuing of my vision. They have always been my source and strengths.

Table of Contents

ABSTRACT	I
ACKNOWLEDGEMENT	III
TABLE OF CONTENTS	IV
LIST OF TABLES	VII
LIST OF FIGURES	IX
LIST OF ABBREVIATIONS	XI
NOTATIONS	XIII
PART I: BACKGROUND	
CHAPTER 1 INTRODUCTION	1
1.1 MOTIVATION.....	1
1.1.1 Issue	1
1.1.2 Limitation of the State of the Art.....	2
1.1.3 The Proposed System.....	3
1.2 OBJECTIVE	5
1.3 ORGANIZATION.....	5
CHAPTER 2 REVIEW OF THE STATE OF THE ART	7
2.1 COMPLEMENTARY LEARNING IN POSITIVE/NEGATIVE CLASSIFICATION.....	7
2.1.1 Positive/Negative Learning System.....	7
2.1.1.1 Pseudo Outer-Product Fuzzy Neural Network	8
2.1.1.2 Concept Learning	9
2.1.2 Neutral Learning System	10
2.1.2.1 Multilayer Perceptron.....	10
2.1.2.2 Support Vector Machine.....	12
2.1.3 Complementary Learning System.....	14
2.1.3.1 Decision Tree.....	15
2.1.3.2 Naïve Bayesian.....	16
2.2 COMPLEMENTARY LEARNING IN OTHER SYSTEM CLASSIFICATION.....	17
2.2.1 Global & Local Learning.....	17
2.2.2 Transductive & Inductive Learning	18
PART II: RESEARCH ISSUES	
CHAPTER 3 COMPUTATIONAL INTELLIGENCE SYSTEM	21
3.1 INTRODUCTION	21
3.2 ISSUES	21
3.2.1 Accuracy	21
3.2.2 Transparency.....	22
3.2.2.1 Interpretability	22
3.2.2.2 Tractability	24
3.2.3 High Dimensionality.....	25
3.3 SUMMARY	26
PART III: PROPOSED SOLUTION	
CHAPTER 4 COMPLEMENTARY LEARNING	27
4.1 INSPIRATION	32
4.1.1 Positive and Negative Learnings	32
4.1.2 Structure and Parameter Learnings.....	37

4.2	FORMALIZATION	40
4.3	CHARACTERISTICS OF CLFNN	47
4.3.1	CLFNN Offers Closer Representation of the Problem Space	47
4.3.2	CLFNN Gives Better Generalization	48
4.3.3	CLFNN Handles Skewed Data Well	48
4.3.4	CLFNN Fires Positive Rules More If Given a Positive Sample	50
4.4	EXPERIMENT	51
4.4.1	Complementary Learning on Linearly Separable Problem	51
4.4.2	Complementary Learning on Multi-class Classification Problem	53
4.4.3	Complementary Learning on a Hard Problem	56
4.5	SUMMARY	61
CHAPTER 5 FALCON-AART		63
5.1	ARCHITECTURE	63
5.2	LEARNING	68
5.2.1	Structure Learning	69
5.2.2	Parameter Learning	73
5.2.3	Learning Termination	75
5.3	TRACTABILITY	76
5.4	INTERPRETABILITY	81
5.4.1	Complexity	81
5.4.2	Distinguishability	83
5.4.3	Normality	84
5.4.4	Coverage	84
5.5	EXPERIMENT	84
5.5.1	Breast Cancer Diagnosis Decision Support	85
5.5.2	Breast Cancer Prognosis Decision Support	93
5.5.3	Diagnosis with Proteomics	98
5.6	SUMMARY	105
CHAPTER 6 GENETIC COMPLEMENTARY LEARNING		106
6.1	GCL ARCHITECTURE	107
6.2	GCL LEARNING	108
6.3	GCL ANALYSIS	111
6.4	GCL INTERPRETABILITY	112
6.5	EXPERIMENT	114
6.4.1	Translation Initiation Sites Prediction	114
6.4.2	Cervical Cancer Diagnosis	123
6.6	SUMMARY	130
CHAPTER 7 HIERARCHICAL COMPLEMENTARY LEARNING		131
7.1	HCL BIOLOGICAL EVIDENCE	134
7.2	HCL LEARNING	136
7.3	HCL INFERENCE	144
7.4	HCL ASSESSMENT	146
7.4.1	Good Classification Accuracy	146
7.4.2	Effective Treatment of High Dimensionality	146
7.4.3	Good Interpretability	147
7.4.4	High Efficiency, Scalability and Usability	148
7.4.5	Robustness	150
7.4.6	Incremental Data Incorporation	151
7.4.7	Minimal Number of Parameters	151
7.4.8	Short Response Time	151

7.4.9	One Pass of Training Data	152
7.5	EXPERIMENT	152
7.5.1	HCL and Proteomics	152
7.5.2	HCL and Hemostasis	158
7.6	SUMMARY	164

PART IV: CONCLUSIONS AND FUTURE WORKS

CHAPTER 8 CONCLUSIONS AND FUTURE WORKS165

8.1	CONCLUSIONS	165
8.2	SUMMARY OF CONTRIBUTIONS	167
8.3	FUTURE WORK	170

REFERENCES.....175

A PUBLICATION ARISES FROM THE RESEARCH WORKA1

B PROPOSITIONAL LOGIC TRANSFORMATION B1

C DETAILED EXPERIMENTAL RESULTSC1

D SYNTACTIC DATASETD1

List of Tables

Table 4.1: Similarity of human and CLFNN inference	45
Table 4.2: Rulebase of complementary learning and positive learning system.....	47
Table 4.3: Performance on iris classification task	53
Table 4.4: Performance on lymphoma classification task	54
Table 4.5: Performance on Leukemia subtype classification task.....	55
Table 4.6: Ten most relevant feature for predicting patient outcome.....	56
Table 4.7: Performance of different system on CNS tumor outcome prediction	57
Table 5.1: Comparison of FALCON-AART learning with psychological theories	69
Table 5.2: Reasoning process of FALCON-AART-AARS, Physicians, & CDDM.....	80
Table 5.3: Breast cancer diagnosis methods.....	85
Table 5.4: Reported accuracy on WDBC dataset	87
Table 5.5: WDBC classification result	88
Table 5.6: Rules generated.....	90
Table 5.7: Reported accuracy on WPBC dataset	94
Table 5.8: Breast cancer prognosis result	95
Table 5.9: Overall performance in ovarian cancer diagnosis	101
Table 5.10: Complementary fuzzy rules and crisp rules.....	101
Table 6.1: Complexity analysis of GCL	111
Table 6.2: Accuracy of eukaryotic TIS prediction.....	116
Table 6.3: The performance of computational intelligence methods in TIS prediction ..	120
Table 6.4: Fuzzy and crisp rules	121
Table 6.5: Inference process of GCL and human beings.....	122
Table 6.6: Features derived from cytological smear images	124
Table 6.7: Performance of popular methods versus GCL.....	125
Table 6.8: Comparison of GCL rules, decision tree rules, and expert rules	126
Table 6.9: Cervical cancer staging.....	127
Table 6.10: Complementary rules generated autonomously by GCL.....	129
Table 6.11: GCL and physician's reasoning process.....	130
Table 7.1: Complexity analysis of HCL	149
Table 7.2: Dataset used in experiments	153
Table 7.3: Classification performance of the systems in the five medical tasks	155

Table 7.4: Performance of SVM under different kernels	156
Table 7.5: Rules generated by HCL, FA, and C4.5	157
Table 7.6: Statistics of Hemostatic attributes	161
Table 7.7: Performance on ovarian cancer prognosis.....	162
Table 7.8: Rules generated on ovarian cancer prognosis.....	162
Table 8.1: The issues and their corresponding proposed solutions	167
Table B1: Performance on Leukemia subtype classification.....	B1
Table B2: Performance of support vector machine using different kernels	B2

List of Figures

Figure 2.1: Architecture of POPFNN	8
Figure 2.2: Segregation of positive and negative knowledge in CLFNN.....	9
Figure 2.3: Multilayer perceptron	10
Figure 2.4: Decision tree.....	15
Figure 4.1: Learning taxonomy.....	28
Figure 4.2: Explicit learning taxonomy	29
Figure 4.3: Implicit learning taxonomy	30
Figure 4.4: Survival learning taxonomy	31
Figure 4.5: Slices of fusiform gyrus of car and bird expert in recognition task	36
Figure 4.6: Complementary learning fuzzy neural network architecture	43
Figure 4.7: Problem modeling of complementary learning and positive learning	47
Figure 4.8: Better generalization of complementary learning than positive learning.....	48
Figure 4.9: Syntactic data	52
Figure 4.10: Discriminating lines of C4.5, POPFNN, MLP and CLFNN	53
Figure 4.11: Relationship between complementary learning & no. of features.	59
Figure 4.12: Relationship between complementary learning time & no. of features.	59
Figure 4.13: Relationship between complementary learning rule & no. of features.	59
Figure 4.14: Relationship between complementary learning rule & training time.....	60
Figure 5.1: FALCON-AART architecture.....	64
Figure 5.2: Human feature analysis model.....	68
Figure 5.3: Visual cortical plasticity as a function of age.....	72
Figure 5.4: Different relationship between fuzzy sets.	83
Figure 5.5: Abstract rules.....	87
Figure 5.6: Accuracy under different number of rules	91
Figure 5.7: The deduction based on complementary rules during decision-making	92
Figure 5.8: Numerical example of FALCON-AART inference process	92
Figure 5.9: Number of features versus accuracy plot	97
Figure 5.10: ROC plot.....	97
Figure 5.11: The original fuzzy set generated by FALCON-AART	98
Figure 5.12: The fuzzy sets tuned	98
Figure 5.13: Accuracy versus number of features	100
Figure 5.14: CLFNN reasoning process	102

Figure 5.15: CLFNN performance based on potential biomarkers.....	104
Figure 5.16: CLFNN performance on training and testing sets of different sources.....	105
Figure 6.1: Architecture of GCL fuzzy neural network.....	108
Figure 6.2: Operations of genetic complementary learning.....	109
Figure 6.3: Performance under different number of features	120
Figure 6.4: Performance of TIS prediction under different combinations of features.....	123
Figure 6.5: Performance of GCL under different number of features	128
Figure 7.1: The concept structure of HCL and HFS	133
Figure 7.2: Building block of HCL fuzzy neural network.....	139
Figure 7.3: The different scenarios for the chunking operation.....	142
Figure 7.4: The tuning of the fuzzy sets supports	143
Figure 7.5: Averaged recall accuracy of the five diagnosis tasks.....	154
Figure 7.6: Averaged predict accuracy of the five diagnosis tasks.....	154
Figure 7.7: The antecedent fuzzy sets formed by HCL and FALCON-AART	158
Figure 7.8: Membership function of D-dimer.....	163

List of Abbreviations

AARS	Approximate Analogical Reasoning Scheme
ANN	Artificial Neural Network
ART	Adaptive Resonance Theory
ATIII	Antithrombin III and Plasminogen Activities
AUC	Area Under the Curve
AVR	Augmented Variance Ratio
BLR	Binary Logistic Regression
CDDM	Clinical Differential Diagnosis Methodology
cDNA	Complementary Deoxyribonucleic Acid
CDSS	Clinical Decision Support System
CI	Computational Intelligence
CL	Complementary Learning
CLFNN	Complementary Learning Fuzzy Neural Network
CNS	Central Nervous System
COA	Center of Area
CV	Cross validation
DM	Distance Measure
DNA	Deoxyribonucleic Acid
EST	Expressed Sequence Tag
FA-AARS	FALCON-AART with AARS
FALCON	Fuzzy Adaptive Learning Control Network
FALCON-MART	FALCON-Modified ART
FLS	Fuzzy Logic System
FNA	Fine Needle Aspiration
FNN	Fuzzy Neural Network
GA	Genetic Algorithm
GCL	Genetic Complementary Learning
HCL	Hierarchical Complementary Learning
HFG	Hybrid Fuzzy Genetic
HFS	Hierarchical Fuzzy System
ID3	Induction-Tree
ILFN	Incremental Learning Fuzzy Neural Network
KNN	k-Nearest Neighbor
LDA	Linear Discriminant Analysis
LR	Logistic Regression
LTD	Long-Term Depression
LTP	Long-Term Potential
MALDI-TOF	Matrix Assisted Laser Desorption and Ionization with TOF
MC	Matthew Correlation
MCES	Monte Carlo Evaluative Selection
MF	Modification Function
MLP	Multilayer Perceptrons
mRNA	Messenger Ribonuclei Acid
MSM	Multi-Surface Method
NB	Naïve Bayesian
OSM	Overall Similarity Measure
PNN	Prototype Neural Network
POPFNN	Pseudo Outer-Product Fuzzy Neural Network
RAM	Random Access Memory
REX-1	Rule Extractor 1
ROC	Receiver Operating Classifier

RSA	Recurrence Surface Approximation
SELDI-TOF	Surface Enhanced Laser Desorption and Ionization with TOF
SM	Similarity Measure
SVM	Support Vector Machine
TAT	Antithrombin III Complex
TFPI	Tissue Factor Pathway Indicator
TIS	Translation Initiation Site
TOF	Time of Flight
UCI	University of California, Department of Information and Computer Science.
US	United State
vWF	von Willebrand Factor
WDBC	Wisconsin Diagnostic Breast Cancer
WPBC	Wisconsin Prognostic Breast Cancer

Notations

T	Number of samples
C	Concept
$\mu_C(\bullet)$	Membership degree of belonging to concept C
$\mu_{\neg C}(\bullet)$	Membership degree of not belonging to concept C
$SM(\bullet)$	Similarity measure
ρ	Threshold
i	Index of input dimension, and $i \in [1, I]$
j	Index of input linguistic term node, and $j \in [1, J]$
k	Index of rule node, and $k \in [1, K]$
l	Index of output linguistic term node, and $l \in [1, L]$
m	Index of output dimension (class index), and $m \in [1, M]$
N	Number of generations
x'_i	Input from i^{th} dimension, and of the pattern t
$\tilde{x}_i$	Fuzzified x_i
y_m	Output of m^{th} -dimension
y'_m	Output of m^{th} dimension, of the t^{th} pattern
$\mathbf{x}'$	Input vector of pattern t
$f_l(\bullet)$	Aggregation function of layer l
$a_l(\bullet)$	Activation function of layer l
u_{ij}, v_{ij}	Weights of linkage that links i^{th} node to j^{th} node
$\min_i, \max_i$	Minimum and maximum feature values for i^{th} dimension
z_{ij}	Output of j^{th} node that is linked to i^{th} precedent node
$z_{\text{rule } k}$	Output of rule node k^{th}
net_{ij}	Net synaptic input to j^{th} node that is linked to i^{th} precedent node
$\mathbf{A}$	Fuzzy set of $\mathfrak{R}$
w_{ij}	(u_{ij}, v_{ij}) fuzzy set or weights of node j^{th} linked to precedent i^{th} node
p, q	Proposition
τ_0	Threshold

$\mathbf{X}$	Input arbitrary set (crisp/fuzzy)
$\mathbf{A}_{ij}$	Antecedent fuzzy set of j^{th} rule and i^{th} input dimension
$\mathbf{B}_j$	Consequent fuzzy set of j^{th} rule
$\mu_{\mathbf{x}}(\bullet)$	Membership function of fuzzy set $\mathbf{X}$
$ \bullet $	Cardinality
U	Input universe of discourse, and $x \in U$. Finite and non-empty set
V	Output universe of discourse, and $y \in V$
$*$	t-norm operation
$J(w)$	Cost function, with parameters w
d	Desired value
η	Learning constant
$P(\bullet)$	Probability
$\text{sgn}(\bullet)$	Sign function
L_p	Primal Langrangian
F	Set of features
f_{user}	Maximum number of feature
f_i	The i^{th} feature.
$Decision(\mathbf{x})$	The system decision when input is $\mathbf{x}$

Part I:

Background

Chapter 1 Introduction

1.1 Motivation

Computational intelligence (CI) is a successor of artificial intelligence. CI embraces elements of learning, adaptation, evolution and fuzzy logic (rough sets) to create programs that are intelligent in some sense [Poole99]. Since its debut, it has been successfully applied in various arenas. CI system is believed to bring forth better results in solving real life problems, be it task automation or decision support. For example, *Clinical Decision Support System* (CDSS) based on CI could potentially reduce the high medication error rate, which may subsequently save annually USD\$182 billions that was spent on medication error, as well as 120,000 lives that died as a result of medication error each year [CIPME07][Rudman04]. Hence, CI research grows tremendously.

1.1.1 Issue

Unfortunately, the development and application of CI method are non-trivial. Many developed CI systems are yet to be accepted in practice. Although many factors such as lack of standardization in design and validation, privacy and security issues, etc, have ushered in this situation, the main reason lies in the psychology of users toward these systems, especially in medical domain [Dinka06]. Apparently, perceptions are the major determinant of CI system adoption [Joy05]. Take for example, most CDSS developed was not interpretable in medical domain. Consequently, users are reluctant to trust these CDSS [Fieschi03]. Moreover, they don't have a reasoning flow that the users are familiar with [Long01]. Thus, it is difficult for the users to relate, and to analyze the system [Dinka06].

This implies a system ought to have high interpretability and tractability, as well as human-like reasoning, on top of superb accuracy [Berlin06][Lisboa06][Long01]. Aside from that, it is desirable that the system is based on functional model of human. This is because human learns effectively and efficiently. As such, system that functionally models human has greater credibility, containing more common ground between user and the system. Functional model is preferred because it is less complicated. It does not require the full understanding of the mechanisms or physiology underlies a particular task.

On the other hand, since the data growth is so rapid at present, it is desirable that the system could implement some strategies to handle the high dimensional and ultra-huge database. With the advancement of instruments and tools, such as DNA microarray in Bioinformatics, the amount of information available increases exponentially. In order for the CI system to assist human, it should have strategy or method to handle such high dimensionality, and the immensity of present database.

1.1.2 Limitation of the state of the art

Accordingly, traditional expert system and statistical methods such as Bayesian network are not suitable because they require manual construction of rules or probability. Popular CI tools such as *Artificial Neural Network* (ANN) and statistical system are not suitable as well because they give no facility to understand their operation and reasoning [Lisboa02]. As more than half of the developed systems offer no explanation for their recommendation [Berlin06], this lack of explication capability causes the limited acceptance of CI systems in practice. Furthermore, since CI systems operate based on probabilistic and logic reasoning, they are burdensome to tract. It is also known that ANN can behave in a non-deterministic manner [Lisboa06], suggesting the ability to explain is paramount.

Viewing from the accuracy aspect, traditional CI tools construct or infer based solely on the target (positive) class, totally ignore the possible contribution from the non-targeted (negative) classes. Some do not consider the class information at all, which is useful in classifying the dataset. This not only weakens the ability of the system in classification performance, but also causes poor performance when the data distribution is imbalanced.

1.1.3 The proposed system

Thus, *Complementary Learning Fuzzy Neural Network* (CLFNN) is proposed. First of all, the proposed system is based on *Fuzzy Neural Network* (FNN) because FNN equip the system with the ability to facilitate validation and interpretation. The data-driven generation of fuzzy rules can be used to validate and to understand the system. However, the provision of rules does not guarantee interpretability. Hence, the proposed systems are designed such that they satisfy the interpretability criteria [Penã-Reyes04][Casillas03] [Ishibushi01]. This will be discussed in Chapters 5, 6, and 7. On the other hand, in order to ensure the transparency of the proposed system, the system is mapped with fuzzy inference schemes. This equips the system a human-like reasoning process (for details, please refer to Chapter 5). Moreover, to make the transparency measurable, some criteria are developed and are stated in Section 3.3.2.

Since psychology of user plays an important part for user acceptance, biological and psychological-inspired functional model is incorporated into CLFNN. Functional models simplify the neuro-physiological process underlying human tasks, and hence, do not significantly increase the complexity of the system. This incorporation of biological model not only gives the credibility, but also gives the user a sense of familiarity. User can identify and relate with the system since it operates similarly as the user. Since most of the tasks involve pattern recognition, a functional model of

human pattern recognition mechanism is proposed. Such pattern recognition mechanism can be observed at the psychological, neurological, and even at the cellular levels. This mechanism concerns the learning from positive and negative samples, the lateral inhibition as well as the segregation of positive and negative knowledge, and is termed as *Complementary Learning* (CL) [Tan07d][Tan06a]. Human-like inference process and recognition mechanism can be obtained through incorporating the CL into FNN (thus the name, CLFNN). This may lead to an open attitude towards CI system, besides improving the performance. The details of the proposed system can be found in Chapter 5.

From the viewpoint of complementary learning, the plethora of CI methods for realizing CDSS can be loosely categorized into

- Positive/negative learning system - system that constructs its knowledge based on positive/negative samples only.
- Neutral learning system - system that does not specifically distinguish the positive and negative samples, and does not consider the difference between them.
- Complementary learning system - system that takes into account the positive and negative samples, as well as the lateral inhibition between the two.

Complementary learning system has the potential of offering greater accuracy because it considers not only the characteristics of positive and negative classes, but also the differences between the two. This information can be useful in discriminating two different classes of samples.

To summarize, CLFNN is a promising tool because it implements human-like reasoning and a biological plausible learning in which the user can identify with.

Furthermore, CLFNN inherited the autonomous fuzzy rules construction from ANN and the expressive power from FNN, providing good interpretability without the time-consuming process of manual rule construction. On top of that, positive and negative rules are generated, which gives a closer representation of the problem. Together, these potentials of CLFNN initiate and motivate this research. From the results on various tasks, CLFNN demonstrated better performance in pattern recognition and in transparency. On this ground, it is believed that CLFNN offers an attractive alternative for enhancing the present CI system.

1.2 Objective

The objectives of this study are:

1. To analyze and establish different types of CL.
2. To formalize and define CLFNN.
3. To assess the feasibility of realizing CDSS using CL.

1.3 Organization

The thesis is divided into four parts. Part I consists of *Chapter 1* and *Chapter 2*. Part II consists of *Chapter 3*. *Chapters 4, 5, 6 and 7* make up the Part III. Part IV contains the last chapter, *Chapter 8*. Part I gives preliminary understanding of the thesis, where *Chapter 1* (this chapter) states the objectives and the motivations of the thesis, and *Chapter 2* gives a brief introduction to some state of the art or common CI methods. They are categorized generally to positive/negative, neutral, or complementary learning. Part II lists down the challenges behind the development of CI system. The identified issues are described in *Chapter 3*. Subsequently, starting from Part III, the different types of proposed CLFNN are described. The overview of CLFNN and its

formalization are described in *Chapter 4*. *Chapters 5, 6 and 7* briefly display the CLFNN that functionally model the different levels of CL:

- ☐ Functional model of hippocampal complementary learning - *Fuzzy Adaptive Learning Control with Another Adaptive Resonance Theory* (FALCON-AART) [Tan05a][Tan05b].
- ☐ Functional model of complementary gene selection process - *Genetic Complementary Learning* (GCL) [Tan06b][Tan05c].
- ☐ Functional model of human hierarchical complementary category learning - *Hierarchical Complementary Learning* (HCL) [Tan07a].

The last part of the thesis, Part IV, which consists of *Chapter 8*, summarizes the research work, its contributions and future work.

Chapter 2 Review of the State of the Art

Computational Intelligence (CI) encompasses many advanced techniques and algorithms from multi-disciplinary communities such as machine learning, statistics, *Artificial Neural Network* (ANN), fuzzy logic, evolutionary computing, data mining, *Fuzzy Neural Network* (FNN), model-based approach, and so on [Witten05]. Recently, there are surging interests in hybrid approach such as genetic fuzzy system, statistical ANN, etc. In the context of the proposed *complementary learning paradigm*, these methods can be classified into three types of system. These systems are categorized based upon whether the system learning includes both positive and negative knowledge, whether the system learning includes the difference between positive and negative knowledge, and whether the system implements strategy to minimize the possible confusion between the positive and negative knowledge.

2.1 Complementary Learning in Positive/Negative Classification

2.1.1 Positive/Negative Learning System

Positive/negative learning system refers to the system that learns or constructs its knowledge solely based on information obtained from a single class. Alternatively, this system infers and makes decisions based on the information obtained from that one class. Any sample that is not described by its positive/negative rulebase is assumed negative/positive (Equation 2.1). This category of learning system does not take into account the relation between the positive and negative classes.

$$y = \begin{cases} +, & \text{if } \mu_r(x) > \rho \\ -, & \text{Otherwise} \end{cases} \quad (2.1)$$

2.1.1.1 Pseudo Outer-Product Fuzzy Neural Network

One example is *Pseudo Outer-Product Fuzzy Neural Network* (POPFNN) [Zhou96]. Its architecture is shown in Figure 2.1.

The neurons in antecedent and consequent layers employ bell-shape membership function. The feedforward inference process is essentially the sum of rules activations, whereas the learning of the network is by the *pseudo outer-product learning algorithm* [Zhou96].

Figure 2.1: Architecture of POPFNN.

The main difference between positive/negative and complementary learning system is found in their antecedent and rule layers. There is segregation of positive and negative knowledge in complementary learning system (see Figure 2.2) but not in the positive/negative learning system. Moreover, the inference process of the positive/negative learning system is either without considering the negative/positive class, or without considering the differences between positive and negative classes.

Figure 2.2: Segregation of positive and negative knowledge in complementary learning.

Although there are enhanced versions of POPFNN, such as POP-Yager [Quek05] and POPFNN-AARS [Quek06], developed over the years, the improvement focuses only on the inference part of the system. The architecture and learning are essentially the same, i.e. positive learning, and hence, the limitation due to the ignorant of negative knowledge is still unresolved.

2.1.1.2 Concept Learning

Concept learning [Akgöbek06] induces a general knowledge based on the specific examples. In other words, by definition, most if not all concept learning is positive learning because they extract general knowledge from the specific samples of the class they want to learn, as shown in Equation 2.2.

$$f_{\text{inductive learning}} : (X, Y^+) \rightarrow R^+ \quad (2.2)$$

This learning is effective and biologically plausible [Tenenbaum06]. Although this learning is capable of enabling a person to recognize an object, he/she will be more competent in recognizing the object if he/she were given the positive and negative examples of that. Many enhanced concept learning algorithms have been introduced, for example, CN2 [Clark89], PRISM [Cendrowska87], ILA [Akgöbek06], Genetic concept learning [De Jong93], *Rule Extractor-1* (REX-1) [Akgöbek06], etc. In positive learning, one only looks at the similarity between the object and positive

knowledge. However complementary learning looks at the similarity between the object and positive knowledge as well as the dissimilarity between the object and the negative knowledge.

2.1.2 Neutral Learning System

Neutral learning system is a term given to method that does not consider the class and the differences between classes during system training and inference. Neutral learning system learns by tuning its parameters to minimize some cost functions. Some well-known methods such as *Multilayer Perceptron* (MLP), and *Support Vector Machine* (SVM) [Witten05] belong to this category.

2.1.2.1 Multilayer Perceptron

MLP is a popular ANN that has found success in various tasks, including medical diagnosis and biomedical control. MLP is formed of layers of neurons, where there are weighted interconnections in-between. Learning is done by adjusting the weights of the connections as well as the bias to the neurons between input, output, and hidden layers. The tuning is done according to the cost function defined at the output layers by means of *backpropagation* [Witten05]. Usually, three-layer perceptron is sufficient to approximate any functions to certain level of accuracy.

Figure 2.3: Multilayer perceptron.

Figure 2.3 shows an example of three-layer perceptron. It is made up of input, hidden, and output layers. Normally, sigmoidal function is chosen as the activation function of hidden and output layers. Input layer only propagates the input. MLP has feedforward and feedback operations. Feedforward operation is the inference process of the network, whereas the feedback operation is the learning process, by which the parameters of the network are tuned according to the backpropagation algorithm. The inference process of MLP can be summarized as Equation 2.3:

$$y_k = f_k \left[\sum_{j=1}^J v_{jk} \left(f_j \left(\sum_{i=1}^I w_{ij} x_i + bias_j \right) \right) + bias_k \right] \quad (2.3)$$

where f_j = activation function of hidden neuron, and f_k = activation function of output neuron; w_{ij} = weight of the connection between the i^{th} input neuron and the j^{th} hidden neuron. Likewise, v_{jk} = weight of the connection between the j^{th} hidden neuron and the k^{th} output neuron.

As for the learning process, MLP first computes the error signal at the output layer, i.e. cost function, and subsequently propagates the error signal to the hidden layers. The error signal is then used to update the weights of the connections, as given in Equations 2.4-2.6.

$$J(w) = \frac{1}{2} \sum_{k=1}^K (d_k - y_k)^2 \quad (2.4)$$

where $J(w)$ is the cost function, and d_k is the desired output. Assuming the activation function is a sigmoid, the weight update equations for the weights between hidden and output layers, and for the weights between input and hidden layers are computed using Equations 2.5 and 2.6 respectively. The derivations can be found in [Witten05].

$$v_{jk}(t+1) = v_{jk}(t) + \eta v_k (1 - y_k) (d_k - y_k) \left[f_j \left(\sum_{i=1}^I w_{ij} x_i + bias_j \right) \right] \quad (2.5)$$

$$w_{ij}(t+1) = w_{ij}(t) + \eta \left[f \left(\sum_{i=1}^I w_{ij} x_i + bias_j \right) \right] \left[1 - f \left(\sum_{i=1}^I w_{ij} x_i + bias_j \right) \right] \sum_{k=1}^K [y_k (1 - y_k) (d_k - y_k) v_{jk}] \quad (2.6)$$

where $f'(\bullet)$ is the derivative of the activation function $f(\bullet)$, and η is the learning constant.

MLP is a very popular tool and has been successful in various applications. However, one major limitation of MLP is that it is not interpretable. It offers no means for user to validate and to understand its computation. Furthermore, its inference process is different from that of human, and hence, user may find it challenging to relate his/her reasoning with the system operations. In addition, backpropagation has some undesirable features such as likely to be trapped in local minima, slow, etc. Some improvements proposed include incorporating rule extraction method to provide interpretability for MLP [Hruschka06]; introducing new adaptive learning and cost function to enhance learning accuracy [Yu06], etc. In spite of that, the original MLP is still the most popular.

2.1.2.2 Support Vector Machine

Support Vector Machine (SVM) is a system that non-linearly maps the input to a higher dimensional space and there it performs the classification. The SVM then performs kernel tricks to maximize the margin of the hyperplane that gives the optimal classification. In strict sense, SVM is a neutral learning system. Despite the use of *support vectors* to determine the separating hyperplane of the positive and negative samples, SVM is essentially a quadratic programming optimization problem. The finding of the hyperplane does not differentiate the positive and negative classes, and

the inference does not exploit the lateral inhibition between the positive and negative classes. Thus, SVM is categorized as neutral learning. Its algorithm is as follows:

Given training set $X = \bigcup_{i=1}^T \{\mathbf{x}', \mathbf{y}'\}$, SVM find the optimal α' for $\mathbf{x}'$ by using

Equation 2.7:

$$\text{sgn} \left[f(\mathbf{x}) = \sum_{i=1}^T y' \alpha' K(\mathbf{x}, \mathbf{x}') + b \right] \quad (2.7)$$

where $\text{sgn}(\bullet)$ is the sign function. To solve α' , the soft-margin SVM maximizes the *primal Lagrangian*, as given in Equation 2.8.

$$L_p = \frac{\|W\|^2}{2} + C \sum_i \xi^{ik} - \sum_{i=1}^T \alpha' [y' (\mathbf{w} \cdot \mathbf{x}' + b) - 1 + \xi^i] - \sum_{i=1}^I \mu' \xi^i \quad (2.8)$$

where $k=1$ or 2 , $\alpha' \geq 0$, and $\mu' \geq 0$. The penalty constant C represents the trade-off between the empirical error and the hyperplane margin. The value α' satisfies the condition stated in Equation 2.9:

$$\begin{aligned} 0 &\leq \alpha' \leq C \\ \sum_{i=1}^T \alpha' y' &= 0 \end{aligned} \quad (2.9)$$

One merit of SVM is that it has no local minima problem. Moreover, it has the capacity of controlling the margin optimization of the separating hyperplane. However, SVM has its limitations as well: poor performance in skewed dataset. SVM is heavily dependent on its parameters such as C (controlling the amount of overlap) and the kernel parameters [Adankon07]. Yet, there is no systematic way of selecting the kernel functions. On top of that, the inference process of SVM may become time-consuming and computationally expensive due to the large number of support vectors [Li07]. Also, SVM is difficult to adapt and to extract knowledge [Kasabov04a]. Although improvements over SVM have been made, only the original

2 Review of the State of the Art - T. Z. TAN

SVM is widely used. Some of the improvements are: (1) using empirical error criterion [Adankon07] and genetic algorithm [Wu07] to guide parameters setting and kernel selection; (2) coupling feature vectors with SVM to handle effectively large number of support vectors [Li07]. There are also efforts in improving the interpretability of SVM [Zhou07].

SVM and MLP are not classified as complementary learning system because they do not exploit the difference between classes during inference. In a two class classification task $y = \{0,1\}$, given a sample $\mathbf{x}$, the inference process of the MLP, SVM, and CLFNN are given in Equations 2.10, 2.11, and 2.12 respectively.

$$y = \begin{cases} 1, & f\left[\sum_{j=1}^J w_j \left(f_j\left(\sum_{i=1}^I w_{ij} x_i + bias_j\right)\right)\right] > \rho \\ 0, & \text{Otherwise} \end{cases} \quad (2.10)$$

$$y = \begin{cases} 1, & \text{sgn}\left[f(\mathbf{x}) = \sum_{i=1}^I y' \alpha' K(\mathbf{x}, \mathbf{x}') + b\right] > 0 \\ 0, & \text{Otherwise} \end{cases} \quad (2.11)$$

$$y = \begin{cases} 1, & \mu_{r \in R^+}(\mathbf{x}) > \mu_{r \in R^-}(\mathbf{x}) \\ 0, & \mu_{r \in R^+}(\mathbf{x}) < \mu_{r \in R^-}(\mathbf{x}) \end{cases} \quad (2.12)$$

As seen from Equations 2.10 to 2.12, the complementary learning inference process takes into account the difference between the positive and negative knowledge, whereas MLP and SVM do not exploit the differences between positive and negative knowledge but a threshold value. In other words, complementary learning is the only method here that considers the behavior of positive and negative class information during classification. In that sense, MLP and SVM are classified as neutral system.

2.1.3 Complementary Learning System

Complementary learning system learns from the positive and negative classes, as well as the differences. During inference, complementary learning system assesses the difference between the positive and negative samples, as well as the features of positive and negative samples. When a positive sample is presented to the system, only positive knowledge is activated, and concurrently, the negative knowledge is inhibited, and vice versa.

Loosely, the well known decision tree can be considered as a class of complementary learning system. Strictly, according to the complementary learning formalization stated in Section 4.2, however, it is not as it does not exploit the lateral inhibition between positive and negative classes. Nevertheless, it is one of the best off-the-shelf tools that has been applied in various fields.

2.1.3.1 Decision tree

Decision tree has a hierarchical structure shown in Figure 2.4. Firstly they split the data based on the best attribute according to some criteria, such as *entropy* (ID3), and *gain ratio impurity* (C4.5) [Witten05]. The splitting process continues recursively until the children tree nodes are pure or almost pure (i.e. having no sample of other class), or there are no more training example. *Pruning*, whereby useless split processes are reversed, can be done to enhance performance.

Figure 2.4: Decision tree.

Despite the popularity of decision tree, it possesses some shortcomings [Witten05][Kasabov04a]: (1) relations between features are lost; (2) it is unstable as small perturbation will result a very different tree; (3) lower level of the tree is less reliable because of the recursive partitioning; (4) it is difficult to adapt; (5) its knowledge base is less expressive because of the use of crisp rule.

Although decision tree makes use of positive and negative samples, it uses crisp rules. As a result, their problem modeling is similar to that of positive/negative system, because crisp rule system makes decision based on single threshold value. In this system, space lower than the threshold is treated as positive and the space higher than the threshold is treated as negative. Thus, strictly speaking, decision tree is not a complementary learning system. One enhancement for decision tree is to incorporate fuzzy logic into it, so that it can handle data uncertainty [Olaru03].

2.1.3.2 Naïve Bayesian

Similar to decision tree, Naïve Bayesian is perceived by others as a complementary learning system because it uses probability of positive and negative classes. However, in this work, Naïve Bayesian is not treated as a complementary learning system because it does not exploit the lateral inhibition between the positive and negative classes. Naïve Bayesian predicts the posterior probability of a class y_m , $P(y_m|x_1, x_2, \dots, x_i)$,

when given the sample from the class prior probability $P(y_m)$, and class conditional probability $P(x_1, x_2, \dots, x_I | y_m)$. $m \in [1, M]$ is the class index, whereas $i \in [1, I]$ is the index of input dimension. Naïve Bayesian classifier is based on the Bayes Theorem, as given in Equations 2.13-2.15:

$$P(y_m | x_1, x_2, \dots, x_I) = \frac{P(x_1, x_2, \dots, x_I, y_m)}{P(x_1, x_2, \dots, x_I)} \quad (2.13)$$

$$P(y_m | x_1, x_2, \dots, x_I) = \frac{P(x_1, x_2, \dots, x_I | y_m) P(y_m)}{P(x_1, x_2, \dots, x_I)} \quad (2.14)$$

$$P(y_m | x_1, x_2, \dots, x_I) = \frac{P(x_1, x_2, \dots, x_I | y_m) P(y_m)}{\sum_{m=1}^M P(x_1, x_2, \dots, x_I | y_m) P(y_m)} \quad (2.15)$$

Since Naïve Bayesian classifier assumes conditional independence, and the denominator is a constant, the posterior probability can be rewritten to Equations 2.16-2.17

$$P(y_m | x_1, x_2, \dots, x_I) = \frac{1}{Z} P(y_m) \prod_{i=1}^I P(x_i | y_m) \quad (2.16)$$

where

$$Z = \sum_{m=1}^M P(x_1, x_2, \dots, x_I | y_m) P(y_m) \quad (2.17)$$

Thus, the Naïve Bayesian classifier is the function $f(\mathbf{x})$ defined in Equation 2.18

$$f(\mathbf{x}) = \arg \max_y P(y = y_m) \prod_{i=1}^I P(x_i = x | y = y_m) \quad (2.18)$$

Naïve Bayesian classifier is simple and it performs well when the data is normally distributed. Also, it gives prediction uniqueness. However, the performance is unsatisfactory when the data is not normally distributed, and when the probability information conveyed in the training data does not reflect the real scenario. A few

extended version of Naïve Bayesian such as weighted Naïve Bayesian [Yager05] has been developed, albeit it is less popular than the original Naïve Bayesian.

2.2 Complementary Learning in Other Classification

2.2.1 Global & Local Learning

A way to classify the plethora of computational intelligence system is to divide them into global and local learning. As the name suggest, global learning looks at the overall picture of the observations, whereas local learning looks at the specific local information of the observations. This is with reference to the inference as well as the learning of the system.

Global learning assumes a model or a set of rules described by a set of parameters estimated from the total sample of observations. Subsequently, the classification of sample is based on this set of parameter. Linear discriminant analysis, decision trees, and generative learning belong to this category and hence is classified as global learning method [Huang05][Tutz05]. Global method has the strengths of not susceptible to local minima [Ye07], and has better representation of the data as the global structure of the data is captured [Silva03]. However, global learning suffers in difficulty in selecting the model to obtain accurate description of the data globally.

Local learning on the other hand, directly employs part of information critical for the specifically oriented tasks and does not assume a model for the data [Huang05]. Since local learning focuses on local information of the observed data, local learning usually gives better performance. Multilayer perceptron, and support vector machine are local learning system [Ye07]. Although, local learning is claimed to be superior in performance and efficiency over global learning, it suffers in local minima problem. Moreover, since it ignores the information regarding the structure of the

data, its performance can be worse than global learning method. This renders the interpretation of the model unreliable [Huang05].

In this global-local classification system, complementary learning can be viewed as the hybrid of global and local system. For local learning, the complementary learning learns the pattern of the positive and negative knowledge independently. Subsequently, complementary learning employs global learning which exploits the difference between the overall structure of the positive and negative knowledge. Putting the global and local learning together allows the strength of both methods to be incorporated, and this is found promising [Tutz05].

2.2.2 Transductive & Inductive Learning

Transductive learning has emerged as one of the latest research area in machine learning community [El-Yaniv05]. Transductive learning deals with learning to predict by making use the information obtained from labeled train data and unlabeled test data [Song05]. In contrast to inductive learning, which concerns the development of model to approximate data from whole problem space, transductive learning concerns the development of model for every new input vector and the model is used to predict the output of the vector [Kasabov04b]. However, note that transductive learning is not equivalent to semi-supervised learning [Angelini08] that induces a model based of test data [Collobert06]. Formally, given a set of labeled samples $\{(x_t, y_t) : t = 1, \dots, T\}$, and a set of unlabeled samples $\{x'_{t+1}, \dots, x'_{t+n}\}$, the inductive learning seeks to derive a model $f(x)$ that describes the samples, $\{(x_t, y_t) : t = 1, \dots, T\} \mapsto f$, where semi-supervised method uses unlabeled data to derive the model, $\{(x_t, y_t) : t = 1, \dots, T\} \cup \{x'_{t+1}, \dots, x'_{t+n}\} \mapsto f$. On the other hand, transductive learning infers the labels $\{y'_{t+1}, \dots, y'_{t+n}\}$ of the unlabeled data,

2 Review of the State of the Art - T. Z. TAN

$\{(x_t, y_t) : t = 1, \dots, T\} \cup \{x'_{t+1}, \dots, x'_{t+n}\} \mapsto (y'_1, \dots, y'_n)$. Majority of the computational intelligence tools are inductive, such as MLP, dynamic fuzzy inference system, ANFIS, and so on. Although inductive learning is useful when global model is needed, it does not consider the new data vector, which discounted the performance [Song05][Kasabov04b][Pang04].

Paradoxically, transductive learning can analyze all data points, which allows it to approximate the distribution and the data structure more effectively [Duchenne06][Lu06]. Also, as transductive learning takes into account the knowledge of each vector, it gives better local generalization [Song05] and allows local region of the data to be focused [Schwaighofer03]. Hence, transductive learning is easier than inductive learning [Collobert06]. However, transductive learning also has its limitations. For example, transductive learning is more computationally expensive, has larger prediction time and memory consumption [Schwaighofer03][Lu06]. Furthermore, as dataset are not usually available in advance, transductive model is difficult to be implemented as online classifier [Kasabov04b]. Transductive SVM [Collobert06], k-NN, transductive neuro-fuzzy inference and classifier [Song05][Ma05] are some examples of transductive system.

In this work, complementary learning is an inductive learning system as no unlabelled data are used for training. Based on the advantages of transductive learning, it may be promising to develop a transductive complementary learning in future.

Part II:

Research Issues

Chapter 3 Computational Intelligence System

3.1 Introduction

The applications of *Computational Intelligence* (CI) have been growing since its debut. CI has been applied in finance, environmental science, medicine, proteomics, computer security, bioinformatics, and many others [Kasabov07][Santos07][Lisboa06][Cherkassky06][Ezziane06][Malone06][Isasi04][Zhou03]. In general, its application can be divided into two: task automation and decision support. This thesis deals with the issues related to decision support, especially the application in clinical domain. Nevertheless, the issues discussed are applicable to both because the two tasks are similar. Since one of the objectives of this thesis is to develop a *Clinical Decision Support System* (CDSS), the discussion is more on the application of CI in medical domain.

3.2 Issues

There are many issues haunting the application of CI in real life. Apart from accuracy, legal, ethical, and psychological issues affect the implementation of CI into real life application [Lisboa06][Anderson07][OpenClinical02]. In this thesis, focus is put on accuracy, transparency, and high dimensionality issues.

3.2.1 Accuracy

Accuracy of a system is closely linked to validation. A system should not be introduced into practice before it has been properly evaluated. This evaluation process should cover all stages of development. Some common performance evaluation measures are [Lavrač99]: sensitivity, specificity, classification accuracy, reliability of negative predictions, precision, rule evaluation using relative measure,

3 Computational Intelligence System - T. Z. TAN

Receiver-Operating Characteristic (ROC) curve [Mac Namee02], and decision boundary analysis [Zhang00]. The most common method is the *cross-validation* approach [Zhang00]. These are the commonly accepted evaluation measures, and are important to generate trust among the users. The validation of system includes quality control of knowledge [OpenClinical02]. Hence, the consistency and accuracy of the system rulebase should be ensured. This will be elaborated more in the interpretability section later on.

There are many inspirations to improve the performance. The most credible comes from the human model because human is the most effective learning system on earth [Kasabov07]. However, the details and mechanisms of human learning are not known yet. Even if they were known, it is almost impossible to model the human learning exactly as it is very complicated.

3.2.2 Transparency

The explicability and tractability of a system play a primary role in determining user's acceptance to the system. A system must be able to provide justification together with explanation that can be validated by the user, in the form familiar to him/her [Lisboa06][NEDST03]. It also includes the extent of allowing the user to understand and critique the decision made [Dinka06]. Human-like reasoning is therefore important to facilitate the user's understanding so as to generate trust [Farukhi04]. Apart from that, the rulebase should be transparent such that user can easily understand and does minor changes. There are two parts in transparency: interpretability and tractability, which are described below.

3 Computational Intelligence System - T. Z. TAN

3.2.2.1 Interpretability

Interpretability or comprehensibility [Zhou03] refers to how easily one can understand the system. However, the problem is that most of the present systems do not have the architecture that facilitates user's understanding. For example, in medical domain, doctor does not understand the reasoning process of CDSS from theoretical or statistical aspect [Fieschi03]. This inability of CDSS to justify and explain itself has contributed to the lack of acceptance in clinical practice [Berlin06]. Therefore, a system must provide a means for user to analyze and to understand why it gives a particular decision with relevant reasons from its knowledge base.

Quoting another example from medical application of CI, it is shown that doctor interprets quantitative data with scant reference to details [Fieschi03]. The doctor refers a "patient of 15 year old" as "adolescent", a "patient of body temperature 37.9 degree" as "normal temperature". It is also shown that doctor prefers to use compact abstraction. For instance, a daily blood glucose profile can be said to be "average-high" if the mean of pre-prandial readings exceeds 6 mmol/l (millimoles per liter, standard unit for blood glucose concentration). Otherwise the daily profile is said to be "average-normal-or-low" [Deutsch04]. Putting it the other way, the knowledge representation should be intuitive, and the system has to encapsulate the unnecessary details.

In improving the interpretability, one needs to take note of the *limited rationality* principle [Fieschi03]: human capacity for reasoning is limited to cope with only a restricted number of facts at a time. Thus, an interpretable system should seek a simple knowledge representation and reasoning process.

That was from psychological point of view. Viewing from the data perspective, the system should satisfy criteria that ensure good interpretability. They are

3 Computational Intelligence System - T. Z. TAN

[Penã-Reyes04] [Casillas03][Ishibushi01]: (1) complexity – the number of antecedents and the number of rule; (2) distinguishability – how easily the antecedents of the rules can be separated and assigned understandable terms; (3) normality – there is a perfect match between the antecedents and samples; (4) coverage – for every sample there is a rule to describe it. These criteria will be discussed in details in Chapter 5.

3.2.2.2 Tractability

One reason for lack of CI system acceptance in real life application is that the system decision cannot be disputed or challenged [Berlin06]. The system does not explain how it arrives at certain decision. Consequently, users felt that they have lost control over the task [Dinka06]. Owing to that reason, the ability of the CI system to explain its decision is critical.

The closer the system reasoning to that of the user, the easier it is for the user to understand and accept the conclusions reached [Long01]. Hence, it is important for a system to have a reasoning process that is similar to human. It is not necessary to imitate exactly the user's reasoning, but it is desirable that the system allows explanation in his/her familiar terms [Fieschi03].

In addition, in the context of tractability, a system should follow the steps proposed in the *Clinical Differential Diagnosis Methodology* (CDDM), where intermediate results are made accessible by the physician in each induction step. The CDDM is a standard of patient examination, medical data aggregation, and experience-based reasoning procedure, which the physician utilizes to reach a diagnosis [Economou00]. Adopting it to CI system, it means support should be provided at every relevant stages and levels of the consultation [NEDST03].

In order to be a tractable system, a CDSS should meet the following criteria:

3 Computational Intelligence System - T. Z. TAN

- ◆ Human-like reasoning – the reasoning process should be highly akin to that of human.
- ◆ Step by step inference – clearly defined steps in deduction.
- ◆ CDDM procedure – the steps of inference should resemble the steps in CDDM (for CDSS): patient examining, medical data aggregating, and experience-based reasoning procedure.
- ◆ Explanation capability – ability to explain its deduction processes.
- ◆ User familiar terms – the explanation and justification must be done in the terms familiar to users.

3.2.3 High dimensionality

Nowadays, the rate at which the data is generated is higher than before. For example, the advance of microbiology such as *microarray* and *proteomics* [Baak03] has revolutionized the current medical practice. Microarray and proteomics allow analysis at the level of genes and proteins respectively. They are the new methods that promise accurate cancer detection, therapeutic response prediction, and so on. In order to make use of these new technologies, it is crucial for the CI system to handle challenges in high dimensionality. The problem may result in combinatorial increase of computation requirements if the relationship between the features were to be assessed [Mullins06]. Furthermore, the data for complex relationship such as cancer outcome are usually sparse and spread across many dimensions [Mullins06]. As a result, extensive data is required to achieve some statistical significance, which is not always possible. Hence, the system should be able to complement these new technologies in providing decision support services. The number of features of proteomics and microarray can be up to 15,000. For instance, the gene expression data typically consists of 10-100 samples, with more than 10,000 features

3 Computational Intelligence System - T. Z. TAN

[Bhaskar06]. As a result, most of the systems that based on conventional CI methods are either not able or they will take enormous amount of time to simulate proteomic and microarray datasets on personal computer. Thus, a CI system ought to have the capacity for dealing with high dimensionality.

3.3 Summary

Despite the many issues that hinder the application of CI system in real life, it is believed that CI systems will be increasingly prevalent in near future [Ohlsson04]. One important issue in CI is the quality of the data used to train the system. Poor quality of the data is often encountered in real life. Noise, uncertainties, and error in data collection are almost inevitable. The differences in data from heterogeneous sources exacerbate the situation. Hence CI is promising because CI can automatically generate the knowledge based on the data, freeing the user from the need of full understanding on the problem domain. Moreover, CI can be used to extract information from large data sources, and helps in identifying the structures of the data. It can be applied to different structured and unstructured data components to evaluate trends and to identify patterns. Soft Computing of CI is particularly promising because soft computing methods are able to capture uncertainties in the data.

Part III:

Proposed Solution

Chapter 4 Complementary Learning

The significant distinction between human and animals is the ready ability to learn and create. Such ability of effectively learning a new skill, applying the skill, and innovating using a set of skills, provides human beings a limited sovereignty over animals. The capacity to learn is a hallmark of intelligence. Thus, the understanding of human learning brings about significant benefits. Apart from gaining insights into how to deal with learning- or memory-related diseases, study of human learning allows the design of effective learning strategies, creation of intelligent machines, and so on.

However, human learning is a highly complex subject. Despite many endeavors, the understanding gained is not as fruitful as expected. This is due not only to the vast amount of data generated, but also a lack of consensus in the theories proposed by the researchers in human study. Nevertheless, the knowledge generated from human learning studies provides powerful and useful insights, and offers valuable solutions to a plethora of problems. Hence, human learning remains as a promising research area.

Even today, the underlying principle is not well understood yet. Many efforts from various disciplines have been doubled in the endeavour: cognitive science, psychology, ethology, anthropology, neuroscience, and artificial intelligence. While this generates enormous data on human learning, it is difficult to summarize what has been achieved thus far. The taxonomy of learning provides a framework to describe the achievements thus far. Taxonomy is also useful for guiding the development of theories [Willingham01]. However, there is yet unanimously accepted taxonomy. Most taxonomies view human learning from a single perspective that describes partially the very complex and multi-faceted human learning process. In this work, the taxonomy of human learning that is adapted from human memory is proposed. Hence,

there is no clear division between learning and memory in this work. For example, associative memory is needed for associative learning. In fact, most of the times, those two terms are used interchangeably. The proposed taxonomy shown in Figure 4.1 links the findings of behavioral psychology and neuroscience. It provides an overview of human learning, and shows the relation between the complementary learning and other learning paradigm.

Figure 4.1: Learning taxonomy.

Generally, learning can be divided into: (1) learning that is controllable and intentional. In that case, the learner actively searches for rules and the learner is aware of the rules considered. This refers to thinking or cognition, and can be viewed as the ability to develop concepts or rules [Call03]. This type of learning is also known as *rule/exemplar learning* [Love02]; (2) learning that is automatic, undirected, passive, and stimulus driven. The former is also known as *explicit learning* [Maddox04]. The later is known as *implicit learning* [Maddox04].

Explicit learning can be categorized into *episodic learning* (learning from events) [Bukach04] and *semantic learning* (learning about concepts, words and meanings) [O’Kane04], as shown in Figure 4.2. Episodic learning can be further broken down into *inference learning* that focuses on the internal structures of each category, and *classification learning* that focuses on the difference between categories [Love02]. Episodic learning consists of *context* [Haney01], and *item learning* [Hoyer03]. One form of context learning is the ability of the human being in navigation, i.e. *spatial*

4 Complementary Learning - T. Z. TAN

learning [Flood98]. In addition, *sequence learning* [Destrebecqz03] that concerns human ability to learn sequential material is yet another type of explicit learning. Apart from that, episodic learning consists of *scientific learning* [Davis86], which is the ability to learn a relationship between two or more events that are separated by more than 60 seconds. There is a class of scientific learning that enables human in anticipating highly probable future events [Davis86], known as *prospective learning* [Einstein95]. There is a category of learning under the umbrella of prospective learning called *predictive learning* [Karazinov04]. In predictive learning, the subject learns to associate stimuli with the occurrence or non-occurrence of certain outcome. Complementary learning belongs to episodic learning. It is closely related to classification and item learning because it involves the learning of the traits of positive and negative example, as well as the differences between them. In fact, most of the human learning employs the complementary learning mechanism. For instance, development of positive and negative concepts, and learning from positive and negative observations, etc. Social learning can be classified under explicit learning as well because it is a conscious process. Social learning comprises learning from instructions, observations, and mistakes [Chialvo99]. Semantic learning, on the other hand, can be divided into *conceptual* and *encyclopaedic* learning [Thompson-Schill03].

Figure 4.2: Explicit learning taxonomy.

For implicit learning, it can be classified into *associative* and *non-associative* learning in general (Figure 4.3). Some classified associative learning into *exemplar-based*, *priming*, and *skill learning* [Maddox04]. Skill learning refers to the increased speed or accuracy when a particular task is continually repeated [Schmidtke96]. Priming refers to the facilitation of recognition, reproduction, or biases in the selection of stimuli that has recently been perceived [Eichenbaum00]. On the other end, exemplar-based associative learning receives most attention and it has the forms of *classical* and *operant conditioning* that associate conditioned or unconditioned stimulus to a particular response. Aside from conditioning, exemplar-based associative learning can be in the form of *interleaved learning* [O'Reilly02], in which interleaved positive and negative examples are presented to the learner. This interleaved learning is a particular case of the *complementary learning* paradigm proposed. As for non-associative learning, it includes [Wright96]: (1) *habituation* — the gradual decrement in response that is repeatedly elicited; (2) *dishabituation* — the facilitation of a habituated response by presentation of a second stimulus; (3) *sensitization* — the facilitation of a non-decremented response by presentation of a second strong or noxious stimulus.

Figure 4.3: Implicit learning taxonomy.

Apart from the explicit and implicit learning, there is a type of learning that is essential for survival, i.e. survival learning. This type of learning concerns the ability to learn from aversive conditions and stimuli that are emotionally significant [Joseph96]. Survival learning (Figure 4.4) is made up of learning that induces avoidance action to shun from unpleasant stimuli (*avoidance learning*) [Smith02], and learning that arises due to the change of stimulus-reward expectation (*reversal learning*) [Clark04], as well as learning that causes arousal or fear to certain neutral stimuli (*fear conditioning*) [Conrad04]. Reversal learning is categorized into this group because it involves the adaptation of what have been learned to the new expectation, which may be caused by unpleasant experience such as a grave mistake.

Figure 4.4: Survival learning taxonomy.

Hence, complementary learning can be accomplished consciously or subconsciously. This least understood positive and negative complementary mechanism can be found in most of the human learning, which suggests the promising role of the mechanism in human learning. Evidences of complementary learning are found in neuroscience and psychological data. Due to the reason that complementary learning considers not only the characteristics of positive and negative examples, but also the differences between them, complementary learning can potentially offer superior performance. In other words, complementary learning exploits the differences, which is the *lateral inhibition* between the positive and the negative

examples. Lateral inhibition in this context refers to the mutually exclusive activation of a cell, where activated positive cells inhibit the activation of negative cells. Furthermore, interleaved learning found in the *Hippocampus*, which is a special form of complementary learning, is shown to be the mechanism that is unsusceptible to *catastrophic interference* [O'Reilly02].

4.1 Inspiration

4.1.1 Positive and negative learnings

Before one learns to distinguish what is abnormal, one would have to identify what is considered as normal, and vice versa. This illustrates the basic principle of complementary learning. For instance, if a child was to be taught to recognize an apple, he/she would be allowed to see, touch, and smell one. This is an example of positive learning. Of course, the child may be able to recognize an apple after he/she has seen, touched, and smelled one. However, if he/she is further allowed to see, touch, and smell fruits other than apple, he/she will be able to learn faster, and most likely he/she will be able to distinguish an apple out of a box of fruits. The latter behaviour is termed negative learning. This alternative learning approach is given the name called the *complementary learning paradigm*.

The complementary learning paradigm essentially consists of three properties: (1) learning from the positive and negative examples; (2) segregation of positive and negative knowledge; (3) lateral inhibition between the positive and negative knowledge. The evidence supporting these properties can be drawn from both cognitive psychology and neuroscience findings, which are discussed in the following sections.

4.1.1.1 Evidence from Psychology

The psychology support for the complementary learning paradigm is summarized in the followings:

Learning from positive and negative examples

- Human learns through positive (reward) and negative (punishment) experiences.
- Memory consolidation process incorporates knowledge through a slow and gradual interleaved learning process, from the hippocampal system to the neocortical system [O'Reilly02]. This interleaved learning is a process that involves a series of presentations interleaved with exposure to other examples from the domain. It is a form of complementary learning where interleaved positive and negative examples of an item are presented repetitively until the item is learned.

Segregation of positive and negative knowledge, as well as the lateral inhibition process during recognition

- Memory traces in the brain compete with one another when a person tries to recall a specific event. Activation of the correct traces is required for successful recalls because activation of other traces will interfere with the retrieval of the desired event. Hence, one possible way of controlling memory retrieval process is through the simultaneous activation of relevant and inhibition of irrelevant memory traces [Levy02].

Thus, cognitive psychological study gives undivided supports to the complementary learning paradigm. Back to the “apple” example, when the child is shown an apple, he/she identifies the features of an apple, which strengthens his/her ability to recognize an apple (positive reinforcement). These identified features are stored in memory. When the child is shown fruits other than apple, he/she extracts features that are not of an apple (negative reinforcement). Likewise, the extracted features are memorized. Hence, his/her ability to recognize what is not an apple is

further strengthened. Having seen apple and other fruits, whenever the child is presented an apple, he/she recalls the memory for apple and suppresses the memory for other fruits at the same time, leading to an accurate identification of an apple. This gives support to the complementary learning paradigm.

4.1.1.2 Evidence from Neuroscience

Complementary learning is also observed in human brain and neural system. In fact, human brain learns complementarily [Grossberg01]. The following summarizes the evidences collected from neuroscience studies at memory structure, neuro-molecular, and cellular levels:

Learning from positive and negative examples

- Hippocampus, a structure involved in human learning and emotion [Lisman05], possesses neurons whose synapses have both positive (long term potential, LTP) and negative feedback signals (long term depression, LTD). The LTP and LTD stimulate and suppress neuron firings, respectively.
- The neocortex is densely innervated with interlaminar excitatory (positive) and inhibitory (negative) projections. Many of the projections of excitatory neurons are accompanied by inhibitory axons [Thomson03].
- The excitatory (positive) adrenergic and inhibitory (negative) GABAergic neurotransmitter systems play a complementary role in regulating brain activities.
- Topographical map of neural circuits (neural maps) underlies structural basis of cognition. They are responsible for different cognitive functions undertaken by that region. The development and formation of these neural maps are dependent on the guided growth of axons. One of the various mechanisms that

guide these axonal growths in the *hippocamposeptal map* is the interaction of positive and negative guidance between the pre- and post-synaptic neurons [Gazzaniga00]. Such positive and negative regulatory mechanism can also be observed in *Protein Kinase A* pathway and the *Cyclic AMP-Response Element Binding Protein*, which is paramount for the memory consolidation processes [Abel98]. This mechanism can be observed at the cellular level, where the neural computation involves a dynamic interaction of excitatory and inhibitory synapses [Liu04].

Segregation of positive and negative knowledge, as well as lateral inhibition

- Synaptic plasticity plays a pivotal role in memory storage, as well as learning, because the key mechanism in how brain processes information lies in the tuning of the synaptic connection [Gazzaniga00]. The consolidation of short-term to long-term memories requires gene expression and synaptic changes. These long-term synaptic changes are governed by the complementary paradigm, i.e. both positive and negative regulators. On the other hand, the consolidation process requires simultaneous removal (negative) and activation of transcriptional activators (positive).
- The human brain is a specialized recognition system: the brain registers different brain areas exclusively for recognizing different objects. The existence of specialized visual recognition are reported for facial identity, facial expression of emotion, places and animals [Duchaine01]. Examples are: (a) the activation areas of car and bird recognitions are different. Brain regions that are responsible for bird recognition will not be activated when a car object is presented. This prevents a person from recognizing a car as a bird, and vice versa. Paradoxically, when a car object (positive sample) is presented, brain

areas that are responsible for recognizing car (positive rules) will be activated.

At the same time, areas registered for other objects (negative rules) will be inhibited (Figure 4.5) [Gauthier00]; (b) Lateral fusiform gyrus (*Brodman's Area 37*) responds maximally to faces whereas lingual/medial fusiform gyrus is strongly activated to houses. Parahippocampal gyrus is the sole site that displays significant activation to human faces, but not to dogs' face and houses [Blonder04]; (c) Face, location, and house recognitions activate different areas in the brain, creating a distinguished neural map for each object [Sala03].

- Likewise, different types of information are registered in different brain areas, be it encoding, retrieval or manipulation. For example, there is a clear differentiation of spatial-word and colour-word Stroop tasks activation in dorsolateral prefrontal cortex [Banich00]. Studies have demonstrated double dissociation of dorsal and ventral prefrontal cortex activation for spatial and non-spatial information, in regard -less of visual [Courtney98] or auditory [Alain01] stimuli. In addition, verbal recognition invokes greater activity in

Figure 4.5: Slices of fusiform gyrus of car and bird expert in face, car, and bird recognition. The rectangular boxes show the activated areas of brain for different recognition tasks (Adapted from [Gauthier00], with permission from *Nature Neuroscience*).

left prefrontal cortex, while figural recognition stimulates greater activity in the right inferior temporal gyrus [Beason-Held05].

- Human emotions are found to excite distinct brain activation patterns as well [Damasio00]. Happiness, sadness, anger, and fear elicit different patterns of brain activation and deactivation. Moreover, the amygdala, structure implicated in human emotion, has neurons dedicated to rewarding (positive) stimuli, and for punishing (negative) stimuli exclusively. This shows that human brain organizes information in “separate” manner: different brain areas, brain cells or neurons, and different patterns of activation/deactivation are registered for different information.
- Event-related activation of cued and uncued task study [Hester04] reported that the concurrent activation of task-relevant cortical region and deactivation of task-irrelevant cortical region is responsible for successful task performance. Likewise, this kind of complementary brain activation/deactivation is observed in *speeded response category learning task*, where subjects are asked to derive categories from the black and white (positive and negative) dot patterns presented [Aizenstein00].

4.1.2 Structure and Parameter Learnings

Structure and parameter learnings are not classified as complementary learning in this work. However, since structure and parameter learnings are often implemented in *Fuzzy Neural Network* (FNN), their biological evidences are described in this section.

Structure learning is responsible for creating and adjusting a network such that it is sufficient to represent the underlying data, whereas parameter learning is responsible for fine-tuning the internal memory elements or parameters of the network so as to

improve the performance of the network. Structure learning self-organizes the network structure by adding new nodes, and tuning the location of its memory elements. On the other hand, the parameter learning adjusts the network parameters (synaptic weights) to enhance performance. These structural and parametric plasticities, known as double plasticity, provide additional flexibility to the system. Accordingly, the double plasticity allows the network to learn with relatively superior performance. Structure and parameter learning can happen concurrently, sequentially (structure learning prior to parameter learning), or independently.

Double plasticity can be observed in many natural biological systems [Bersini99], one example is the immune network. Immune network is structured as networks of interacting units (antibodies). Each unit that is already present in the network can alter their concentration (parameter plasticity), as well as simply appearing or disappearing (structural plasticity). When an antigen (the external data) enters the human body, the antibody is recruited to bind the antigen. New set of antibodies will be produced to cater to a new set of antigen. This happens autonomously and alters the immune network structure. Antibody concentration grows if stimulated by an antigen, which is analogous to the tuning of parameter by an external data. On the other hand, ecosystems are also examples of double plasticity. While parametric plasticity amounts to the species density and the interaction between species, structural plasticity refers to the introduction of new species through crossover or mutation, and extinction of existing species [Bersini99].

Human brain also possesses both structural and parametric plasticities. The human brain structure is modulated by experience. New neurons are continually generated (*neurogenesis*) when novel information arrives, demonstrating the structural plasticity of the human brain. This neurogenesis is paramount to learning and memory

in human [Gross00]. On the other end, the existing interneuron connections, which store information in the forms of synaptic weights, are adjusted by chemical substances, exhibiting parametric plasticity in the human brain. However, there are equivocal findings of whether the two plasticities occur simultaneously or sequentially. The classical view of the human brain development is that neurogenesis occurs only within a restricted developmental period and stops before puberty [Kornack01]. There is no postnatal neurogenesis, which is to say the retention of memory and learning are through the modification of the synaptic weights between the neurons. It is argued that the neurogenesis observed in the adult brain is actually non-neuronal cells [Kornack01]. This supports the conjectures that structural and parametric learnings happen sequentially. The other view is that, postnatal neurogenesis does occur in adult brain, including area in olfactory bulb [Bayer83], dentate gyrus [Kaplan77], neocortex [Gould99a], and hippocampus [Gould99b]. Furthermore, it is proposed that adult neurogenesis is responsible for the social and cognitive differences between primate and non-primate. More generally, adult neurogenesis plays a critical role in structural stability of the brain, as well as in human learning and memory [Gross00]. Hence, this gives weight to structural and parametric learnings occurring concurrently in the human brain. Although these findings are controversial and non-conclusive, they nevertheless affirm the presence of structural and parametric plasticities in the human brain. This provides a neurobiological basis to the double plasticity.

Another example of double plasticity can be found in the human visual pathway where developmental reorganization is observed in rod monochromats (a rare group of congenitally colour-blind individuals who lack cone photoreceptor function). It is reported that the cortical maps of normal and monochromats are different. Monochromats fixate stimuli at their inactive fovea rather than eccentrically. This is

due to the abnormal visual experience, which changes the cortical organization. These changes are mediated by a variety of mechanisms, including the reorganization of neural connectivity (structure learning), and changes in the efficacy of established connections (parameter learning) [Baseler02].

4.2 Formalization

The complementary learning paradigm can be formalized as follows:

Given universe of discourse U consisting T elements, $U = \{x^1, \dots, x', \dots, x^T\}$, a fuzzy set A_C representing a particular concept C , the elements in U will have unity membership if it belongs to the concept C .

$$\mu_{A_C}(x) = \begin{cases} 1, & \text{if } SM(x, A_C) \geq \rho \\ 0, & \text{otherwise} \end{cases} \quad (4.1)$$

where $SM(\bullet)$ is a function for computing the similarity measure between input and fuzzy set A_C and ρ is the predefined threshold. Similarly, elements that do not belong to concept C will have unity membership function for the concept $\neg C$.

$$\mu_{A_{\neg C}}(x) = \begin{cases} 1, & \text{if } SM(x, A_{\neg C}) \geq \rho \\ 0, & \text{otherwise} \end{cases} \quad (4.2)$$

Complementary Learning Fuzzy Neural Network (CLFNN) is FNN based on the complementary learning paradigm. By complementary learning, it refers to the followings in this work:

1. It learns from positive and negative samples.
2. It segregates positive and negative knowledge.
3. It exploits the lateral inhibition between the positive and negative knowledge.

According to this formalization, the learning models can be categorized into: (1) positive/negative system, where the system constructs knowledge, or infer, solely

4 Complementary Learning - T. Z. TAN

based on the class of interest; (2) neutral learning system, where there is no concept of positive and negative; (3) complementary learning system, where the system generates knowledge based on positive and negative classes, along with exploitation of the lateral inhibition between positive and negative examples.

Formally, CLFNN is a 9-tuple, $\langle X, Y, D, A, R, B, l, s, p \rangle$. The definition and description of each CLFNN component is given in the followings:

X is an observation described by a set of attributes $\mathbf{x} = (x_1, \dots, x_i, \dots, x_l)$, and

$X \in U_1 \times U_2 \times \dots \times U_l = U$, where U denotes the input space.

Y is the outcome(s) associated with X . $Y = Y^+ \cup Y^-$, where Y^+ and Y^- denote positive and negative outcomes respectively. $Y \in V$, where V is the output space,

and $V = V_1 \times V_2 \times \dots \times V_M$. M is the total number of possible outcomes. Note that

the outcomes are often *mutually exclusive*, that is, for all $(m, n) \in M$ and $m \neq n$,

$Y_m \cap Y_n = \emptyset$. This is to say, if $Y^+ = \{Y_m\}$, then $Y^- = Y - Y^+ = \bigcup_{n \in M, n \neq m} Y_n$.

D is the source knowledge described by a set of observation X and its corresponding outcome Y . That is, $D = \bigcup_{i=1, \dots, T} \langle X, Y \rangle$, where $T = |D|$ is the total number of

observation and outcome pairs. Since $Y = Y^+ \cup Y^-$, D can be divided into positive

and negative data, i.e. $D = D^+ \cup D^-$.

A is a set of knowledge describing the relationships amongst observations X in D .

These relationships can be represented by linguistic, logistic or mathematical term.

B is a set of knowledge describing the relationships amongst observations Y in D .

R is a rule that describes the structural relationship between A and B . In other words,

R represents the knowledge learnt from the data D , linking the inputs to the correct

outcomes. Therefore, the membership function of R , μ_R , maps A to B , that is, $\mu_R : A \rightarrow B$. As a result, $R = R^+ \cup R^-$, where R^+ contains positive knowledge constructed from D^+ and R^- contains negative knowledge constructed from D^- .

l is a learning function that constructs the CLFNN based on D . That is, $l : D \rightarrow \langle A, B, R \rangle$. It comprises two learning functions: structure learning s , and parameter learning p . The two can be carried out concurrently, or sequentially (s , then p), or independently.

s is a structure learning that constructs the network. It can be any clustering algorithm that derives the structures from data autonomously. It constructs the fuzzy sets and memory elements that describe the underlying data. It adds new node if the existing knowledge is not sufficient to describe the data. The process ends when it deems that the structure of the network is appropriate to reflect the data D . Since s is performed on X and Y spaces, s can be written as the confluence of $\langle s_X, s_Y, s_R \rangle$, where $s_X : X \rightarrow A, s_Y : Y \rightarrow B, s_R : A \rightarrow B$. Normally, the same structure learning is chosen for s_X and s_Y . Function s_R relates the input space A to output space B . Weights can be applied to the linkages if there is a need. Since s is responsible for implementing the positive and negative learnings architecture, s constructs positive and negative knowledge separately. That is,

$$s_X : \begin{cases} X^+ \rightarrow A^+ \\ X^- \rightarrow A^- \end{cases}, s_Y : \begin{cases} Y^+ \rightarrow B^+ \\ Y^- \rightarrow B^- \end{cases}, s_R : \begin{cases} A^+ \rightarrow B^+ \\ A^- \rightarrow B^- \end{cases}.$$

p is a parameter learning which fine-tunes the memory elements (u, v) of the network. Parameter learning is performed after structure learning. Parameter learning can be any supervised learning algorithm. Normally, learning algorithm

that seeks to minimize certain cost function $p(X, Y; u, v) : E_s + E_p \rightarrow \epsilon$ is employed. E_s is the structure learning error, E_p is the parameter learning error, and ϵ is a constant.

In CLFNN, the set of knowledge that describes the relationship between observation and input data is represented by a linguistic term characterized by fuzzy set A . A is a mapping of X to the linguistic representation of the CLFNN. Every X is described by J number of A . In other words, A defines a set of locations in the input space. A is characterized by its membership function μ_A . μ_A is described by memory elements (u, v) . An example is given in Equation 4.3. $\min_i$ and $\max_i$ refer to minimum and maximum value of i^{th} input dimension respectively.

$$\mu_{A_{ij}}(x_i \in X) = \begin{cases} 1, & u_{ij} \leq x_i \leq v_{ij} \\ \frac{x_i - \min_i}{u_{ij} - \min_i}, & x_i < u_{ij}, x_i \geq \min_i \\ \frac{\max_i - x_i}{\max_i - v_{ij}}, & x_i > v_{ij}, x_i \leq \max_i \\ 0, & \text{otherwise} \end{cases} \quad (4.3)$$

With reference to Equations 4.1 and 4.2, the concept C can be interpreted as R^+ , while concept $\neg C$ as R^- , and the similarity measure is the membership function, μ_R . Since positive and negative knowledge are “separated”, this induces a *lateral inhibiting* effect when one of the positive or negative knowledge is matched with the input. In other words, when D^+ is presented, R^+ will be activated, and concurrently, R^- will be deactivated, and vice versa (Equations 4.4 and 4.5).

$$\mu_{R^+}(X) = \begin{cases} 1, & X \in D^+ \\ 0, & \text{otherwise} \end{cases} \quad (4.4)$$

$$\mu_{R^-}(X) = \begin{cases} 1, & X \in D^- \\ 0, & \text{otherwise} \end{cases} \quad (4.5)$$

This is the functional model of the complementary learning paradigm, which minimizes confusion during the inference and may offer superior performance in recognition. Furthermore, modeling of the source space $\langle X, Y \rangle$ with R^+ and R^- gives a better representation of the problem.

Figure 4.6 shows the CLFNN model. It has five layers with each layer corresponds to the Mamdani's fuzzy rule model as described in Equation 4.6.

Figure 4.6: Complementary learning fuzzy neural network architecture.

This direct mapping of network structure to the fuzzy rule model allows rapid and simple rule derivation from the network. There are two data flow directions at the output layer (Y): feedforward, and feedback operations. The feedback operation is used during system learning, whereas the feedforward operation is used during system

inference. All the layers are constructed autonomously by the structure learning s , and is finetuned subsequently using parameter learning p . Since CLFNN has memory elements in Layers A and B only, the tuning is done solely on these layers. Note that the positive and negative rules are “separated”. This functionally models the lateral inhibiting relationship between positive and negative knowledge, taking advantage of the complementary effect between positive and negative rules. The CLFNN inference process is given below.

Step 1. Given input $\mathbf{x} \in X^+ \in D^+$, which is fed into Layer A .

Step 2. Compute the *Overall Similarity* (OSM) of every stored fuzzy representation for every dimension.

$$\text{OSM}_j = \frac{1}{I} \sum_1^I \mu_{A_j}(x_i), \quad j = 1, 2, \dots, J \quad (4.7)$$

Step 3. Select the winning rule r^* , which has the maximum OSM.

$$r^* = \max_j \mu_{R_j}(\mathbf{x}) \quad (4.8)$$

Step 4. Since $\mathbf{x}$ is positive sample, positive rules have greater firing strength than negative rules. Thus, positive rule is selected. This feature implements the lateral inhibiting relationship between positive and negative rules. In this case, negative rule will not fire because they are inhibited by positive rules, and this leads to correct decision. Similarly, if the input $\mathbf{x}$ is a negative sample, negative rules have greater firing strength than positive rules. As a result, negative rule is selected. That is

$$r^* = \begin{cases} r^+, & \text{if } \mu_{R^+}(X^+) > \mu_{R^-}(X^-) \\ r^-, & \text{if } \mu_{R^-}(X^-) > \mu_{R^+}(X^+) \end{cases} \quad (4.9)$$

Step 5. Determine the conclusion implied by this winning rule and output the conclusion.

Thus, every inference step of CLFNN has a one-to-one mapping to the human inference step, as shown in Table 4.1. This suggests that CLFNN has an inference process that is highly akin to that of human.

Table 4.1: Similarity of human and CLFNN inference.

Human	CLFNN
Making observations.	Feeding input to the system.
Weighting the past experience or knowledge that describes the current situation.	Computing overall similarity.
Choosing the experience or knowledge that best reflects the current situation.	Selecting maximally fired rule.
Retrieving the actions of best-matched past experience or knowledge.	Determining the conclusion implied by the maximally fired rule.
Acting out the actions.	Output the conclusion.

In multi-class classification setting, since CLFNN uses both positive and negative knowledge in its decision making, CLFNN approach multi-class classification in a way similar to divide-and-conquer. CLFNN can be viewed as a combination of multi-subcomponent where each subcomponent treats its target class positive, and other classes negative. For example, given a three-class classification task, $Y = \{A, B, C\}$, CLFNN generates its rulebase, $R = \{R_A, R_B, R_C\}$. The positive and negative rules for class A will be $R_A^+ = R_A, R_A^- = R_B \cup R_C$. During inference, CLFNN will make a decision of

$$y = \begin{cases} A & \text{if } \mu_{R_A^+}(\mathbf{x}) > \mu_{R_A^- = R_B \cup R_C}(\mathbf{x}) \rightarrow \mu_{R_A^+}(\mathbf{x}) > \mu_{R_B^+}(\mathbf{x}) \text{ AND } \mu_{R_A^+}(\mathbf{x}) > \mu_{R_C^+}(\mathbf{x}) \\ B & \text{if } \mu_{R_B^+}(\mathbf{x}) > \mu_{R_B^-}(\mathbf{x}) \\ C & \text{if } \mu_{R_C^+}(\mathbf{x}) > \mu_{R_C^-}(\mathbf{x}) \end{cases}$$

Hence, in other words, the way CLFNN treated multi-class classification can be viewed as a divide-and-conquer approach. The performance of CLFNN on three multi-class classification tasks is provided in Section 4.4.2.

4.3 Characteristics of CLFNN

The following sections list down the properties of CLFNN.

4.3.1 CLFNN offers closer representation of the problem space.

Positive/negative system makes assumptions for the uncovered space by its rulebase. The validity of these assumptions is not always justifiable. For instance, for a malignant and benign diagnosis problem, the positive system rulebase covers only the malignant class and assumes any other space not covered as benign. This renders the rulebase ambiguous when a new category such as borderline is brought in. Conversely, CLFNN explicitly generates positive and negative knowledge and makes no assumption about the space uncovered. This allows greater flexibility and extendibility, besides offering better representation of the problem space. Moreover, explicitly generating positive and negative knowledge improves the generalization performance. Illustration of the rulebase for a diagnosis problem (single output and two inputs system) is given in Table 4.2 and Figure 4.7 respectively.

Table 4.2: Rulebase of complementary learning and positive learning system.

Rules	Complementary Learning Rulebase	Positive System Rulebase
Positive	IF x_1 is very low (0-2), AND x_2 is moderately high (5.5-8.8), THEN Malignant.	IF x_1 is low (3-5), AND x_2 is high (3.7-5.6), THEN Malignant.
Negative	IF x_1 is high (9-11), AND x_2 is low (2.3-3.3), THEN Benign.	ELSE Benign.

Figure 4.7: Problem modeling of (a) complementary learning (b) positive learning.

From Figure 4.7, it can be seen that complementary learning provides a more concise modeling of the input space. It can be easily extended to include borderline information.

4.3.2 CLFNN gives better generalization.

Positive and negative learnings will give better generalization performance, as shown in Figure 4.8.

Figure 4.8: Better generalization of (a) complementary learning than (b) positive learning.

In Figure 4.8, if an unseen sample is presented to the system, positive learning system gives a misclassification, whereas for complementary learning system, it first computes the firing strength of the positive and negative rules induced by the sample. If the sample is closer to the space covered by positive rulebase, then positive rules are fired with greater strength, and therefore, the system outputs a decision of positive class. Thus, this gives a correct classification. If the unseen sample is negative, by definition, it is closer to space covered by negative rules. Thus, in this case, both systems give correct classification.

4.3.3 CLFNN handles skewed data well.

A skewed or imbalanced dataset is the one having very few instances of the positive class compared to the negative class, or vice versa. The traditional machine learning algorithm such as Naïve Bayesian, C4.5, *Support Vector Machine* (SVM), and

Artificial Neural Network (ANN) perform unsatisfactorily when the dataset is imbalanced. They either biased against the minority or the majority [Tang05]. This is problematic especially in medical data where the positive cases are often much less than the negative cases [Zhou06a]. Consequently, the highly skewed distribution poses severe hardship for ANN. It either has a region of inconsistent network performance (where its behavior is unpredictable), or it just fail to learn anything.

The unsatisfactory performance can be seen by the way many other systems make decision. In SVM, output y is decided as follow:

$$y = \begin{cases} +, & \text{if } \text{sgn}\left[f(\mathbf{x}) = \sum_{i=1}^I y' \alpha' K(\mathbf{x}, \mathbf{x}') + b\right] = 1 \\ -, & \text{if } \text{sgn}\left[f(\mathbf{x}) = \sum_{i=1}^I y' \alpha' K(\mathbf{x}, \mathbf{x}') + b\right] = -1 \end{cases} \quad (4.10)$$

The α' of the minority class tends to become much larger than those of majority class because the low number of minority class instances make them appear to be further away from the hyperplane.

In Naïve Bayesian, output y is decided as follow:

$$y = \begin{cases} +, & \text{if } P(y=+) \prod_{i=1}^I P(x=x_i|y=+) > P(y=-) \prod_{i=1}^I P(x=x_i|y=-) \\ -, & \text{otherwise} \end{cases} \quad (4.11)$$

Naïve Bayesian may select poor weights for the decision boundary due to the much lesser training examples of the minority class. The weight vectors are biased to either one of the classes.

In *Multilayer Perceptron* (MLP), the output y is decided as follow:

$$y = \begin{cases} +, & \text{if } f\left[\sum_{j=1}^J w_j \left(f_j\left(\sum_{i=1}^I w_{ij} x_i + \text{bias}_j\right)\right)\right] > \rho \\ -, & \text{otherwise.} \end{cases} \quad (4.12)$$

Since MLP adjusts the weights according to the error signal at the output layer, the weights of the minority class is tuned relatively lesser than that of the majority class. Hence, the weights are biased to the majority class, and as a result, the minority class fails in the threshold test.

As for C4.5 or any decision tree algorithm that uses recursive partitioning algorithm, minority class is difficult to learn. The performance becomes worse if the pruning method is not used properly, as the pruning may remove the nodes of minority class.

On the contrary, CLFNN inference is based on the difference between the positive and negative. Furthermore, the positive and negative spaces are separated, as illustrated in Equation 4.13.

$$y = \begin{cases} +, & \text{if } \mathbf{x} \in X^+ \text{ AND } \mu_{R^+}(\mathbf{x}) > \mu_{R^-}(\mathbf{x}) \\ -, & \text{if } \mathbf{x} \in X^- \text{ AND } \mu_{R^-}(\mathbf{x}) > \mu_{R^+}(\mathbf{x}) \end{cases} \quad (4.13)$$

Since the positive and negative regions are separated, the performance of CLFNN is less influenced if the number of minority sample is not sufficient. If $\mathbf{x}$ is not in the space covered by the positive and negative rule spaces, it is inferred based on the distance from the positive and negative rule spaces. Together with the inference involving the lateral inhibition between positive and negative samples, the segregation of positive and negative spaces offers complementary learning system more tolerances toward the imbalanced data distribution [Tan07c]. On the other hand, CLFNN represents the space more precisely and it is easier to extend its space to incorporate a new class.

4.3.4 CLFNN fires positive rules more if given a positive sample.

When a positive sample is presented to the CLFNN, the CLFNN fires the positive rules more than the negative rules, and vice versa. This property of CLFNN is in regardless

of whether the data is linearly separable or not. The sections below describe the responses of CLFNN under linearly separable data and nonlinearly separable data.

4.3.4.1 Linearly separable data

When the input space is linearly separable, the positive rules cover only the positive sample space. Likewise, the negative rules cover only the negative sample space. Thus, the rules are maximally activated if the sample presented is within the space covered by them.

4.3.4.2 Nonlinearly separable data

Samples are usually nearer to the space of their class. Sample that lies in the space different from the space of its class is known as outlier or special case. In other words, if there is no outlier or special case, the positive samples lie in the space nearer to the coverage of positive rules, and similarly for negative samples. Since positive sample is nearer to the space covered by the positive rules, one of the positive rules has greater activation than all the negative rules, and vice versa. That is, for $x \in X^+$, $\exists r^+ \in R^+$, such that $\mu_{r^+}(x) > \mu_{r^-}(x)$.

4.4 Experiment

All the experiments in this thesis are simulated on a system of Intel Pentium IV 2 GHz processor, and 512 Mb *Random Access Memory* (RAM).

4.4.1 Complementary learning on linearly separable problem

In this experiment, a representative of each learning system is selected: complementary learning-*Fuzzy Adaptive Control Network with Another Adaptive Resonance Theory* (FALCON-AART) [Tan05b]; positive learning-*Pseudo Outer Product Fuzzy Neural Network* (POPFNN) [Zhou96]; neutral learning-*Multilayer Perceptron* (MLP)

4 Complementary Learning - T. Z. TAN

[Witten05], and a decision tree – C4.5. A linearly separable syntactic dataset (given in Appendix D) is used to assess the system performance (see Figure 4.9).

Figure 4.9: Syntactic data.

All systems are able to attain 100% recognition accuracy in this dataset. As seen from Figure 4.10, the CLFNN offers a closer representation of the problem space. CLFNN constructs positive and negative rules to cover positive and negative samples respectively. In contrast, positive system such as POPFNN only stores the positive rules, and assumes the uncovered as negative. This prohibits a finer categorization of classes as well as increases the risk of misclassification. In other words, complementary learning has greater extendibility when compared against positive system. On the other hand, C4.5 uses crisp rules, which cannot handle data uncertainty. C4.5 behaves like positive system in this case because the rules are generated on the basis of a single feature. The discriminating line of MLP nicely separates the positive and negative classes. However, no rules are generated to explain its computation. When the data is non-linearly separable, the discriminating lines (decision boundaries) are very complicated, and therefore, difficult to understand the system. Figures 4.7 and 4.10 shows that CLFNN offers a better representation of the problem space, in which the positive and negative spaces are defined unambiguously. This eases the understanding of the system and improves the system performance.

4 Complementary Learning - T. Z. TAN

Figure 4.10: The discriminating decision boundaries of (a) decision tree C4.5; (b) POPFNN; (c) MLP; (d) CLFNN. Shaded area is positive space.

4.4.2 Complementary learning on multi-class classification problem

When complementary learning is applied on multi-class classification, it generates positive and negative rules for each class. For example, given a three-class classification task, $Y = \{A, B, C\}$, CLFNN generates its rulebase, $R = \{R_A, R_B, R_C\}$.

The positive and negative rules for class A will be $R_A^+ = R_A, R_A^- = R_B \cup R_C$. During inference, complementary learning response is similar to the divide-and-conquer approach. Since the rulebase of each class is formulated in positive and negative rules, the rules generated for each class is the same as the case of approaching the problem using divide-and-conquer method. The following shows the performance of complementary learning in multi-class classification.

4.4.2.1 Iris

Fisher's Iris data [Fisher36] is a well-known benchmark data for classification. Measurements are done on sepal length, sepal width, petal length and petal width to determine species of Iris. There are three species all together, namely, Sentosa, (Y_A), Virginica, (Y_B), and Versicolor, (Y_C). The performance of complementary learning and common in Iris classification, a three-class classification problem is summarized in Table 4.3.

Table 4.3: Complementary learning performance on iris classification task.

Methods	Training Accuracy (%)	Testing Accuracy (%)	Number of Rules	Training time (s)
Naïve Bayesian	92.15	97.97		0.33
MLP	98.03	97.97		0.14
SVM	94.16	97.97		0.14
C4.5	100.0	94.94	5	0.02
FALCON-AART	100.0	94.94	5	0.14
GCL	98.03	97.97	5	0.08
HCL	100.0	94.94	6	2.31

Most of the methods have similar performance. The sensitivity of each class for all the methods is about 90% ($Y_A \approx 96\%$, $Y_B \approx 90\%$, $Y_C \approx 94\%$), suggesting the iris classification is highly separable. Attempting the iris classification using divide and conquer also gives similar sensitivity for each class ($Y_A \approx 100\%$, $Y_B \approx 94\%$, $Y_C \approx 94\%$). This suggests the number of rules generated by decision tree and complementary learning is similar. However, as complementary learning generates positive and negative rules, the number of rules generated is generally higher. HCL takes the longest time to learn the problem. This is because HCL learns the problem feature by feature, and hence slows down the learning process.

4.4.2.2 Lymphoma

Lymphography is the clinical standard for accessing problem with human lymphatic system. However, it has been phased out in the past 15-20 years by less invasive techniques. Computed tomography scanning is one of them. This dataset is taken from the University Medical Centre, Institute of Oncology, Ljubljana, Yugoslavia [Kononenko84]. A total of 148 instances can be found in this dataset. Class distribution: 2 instances of normal, 81 instances of metastases, 61 instances of malign lymph, and 4 instances of fibrosis. The result of the classification is shown in Table 4.4.

Table 4.4: Complementary learning performance on lymphoma classification task.

Methods	Training Accuracy (%)	Testing Accuracy (%)	Number of Rules	Training time (s)
Naïve Bayesian	90.56	85		0.01
MLP	100	84		0.7
SVM	94.36	84		0.61
C4.5	94.36	82	8	0.03
FALCON-AART	100	84.0	37	15.89
GCL	90.75	82	12	0.17
HCL	77	77	55	15.39

As seen from Table 4.4, complementary learning in general constructs more rules.

This is because positive and negative rules are specifically generated for every class. HCL in this case does not perform as good, owing to the reason that the input is categorical. Since HCL infers based on single dimension, the accuracy is significantly lower as the inter-relationship between the features are not used for classification.

4.4.2.3 Leukemia subtype classification

The same dataset is used in Section 7.5.1. The result of multi-class classification is shown in Table 4.5. In this experiment, to reduce the computation cost and to improve the system performance, the ten most relevant genes are selected using SVM feature selection method.

4 Complementary Learning - T. Z. TAN

Table 4.5: Complementary learning performance on Leukemia subtype classification task.

Methods	Training Accuracy (%)	Testing Accuracy (%)	Number of Rules	Training time (s)
Naïve Bayesian	99	96.93		0.02
MLP	98	97		1.36
SVM	96.93	96.93		1.7
C4.5	100.0	93.02	8	0.03
FALCON-AART	98.83	98.15	40	3.57
GCL	98.25	98.15	61	1.34
HCL	100	81.7	130	198.0

Comparing the results of Tables 4.5 and 7.3, it is shown that the accuracies of the methods in multi-class classification improve. This is unusual because multi-class classification is a more difficult task. This happens because of different gene sets used, and these gene sets work better with the methods. The other reason is that the differences between the classes are more obvious as there are more classes to contrast with. Generally, complementary learning takes longer time to learn the problem, owing to its relatively larger rulebase. In particular, HCL takes longer time to learn the problem due to its huge rulebase, where positive and negative rules are generated for all six different classes. As only single dimension is taken into account, HCL suffers in poor accuracy in this case.

4.4.3 Complementary learning on a hard problem

This experiment seeks to demonstrate how complementary learning would perform on a hard problem—*Central Nervous System* (CNS) embryonal tumor outcome prediction [Pomeroy02]. Outcome prediction is a hard problem because little is known biologically about the disease [Pendharkar06][Duval85]. Since the embryonal tumor pathogenesis is not known, and no consensus on its relationship with other embryonal tumor, the response of patients to therapy is difficult to predict. This is because the response of patients to therapy is highly variable. A therapy that works for one patient may worsen the disease of the other. Unfortunately, the current prognostic factor, i.e.

4 Complementary Learning - T. Z. TAN

staging, is not sufficiently informative to accurately predict the clinical patient outcome. It is found that there are genes expressions that correlate with the medulloblastoma CNS tumor. Thus, it is promising to look at DNA gene expression to predict the patient outcome.

The dataset adopted consists of DNA gene expression of 68 children with medullablastomas [Pomeroy02]. Each of the patients with medullablastomas was treated with carniospinal irradiation to 2400-3600 centiGray (cGy), and chemotherapy of cisplatin and vicristine and combinations of carboplatin, etoposide, cyclophosphamide, procarbor lomustine. A total of 7129 genes are found in this dataset and the ten most relevant genes are chosen for outcome prediction, as shown in Table 4.6. It is selected by using the signal-to-noise ratio criteria [Pomeroy02]. Not all genes are used because not every gene is useful in terms of determining the outcome. Irrelevant genes will disrupt the learning process as it is noise to the system.

Table 4.6: Ten most relevant features for predicting patient outcome.

1. Ribosomal protein S18	6. PLOD Lysyl hydroxylase
2. RPL7A Ribosomal protein L7a	7. APOD Apolipoprotein D
3. MYBL2 V-myb avian myeloblastosis viral oncogene homolog-like 2	8. KIAA0220 gene, partial cds
4. Ribosomal protein S10 mRNA	9. Neuron-specific vesicle coat protein and cerebellar degeneration antigen (beta-NAP) mRNA
5. Chorionic gonadotropin (hcg) beta subunit mRNA	10. Gps2 (GPS2) mRNA

Three-fold cross-validation is used to assess the systems performance in the outcome prediction task, and it is given in Table 4.7. Recall is the training accuracy and predict is the testing accuracy.

4 Complementary Learning - T. Z. TAN

Table 4.7: Performance of different systems on CNS embryonal tumor outcome prediction.

Method	Recall (%)	Sensitivity (%)	Specificity (%)	Testing (%)	Time (s)
C4.5	93.54	60	73.7	68.96	0.03
kNN	100	80	73.7	75.86	0.01
MLP	96.77	70	89.5	82.75	0.17
SVM	83.87	50	100	82.75	0.02
FALCON-AART	100	30	100	75.86	0.8
GCL	90.32	90	100	93.1	0.05
HCL	100.0	84.21	40	69.0	2.96

As shown in the Table 4.7, the task of predicting the outcome is not trivial. Most of the methods attain about 70-80% of accuracy, suggesting the difficulty in outcome prediction. The number of rules generated by C4.5, FALCON-AART, GCL and HCL are 4, 12, 17 and 14 respectively. From the accuracy of the C4.5, it indicates that four rules are not sufficient to describe this complicated dataset. The performance of complementary learning is as competent as the other methods, albeit the accuracy is not that good for all methods. In this problem, FALCON-AART and GCL have training time similar to conventional methods. HCL processes feature by feature sequentially, and this resulted its long training time. Viewing from the other perspective, the number of rules generated by complementary learning system is higher than the rest. In this case, complementary learning has rule base of 30% larger than decision tree. To assess how the system perform on this hard problem, the relationship between number of feature and time, the number of rules, and accuracy is shown in Figures 4.11 – 4.14.

4 Complementary Learning - T. Z. TAN

Figure 4.11: Relationship between complementary learning accuracy and number of features.

Figure 4.12: Relationship between complementary learning time and number of features.

Figure 4.13: Relationship between number of rules and number of features.

Figure 4.14: Relationship between number of rules and training time.

Figures 4.11 – 4.13 indicates the performance of complementary learning when the number of features increases. In general there is no obvious trend for the number of rules and the accuracy, suggesting that not all the features are relevant in performing the outcome prediction. As for the training time, HCL training time increases as the number of features linearly. This is because HCL processes features sequentially. Paradoxically, GCL and FALCON-AART approach the features collectively. No observation on the increases of training time for GCL and FALCON-AART because the number of features is small. The number of rules does not correlate with the number of features. However, since GCL and FALCON-AART treat the features collectively, the complexity of the rules elevates when the number of features increases. From Figure 4.14, it is shown that as the number of rules constructed by the complementary learning system increases, the training time increases as well.

How one system performs on a hard problem depends heavily on the system computation and resource efficiency. From the relationships of the complementary learning performances in Figures 4.11 – 4.14, the number of rules emerges as one of the most influencing factors on the efficiency and the capability of complementary learning system in approaching hard problem. Apart from the number of rules, a closely related

factor, is the way of rule assessment, in particular, the fitness function in GCL. This is because fitness function has to be run once for every rule in the population. Hence, selecting a less computational expensive fitness function is important. The other factor that influences significantly the efficiency and effectiveness of the system is the number of features. The present HCL implements a simple feature selection method, whereas the FALCON-AART and GCL relies on an external preprocessing method to select the best feature, which discounted the efficiency of the system. Thus, in order to effectively approach hard problem using complementary learning system, one needs to reduce the number of features, the number of rules, as well as carefully select an appropriate fitness function to evaluate the rule.

4.5 Summary

Complementary learning is a biological plausible mechanism that underlies various biological activities. This complementary learning can be explicit and implicit. A formalization of the *Complementary Learning-based Fuzzy Neural Network* (CLFNN) is given in this chapter. Theoretically, CLFNN performs better in classification than that of positive/negative and neutral learning systems. CLFNN provides positive and negative rules for modeling of the problems space. It has greater interpretability than neutral learning system because neutral learning does not have the concept of positive and negative samples. However, the provision of positive and negative rules in CLFNN also poses an interpretability problem. Since CLFNN explicitly generates positive and negative rules, its rulebase is generally larger than positive/negative systems. This problem may be alleviated by implementing some constraints to the learning. In order to further analysis of CLFNN, experiment of CLFNN on a hard problem is conducted and is summarized in Appendix E.

4 Complementary Learning - T. Z. TAN

learning. In order to further analysis of CLFNN, experiment of CLFNN on a hard problem is conducted and is summarized in Appendix E.

Chapter 5 FALCON-AART

FALCON-AART [Tan05b] functionally models the complementary learning paradigm of human pattern recognition. FALCON-AART incorporates the complementary learning into the FALCON-ART [Lin96]. FALCON-ART is a system with *Fuzzy Adaptive Learning Control Network* (FALCON) structure, which uses fuzzy *Adaptive Resonance Theory* (ART) to construct the network. The FALCON is a 5-layer architecture, which maps onto the elements of fuzzy rules. This will be described in Section 5.1. However, FALCON is a structure. It requires a learning algorithm to construct its network. In FALCON-ART, fuzzy ART algorithm is used to perform fuzzy clustering on the input-output space. Fuzzy ART is an online similarity measure algorithm, which resonates to include a sample into a particular cluster if the sample passes the similarity threshold test.

5.1 Architecture

One advantage of CLFNN is its ability to generate positive and negative fuzzy rules in the form given in Equation 5.1:

$$\text{IF } x_1 \text{ is } A_1 \text{ and } x_2 \text{ is } A_2, \text{ THEN } y_1 \text{ is } B_1 \text{ and } y_2 \text{ is } B_2 \quad (5.1)$$

The fuzzy rule in Equation 5.1 is an example of a system with two inputs and two outputs. It consists of five elements:

- Input linguistic variables (x_1, x_2) .
- Input linguistic terms (A_1, A_2) . This represents fuzzy entities such as tall, short, thin, fat, and so on. FALCON-AART represents input linguistic terms by using trapezoidal membership function.
- If-Then rule: links the antecedent part (i.e. input linguistic variables and terms as above) with the consequent part (i.e. output linguistic variables and terms as below).

- Output linguistic variables (y_1, y_2).
- Output linguistic terms (B_1, B_2).

Each layer of FALCON-AART, shown in Figure 5.1, is mapped onto the elements of fuzzy rule listed in Equation 5.1. Every link in FALCON is unity weight. In other words, output of preceding layer is fed in directly. The followings show the node operations.

Let $\mathbf{x}^t = (x_1^t, \dots, x_i^t, \dots, x_I^t)$ denotes the input vector to the network, where $t=1,2,\dots,T$ denotes the number of pattern, $i \in \{1,2,\dots,I\}$ is the index of input dimension. Let (u_{ij}, v_{ij}) and (u_{lm}, v_{lm}) be the weights of input and output linguistic terms respectively. u_{ij} denotes link weight from node i to node j .

Figure 5.1: FALCON-AART architecture.

Let the desired output vector of pattern t be $\mathbf{d}' = (d'_1, \dots, d'_m, \dots, d'_M)$, and the index of output dimension be denoted by m . Index for input linguistic term node is $j \in \{1, 2, \dots, J\}$. $k \in \{1, 2, \dots, K\}$ is used to index rule node, whereas $l \in \{1, 2, \dots, L\}$ is used to index output linguistic term node. Output of node is represented by z .

Layer 1-Input Linguistic node: Each node in this layer corresponds to one input linguistic variable. The transfer function of each node is shown in Equation 5.2:

$$\begin{aligned} z_i &= a_1(x_i) \\ &= x_i \end{aligned} \quad (5.2)$$

where x_i is the i^{th} element of input vector $\mathbf{x}$, $a_1(\bullet)$ is the activation function, and z_i is the output of the i^{th} node.

Layer 2-Input Terms node: Each node in Layer 2 represents fuzzy sets of the input space characterized by trapezoidal activation function. It is computed using Equations 5.3 and 5.4.

$$\begin{aligned} z_{ij} &= a_2(\text{net}_{ij}) \\ &= 1.0 - g(\text{net}_{ij} - v_{ij}, \gamma) - g(u_{ij} - \text{net}_{ij}, \gamma) \end{aligned} \quad \begin{matrix} (u_{ij}, v_{ij}) \\ (5.3) \end{matrix}$$

where net_{ij} is the total synaptic input from the i^{th} node in Layer 1 to the j^{th} node in Layer 2, $a_2(\bullet)$ is the activation function, z_{ij} is the output of the j^{th} node, u_{ij} is the lower bound of the weight, and v_{ij} is the upper bound of the weight. The function $g(\bullet)$ is defined by Equation 5.4.

$$g(s, \gamma) = \begin{cases} 1, & \text{if } s\gamma > 1 \\ s\gamma, & \text{if } 0 \leq s\gamma \leq 1 \\ 0, & \text{if } s\gamma < 0 \end{cases} \quad (5.4)$$

In Equation 5.4, the γ parameter controls the fuzziness of the trapezoidal membership function. Larger γ means crisper fuzzy set. Each input term node is connected to a node in Layer 3, forming an I-dimensional membership function in the input space.

Layer 3-Rule node: The rule node maps the antecedent and the consequent. To compute the overall matching degree, it performs the product operation defined in Equation 5.5.

$$\begin{aligned} z_{\text{rule } k} &= a_3(z_{ik}) \\ &= \prod_{i=1}^I z_{ik} \end{aligned} \quad (5.5)$$

where $z_{\text{rule } k}$ is the output of the k^{th} rule node, $a_3(\bullet)$ is the activation function, and I is the number of input nodes.

Layers 2 and 3 implement the complementary learning paradigm. Layers 2 and 3 form the membership functions of the positive and negative knowledge separately, so that the lateral inhibition between the positive and negative classes can be exploited. This is to functional model the complementary learning in *hippocampus* of human brain. Hippocampus is often implicated in human learning and memory processes. It has positive (*long-term potentiation*, LTP) as well as negative feedback signals (*long-term depression*, LTD) that stimulate and suppress the firing of a neuron respectively. FALCON-AART functionally models this mechanism, such that when a positive sample is presented, only positive rules are activated, and concurrently, the negative rules are inhibited.

Layer 4-Output Linguistic Term node: Nodes in Layer 4 can operate in training and inference modes. During training, the desired output is fed into the network, nodes in this layer perform the fuzzy OR operation on the fired rule nodes linking to the same consequent, as shown in Equation 5.6.

$$\begin{aligned} z_{lm} &= a_4(z_{\text{rule } k}) \\ &= \max(z_{\text{rule } 1}, z_{\text{rule } 2}, \dots, z_{\text{rule } k}) \end{aligned} \quad (5.6)$$

where z_{lm} is the output of the l^{th} output linguistic term node, and $a_4(\bullet)$ is the activation function. During the inference phase, nodes in Layer 4 operate like those in

Layer 2. Layer-3-links define the precondition of the rule nodes, and Layer-4-links define the consequents of the rules. In other words, links between Layers 3 and 4 are the connectionist inference engine of FALCON-AART.

Layer 5-Output Linguistic Variable: Layer-5-node operates in two modes as well. During training, the Layer-5-node performs exactly like Layer-1-node. During the inference phase, Layer-5-node computes the output by using the *centre of area* defuzzification operator defined in Equation 5.7.

$$y_m = a_5(z_{lm})$$

$$= \frac{\sum_{l=1}^L m_{lm} z_{lm}}{\sum_{l=1}^L z_{lm}} \quad (5.7)$$

where y_m is the output of m^{th} node, $a_5(\bullet)$ is the activation function, and

$$m_{lm} = \frac{u_{lm} + v_{lm}}{2}.$$

These operations of FALCON-AART are highly akin to human learning. When presented some stimuli, human performs *feature analysis* prior to the consolidation of stimuli-response into memory. Stimuli are broken down into a series of component features such as angles, curves, shapes, edges, colour, etc (i.e. $\mathbf{x}' = (x'_1, \dots, x'_l)$), as described by *Gestalt's laws of organization* [Hart83] (Figure 5.2).

Figure 5.2: Human feature analysis model.

As seen in Figure 5.2, the construct of human recognition is analogous to the FALCON-AART architecture. The stimuli presented to the model are the inputs to FALCON-AART. After the stimuli are passed on to the retina, retina cells measure the similarity based on their receptive fields. This operation is the same in FALCON-AART Layer 2, where each node compares the input with its stored fuzzy sets. The feature detection phase that assays the overall matching degree is a one-to-one mapping to the FALCON-AART rule matching process. Subsequently, it determines which knowledge best represents the stimuli, and then decision is made based on the selected knowledge. This closely resembles the process of FALCON-AART selecting the maximum firing rule, and deducting the conclusion linked by the rule. Together, this supports the plausibility of FALCON-AART structure for simulating human-like recognition.

5.2 Learning

Learning theory [Thompson87] states that knowledge and language acquisition follows the principles of reinforcement and conditioning. These psychology theories provide the operations of FALCON-AART a natural interpretation, since FALCON-AART

learns from the input x (i.e. stimulus in classical conditioning), as well as from the mistakes that it makes (i.e. reinforcement in operant conditioning). The *stimulus generalization* in psychology describes the phenomena that human responds to a similar stimulus in a similar way, supporting the generalization by means of fuzzy sets in FALCON-AART. Together, these show that human learning involves the classical and operant conditionings. Owing to that reason, FALCON-AART has two states of learning. The relation of the FALCON-AART learning and human learning is listed in Table 5.1.

Table 5.1: Comparison of FALCON-AART learning with psychological theories.

Psychological Model	FALCON-AART Model
Classical Conditioning	Structure Learning, where stimuli are remembered
Operant/ Instrumental Conditioning	Parameter Learning, where consequence of response are learned

This section outlines the structure and parameter learnings in FALCON-AART. Initially, FALCON-AART is without any rule or membership function apart from input and output layers. As training commences, FALCON-AART uses the modified version of fuzzy ART learning algorithm [Tan05b] to construct its network and membership functions. FALCON-AART learning consists of two phases, namely structure and parameter learnings. They are described in the following sections. These learning procedures are applied on a dataset composed of T data pairs $\{(\mathbf{x}^t, \mathbf{d}^t), t = 1, 2, \dots, T\}$, where input vector $\mathbf{x}^t = (x_1^t, x_2^t, \dots, x_1^t)$ and desired output vector $\mathbf{d}^t = (d_1^t, d_2^t, \dots, d_M^t)$.

5.2.1 Structure Learning

FALCON-AART works in a two-sided manner, i.e. both input and output data are simultaneously fed into the network at Layers 1 and 5 respectively. Accordingly,

input and output clustering can be performed at the same time by using the modified fuzzy ART algorithm. After which, mapping of the input and output clusters are carried out. Besides, FALCON-AART learns incrementally, as human are continuously adapting themselves to environment and external stimuli.

The structure learning algorithm of FALCON-AART is described as follows:

Step 1: Present the training data pair $(\mathbf{x}', \mathbf{d}')$.

Step 2: Create input and output clusters using the current datum if there is no rule node. Then, create a new rule to link the two clusters, and go to Step 6.

Step 3: Select a rule r^* with maximum firing strength as described in Equation 5.8:

$$r^* = \arg \max(z_{\text{rule } 1}, z_{\text{rule } 2}, \dots, z_{\text{rule } K}) \quad (5.8)$$

Step 4: With rule r^* , simultaneously find its associated input and output clusters at Layers 2 and 4 respectively. Mark the input and output clusters as the winning clusters. For the winning input and output clusters, employ Equations 5.9 and 5.10 as the vigilance test respectively. If the winning input and output clusters pass the vigilance test, go to Step 6. Otherwise go to Step 7.

$$\sum_{i=1}^I (u_{ij} - v_{ij}) \leq (1 - \rho) \sum_{i=1}^I \text{Max}_i \quad (5.9)$$

where $\text{Max}_i = \max(x_i^1, x_i^2, \dots, x_i^T)$, and ρ is a vigilance parameter for clustering (large ρ results in fine clustering).

$$\sum_{m=1}^M (u_{lm} - v_{lm}) \leq (1 - \rho) \sum_{m=1}^M \text{Max}_m \quad (5.10)$$

where $\text{Max}_m = \max(d_m^1, d_m^2, \dots, d_m^T)$.

Step 5: Reset the firing strength of the rule associated with the winning clusters to zero so that this rule will not be selected again. If there is a rule whose firing strength is greater than zero, perform Step 3. Otherwise, create a new input and output clusters using the current training data, i.e. $(\mathbf{x}', \mathbf{d}')$. Create a new rule to link these new clusters.

Step 6: The input cluster is updated using Equations 5.11 and 5.12, whereas the output cluster is updated using Equations 5.13 and 5.14.

$$\mathbf{u}^{t+1} = \mathbf{u}^t - \beta(\mathbf{u}^t - \mathbf{x}^t) \quad (5.11)$$

$$\mathbf{v}^{t+1} = \mathbf{v}^t + \beta(\mathbf{x}^t - \mathbf{v}^t) \quad (5.12)$$

$$\mathbf{u}^{t+1} = \mathbf{u}^t - \beta(\mathbf{u}^t - \mathbf{d}^t) \quad (5.13)$$

$$\mathbf{v}^{t+1} = \mathbf{v}^t + \beta(\mathbf{d}^t - \mathbf{v}^t) \quad (5.14)$$

where β is a learning function defined in Equation 5.15:

$$\beta = e^{-\phi/\lambda} \quad (5.15)$$

where

$$\phi = \begin{cases} 0, & \text{if } r^* \text{ is a new rule} \\ 1, & \text{if } r^* \text{ is a best-fit rule} \\ 2, & \text{Otherwise} \end{cases} \quad (5.16)$$

and

$$\lambda = \lambda_{ini} (\lambda_{final} / \lambda_{ini})^{q/g} \quad (5.17)$$

In Equation 5.17, q is the epoch number, g is the age of the cluster, λ_{ini} is the initial degree of fuzziness, and λ_{final} is the final degree of fuzziness. In Equation 5.17, $g \geq 1$, $\lambda_{final} \approx 0$ and $\lambda_{ini} \geq \lambda_{final}$. Since $\lambda_{ini} \geq \lambda_{final}$, λ is a monotonically decreasing function. β is a distance-weighting function [Baraldi02] that is dependent on the current epoch number, q . The system has greater plasticity initially, and as the age (epoch number) grows, the plasticity decreases. Thus, the incorporation of q , diminishes the stability-plasticity dilemma. Excessive plasticity that leads to *catastrophic forgetting* is also avoided. FALCON-AART allows cluster growing to include new training data as long as its ‘age’ is younger than a predefined age, g . After that the cluster will not grow further. This is a model of the human visual cortical plasticity. In general, at younger age, people learn faster, where at later age, people learn slower. This is because memory is stored in the interconnection between neurons, and synapses, where the neuronal plasticity decreases over time. We learn slower and more difficult as we grow older. This is supported by neuroscience evidence: (1) the exposure of an adrenal stress hormone *glucocorticoids* increases as age increases, which inhibits the neurogenesis in hippocampus [Lupien98]; (2) *Adenylyl Cyclase subtype 1*

(AC1), which is implicated in memory process and spatial learning, decreases as age progresses and therefore reduces the learning capability [Mons04]; (3) Gan et. al. have proved that the plasticity of interneuronal connection deteriorates, as animal grows older [Gan03]. In other words, human learning is a non-linear curve. The plot of visual cortical plasticity in Figure 5.3 justifies this statement [Cooper95]. FALCON-AART structure learning is a functional model of this concept. It is a function of time, and the learning rate decreases exponentially as “age” grows.

Figure 5.3: Visual cortical plasticity as a function of age.

Adopting the concept of age, the system learns more in the first few passes of the training samples and learns little in later stage of the training. In this way, the stability and robustness against data presentation order are ensured.

Step 7: If there is no more training data, then terminate learning. Otherwise, go to Step 1 and present the next training data pair $(\mathbf{x}^{t+1}, \mathbf{d}^{t+1})$.

FALCON-AART tunes its membership functions by adjusting the slope parameter γ as well as the location of its kernel, (u, v) . The tuning of γ for the input/output terms is shown in Equations 5.18 and 5.19 respectively:

$$\gamma = [(Max_i - v_{ij}) + (u_{ij} - Min_i)] / 2 \quad (5.18)$$

where $Max_i = \max(x_i^1, x_i^2, \dots, x_i^T)$, and $Min_i = \min(x_i^1, x_i^2, \dots, x_i^T)$.

$$\gamma = [(Max_m - v_{lm}) + (u_{lm} - Min_m)] / 2 \quad (5.19)$$

where $Max_m = \max(d_m^1, d_m^2, \dots, d_m^T)$, and $Min_m = \min(d_m^1, d_m^2, \dots, d_m^T)$.

By tuning the slope and location simultaneously, one more degree of freedom is gained, and hence, the system may learn the problem better and faster. Note that the clusters are assumed to be overlapping. Each input belongs to every cluster to a certain degree. Through γ tuning, non-convex data can be effectively handled. Since γ is determined by the system itself, users are freed from having to search for the optimal value of γ .

5.2.2 Parameter Learning

The second stage of FALCON-AART learning aims at fine-tuning the membership functions and fuzzy rules learned. The backpropagation is adopted, with slight modification guided by psychology theory. The network is tuned until its learning error is below a threshold. The learning error is computed using Equation 5.20:

$$E = 0.5[\mathbf{d}' - \mathbf{y}']^2 \quad (5.20)$$

where $\mathbf{d}'$ is the desired output and $\mathbf{y}'$ is the computed output. Error is propagated from Layers 5 to 2, and while the error is propagated, weights at Layers 5 and 2 are updated. According to *Proster Theory* [Hart83], all current learning is heavily influenced by previous learning and by a vast array of stored bias. Hence, the previous weight updates are also considered during current weight updates in Layers 5 and 2.

For simplicity, let w'_{ij} denotes (u'_{ij}, v'_{ij}) , w'_{lm} denotes (u'_{lm}, v'_{lm}) , and δ^k denotes the error at layer k . Let η be the learning rate. The learning algorithm is described from Equations 5.21 to 5.28:

Layer 5: The updating rule for the weight at Layer 5 is given in Equation 5.21:

$$w'_{lm}{}^{t+1} = (1 - \alpha)[w'_{lm}{}^t + \eta \frac{\partial E}{\partial w'_{lm}}] + \alpha w'_{lm}{}^{t-1} \quad (5.21)$$

where α is the momentum constant, and

$$\begin{aligned}
\frac{\partial E}{\partial w_{lm}} &= \frac{\partial E}{\partial a_5} \frac{\partial a_5}{\partial w_{lm}} \\
&= -[\mathbf{d}^t - \mathbf{y}^t] \frac{z_{lm}}{2 \sum_{l=1}^L z_{lm}}
\end{aligned} \tag{5.22}$$

Error at Layer 5 that will be propagated to Layer 4 is computed using Equation 5.23:

$$\begin{aligned}
\delta^5 &= -\frac{\partial E}{\partial a_5} \\
&= -[\mathbf{d}^t - \mathbf{y}^t]
\end{aligned} \tag{5.23}$$

Layer 4: The error at Layer 4 is computed by Equation 5.24:

$$\begin{aligned}
\delta^4 &= -\frac{\partial E}{\partial a_4} = -\frac{\partial E}{\partial a_5} \frac{\partial a_5}{\partial a_4} \\
&= \delta^5 \left(m_{lm} \sum_{l=1}^L z_{lm} - \sum_{l=1}^L m_{lm} z_{lm} \right) / \left(\sum_{l=1}^L z_{lm} \right)^2
\end{aligned} \tag{5.24}$$

Layer 3: The nodes at this layer propagate the error to Layer 2. The error of the rule node is computed by Equation 5.25. Note that $\frac{\partial a_4}{\partial a_3} = \frac{z_{\text{rule } k}}{z_{\text{rule max}}}$ because only the winning rule fires.

$$\delta^3 = -\frac{\partial E}{\partial a_3} = -\frac{\partial E}{\partial a_4} \frac{\partial a_4}{\partial a_3} = \delta^4 \frac{z_{\text{rule } k}}{z_{\text{rule max}}} \tag{5.25}$$

Layer 2: The updating rule for weight in Layer 2 is given in Equation 5.26.

$$w_{ij}^{t+1} = (1 - \alpha) \left[w_{ij}^t + \eta \frac{\partial E}{\partial w_{ij}} \right] + \alpha w_{ij}^{t-1} \tag{5.26}$$

where

$$\begin{aligned}
-\frac{\partial E}{\partial w_{ij}} &= -\frac{\partial E}{\partial a_3} \frac{\partial a_3}{\partial a_2} \frac{\partial a_2}{\partial w_{ij}} \\
&= \delta^3 \prod_{k \neq i} z_k \frac{\partial a_2}{\partial w_{ij}}
\end{aligned} \tag{5.27}$$

and

$$\frac{\partial a_2}{\partial w_{ij}} = \begin{cases} \gamma, & \text{if } 0 \leq (x_i' - w_{ij}')\gamma \leq 1 \\ 0, & \text{otherwise} \end{cases} \tag{5.28}$$

Note that both learning rate η and momentum α are adaptive. They are adjusted according to the difference between current and previous errors. If the current error is greater than the previous, η is decreased, and α is set to zero, as this suggests the direction along the gradient may be erroneous. Thus, the learning is slower to ignore previous knowledge. Otherwise, η is increased (large step means quick learning in the direction conveyed by the sample) and α is set as a positive value (the larger the α value, the more is learnt from the gradient of previous learning). This indicates that the current direction may be promising. Thus, the learning is faster to take into account the previous knowledge. This corresponds to the *Thorndike's Law of Effect* [Hart83], where the system rewards satisfying response and punishes unsatisfying response so that it is less likely to be repeated.

The purpose of using momentum α is to improve the speed of convergence. By adding a fraction of the previous weight change to the current weight change, not only this coincides with psychology theory, but also reinforces the movement in the same direction on successive steps. Furthermore, the factor $(1-\alpha)$ in Equation 5.26 smoothens the descent path by preventing abrupt changes in the gradient due to local anomalies. Hence, it is likely to suppress any oscillations resulting from changes in the slope of the surface. Another advantage of $(1-\alpha)$ is that it protects the synaptic weights from too quick adaptation and hence prolongs the synaptic memory. This subsequently improves the robustness of the network against noise.

5.2.3 Learning Termination

Normally, learning of iterative neural network is stopped when the mean square error converges. However, this may take a very long time or the network may just unable to converge. FALCON-AART learning terminates when the mean square error between

two consecutive epochs is sufficiently small, as defined in Equation 5.29. This offers a faster learning.

$$\text{training} = \begin{cases} \text{stop,} & \text{if } E^q - E^{q-1} \leq \varepsilon \\ \text{next iteration,} & \text{otherwise} \end{cases} \quad (5.29)$$

where ε is the predetermined threshold value that defines “sufficiently small”.

5.3 Tractability

From Chapter 3, a tractable system has to meet the criteria:

- Step by step inference
- Human-like reasoning
- CDDM procedure
- Explanation capability
- User familiar terms.

Step by Step inference

In order to ensure the tractability of FALCON-AART, it is mapped onto the fuzzy inference schemes called *Analogy Approximation Reasoning Scheme* (AARS) [Turksen90], so as to give FALCON-AART a clearly defined step by step inference process. The mapping formalizes the inference process of FALCON-AART, and equips it with stronger theoretical foundation as well as tractability. Functions performed by each layer of FALCON-AART-AARS are derived from processing steps of AARS fuzzy rule-based system. FALCON-AART-AARS inference steps are listed here:

Step 1:Input Fuzzification

Inputs $\mathbf{x}' = (x'_1, x'_2, \dots, x'_I)$ are fuzzified to $\overline{\mathbf{x}'} = (\overline{x'_1}, \overline{x'_2}, \dots, \overline{x'_I})$ before they are fed into the FALCON-AART-AARS. FALCON-AART-AARS uses singleton fuzzifier to perform this task.

Step 2: Antecedent Matching

Fuzzified input fuzzy sets are compared with their corresponding rule antecedents. Subsequently, *Similarity Measure* (SM) is reckoned from different *Distance Measure* (DM) such as disconsistency measure and Houdroff measure. However, when disconsistency measure is used, the SM will be translated to range [0.5, 1] [Turksen90]. This is undesirable because this reduces the differences between rules. Furthermore, the definition of SM given by Turksen [Turksen90] involves division, which is computationally expensive. Hence, the definition of SM in [Zhou99] is used instead (see Equation 5.30):

$$SM = 1 - DM \quad (5.30)$$

It is less expensive in terms of computation, yet it preserves the relation of SM and DM. Thus, Equation 5.30 can be rewritten as Equation 5.31

$$\begin{aligned} SM &= \mu_{\overline{x_i^t \cap w_{ij}}} (x_i^t) \\ &= \overline{x_i^t} \cap w_{ij} \\ &= \mu_{\overline{x_i^t}} (x_i^t) * \mu_{w_{ij}} (x_i^t) \end{aligned} \quad (5.31)$$

Since $\overline{x_i^t}$ is singleton, Equation 5.31 can be simplified to Equation 5.32:

$$\begin{aligned} SM &= \mu_{\overline{x_i^t}} (x_i^t) * \mu_{w_{ij}} (x_i^t) \\ &= 1 * \mu_{w_{ij}} (x_i^t) \\ &= \mu_{w_{ij}} (x_i^t) \end{aligned} \quad (5.32)$$

Step 3: Rule Fulfillment

In this step, the *Overall Similarity Measure* (OSM) between fuzzified inputs and rule antecedents are calculated. For fuzzy rule with composite “AND”, the OSM can be taken as the average of similarity measure of its antecedents [Turksen90], as described by Equation 5.33.

$$SM_{rule\ k} = \frac{1}{I} \sum_i^I SM_{ik} \quad (5.33)$$

where $SM_{rule\ k}$ = OSM of k^{th} rule, SM_{ik} = SM of i^{th} antecedent of rule k , and I = Input dimension.

Alternatively, the OSM can be computed by taking the minimum SM of a rule antecedent [Zhou99], as defined in Equation 5.34.

$$SM_{rule\ k} = \min_{i=1,2,\dots,I} \{SM_{ik}\} \quad (5.34)$$

The decision of firing a rule is made based on the computed OSM and the threshold τ_0 .

This check is performed on every rule. If a rule has OSM smaller than τ_0 , then this rule is considered as irrelevant and will not be fired (Equation 5.35).

$$z_{rule\ k} = \begin{cases} SM_{rule\ k}, & \text{if } SM_{rule\ k} \geq \tau_0 \\ 0, & \text{otherwise} \end{cases} \quad (5.35)$$

Step 4: Consequent Derivation

The consequents of firing rules are derived using *Modification Function* (MF) based on the rules OSM and consequents. Expansion MF (mf_+) and reduction MF (mf_-) are adopted in FALCON-AART-AARS. mf_+ captures additional uncertainties due to dissimilarity between inputs and rules, whereas mf_- filtered off unnecessary information when there is high degree of matching between inputs and rules. One way to define the expansion and reduction MF are described in Equations 5.36 and 5.37.

$$mf_+: \mu_{w_{(lm)_k}}(y_m) = \min \left[1, \frac{\mu_{w_{(lm)_k}}(y_m)}{z_{rule\ k}} \right] \quad (5.36)$$

$$mf_-: \mu_{w_{(lm)_k}}(y_m) = \mu_{w_{(lm)_k}}(y_m) \times z_{rule\ k} \quad (5.37)$$

Either forms of MF function can be used. Generally, expansion MF is preferred as high matching degree is rare. Both forms of MF function can be applied coincidentally

by selecting the appropriate form of MF based on the SM. If SM exceeds the threshold, then applies mf_- , else, applies mf_+ . However, this method posts a problem as there may be more than one rule linked to the same consequent. There could be one rule using expansion whereas the other using reduction forms. In this situation, take the expansion form as it captures more information.

There may have more than one rule connected to a consequent. Hence, the final inferred consequent derived is the aggregate of different consequents inferred by different rules. The final inferred consequent is given in Equation 5.38:

$$\begin{aligned}\overline{w_{lm}} &= f[\overline{w_{(lm)_k}}] \text{ , where } f \text{ can be } \cap, \cup, \Sigma \\ &= \bigcup_{k \in \{1, 2, \dots, K\}} \overline{w_{(lm)_k}} \text{ , choose } f \text{ to be } \cup \\ &= \max_{k \in \{1, 2, \dots, K\}} \left\{ \mu_{\overline{w_{(lm)_k}}} (y_m) \right\}\end{aligned}\quad (5.38)$$

If expansion form of MF is used, Equation 5.38 can be expanded to Equation 5.39.

$$\begin{aligned}\overline{w_{lm}} &= \max_{k \in \{1, 2, \dots, K\}} \left\{ \min \left[1, \frac{\mu_{w_{(lm)_k}} (y_m)}{z_{\text{rule } k}} \right] \right\} \\ &= \min \left\{ 1, \max_{k \in \{1, 2, \dots, K\}} \left[\frac{\mu_{w_{(lm)_k}} (y_m)}{z_{\text{rule } k}} \right] \right\} \\ &= \min \left\{ 1, \mu_{w_{lm}} (y_m) \times \max_{k \in \{1, 2, \dots, K\}} \left[\frac{1}{z_{\text{rule } k}} \right] \right\} \\ &= \min \left[1, \mu_{w_{lm}} (y_m) \times \min_{k \in \{1, 2, \dots, K\}} \{ z_{\text{rule } k} \} \right] \quad \mu_{w_{lm}} (y_m) \leq 1, z_{\text{rule } k} \leq 1 \\ &= \mu_{w_{lm}} (y_m) \times \min_{k \in \{1, 2, \dots, K\}} \{ z_{\text{rule } k} \}\end{aligned}\quad (5.39)$$

If reduction form of MF is used, then Equation 5.38 becomes Equation 5.40

$$\begin{aligned}\overline{w_{lm}} &= \max_{k \in \{1, 2, \dots, K\}} \left\{ \mu_{w_{(lm)_k}} (y_m) \times z_{\text{rule } k} \right\} \\ &= \max_{k \in \{1, 2, \dots, K\}} \{ z_{\text{rule } k} \} \times \mu_{w_{lm}} (y_m)\end{aligned}\quad (5.40)$$

For AARS scheme, there are a few combinations that can be applied on the consequent derivation. One can use reduction, or expansion form of MF, or the average of the two. Alternatively, one can apply a threshold on the consequent derivation. If this

particular threshold is exceeded, then use reduction form, else, use expansion form. This threshold can be set to the vigilance parameter ρ or complement of it ($1 - \rho$). The later is more meaningful as the coarser the clusters are, the likelihood of rule firing is higher.

Step 5: Output Defuzzification

The final consequent derived from the output fuzzy sets are defuzzified to obtain crisp output. Defuzzification used by FALCON-AART-AARS is the *Center of Area* (COA) method.

Human-like Reasoning & CDDM procedures

The mapping of FALCON-AART onto AARS equips the system with human-like reasoning process, which conforms to the CDDM procedures. This is because AARS is a hypothetico-deductive reasoning that is adopted by the physicians [Evans05]. Accordingly, FALCON-AART-AARS has a one-to-one mapping to the physician reasoning process, as shown in Table 5.2.

Table 5.2: Reasoning process of FALCON-AART-AARS, Physicians, and CDDM.

Steps	Physician	FALCON-AART-AARS	CDDM
1	Observe the symptoms and ask the patient.	Take in the inputs from the data.	Patient examining.
2	Assess similarity of present input with past knowledge and experience.	Perform the antecedent matching process, which compares existing knowledge with current input.	Medical data aggregating.
3	Select the best experience or knowledge that describes current situation.	Select the best rule that describes the current input.	Experience-based reasoning.
4	Decide the appropriate response based on the selected knowledge.	Choose the appropriate output representation based on the selected rule.	Medical data aggregating.
5	Act upon the decisions.	Output the decision suggested by the output representation.	Diagnosis.

Explanation capability and user familiar terms

Since FALCON-AART is a FNN, it has the capability to generate rules autonomously. As each step of the inference process is clearly defined, and the positive and negative rules are segregated in FALCON-AART, FALCON-AART can be traced relatively easier. Thus, how and why a particular output is given can be explained.

On the other hand, the rules generated by FALCON-AART are fuzzy rules. Fuzzy rule allows the usage of linguistic terms, which encapsulates unnecessary details from the user. This generation of positive and negative rules enhances the tractability of the system, and exhibits FALCON-AART as a promising tool for CDSS.

5.4 Interpretability

There are two aspects of interpretability: psychology and data. The psychological part of interpretability is satisfied by FALCON-AART due to the nature of FNN: the self-organizing of rulebase, the extractable rules, and the use of fuzzy sets. The challenge lies in the interpretability from data point of view.

Interpretability [Pedrycz96] is the major strength of FNN in realizing DSS. However, the provision of rules does not necessarily guarantee interpretability. Some criteria have been proposed by other authors to assess a system for interpretability [Penã-Reyes04][Casillas03] [Ishibushi01]: (1) complexity; (2) distinguishability; (3) normality; and (4) coverage or completeness. The following sections describe these criteria and the FALCON-AART strategies in dealing with these criteria.

5.4.1 Complexity

There are three types of complexity: (i) complexity of the rule, i.e. the number of antecedents allowable for each rule without deteriorating the efficiency of human in understanding them. It was proposed that human beings can only handle 7 ± 2 distinct

elements in short-term memory [Casillas03]. Thus, the features dimension should be kept low; (ii) the number of elements – the number of linguistic terms per feature. The number of linguistic terms associated with each input dimension is to be kept low to ease understanding and interpretation; (iii) complexity of rulebase or *compactness*. This refers to the size of rulebase. Since the rules can be extracted out separately, the rules function as records or database, and users do not need to store them in their short-term memory. As FALCON-AART generates the same number of linguistic terms as the number of rules, the number of elements is the same as the complexity of rulebase. From the perspective of interpretability, cutting down the number of antecedents has deeper impact in improving the readability of the rule than constraining the rulebase size. However, it is desirable that the rulebase is compact (i.e. have sufficient number of relevant rules) and consistent [Casillas03], because irrelevant rules may cause confusion and waste computational resource.

The strategy adopted to ensure complexity is to reduce the number of rule antecedents through data preprocessing that selects the most relevant features. As for the number of rules, the strategy employed is to remove redundant and conflicting rules, and to combine similar rules. Conflicting rules are rules that have very similar antecedents, but very different consequents. That is, $\forall i = 1, 2, \dots, I, k \neq k', SM(A_{ik}, A_{ik'}) > \rho_{close}$, but $SM(B_{(lm)_k}, B_{(lm')_{k'}}) < \rho_{close}$. $SM(\bullet) \in [0, 1]$ is a function that measures how close two fuzzy sets are (see Equation 5.41 [Wang97]), and $\rho_{close} \in [0, 1]$ is a threshold defining “close”.

$$SM(A_{ik}, A_{ik'}) = \frac{\sum_{i=1}^T [1 - |\mu_{A_{ik}}(x_i) - \mu_{A_{ik'}}(x_i)|]}{T} \quad (5.41)$$

Redundant rules are defined as rules where $\forall i = 1, 2, \dots, I, k \neq k', A_{ik} \subseteq A_{ik'}$. The rulebase is allowed to be full grown, and then the removal of the conflicting and redundant rules starts.

This, apart from reducing the number of rules, ensures the consistency of the rulebase. Combining similar rules is done using the SM function as well: if $\forall i=1,2,\dots,I, k \neq k', SM(A_{ik}, A_{ik'}) > \rho_{close}$, then combine the rules. Aside from combining similar rules, similar linguistic term for each feature is also combined if it satisfies the condition $SM(A_{ik}, A_{ik'}) > \rho_{close}$ so as to reduce the number of elements for each rule. Rule weights are not employed because they convey ambiguous information regarding the importance of rules and may decrease the interpretability of the system [Ishibushi01].

5.4.2 Distinguishability

This criterion ensures that every membership function has a semantic meaning. Membership functions should convey a clear linguistic term. Since the AART learning algorithm assumes that there are overlapping between classes, further work has to be done to satisfy the distinguishability criterion.

To ensure interpretability, the following strategy is adopted. If two membership functions are overlapping $u_{ik} < v_{ik'}$ or $u_{ik'} < v_{ik}$ (see Figure 5.4), membership functions are adjusted in a way given by Equation 5.42 (for case $u_{ik} < v_{ik'}$) such that the criterion is satisfied, and clear semantic meaning can be associated to each linguistic term.

$$\begin{aligned} u_{ik}^{t+1} &= u_{ik}^t + \frac{|v_{ik'}^t - u_{ik}^t|}{2} \\ v_{ik'}^{t+1} &= v_{ik'}^t - \frac{|v_{ik'}^t - u_{ik}^t|}{2} \end{aligned} \quad (5.42)$$

Figure 5.4: The different relations between two fuzzy sets (dash line is $(u_{ik'}, v_{ik'})$, solid line is (u_{ik}, v_{ik})): (a) non-overlapping, $v_{ik} < u_{ik'}$; (b) overlapping, $u_{ik'} < v_{ik}$; (c) overlapping, $u_{ik} < v_{ik'}$; (d) fully overlap, subset, $u_{ik'} > u_{ik}$ and $v_{ik'} < v_{ik}$. The rule in (d) case will be combined by the complexity reduction process.

5.4.3 Normality

This criterion is to ensure at least one datum in the universe of discourse matches perfectly with every fuzzy set constructed by the system. The strategy employed by FALCON-AART to meet the normality criterion is by using the ART learning algorithm, which constructs the membership function based on data. The data used to construct the fuzzy sets matches with the constructed fuzzy sets.

5.4.4 Coverage

This criterion is to ensure that each possible input has at least one rule describing it. Since membership degree of any data outside the space covered by the rules is approximated by the distance from the rules, FALCON-AART satisfies this criterion.

5.5 Experiments

The dataset is divided into three mutually exclusive, stratified cross-validation sets, with training set made up of 33% of whole data, and testing set made up of the remaining 67%. The performances presented are averaged results over the three training/testing cross-validation sets. The measures used to assess the performance are: (1) Recall – the classification accuracy on the training set; (2) Predict – the classification accuracy on the testing set; (3) Sensitivity and Specificity, as described in Equations 5.43 and 5.44.

$$\text{Sensitivity} = \frac{\text{Positive samples correctly classified}}{\text{Total number of positive samples}} \quad (5.43)$$

$$\text{Specificity} = \frac{\text{Negative samples correctly classified}}{\text{Total number of negative samples}} \quad (5.44)$$

The positive samples are those the classifier seeks to classify out. In this case, it is the cancer (for diagnosis) or recurrent (prognosis) patient. Sections 5.5.1, 5.5.2 and 5.5.3 describe the case studies of using CLFNN as CDSS.

5.5.1 Breast Cancer Diagnosis Decision Support

Breast cancer is the second most deadly cancer among women. In 2007, 240,510 new cases of breast cancer and 40,910 deaths are anticipated in United States alone [ACS07]. This overwhelming figure indicates the importance of early detection for breast cancer. Breast cancer diagnosis is traditionally carried out using surgical biopsy. However, this method is costly, time-consuming, painful and highly invasive [Mangasarian95]. *Fine Needle Aspiration* (FNA) is the best alternatives to surgical biopsy. It is cost effective, non-traumatic and non-invasive. Apart from surgical biopsy and FNA, mammogram, thermogram, and gene profiling, are recent advancements in breast cancer diagnosis. The reported sensitivity (correct diagnosis of cancer when the disease is present) for each of the breast cancer diagnosis techniques are presented in Table 5.3. As illustrated in Table 5.3 FNA is one of the best alternatives for breast cancer diagnosis. Unfortunately, FNA sensitivity varies widely [Mangasarian95].

Table 5.3: Breast cancer diagnosis method.

Technique	Sensitivity	Reference
Surgical Biopsy	$\approx 100\%$	[Mangasarian95]
Clinical Examination	$< 65\%$	[Sickles84]
FNA	65%-98%	[Giard92]
Mammogram	68%-79%	[Fletcher93]
Thermogram	90%	[Amalu03]
Gene Profiling	90%	[van't Veer02]

The *Wisconsin Diagnostic Breast Cancer* (WDBC) dataset [Blake98] from the University of Wisconsin Hospital is a benchmarking dataset for diagnosing breast

masses based on FNA test [Mangasarian90]. The diagnosis was based on nine visually assessed characteristic of FNA sample. After removing 16 samples with missing values, this experiment is conducted using the dataset of 683 samples (444 benign and 239 malignant).

The WDBC dataset has been used in many studies. A brief summary of some studies is given in Table 5.4. Although the reported accuracies are impressive, some of these methods possess a few limitations. For example, linear programming method such as *Multi-Surfaced Method* (MSM), *Support Vector Machine* (SVM) [Kwok00] and Kernel-based clustering [Camastra05] do not provide interpretation for their computations. Boolean rules, NeuroRule [Setiono00], and feedforward neural network require a time-consuming and error prone process of manual rules extraction. *Hybrid Fuzzy-Genetic* (HFG) (i.e. a positive learning algorithm), *Incremental Learning Fuzzy Neural Network* (IFLN)-fuzzy rules and *Fuzzy CoCo* algorithms adopt genetic algorithm in rule construction and extraction. When stochastic algorithm (e.g. genetic algorithm) is used for rule construction or extraction, cares have to be taken as the stochastic might generate: (1) conflicting rules, which may result in unstable performance; (2) “abstract rules” that is not data-driven and cover unpopulated input space. These rules may arise due to the crossover operation of the stochastic algorithm (see Figure 5.5). Moreover, as stochastic algorithms require solution search in the large input space, stochastic algorithms tend to be computational costly.

Table 5.4: Reported accuracy on WDBC dataset.

Method	Accuracy (%)	Remark on Training Set	Number of Rules	Reference
Multi-Surface Method (MSM)	93.5	Use 50% data on the reduced set (369 samples available at the time) for training.	No rules	[Wolberg90]
1-Nearest Neighbour (1-NN)	93.7	Use 50% data on the reduced set (369 samples available at the time) for training.	No rules	[Zhang92]
Boolean rules	98.21	Not Available.	3	[Setiono96]
Feed-forward Neural Network	97.19	Not Available.	5	[Taha97]
Hybrid Fuzzy-Genetic (HFG)	98.8	Not Available.	4	[Penã-Reyes99]
NeuroRule	97.7	Use 50% data for training (Rules are manually extracted).	No rules or 6	[Setiono00]
SVM	93.6	Use 33. 4% data for training.	No rules	[Kwok00]
Global Optimization-based method	97.5	Use 80% data for training.	No rules	[Bagirov01]
Incremental Learning Fuzzy Neural Network (IFLN) and Fuzzy Rules	98.36	Use 100% data for training.		[Meesad03]
	98.5	Use 50% data for training.	4	
Fuzzy Genetic Approach	97.97	Use 100% data for training.	5	[Penã-Reyes04]
	97.76	Use 75% data for training.	5	
	97.23	Use 50% data for training	5	
Fuzzy CoCo	98.25	Use 100% data for training.	7	[Penã-Reyes04]
VISIT	97.5	Use 60% data for training.	3	[Chang04]
IRSS	96.89	Use 50% data for training.	5	[Chatterjee04]
Kernel-based Clustering	97	Use 100% data for training.	No rules	[Camastra05]

Figure 5.5: Before crossover, the rules cover the sample space nicely. During crossover, Rule 1 exchanges the membership function in x_2 with Rule 2, forming new rules: Rule 1' and Rule 2'. After crossover, the new rules cover the space that has no sample, and therefore, the rules are “abstract” and irrelevant.

Three stratified cross-validation training and testing sets are created. Training sets constitutes 33.24% of the entire dataset (benign = 148, malignant = 79), whereas the testing sets unseen to the system consist of the remaining 67.76% (benign = 296, malignant = 160). FALCON-AART, *Genetic Complementary Learning* (GCL), and *Hierarchical Complementary Learning* (HCL) are benchmarked against some popular systems such as *Binary Logistic Regression* (BLR), *Prototype Neural Network* (PNN), *k-Nearest Neighbour* (kNN), *Linear Discriminant Analysis* (LDA), *Multilayer Perceptron* (MLP), *Naïve Bayesian*, ID3, C4.5 [Witten05]. FALCON-AART, GCL, and HCL are also compared with FALCON-ART [Lin96] and FALCON-MART [Quek00]. The results averaged over three training/testing sets are summarized in Table 5.5. Since HCL analyses the data feature by feature, and contrasts the difference between the classes according to the information conveyed in individual feature, HCL is not able to classify categorical dataset. For that reason, HCL is taken out.

Table 5.5: WDBC classification result.

Method	Recall (%)	Sensitivity (%)	Specificity (%)	Predict (%)	Number of Rules	Training Time (s)
BLR	97.93	93.66	98.77	97.35	Not Applicable	0.66
kNN	98.8	93.79	98.31	97.71		8.06
LDA	97.05	92.66	98.5	97.35		0.047
MLP	98.51	95.86	98.17	97.35		1.86
Naïve Bayes	98.66	98.89	98.09	98.37		0.063
PNN	95.44	99.79	89.8	93.35		0.079
ID3	92.68	90.93	91.39	91.23	3	0.047
C4.5	98.95	95.4	96.3	95.98	11	0.09
FALCON-ART	62.99	45.31	72.13	62.37	31	33.75
FALCON-MART	97	92.86	96.86	97.12	97	62.79
FALCON-AART	98.95	98.12	98.18	98.51	11	5.79
GCL	95.5	83.65	99.66	95.29	21	3.02

From Table 5.5, FALCON-AART outperforms the others, suggesting superior generalization capability. This is highly desirable as using lesser training data, not only the number of rules formulated can be reduced, but also the training time can be significantly reduced. This is especially important in cases of limited samples, which

is prevalent in medical domain. Thus, it is important for CDSS to have good generalization ability. In this task, GCL does not perform as good. This may due to the data being categorical, where the differences between positive and negative are diminished. FALCON-AART and GCL achieve 100% and 99% on testing set when trained with 100% data and 50% data respectively. Thus, CL-based CDSS offers attractive solution to the breast cancer diagnosis using FNA as compared to the methods in Table 5.4.

Despite the good classification rate, statistical method such as Naïve Bayesian provides no interpretation. On the contrary, CLFNN autonomously generates fuzzy rules for reasoning. The physician can refer to its reasoning process to counter-check his/her decision, and these rules justify its computation. Comparing to its ancestors FALCON-MART and FALCON-ART, FALCON-AART improves significantly in rule generation and training time. The rulebases of FALCON-AART and GCL are compact, achieving better accuracy while keeping the rulebase small. This improves the interpretability of the system. However, the complexity of CLFNN rules is higher than seven elements, which according to psychological studies, could not be handled efficiently by human beings [Valente de Oliveira99]. Hence, in order to improve the interpretability of FALCON-AART and GCL, the number of antecedents should be decreased. This can be done through feature selection prior to the training, albeit this feature selection sometimes comes at the cost of accuracy. For illustration, feature selection is performed on the same training set using *Monte Carlo Evaluative Selection* (MCES) [Quah07], which selected the five most relevant features (only FALCON-AART is used for illustration purpose). FALCON-AART attained a Recall accuracy of 93.39% and a Predict accuracy of 97.06%. It generated 10 rules and the training process took 0.656 seconds. This slight drop in accuracy is because only five

features are used, but the complexity of the rules generated is lower and of greater interpretability. The rules generated by FALCON-AART, positive learning algorithm, and decision tree, are shown in Table 5.6.

Table 5.6: Rules generated. Complementary learning rules are fuzzy rules, and consist of positive and negative rules. C4.5 rules are crisp rules.

	Complementary Learning (FALCON-AART)	Positive Learning (HFG)	C4.5
Rule 1	IF uniformity of cell shape is high AND bare nuclei is rather high AND clump thickness is medium AND bland chromatin is very high AND single epithelial cell size is medium THEN Malignant.	IF uniformity of cell size is low AND single epithelial cell size is high AND bland chromatin is low AND mitosis is low THEN benign.	IF uniformity of cell size > 2 AND uniformity of cell shape > 2 AND uniformity of cell size ≤ 4 AND bare nuclei ≤ 2 AND marginal adhesion ≤ 2 THEN Benign.
Rule 2	IF uniformity of cell shape is low AND bare nuclei is rather low AND clump thickness is very low AND bland chromatin is low AND single epithelial cell size is medium THEN Benign.	Else Malignant.	IF uniformity of cell size ≤ 2 AND bare nuclei > 3 AND clump thickness > 3 AND bland chromatin ≤ 2 AND marginal adhesion ≤ 3 THEN Malignant.

As illustrated, the CL-based system generates rules that describe the input space more closely. On the contrary, the positive learning system even though has a smaller rulebase, it assumes that the uncovered space is negative. Therefore, in this aspect, CL-based system gives a better modeling of the input space. It is likely to perform well in multi-class problem because it generates a distinct set of positive and negative rules for each class. Comparing with rules of decision tree, FALCON-AART rules encapsulate unnecessary details from users. The FALCON-AART rules capture uncertainty in the data, a feature where decision tree rules do not offer. Ability to handle uncertainty is important because the medical domain is inherently uncertain. Furthermore, different physicians may have different sets of reference values. The rules generated by the decision tree may seem to be more intuitive than those generated by FALCON-AART. The reason is that, FALCON-AART uses all input dimensions to formulate its rule, whereas the decision tree uses only part of the input dimensions.

However, using only part of the input dimensions may create confusion because the same features may be used twice in the rule (as shown in Rule 2 of C4.5). Worse still, sometimes it may lead to contradicting conclusions. Therefore, rules generated by CL paradigm are a better approach to CDSS.

As shown, the complexity criterion can be done by carrying out feature selection. The normality and coverage are met because of data-centred learning. Figure 5.6 shows the accuracy of the five-input system using different number of rules. The minimum number of rules in complementary learning system is two. As shown, the system yields good performance (92% prediction accuracy) even when only two rules are used. The addition of other rules for inference does not significantly improve the performance, suggesting these rules are for special or extreme cases.

Figure 5.6: Accuracy under different number of rules.

Figure 5.7 illustrates the decision making of FALCON-AART based on these complementary rules. If the input is a negative sample, then the negative rules are activated and the positive rules are inhibited. Likewise a positive sample activates only the positive rules, and concurrently deactivates the negative rules. Figure 5.8 provides a numeric example of the reasoning process.

Figure 5.7: The deduction based on complementary rules during decision-making process.

Figure 5.8: Numerical example of FALCON-AART inference process.

From Figures 5.7 and 5.8, when a positive sample is presented, it is propagated to Layer 2. Subsequently, the sample is compared with the rule and the overall firing strength of each rule is computed. Since positive and negative rules are created separately, negative rule has a lower firing strength than the positive rule, which resulted the inhibition of negative rule. This minimizes confusion in the inference process. Accordingly, the consequent linked by the maximally fired rule is activated, and defuzzification is performed to derive output. Since the output is close to one, conclusion is positive class.

5.5.2 Breast Cancer Prognosis Decision Support

Prognosis is a medical opinion of the likely course and outcome of a disease. Prognosis is important for predicting the risk of recurrence for a particular treatment, or for a particular type of tumour. Knowing the risk of recurrence would assist the physician in deciding which treatment to apply. This is critical because the side effects of treatments are traumatic, dangerous and unpleasant. Furthermore, the cost of these treatments is generally high. Hence, the estimation of the recurrence risk is paramount prior to the choice of the treatment.

The *Wisconsin Prognostic Breast Cancer* (WPBC) dataset is obtained from UCI machine learning repository [Blake98]. The dataset contains 151 non-recurrent patterns and 47 recurrent patterns. Each pattern has 32 features. Training set comprises 1/3 (Non-recurrent = 50, Recurrent = 18), and testing set comprises the remaining 2/3 (Non-recurrent = 101, Recurrent = 29). Summary of a few studies on this dataset is given in Table 5.7.

Table 5.7: Reported accuracy on WPBC dataset.

Method	Accuracy (%)	Remark on Training Set	Number of Rules	Reference
Recurrence Surface Approximation (RSA)	81.7	Reduced set, leave-one-out test result.	No rules	[Mangasarian95]
Pooled RSA	87.1	Reduced set, leave-one-out test result.	No rules	[Mangasarian95]
Least-1-Norm Error	78.2	Reduced set, leave-one-out test result.	No rules	[Mangasarian95]
Global optimization-based method	87.8	Leave-one-out test result.	No rules	[Bagirov01]
Genetic algorithms with hyper-cone membership functions.	77.29	Use 50% data for training.	2	[Hng03]
FALCON-AART	77		36	[Tan05b]
GCL	92		17	[Tan06e]
HCL	79		33	[Tan07a]

Although linear programming methods such as RSA are capable of good prognosis, they are incomprehensible. They do not provide explanation for their computations. On the contrary, CLFNN not only put forward fuzzy rules to explain its reasoning process, but also good generalization capability. The leave-one-out accuracies of FALCON-AART, GCL, and HCL are 100%, outperforming the leave-one-out result of other methods. This suggests that CLFNN is more suitable as breast cancer prognosis CDSS.

FALCON-AART, GCL, and HCL of CLFNN are benchmarked against FALCON-ART [Lin96], FALCON-MART [Quek00], MLP [Witten05], Assistant 86 [Cestnik87], Bayesian classifier [Clark89], AQR [Clark89], CN2 [Clark89], fuzzy expert system [Meesad03], AQ15 [Michalski86], CLILP2 [Liu92], and PRISM [Cendrowska87]. Since PRISM algorithm can only handle nominal values, the data is discretized using simple binning prior to the learning. The results, averaged over three different stratified cross-validation sets, are shown in Table 5.8.

Table 5.8: Breast cancer prognosis result.

Architecture	Proportion of Training Data (%)	Classification Rate (%)
Assistant86	100.0	72.0
Bayes	100.0	66.0
MLP	70.0	77.27
AQR	100.0	72.0
CN2	100.0	71.0
AQ15	100.0	68.0
Fuzzy Expert System	70.0	75.83
CLILP2	100.0	77.0
PRISM	30.0	58.99
FALCON-ART	30.0	22.31
FALCON-MART	30.0	66.38
FALCON-AART	30.0	77.61
	50.0	77.0
	100.0	91.92
GCL	30.0	97.85
	50.0	92.78
	100.0	92.8
HCL	30.0	76.78
	50.0	79.38
	100.0	100.0

Again, Table 5.8 reveals that CLFNN performed well for prognosis. CLFNN uses a smaller set of training data, yet CLFNN achieves comparatively good performance. This implies that CLFNN has greater generalization ability than other techniques, and CL paradigm is indeed able to enhance the performance of the classifier. One can observe that even using all the data for training, all the classifier presented cannot recall with 100% accuracy. This suggests that the dataset may contain contradicting cases. This does not necessarily mean the contradicting cases are noise or outliers; rather they may be special cases. Owing to this reason, the accuracy using 50% data for training may give a poorer accuracy than using 30% data for training.

In order to demonstrate that CL offers superior classification performance than positive and neutral learning, FALCON-AART is compared against PRISM [Cendrowska87], POPFNN [Zhou96] and MLP [Witten05] (since GCL and HCL have similar performance as FALCON-AART, only FALCON-AART is used for

illustration). PRISM and POPFNN construct knowledge on the basis of positive samples only. In contrast, neutral learning algorithm such as MLP learns only from mistakes, and stores the knowledge without considering whether it is positive or negative. Due to the structure of POPFNN (each rule has different sets of input linguistic terms), POPFNN could not classify dataset with more than 13 input dimensions using the lab system. To enable comparisons, feature selection is done on the prognosis training set using MCES [Quah07]. The prediction accuracy versus number of features plot is shown in Figure 5.9. The same feature set is used for all the methods.

It can be observed from Figure 5.9, CL approach outperforms positive and neutral learning, except for a few instances where MLP performs better. This is due to the data imbalance. In those cases (number of features = 23 – 25, area enclosed by the dashed line rectangle in Figure 5.9), the sensitivity of MLP is lower than FALCON-AART, but MLP have higher specificity than FALCON-AART. Since the number of negative samples is a lot greater, MLP has higher accuracy. From Figure 5.9, as number of features exceeds 17, the performance of the system fluctuates in smaller amplitude, suggesting that these features are irrelevant to the prognosis. These features are not informative to the particular algorithm in this prognosis task..

To assess classifier performance, *Receiver Operating Characteristic* (ROC) plot is often used. It is a plot of sensitivity against 1-specificity under different classifier thresholds. Since PRISM do not have a classifier threshold, only ROC plot of POPFNN, MLP and FALCON-AART is given in Figure 5.10. The larger is the *Area Under the Curve* (AUC), the better is the classifier performance. From Figure 5.10, FALCON-AART has the best performance. All the classifiers are close to the

45-degree line (symbolizing 50-50 random guessing), which shows that the prognosis task is difficult.

Figure 5.9: Number of features versus accuracy plot.

Figure 5.10: ROC plot.

Moreover, the results show that CL (FALCON-AART) outperforms positive (POPFNN, PRISM) and neutral learnings (MLP). Apart from greater accuracy, CL has improved interpretability. The strategy employed by CL ensures the interpretability of the system, meeting the criterion such as distinguishability and compactness. Figures 5.11 and 5.12 display the original generated fuzzy sets and the

tuned fuzzy sets, with $\rho_{close} = 0.9$. As shown, the strategy improves the readability and complexity of the rules.

Figure 5.11: The original fuzzy sets generated by FALCON-AART.

Figure 5.12: The fuzzy sets tuned after learning. The distinguishability of the fuzzy sets is improved, and thus allowing a clear semantic meaning to be assigned for each linguistic terms. As shown, there is no redundant term in the tuned fuzzy sets.

5.5.3 Diagnosis with Proteomics

The debut of proteomics has given hope for the understanding and study of disease.

Proteome profiling, which is the simultaneous measurement of a large number of

expressed proteins, has facilitated the hypotheses generation, medical diagnosis, and biomarker identification [White04]. This protein signature contributes to new discovery and understanding of diseases. In addition, it allows therapeutic response prediction, survival forecast, novel subtype detection, as well as drug target identification [Montironi03]. The followings are some applications that show how CLFNN can complement proteomics in medicine. The FALCON-AART [Tan05b] is used for illustrating the CLFNN features. The datasets used are: (i) the proteomic pattern of normal and ovarian cancer patients obtained from serum samples using the *Ciphergen Protein Biology System 1* SELDI-TOF (*Surface-Enhanced Laser Desorption/ Ionization -Time of Flight*) mass spectrometer [Petricoin III02], consisting of 100 normal, 16 benign, and 100 ovarian cancer samples (in this work, 16 benign samples are grouped with normal samples); (ii) The proteomics pattern obtained using *Ciphergen Protein Biology System 2* SELDI-TOF mass spectrometer, consisting of 91 normal and 162 ovarian cancer samples [NCI01].

5.5.3.1 CLFNN as cancer diagnosis and staging adjunct

The manual inspection of such proteomics pattern is time-consuming and difficult. Furthermore, the detection of cancer involves many variables such as symptoms' duration, tumor biology, etc [Röcken04]. Hence, computational aid is brought into picture. The confluence of proteomics and computational intelligence is not new, and has been applied in wide variety of diseases and cancers. However, the technologies commonly adopted such as *Multilayer Perceptron* (MLP), *Logistic Regression* (LR), or *Support Vector Machine* (SVM), are not interpretable and lack of transparency. Consequently, physicians are not able to validate the system output. Hence, CLFNN is better suited as adjunct of cancer diagnosis as it offers good pattern recognition performance, and at the same time, provides interpretable and tractable operation.

The two datasets are divided into three randomized, stratified cross-validation training/testing sets, with training set made up of 33% of the data, and testing set made up of the remaining unseen 67%. Since the number of features is ultra-huge (15154), *Augmented Variance Ratio* (AVR) [Zhang02] is used to extract the top 20 features. Subsequently, *Support Vector Machine* wrapper feature ranking method [Liu02] is applied on the top 20 most relevant features. The averaged performances of the CLFNN under different number of features are shown in Figure 5.13. The performance is benchmarked against common bioinformatics tools such as SVM, MLP, and LR, C4.5 [Witten05]. The overall performance of each algorithm is summarized in Table 5.9. The overall accuracy is taken as the weighted average of the two datasets, so as not to introduce bias in estimating the overall accuracy. This is defined in Equation 5.45.

$$\text{Overall Accuracy} = \frac{\text{accuracy}_1 \times n_1 + \text{accuracy}_2 \times n_2}{n_1 + n_2} \quad (5.45)$$

where accuracy_j is accuracy on the j^{th} dataset, and n_j is number of data samples in the j^{th} dataset.

Figure 5.13: Accuracy versus number of features.

Table 5.9: Overall performance in ovarian cancer diagnosis.

Methods	Recall (%)	Sensitivity (%)	Specificity (%)	Predict (%)	Training Time (s)
SVM	91.83	81.22	92.35	87.66	1.05
MLP	95.56	86.41	89.96	88.63	0.25
LR	95.56	88.79	83.4	86.63	0.07
C4.5	93.71	87.0	80.81	83.57	0.02
FALCON-AART	91.41	93.48	85.42	88.93	0.48

As shown, the CLFNN performance is as competent as the conventional methods in pattern recognition ($p\text{-value} < 0.001$, $\alpha = 0.05$). It can be seen that every method has its optimal number of features to classify the dataset. The performance of each system is not improved despite more features are used. This signaled that some features may be noisy and these features render the diagnosis difficult. A significant strength of CLFNN is its provision of intuitive, complementary fuzzy rules to enable user's validation and to facilitate the knowledge extraction process. Examples of rules are shown in Table 5.10, and are compared against rules generated by decision tree (crisp rule). Fuzzy rule is capable of handling uncertainty that may present in the data, a feature lacking in crisp rule. As proteomics data is known to be highly variable, uncertain, and noisy, fuzzy rule is more suitable for supporting proteomics. Moreover, the fuzzy rules allow the use of linguistic terms to encapsulate unnecessary details from the user, which makes the system easier to be interpreted. Besides, in contrast to crisp rule, which sometimes may have repeated antecedents, CLFNN rule applies each antecedent once, and hence, avoids the possible confusion.

Table 5.10: Complementary fuzzy rules and crisp rules.

	Fuzzy rules (CLFNN)	Crisp rules (C4.5)
Positive Rule	IF peak1593 is low to very low, peak1736 is quite low, peak 2237 is medium, THEN Cancer.	IF peak1593 ≤ 0.04 , peak1736 > 0.12 , peak2237 ≤ -0.34 , peak1736 > 0.01 , THEN Cancer.
Negative Rule	IF peak1593 is rather low, peak1736 is quite high, peak 2237 is very low, THEN Normal.	IF peak1593 ≤ 0.04 , peak1736 ≤ 0.12 , peak2237 ≤ -0.76 , THEN Normal.

Apart from that, CLFNN has a human-like reasoning process, which closely resembles how human arrives at his/her conclusion. This ensures transparency of the system and allows user to analyze the system using his/her familiar terms. To illustrate this, an example of CLFNN reasoning is given in Figure 5.14. As shown, CLFNN has a one-to-one mapping of human reasoning process, suggesting that CLFNN deduction process is highly akin to that of human.

Figure 5.14: CLFNN reasoning process.

5.5.3.2 CLFNN as biomarker assay tool

A biomarker is a biologically derived molecule in the body that is indicative of the progress or status of a disease [Shin06]. Proteomics is a promising tool for biomarker

discovery because a biomarker should be measurable in easily accessible body fluid such as serum, urine or saliva. Since these fluids contain protein-rich information, proteomics provides an attractive technology for biomarker discovery [Petricoin04]. Protein profiling can aid the biomarker discovery through identifying the proteins under-expressed or over-expressed in normal and cancer protein expressions — these are potential biomarkers [King03].

There are two main approaches to identify biomarkers in proteomics. The first one is from molecular biology, which focuses on identifying and characterizing a specific biomarker at the molecular level, and then investigates its relationship between structure and its role in cancer development. However, this approach is slow, and lack of sensitivity [Li04]. The other approach is bioinformatics: coupling computational intelligence and proteomics (especially mass spectra) to extract biomarkers that are sensitive and specific to disease with high reproducibility [Li04]. This approach allows the detection of biomarkers collectively, which could promise higher sensitivities and specificities [Xiao05]. The following shows that CLFNN can be used as a bioinformatics tool to validate the biomarker identified. From the feature ranking process, the peak of the 1593th key mass/charge ratio in the spectra is potentially the biomarker for ovarian cancer diagnosis. To assay this potential biomarker, CLFNN is trained based on (i) this feature alone (the performance is assessed against CLFNN performance based on other features); (ii) the second and third most relevant features identified with and without this feature. Figure 5.15 summarized the result.

Figure 5.15 shows that the peak1593 is not as good as peaks 1736 and 2237 combined in detecting ovarian cancer. This indicates that the peak1593 is not the biomarker for the ovarian cancer diagnosis, but peaks 1736 and 2237 combined. Further investigation is required to validate this.

Figure 5.15: CLFNN performance based on potential biomarkers.

5.5.3.3 CLFNN as concept validation tool

One possible application of CLFNN is as concept validation tool, i.e. tool to affirm or support certain hypothesis. For example, to confirm the variability of instrumentation does exist across labs [Rodland04], CLFNN is trained on two different datasets. The CLFNN learned on the training file from the first dataset, and subsequently tested on testing file of the second dataset. The performances under different number of features are compared against the performance of usual setting (same source for training and testing sets). Figure 5.16 summarized the results.

From Figure 5.16, the performance of CLFNN on same data source against the performance of CLFNN on different data source is very different. The paired t-test analysis shows that the two performances are significantly different (p -value < 0.0001 , $\alpha = 0.05$). This confirms the fact that, there is data variability across research centers and instruments.

Figure 5.16: CLFNN performance on training and testing sets of different sources.

5.6 Summary

FALCON-AART functionally models the complementary learning mechanism of human pattern recognition. Due to the use of positive and negative learnings, FALCON-AART exhibits promising performance in the application of medical diagnosis as described in this chapter. Unfortunately, FALCON-AART possesses certain shortcomings, such as high computational load (the complexity analysis of FALCON-AART can be found in [Tan06c]), difficulty in hardware realization, etc.

Chapter 6 Genetic Complementary Learning

Genetic complementary learning (GCL) is the integration of *Genetic Algorithm* (GA) and complementary learning. GA is an evolutionary algorithm that is inspired by the genetic operations in chromosomes. It mimics the natural selection and adaptation of species in nature. Such optimization or construction of fuzzy neural network by using GA is similar to those in [Kubota07][Kubota04][Furuhashi02][Furuhashi97][Kasabov97].

GCL functionally models the complementary genes selection processes in human beings. In the context of evolution, certain traits or *alleles* (a member of a pair or series of genes that is located at specific position on the chromosome) are bound to selection. This selection is dependent on several factors known as *selective pressure*. Moreover, deleterious and advantageous gene populations exist in human genome due to mutations. Hence, the selection process is required to maintain the right composition of the populations, i.e. to promote beneficial genes and to reduce deleterious genes [Fry05]. Therefore, there exists two complementary gene selection processes [Fay01]: (1) *positive/directional selection*, which increases the prevalence of desirable traits; (2) *negative/balancing selection*, which decreases the prevalence of traits that diminish the individual's fitness. By functionally model this concept, GCL favors the firing of positive rules (desired traits) when the positive samples (selective pressure) are presented to the system, and at the same time, the negative rules' (deleterious traits) chance of firing are diminished, which is analogous to positive and negative selections respectively.

Viewing from the other end, the GCL can be viewed as a functional model of the genes interaction called *epistasis*, which refers to the suppression of a gene effect by a nonallelic gene. Epistasis can alter the additive gene variances and covariances, and

hence the response to selection. The interaction between the environment, Allele frequencies, biological genes interaction, physiology, and traits, determines the epistasis process [Carter05]. Again, the epistasis process is complementary: positive epistasis results in genes reinforcing each other, whereas negative epistasis causes genes to diminish each other. Thus, whenever the positive input (environment) interact with GCL, the positive rules are fired (positive epistasis) and concurrently, the negative rules are inhibited (negative epistasis), leading to a positive response, and vice versa.

6.1 GCL Architecture

GCL autonomously generates fuzzy rule in the form described by Equation 6.1.

$$\text{IF } x_1 \text{ is } \mathbf{A}_1 \text{ and } x_2 \text{ is } \mathbf{A}_2, \text{ THEN } y_1 \text{ is } \mathbf{B}_1 \text{ and } y_2 \text{ is } \mathbf{B}_2 \quad (6.1)$$

The fuzzy rule in Equation 6.1 is an example of two-input and two-output system. It consists of five elements:

- Input linguistic variables (x_1, x_2) .
- Input linguistic terms $(\mathbf{A}_1, \mathbf{A}_2)$. This represents fuzzy entities such as tall, short, thin, fat, and so on. The Input linguistic terms are represented by trapezoidal membership functions in GCL.
- If-Then rule: links the antecedent part (i.e. input linguistic variables and terms such as above) with the consequent part (i.e. output linguistic variables and terms such as below).
- Output linguistic variables (y_1, y_2) .
- Output linguistic terms $(\mathbf{B}_1, \mathbf{B}_2)$.

GCL is a five-layer network in which each layer corresponds to an element of the fuzzy rule. Its architecture is shown in Figure 6.1. In contrast to the original GA,

GCL specifically and separately generates populations on positive and negative samples. GCL and standard GA differed in the number of parameters as well. GCL has only one parameter: the maximum number of generations. Other parameters are determined either by the network itself or obtained randomly. This frees the user from having to search for the optimal set of parameters, which is time-consuming.

Figure 6.1: Architecture of GCL fuzzy neural network.

6.2 GCL Learning

Given a dataset $D \in \{D^+, D^-\}$, where D^+ = positive samples, D^- = negative samples.

$D = \bigcup_{t=1}^T (\mathbf{x}, \mathbf{d})'$, and $|D| = T$. $\mathbf{x}' = (x_1, \dots, x_i, \dots, x_I)'$, and $\mathbf{d}' = (d_1, \dots, d_m, \dots, d_M)'$. I is

the number of input dimension, whereas M is the number of output dimension. The

GCL learning is summarized in the followings, as depicted in Figure 6.2:

Figure 6.2: Operations of genetic complementary learning. A rule is analogous to a chromosome in GCL. When a data is given, the rule can mutate its antecedent or consequent to fit the data. As for crossover, it is the antecedent exchanging between the best and the second best rules of that generation.

Step 1: Select initial populations.

This refers to generating initial populations of positive and negative rules. In GCL, each rule is analogous to a chromosome. This can be done in many ways. In this work, this selection is done randomly. That is, generate K^+ number of rules from D^+ and K^- number of rules from D^- . The K^+ and K^- are randomly selected. The initial rulebase consists of positive rulebase $R^+ = \bigcup_{k \in [1, K^+]} r_k$ and negative rulebase $R^- = \bigcup_{k \in [1, K^-]} r_k$.

The system is to evolve from these initial populations.

Step 2: Mutate the rules.

The mutation operation is defined as the adjustment of the membership functions that characterize the antecedents of the fuzzy rule. There are many types of membership functions, and the type of membership function influences the system performance. In GCL, trapezoidal membership function is adopted due to its computational simplicity. The mutation operation is probabilistic, not every generation will mutate, and not every input dimension will mutate. The mutation operator is given in Equation 6.2.

$$u_{ik}(t+1) = u_{ik}(t) + \beta_i(t)[u_{ik}(t) - x_i]$$

$$v_{ik}(t+1) = v_{ik}(t) + \beta_i(t)[x_i - v_{ik}(t)] \quad (6.2)$$

where (u_{ik}, v_{ik}) are the weights of the k^{th} rule in i^{th} dimension. $\beta_i(t)$ is an ever-changing learning constant, $\beta_i(t) \in [0,1]$. As $\beta_i(t)$ and the mutation probability changes randomly every time, care should be taken to ensure the weights are in proper relation, i.e. $u < v$. Whenever u becomes greater than v , the weights are swapped, that is, u becomes v , and the other way round. Note that the supports, i.e. (Min_{ik}, Max_{ik}) of the fuzzy trapezoidal membership function are preset to the minimum and maximum of the dimension respectively. They are not tuned in this phase of learning.

Step 3: Assess the rules.

This requires the definition of the fitness function. The fitness of a rule is given by the strength of the rule to accurately classify the training set. Rules that have fitness lower than a specific threshold are removed.

When classifying the training set, the positive and negative rules carry out the lateral inhibition. A positive training sample activates the positive rules R^+ and at the same time inhibits the negative rules R^- , and vice versa. In this lateral inhibiting way, whenever a positive sample is presented, the firing strength of positive rules is greater than that of negative rules, which leads to a positive decision. Hence, the lateral inhibition can minimize confusion during the inference, and make good performance possible. If the output is correct, the rule with the maximum firing strength is rewarded by increasing its fitness.

Step 4: Crossover best and second best rules.

This is similar to the crossover operation in the chromosome. For this operation, the best and second best rules are used as parents for the crossover. The crossover is done

by exchanging the antecedents of the rules, and the number of antecedents to be exchanged is randomly selected. Note that the operation is done separately for the positive and negative populations.

Step 5: Terminate learning.

Stop the training when it reaches the maximum number of generations provided by the user, or when it achieves the desirable accuracy. Otherwise, go to Step 2.

6.3 GCL Analysis

Given T = number of samples, I = number of dimensions, K = number of rules, and N = number of generations, the estimate of GCL computational complexity in terms of number of multiplications/divisions and additions/subtractions is given in Table 6.1. The cost of assignment and comparison is assumed negligible. It can be seen that complexity of GCL is $O(NIK)$, which shows that GCL is scalable in terms of the number of dimension and number of samples (in worst case, $K=T$, i.e. each sample is described by one rule). As for memory requirement, each rule contains four memory elements (trapezoidal membership function) for every dimension. Hence, the memory requirement of GCL is $4IK$.

Table 6.1: Complexity analysis of GCL.

Description	Number of additions/subtractions	Number of Multiplications/divisions	Remark
Generate initial population	0	0	Generate K rule from T samples from D . In general, $K < T$. In, worst case, $K = T$.
Mutate the population	$2NIK$	$2NIK$	In worst case, each dimension of all rules requires update, for both u and v .
Assess the rulebase	NIK	NK	Requires computation of the rule firing strength and comparison of rules.
Crossover	0	0	Only swapping the antecedents, no arithmetic.
Total	$3NIK$	$NK(2I+1)$	

6.4 GCL Interpretability

Interpretability of a system is the capability of the system to explain its operation in an understandable way [Casillas03]. Interpretability is one sought after feature in intelligent systems, especially those involved in decision support. This is because any reasonable human being will not adopt a suggestion without carefully assessing the feasibility and rationality of the suggestion given. However, providing fuzzy rules does not necessarily guarantee the interpretability of the system. This is especially true for stochastic algorithm such as GA. Hence, several strategies are proposed to ensure the interpretability of GCL system, that it satisfies some of the interpretability criteria [Penã-Reyes04] [Casillas03][Ishibushi01]:

1. *Distinguishability* – Each fuzzy set (linguistic term) should have clear semantic meaning, i.e. an unambiguous definition of a range in the universe of discourse. In other words, the overlapping between each fuzzy set should not be too high to an extent that allocation of linguistic term (semantic meaning) is difficult. In order to meet this criterion, the following strategy is proposed: after assessment of the rulebase (Step 3 of GCL algorithm), the rules are compared against one another. For the rules that are similar, i.e. their fuzzy sets are close to each other in input space, the tail of the fuzzy sets are adjusted according to the distance among them. This is to ascertain that the overlapping between the two fuzzy sets is kept to the minimum possible. The strategy is illustrated in Equations 6.3 and 6.4.

$$Min_{ik}(t+1) = \begin{cases} Min_{ik}(t) - \frac{d(A_{ik}, A_{ik^*})}{2} & \text{if } A_{ik} > A_{ik^*} \\ Min_{ik}(t) & \text{Otherwise} \end{cases} \quad (6.3)$$

$$Max_{ik}(t+1) = \begin{cases} Max_{ik}(t) + \frac{d(A_{ik}, A_{ik^*})}{2} & \text{if } A_{ik} < A_{ik^*} \\ Max_{ik}(t) & \text{Otherwise} \end{cases} \quad (6.4)$$

where A_{ik} is the fuzzy set (linguistic label) associated with the k^{th} rule in i^{th} dimension, k^* is the most similar rule to k and $k^* \neq k$. $(\text{Min}_{ik}, \text{Max}_{ik})$ denotes the left and right tails of the fuzzy set A_{ik} , and $d(\bullet)$ is the distance function defined in Equation 6.5.

$$d(A_{ik}, A_{ik^*}) = \begin{cases} |v_{ik} - u_{ik^*}| & \text{if } A_{ik} < A_{ik^*} \\ |v_{ik^*} - u_{ik}| & \text{Otherwise} \end{cases} \quad (6.5)$$

where (u_{ik}, v_{ik}) are the weights characterizing the fuzzy set A_{ik} .

2. *Complexity* – There are two facets to this criterion: one refers to the number of antecedents of the rules, and the other refers to the number of rules in the rulebase. It is often brought up that this number should not exceed 7 ± 2 distinct terms because human being cannot handle more than 7 items in the working memory. However, there is no evidence that all the rules must be held in working memory prior to the interpretation of the system. The rule can be viewed as an offline record that user can refer to when he/she needs to. Nevertheless, GCL detects and removes redundant rules, i.e. rules with the same or highly similar antecedents and consequents.
3. *Normality* – This refers to the requirement that there must be at least one datum in the universe of discourse to match perfectly with each fuzzy set. This is satisfied by the data-centered rule generation algorithm of GCL.
4. *Coverage* – For any input, at least one rule is fired. This is also satisfied by the GCL algorithm since the GCL rule generation is data-centered, and it uses the maximum and minimum values of the input space. Moreover, data not covered by the rulebase is decided upon based on the distance of the data with the positive and negative rulebase.

-
5. *Consistency* – This means that there are no rules with similar antecedents but different consequents. This is ascertained by a rulebase validation that detects and removes any contradicting or redundant rules. Detecting the inconsistency is relatively easier in GCL as it separates the positive and negative knowledge. For GCL, the rule removal policy is that, whenever contradicting rules are detected, the rules with a lower fitness value are removed. Moreover, to ensure the rules generated are not redundant, rules with very low fitness values are removed after each generation.

6.5 Experiments

6.5.1 Translation Initiation Sites Prediction

Translation Initiation Sites (TIS) is the key point of protein synthesis [Wong03]. It signifies the beginning of a gene coding region, and most of it contains a conserved triplet of nucleotides ATG (in DNA or cDNA) or AUG (in mRNA), known as start codon [Rajapakse05]. Translation starts at TIS in a cDNA/ mRNA sequence and ends at the first in frame stop codon downstream [Li05a]. Accurate identification of the TIS facilitates the understanding of protein coding, gene structures, and product from nucleotides sequences. Moreover, precise TIS recognition can improve the reliability and accuracy of amino acid predictions from nucleotide sequences [Zien00].

Unfortunately, identifying or predicting the TIS is non-trivial [Pedersen97], especially in eukaryotes. Translation in eukaryotes does not always start at the first codon ATG. It was reported that less than 10% of all eukaryotic mRNA utilize downstream ATG as start codon. In other words, 90% accuracy should be achieved by only selecting the first ATG encountered. However, this is possible only if one is given a complete and error free mRNA or cDNA sequence. The difficulty in TIS

identification surges, especially when using unannotated genome data, or when using *Expressed Sequence Tag* (EST). It is not known whether the EST fully represents the upstream or downstream of the asymmetric ends of a strand of DNA bases, and the sequence can potentially be contaminated [Pedersen97]. The fact that the sequences may be incomplete and erroneous further complicates the matter. On top of that, the biological mechanisms underlying translation process are yet to be fully understood, rendering the task at hand difficult [Zeng02].

Thus, computational biology methods emerge for TIS detection. In contrast to biology methods [Li05b], which has high error rate, computational biology methods are less costly and less time-consuming [Li05a]. Generally, there are two types of computational biology approaches [Li05b]: generative and discriminative. The former builds models for the target pattern and subsequently, see how well the candidate sequence fits the model. For example, *hidden Markov models* [Li05b]. The later learns to discriminate a sample, for instance, neural network. With advancement in these technologies, high accuracy has been claimed in TIS prediction, as given in Table 6.2.

Apart from the high accuracy, some findings in regards to TIS have been reported: (1) consensus motif GCCACCATGG is the eukaryotic TIS, albeit such consensus alone is not sufficient to identify the ATG initiator codon [Li05b]; (2) position -3 (the underlined nucleotide A to the left of codon ATG of consensus motif mentioned in (1)) is crucial for TIS recognition [Pedersen97]; (3) ATG in frame to TIS is less likely to be the TIS [Pedersen97]; (4) The hexanucleotide difference before and after the ATG in a window of 50 bases long is important for TIS identification [Salamov98]; (5) local correlations are more critical than long-range correlations in TIS prediction [Zien00]. Hence, computational biology method not only provides good accuracy, but also gives

6 Genetic Complementary Learning - T. Z. TAN

insights into the characteristics of TIS, despite the knowledge has to be derived manually.

Table 6.2: Accuracy of eukaryotic TIS prediction.

Methods	Accuracy (%)				Remark
	Sensitivity	Specificity	Overall	MC	
Positional conditional probabilities (Salzberg) [Zien00]	68.1	73.7	72.32	0.63	Six-fold CV
SVM + Salzberg kernel [Zien00]	78.4	76	77.62	0.7	Six-fold CV
Feedforward Neural Network [Pedersen97]	90	87	88	0.71	Six-fold CV
Ensemble (NB +SVM + Neural Networks + decision tree) [Zeng02]	76.7	81.2	89.7	NA	Three-fold CV
LDA + characteristic of flanking sequences around TIS + conditional probability matrix [Wong03]	88.96	89.09	89.02	NA	Six-fold CV
100 generated features + data mining tools [Liu03]: (i) NB (ii) SVM (iii) C4.5.	75.9	93.7	92.5	NA	Three-fold CV
SVM with Edit Kernel [Li05b]	95.5	98	97.5	0.91	Six-fold CV
Scanning Model + (a) NB (b) SVM [Zeng02]	(a) 88.3 (b) 88.5	(a) 88.9 (b) 88.6	(a) 93.9 (b) 95.4	NA	Three-fold CV
Mixture Gaussian Model [Li05a]	98.06	93.6	92.89	NA	NA
Two Markov Chains Model + One protein coding + MLP [Rajapakse05]	93.8	97.9	97.1	NA	Three-fold CV

Abbrev: MC=Matthew Correlation, CV= Crossvalidation, NA=Not Applicable.

From Table 6.2, it can be seen that the main approach for TIS detection is discriminative method, albeit there are hybrid of generative and discriminative methods such as the combination of Markov model and MLP [Rajapakse05]. The major computational methods for TIS prediction are from statistics and machine learning, including artificial neural networks. Although these methods have offered excellent results, they have some shortcomings: statistical methods such as positional conditional probability matrix [Zien00] suffer from high rates of false positives [Li05b]; SVM requires selection of kernel function, which at present there is no systematic way of doing it. Besides, introducing new kernel such as *Salzberg* [Zien00] or *edit kernel* [Li05b] requires additional steps to ensure the kernel is positive-definite. The main

disadvantage of these methods (statistics, artificial neural networks, SVM, etc) is, they are not interpretable, and they do not possess a reasoning process that is human-like. It is difficult to derive knowledge from a trained system based on these approaches. On the other hand, the reasoning process is highly logic-based and mathematical, which could be difficult to understand and follow. Furthermore, some of these systems do not have biological or natural interpretation [Li05b], which makes the understanding or tracking difficult. Since the goal of TIS detection is to identify the TIS accurately, as well as to gain understanding of TIS, the capability of the system to provide some form of knowledge representation to facilitate the knowledge extraction is important. The knowledge generated is also useful for validation purpose. Hence, GCL is proposed as prediction tool for TIS.

The dataset in [Pedersen97] was used to assess the GCL system. Since vertebrates have similar start codon contexts, sequences from *Bos taurus* (cow), *Gallus gallus* (chicken), *Homo Sapiens* (man), *Mus musculus* (mouse), *Oryctolagus cuniculus* (rabbit), *Rattus norvegicus* (rat), *Sus scrofa* (pig), and *Xenopus laevis* (African clawed frog) were extracted and used to assess the system.

All introns of the sequences were removed, and the remaining exon parts were joined. Only sequences containing at least 10 nucleotides upstream and 150 nucleotides downstream (with reference to the A of the start codon ATG) were selected. Subsequently, redundancy reduction was performed, where the sequences belong to same gene families, homologous genes from different organisms, and redundant sequences are removed, so as to avoid possible biased estimates in recognition performance.

The reduced dataset consists of 3312 vertebrate sequences. These sequences contain 3312 true TIS (functional ATG) and 10,063 false TIS (non-functional ATG).

A data point is generated for each potential start codon. Each data point is represented by a sequence window of 200 nucleotides centered at the respective ATG triplet. In other words, each window comprises potential TIS at the center of the window. Thus, every window contains 203 nucleotides. Every three nucleotides were matched into one amino acid. The frequencies of the amino acids exist in upstream and downstream were computed. Besides, frequency of a pair of amino acids is also taken into account. Three more features were added:

1. f1 — whether a G is present at the +1 position.
2. f2 — whether an A or G present at position -3.
3. f3 — whether there is another ATG in upstream.

Hence, the number of features is 927.

The training set is made up of 1104 positive and 4454 negative samples, whereas the testing set made up of the remaining unseen 2208 positive and 5609 negative samples. Since the number of features is ultra-huge (927), *Augmented Variance Ratio* (AVR) was used to extract the top 20 features. This is to reduce the computational load, and the development time. In addition, the fuzzy rules generated are not interpretable if the number of antecedents of a rule is 928. Hence, feature selection is done to decrease the number of features.

There are two sets of experiments. In both experiments, as GCL is a stochastic algorithm, 200 simulations are run on each of the training/testing sets.

The first experiment is to assess the GCL performance under different number of features, as well as the role of contextual information in TIS prediction.

The second experiment is to assay the feasibility of applying GCL as a concept validation tool. In this experiment, the GCL is tested with training/testing sets with the

three features f1, f2 and f3. The experiments are repeated using one of the features alone, and other combinations of the three features f1, f2 and f3.

The performance measures used are Recall, Predict, Sensitivity, Specificity, Overall Accuracy, and *Matthew correlation* (MC) *coefficient* [Li05b]. MC is a more stringent measure of accuracy that avoids the effect of imbalanced class distribution in the testing set. The overall accuracy is estimated using Equation 6.6.

$$\text{Overall Accuracy} = \frac{\text{Accuracy}_{tr} \times T_{tr} + \text{Accuracy}_u \times T_u}{T_{tr} + T_u} \quad (6.6)$$

where Accuracy_{tr} and Accuracy_u are the accuracy on training and testing sets respectively. T_{tr} is the number of samples in training set and T_u is the number of samples in the testing set. MC is defined in Equation 6.7.

$$MC = \frac{TP \times TN - FP \times FN}{\sqrt{(TP + FP)(TP + FN)(TN + FP)(TN + FN)}} \quad (6.7)$$

where TP is the true positive, TN is the true negative, FP is the false positive, and FN is the false negative.

The performance of GCL is benchmarked against the common computational biology methods [Witten05]: *Naïve Bayesian* (NB), *Logistic Regression* (LR), *Multilayer Perceptron* (MLP), *Support Vector Machine* (SVM), *Linear Discriminant Analysis* (LDA), and decision tree C4.5. The averaged performance over the three cross-validation sets of the systems under different number of features are shown in Figure 6.3, and the best results obtained by each method are summarized in Table 6.3.

Figure 6.3: Performance under different number of features.

From Figure 6.3, the performance using different number of features has very different performance. Generally, the more features are used, the greater is the accuracy. This indicates that the characteristics of sequences around the TIS convey useful information in predicting the TIS and understanding the translation of mRNA sequences [Li05b]. This also suggests that the context and global information play a part in determining the TIS [Li05a][Pedersen97].

Table 6.3: The performance of computational intelligence methods in TIS prediction.

Methods	Recall (%)	Sensitivity (%)	Specificity (%)	Predict (%)	Overall Accuracy (%)	MC
NB	85.14	89.1	67.6	80.17	81.94	0.52
LR	88.77	92.5	72.6	85.55	86.98	0.6
MLP	89.61	91.7	78.4	87.03	88.62	0.65
SVM	88.37	93.3	69.3	83.8	86.39	0.58
LDA	100	58.24	95.84	79.98	88.89	0.62
C4.5	96.67	98.6	66.2	85.69	89.58	0.58
GCL	91.77	80.07	98.61	90.65	91.06	0.83

From Table 6.2 and Figure 6.3, it is shown that GCL is competent in TIS prediction. GCL has a good competency as those reported in Table 6.2, in spite of the fact that it is using only one third of the data for training, and only 20 features, suggesting GCL is a competent tool in TIS prediction.

The significant strengths of GCL are the ability to construct intuitive fuzzy rules, and human-like inference. Together with the rules generated by decision tree C4.5, example of the rules generated by GCL is shown in Table 6.4. Both types of rules can be easily understood by user. However, the rules by decision tree are crisp rules and cannot handle uncertainty. Consequently, a small perturbation in the dataset creates a very different tree. In contrast, fuzzy rules have the capacity to capture uncertainties. These uncertainties could arise due to the nature of the problem, or during data sampling process. Furthermore, fuzzy rules allow the application of linguistic terms (e.g. high, low, etc), which encapsulates unnecessary details from users, and hence, ease the users in understanding and analyzing the system. Moreover, GCL rules does not have repeated antecedents as those in decision tree rules, which are confusing to users.

Asides from that, GCL possesses a reasoning process that is highly akin to human. This is important to facilitate the user in assessing and validating the system. With human-like inference process, the reasoning of the system can be understood and analyzed in his/her familiar terms. The inference processes of GCL and human are listed in Table 6.5. As shown, GCL has a one-to-one mapping onto the human reasoning process.

Table 6.4: Fuzzy and crisp rules.

Rule	Fuzzy rules (GCL)	Crisp rules (C4.5)
1	IF upstream amino acid pairs AA is very infrequent (0-2), downstream amino acid L frequency is low to medium (1-5), upstream amino acid A is very infrequent (0-1), THEN true TIS.	IF upstream amino acid pairs AA frequency < 0, downstream amino acid L frequency > 1, upstream amino acid A frequency $\leq$ 2, downstream amino acid L frequency $\leq$ 4, upstream amino acid A frequency > 0, downstream amino acid L frequency > 2, upstream amino acid A frequency > 1 THEN true TIS.
2	IF upstream amino acid pairs AA is rather frequent (6-7), downstream amino acid L is quite infrequent (1-2), upstream amino acid A is very frequent (8-10), THEN false TIS.	IF upstream amino acid pairs AA frequency < 0, downstream amino acid L frequency $\leq$ 1, upstream amino acid A frequency > 3, downstream amino acid L frequency $\leq$ 1, THEN false TIS.

Table 6.5: Inference process of GCL and human beings.

Step	GCL	Human
1	Present a sample.	Observe a situation, context, or problems.
2	Compute the similarity of current sample with previous experience and knowledge.	Assess the similarity of present observations based on past experience and knowledge.
3	Select the rule with maximum overall similarity.	Choose the experience or knowledge that best described current observations.
4	Derive conclusion based on the similarity of the sample and experience.	Determine the action based on the similarity of current observation compared to past experience and knowledge.
5	Output the decision.	Act out the decision.

The followings delineate an example of the inference process of GCL:

Step 1: Present a positive input sample $\mathbf{x}^+ = (0, 5, 0)$, and desired output is positive.

Step 2: Compute the similarity of the sample and the antecedents of the two rules in Table 6.4.

$$\text{Similarity}^+ = (1 + 1 + 1)/3;$$

$$\text{Similarity}^- = (0 + 1 + 0)/3;$$

Step 3: Select the rule with maximum firing strength. Since positive rule have overall similarity ($\text{Similarity}^+ = 1$) greater than the negative rule ($\text{Similarity}^- = 0.34$), positive rule is activated and negative rule is inhibited.

Step 4: Derive conclusion based on the selected positive rule.

Step 5: Output the 'positive' decision, which is the same as the desired output.

In the second part of the experiments, the capacity of GCL as a concept validation tool is assayed. In this work, GCL is used to validate the concept that, in motif GCCACCATGG, the nucleotide A at position -3 and nucleotide G at position +1 with reference to the triplet ATG are good identifiers for true TIS. The averaged result is summarized in Figure 6.4 using the different combinations of the three features f1, f2 and f3.

From Figure 6.4, it can be shown that using the three features alone do not give acceptable accuracy, confirming that the consensus alone is not sufficient to identify the true TIS [Li05b]. Nevertheless, the combinations of features do provide useful

information in detecting the true TIS, especially combination of f3 with f1 or f2. This suggests the knowledge of whether a nucleotide A presents at position -3 and G presents at position +1, and whether there is an ATG in upstream, helps in detecting the true TIS, albeit the presence of these observations do not offer any warranty.

Figure 6.4: Performance of GCL in TIS prediction under different combinations of features.

6.5.2 Cervical Cancer Diagnosis

Cervical cancer is a major health problem worldwide due to its high incidence and mortality rates [Ho04]. About 500,000 new cases of invasive cervical cancer are diagnosed worldwide each year [O'Leary02]. It is the most common form of cancer in women under 35 years of age, and is the second most common cancer in women of all age [Balas01]. It is the number one killer of young women in undeveloped countries [Ho04]. This may be due to the absence of proper screening, or the high incidence rate in those countries. Cervical cancer represents about 10% of all cancer in western world, but it is the most commonly occur cancer in many Asia countries [Sitter04] such as Korea (third most common cancer) [Ho04], and Taiwan (most common cancer) [Chu03]. Even for developed country such as United State, approximately 10,370 new cases of invasive cancer are expected in year 2007, and 3,670 cervical cancer deaths are

anticipated [ACS07]. The routines screening program, and the advancement of therapy for pre-cancerous lesions, have decreased the mortality rate over the past several decades.

Yet cervical cancer remains as the major cause of cancer death. This prompts the need for a cost-effective and sensitive method for cervical cancer detection. Pap cytology if carried out well can be a very effective tool for cervical cancer detection. However, because the inherent difficulties associated with Pap smear (Papanicolaou test), as well as the hardship in Pap smear image analysis, it often yields an unacceptable false negative rate. Automation of the analysis process using computational intelligence method could potentially ameliorate the situation.

The dataset is collected from The Department of Pathology, Herlev University Hospital [Byriel99]. It comprises a total of 500 cytological smear images of a single cell, captured using a digital camera connected to the microscope and a personal computer. The dataset comprises 200 images of normal cells, and 300 images of cancerous cells. Data are derived from these images using CHAMP software [Byriel99], and the features are listed in Table 6.6.

Table 6.6: Features derived from cytological smear images.

1. Nucleus area	8. Nucleus elongation	15. Cytoplasm perimeter
2. Cytoplasm area	9. Nucleus roundness	16. Nucleus position
3. Nucleus/ cytoplasm area ratio	10. Cytoplasm shortest diameter	17. Maxima in Nucleus
4. Nucleus brightness	11. Cytoplasm longest diameter	18. Minima in Nucleus
5. Cytoplasm brightness	12. Cytoplasm elongation	19. Maxima in Cytoplasm
6. Nucleus shortest diameter	13. Cytoplasm roundness	20. Minima in Cytoplasm
7. Nucleus longest diameter	14. Nucleus perimeter	

Since the parameters are determined by the GCL system itself, and the inherent randomness of GCL, the system is run 200 times on a training/testing set. The dataset is divided into a total of three stratified cross-validation training/testing sets, with training set made up of 1/3 of the total dataset, and testing set made up of the remaining

6 Genetic Complementary Learning - T. Z. TAN

unseen 2/3. The performance of GCL is benchmarked against *k*-Nearest Neighbour (kNN), C4.5, *Multilayer Perceptron* (MLP), *Bayesian Network*, *Support Vector Machine* (SVM), and *Logistic Regression* (LR) [Witten05]. The benchmark result of GCL is given in Table 6.7.

Table 6.7: Performance of popular methods versus GCL.

Method	Recall (%)	Sensitivity (%)	Specificity (%)	Predict (%)	Training Time (s)	Number of Rules
C4.5	99.4	91.7	95.0	93.69	0.09	8
kNN	100.0	88.7	97.0	93.69	0.02	Not Applicable
SVM	89.82	94.7	97.0	96.1	0.39	
MLP	100.0	97.7	96.5	97.0	7.38	
Bayes	91.02	94.0	97.5	96.1	0.05	
LR	100.0	93.2	92.0	92.49	3.75	
GCL	95.21	98.0	100.0	98.8	3.734	30

From Table 6.7, it can be seen that GCL has greater generalization capability compared to others. GCL outperforms most of the popular tools for aiding diagnosis decision with high accuracy. Apart from that, GCL generally requires lesser training time than MLP, and the difference between GCL training time and other popular methods is negligible. Moreover, GCL outperforms the conventional methods in term of system interpretability. GCL has a human-like reasoning process as GCL constructs fuzzy rules that resemble its knowledge. This is important because physician is able to investigate the system operation and validate the output of the system. On the contrary, statistical methods act as black box to user, which results the hardship for the physician to trust the output of the system. Since the rules are generated autonomously, the developing time of GCL is greatly reduced, compared to expert systems that require manual construction of algorithmic rules. GCL provides fuzzy rules that are highly analogous to expert rules. In contrast to decision tree rules, fuzzy rules allow the use of linguistic hedges such as ‘very’, ‘rather’ as well as linguistic terms such as ‘small’, ‘high’, etc. This offers intuitive rule, and is more

expressive for the knowledge represented. In addition, the use of linguistic term encapsulates unnecessary details from the user, and enables uncertainty handling. The comparison between the different rules is given in Table 6.8. Note that the GCL rules listed in Table 6.8 are in simplified form, the original has 24 antecedents. Performing feature selection prior to the construction of the GCL network can reduce the complexity of GCL rules.

Table 6.8: Comparison of GCL rules, decision tree rules, and expert rules.

GCL rules	Decision tree rules	Expert rules [Voss96]
IF nucleus area is medium to high, AND cytoplasm area is medium to high, AND ratio is very high, ...AND minima in cytoplasm is rather high, THEN Normal (simplified).	IF nucleus longest diameter > 12.62, AND cytoplasm brightness ≤ 0.63 THEN normal.	IF area is small, AND brightness is dark THEN normal.
IF nucleus area is very small, AND cytoplasm area is medium to high, AND ratio is medium, ...AND minima in cytoplasm is small to medium, THEN Cancer (simplified).	IF nucleus longest diameter > 12.62, AND cytoplasm brightness > 0.63, AND nucleus area ≤ 150.49 , AND cytoplasm shortest diameter > 16.68, AND nucleus roundness > 0.6239, AND cytoplasm perimeter > 185.63, AND nucleus shortest diameter > 12.33 THEN cancer.	IF area is large, AND brightness is light, THEN mild dysplastic.
		IF area is large, AND brightness is medium, THEN moderate dysplastic.
		IF area is large, AND brightness is dark, THEN severe dysplastic.

Identifying the severity of the disease or the staging of cancer sometimes is important for determining the treatment option or treatment regimen. Hence, to access the capability of GCL in disease staging, the same training/testing sets are used. The task is to classify whether the disease is mild, moderate, or severe dysplastic. The result is given in Table 6.9.

It is shown that GCL outperforms the methods presented in Table 6.9. GCL has a competent sensitivity and specificity when compared against the conventional methods listed in Table 6.9. It is relative faster in training than MLP and LR. Interpretability of the system is one of the strengths of GCL. Unfortunately, the number of rules generated by GCL is more than that of decision tree. This is because GCL specifically generates rules for both positive and negative samples, and mutates separately for the

6 Genetic Complementary Learning - T. Z. TAN

two populations. Nevertheless, GCL exhibits itself a competent *Clinical Decision Support System* (CDSS) for cervical cancer staging.

Table 6.9: Cervical cancer staging.

Method	Disease Stages	Recall	Sensitivity	Specificity	Predict	Training Time	Number of Rules
GCL	Mild	91.57	65.67	100.0	93.11	3.36	84
	Moderate	91.62	60.6	100.0	92.19	3.344	119
	Severe	92.17	61.19	100.0	92.22	2.578	5
	Mean	91.79	62.49	100.0	92.51	3.094	69
C4.5	Mild	99.4	41.2	88.1	50.6	0.05	8
	Moderate	99.4	70.4	56.1	67.57	0.05	12
	Severe	84.94	97.4	13.4	80.54	0.03	3
	Mean	94.58	69.67	52.53	66.24	0.043	8
SVM	Mild	95.78	56.6	86.6	62.57	0.39	Not Applicable
	Moderate	82.04	83.9	25.8	72.37	0.2	
	Severe	80.12	100.0	0.0	79.94	0.39	
	Mean	85.98	80.17	37.47	71.63	0.327	
kNN	Mild	100.0	49.1	83.6	55.99	0.001	
	Moderate	100.0	65.9	42.4	61.26	0.001	
	Severe	100.0	90.3	28.4	77.84	0.001	
	Mean	100.0	98.43	51.47	65.03	0.001	
MLP	Mild	100.0	49.8	98.6	57.78	9.83	
	Moderate	98.8	66.7	48.5	63.06	8.78	
	Severe	97.59	97.0	44.8	86.53	9.13	
	Mean	98.8	71.17	63.97	69.12	9.247	
LR	Mild	100.0	49.1	91.0	57.49	16.19	
	Moderate	94.01	59.2	59.1	59.16	14.5	
	Severe	84.94	97.4	35.8	85.03	11.05	
	Mean	92.98	68.57	61.97	67.23	13.91	
Bayes	Mild	96.39	56.9	77.6	61.08	0.08	
	Moderate	87.43	65.5	72.7	66.97	0.02	
	Severe	81.33	92.9	44.8	83.23	0.02	
	Mean	88.38	71.77	65.03	70.43	0.04	

The accuracy of the diagnosis, as well as the interpretability of the GCL decision support system can be improved by selecting the optimal set of features for GCL prior to the training. Not all the features used are relevant for the diagnosis and the irrelevant features may act as noise that disrupts the learning process. Therefore, by removing those irrelevant features, not only accuracy could potentially be improved, but also the interpretability of the system because the complexity of the fuzzy rules is reduced. In order to illustrate that, the performance of GCL under different number of

features is shown in Figure 6.5. The features are ranked using the SVM wrapper feature selection method [Liu02].

As shown in Figure 6.5, with optimal number of features equals to five, GCL is able to attain 100% for both Recall and Predict, i.e. the optimal performance. Three most important features for detecting cervical cancer are cytoplasm brightness, as well as nucleus shortest and longest diameter.

Figure 6.5: Performance of GCL under different number of features.

One benefit of using lesser feature is that the rules generated are less complex. For example, when only three features are used, the GCL rule has only three antecedents, as shown in Table 6.10. This is in contrast to the GCL rules listed in Table 6.8, where the number of antecedents is high. Note that the full range value is given instead of the usual linguistic term because the rules are to be used to illustrate the reasoning process of GCL.

6 Genetic Complementary Learning - T. Z. TAN

Table 6.10: Complementary rules generated autonomously by GCL.

Positive Rule	Negative Rule
IF cytoplasm brightness is (0.68, 0.94) AND nucleus shortest diameter is (8.7, 19.52) AND nucleus longest diameter is (10.86, 28.51) THEN Cancer.	IF cytoplasm brightness is (0.61, 0.65) AND nucleus shortest diameter is (10.48, 11.87) AND nucleus longest diameter is (19.88, 22.48) THEN Normal.

The following shows an example of the GCL reasoning:

Step 1: Present a positive sample to the system,

$$(\mathbf{x}, d) = [(0.75, 13.41, 19.32), 1]$$

Step 2: Compare the current input with the system knowledge base. This is equivalent to computing the sum of each input membership degree for every rule.

$$\text{Positive rule} = (1 + 1 + 1)/3 = 1$$

$$\text{Negative rule} = (0.3 + 0.1 + 0.9)/3 = 0.43$$

Step 3: Select the rule with maximum overall membership degree.

Step 4: Determine the output linguistic term linked to the rule with maximum overall membership degree. In this case, positive rule has greater sum of overall membership degree, and it is selected.

Step 5: Output the decision, which is cancer for this case.

This reasoning process of GCL provides another advantage as it is highly similar to that of a physician. The five-step reasoning flow of GCL closely resembles physician's reasoning process, as given in Table 6.11. This allows the physician to assess the system in his/her familiar terms, which is an important feature of GCL.

Table 6.11: GCL and the physician's reasoning process.

Steps	Reasoning Process	
	GCL	Physician
1	Present data sample to system.	Observe, and ask questions.
2	Assess current data sample with systems knowledge base.	Evaluate current situation with his/her own experience or knowledge.
3	Select the best-fit rule.	Choose the best-matched experience or knowledge.
4	Determine the consequent of the best-fit rule.	Determine the response referred by the best-matched experience.
5	Output based on the decision deducted from the best-fit rule.	Act out the response, decision associated with the best-matched experience or knowledge.

6.6 Summary

The strengths of GCL are inherited from GA, complementary learning, and *Fuzzy Neural Network* (FNN). The dynamicity and potential of optimal solution from GA, the good recognition performance from the gene-inspired complementary learning, as well as the human-like and interpretable operations from FNN are the advantageous features gained. Experimental results show that GCL is a competent tool in TIS prediction and cervical cancer detection, offering good recognition and interpretability as well as human-like operations. This facilitates the knowledge extraction and system validation processes.

Chapter 7 Hierarchical Complementary Learning

Human uses different knowledge representations to encode the information collected from the environment. One of them is hierarchical organization. In fact, many brain structures are organized in a hierarchical fashion. At the top of the hierarchy is a more general/complex component. Moving down the hierarchy, the subcomponents become more and more specific or simpler. The higher-level component in the hierarchy carries the contextual information of the subcomponents below it.

Hierarchical model is one of the popular computational intelligence tools because of its relative superior performance. The divide-and-conquer which recursively breaks down a complicated problem to a few simpler problems is an effective technique. Although the relations between the subcomponents are lost, the effectiveness of the hierarchical processing allows it to be widely adopted. Decision tree is the most well known hierarchical model at present. However, despite its popularity, decision tree has a few shortcomings [Zhang97]: (1) relations between features are lost; (2) it is not robust because small perturbation may produce a very different tree; (3) deeper level of the tree is less reliable because of recursive partitioning [Buntine91]. The fuzzy decision tree has been proposed to improve on the second limitation [Olaru03].

Note that recursive partitioning is not used in this work. This is because the number of samples used to form the decision boundary in recursive partitioning is lesser as it progresses down the hierarchy. Consequently, the knowledge extracted from the lower part of the hierarchy is comparatively less reliable. Thus, recursive partitioning is not adopted, but the number of samples is kept the same when traversing down the hierarchy. This is the main difference between *Hierarchical Complementary Learning* (HCL) and recursive partitioning.

Complementary learning [Tan06a] can be considered as a special kind of concept learning. Both learning observe the positive and negative samples. However, complementary learning possesses three unique characteristics when compared against conventional concept learning:

1. Explicitly generating positive and negative rulebase.
2. Segregating positive and negative rulebase.
3. Exploiting the lateral inhibition between the positive and negative.

These three characteristics are not found in traditional concept learning. Concept learning, on the other hand, does not make such distinction when learning the target concepts. Most of them such as AQ14 [De Jong93], IACL [De Jong93], C4.5 [De Jong93], ELEM2 [An03], PRISM [Cendrowska87], fuzzy clustering relevance feedback [Bhanu02] and CLIP [Yoshida95] generate the target concept (positive) and then complement it with the negation of the target concept (negative). In other words, the knowledge from negative samples is employed neither in building the rulebase nor in the inference process. This is the major difference between complementary and the traditional concept learning. Furthermore, in contrast to complementary learning, traditional concept learning does not exploit the lateral inhibition between positive and negative during inference. The segregation property of complementary learning is inspired from human pattern recognition. Moreover, the use of general and specific fuzzy sets to describe the knowledge ease the abstract level inference of the proposed learning method, which traditional concept learning is generally slower [Yoshida95].

Since it is possible that human recognizes by means of complementary learning, and subsequently stores the knowledge in a hierarchical organization, a FNN named *Hierarchical Complementary Learning* (HCL) is proposed. The hybrid system uses complementary learning to construct the fuzzy sets and rules in a hierarchical manner.

The system exploits the benefits of hierarchical processing and complementary learning. In a broader sense, HCL can be viewed as a special class of *Hierarchical Fuzzy System* (HFS) [Joo05][Wang99]. A HFS comprises a number of low-dimensional fuzzy systems connected in a hierarchical fashion [Wang99]. HFS is a promising approach to high-dimensional problem because it reduces the number of fuzzy rules and the computational requirements. Apart from that, it is shown to exhibit universal approximation property [Joo05][Liu05][Wei00][Wang99][Huwendiek99]. The conceptual structure of HCL and HFS are shown in Figure 7.1. Conventional HFS has the problem where the fuzzy rule in the middle of hierarchical structure has little meaning, and consequently, it is difficult to design [Joo05]. Some solutions have been proposed, for example, incorporating outputs of the previous layer with the fuzzy rules of the next layer [Wei00], introducing rulebase mapping scheme [Lee03], etc. However, this also introduces additional computation requirement and renders the rule extraction process tedious. Furthermore, it may make the rule difficult to be understood. In contrast, the fuzzy rule of HCL can be extracted directly from its structure, without the need of rulebase mapping. Moreover, the extracted IF-THEN fuzzy rule is in the usual form of a fuzzy system. Thus, the knowledge derivation from HCL is simpler than the conventional HFS.

Figure 7.1: The concept structure of (a) HCL and (b) HFS. FLS = Fuzzy Logic System; x = input variable; y = output of FLS.

7.1 HCL Biological Evidence

One of the reasons why human is capable of learning effectively is that human uses hierarchical representation for his/her knowledge. This hierarchical organization of knowledge allows the decomposition of complex problems to its subparts. Human, when learning highly organized serial patterns, abstracts the information onto a hierarchical structure. Humans represent their knowledge in a hierarchical rule structure that is composed of many levels. The rules at higher levels relate rules at lower levels, and they cover the largest number of patterns and set of elements. On the contrary, the rules at a lower level describe simpler patterns. In other words, the higher the hierarchy, more complicated patterns are described. This hierarchical coding is claimed to reduce memory loading [Fountain95]. Similar hierarchical representation is adopted for different tasks and judgments [Kleinsorge04], including categorization task, serial-spatial task [De Lillo04], and various cognitive tasks [Zhang94].

Hierarchical organization can also be observed in human brain structure. For instance, cortical systems are organized in hierarchical manner, i.e. some regions are hierarchically above some others. A well-studied hierarchical example is visual cortex, where the receptive field size increases as one goes up the hierarchy. This hierarchical organization has been implicated in object recognition [Riesenhuber99]. The higher-level neurons of visual cortex represent more complex features in contrast to the neurons in inferotemporal cortex that represent objects [George05]. Complex cells in visual cortex combines the signals from the simple cells in the lower hierarchy to specifically tuned to different features such as motion-in-depth and retinal disparity [Sabatini02]. The neurons in the lower level hierarchy treat the information from

above as contextual information. When an object is perceived in the higher level, the activity of the lower levels of the hierarchy is reduced [George05].

Another brain area that is organized in hierarchical manner was the emotion processing and motivational control centre. This brain area is responsible for goal-directed and reward-oriented behaviour. It consists of amygdala, orbitofrontal cortex, basal ganglia, and the dopamine neurons that form a multiple-level hierarchical learning system. The orbitofrontal cortex is situated at the top level that makes abstract decision on selecting an action to achieve a goal. The amygdala at the next level decodes the sensory cues as reward-related variables, whereas the basal ganglia at a level below execute the selected action. The dopamine neurons then reinforce the amygdala and orbitofrontal cortex in their adaptation [Zhou02]. Another study confirms that neural representations of motivational drives such as fear, power-dominance are arranged in a hierarchical manner [Sewards03]. Mesial cortex representing the subjective awareness of motivational drives is situated at the highest level, with hypothalamic representation illustrating involuntary responses of motivational drives at a lower level.

In addition, brain structures in tactile shape learning are also organized in a hierarchical fashion. The somatosensory areas are interconnected in hierarchy, and have been implicated in the computation of tactile shape [Bodegård01]. The somatosensory processing is observed not only in shape discrimination, but also in various body parts such as the trunk, the arm, the hindlimb, and the oral representations [Toda02]. It is complementary in nature, i.e. the positive and negative examples are learned and different brain areas are registered for different kind of sensory discriminations. These hierarchical brain structures provide the biological evidence for the hierarchical complementary learning.

Formally, a hierarchical learning can be formulated as follows: for a particular concept C represented by a fuzzy set $\mathbf{A}$, it is defined by a membership function $\mu_{\mathbf{A}_C}$. Given a set of observations $\mathbf{x} = (x_1, x_2, \dots, x_l)$, a concept can be represented as in Equation 7.1.

$$\begin{aligned} \mathbf{A}_{C^{level}} &= \bigcup_{C_i^{level-h} \subseteq C^{level}} \mathbf{A}_{C_i^{level-h}}, h=1, 2, \dots, H \\ \mu_{\mathbf{A}_{C^{level}}}(\mathbf{x}) &= \sum_{C_i^{level-h} \subseteq C^{level}} \left[\sum_{l=1}^l \mu_{\mathbf{A}_{C_i^{level-h}}}(x_l) \right] \end{aligned} \quad (7.1)$$

where h is the hierarchical level index, and H is the number of levels in the hierarchy. In other words, a concept C^{level} at higher level is described by the combination of concepts

$C^{level-h}$ from lower hierarchy levels, i.e. $\mathbf{A}_{C^{level}} = \bigcup_{l=1}^l \mathbf{A}_{C^{level-l}}$.

7.2 HCL Learning

The *Hierarchical Complementary Learning* (HCL) integrates the complementary learning and hierarchical organization. By functionally model the complementary and hierarchical category learning processes, better recognition performance can be obtained as the former exploits the characteristics of both positive and negative concepts as well as the lateral inhibition between the two. On the other hand, the breaking down of complex concepts through the hierarchical organization allows a simpler representation of the problem. Hence, the combination of complementary and hierarchical learning, i.e. HCL is promising. It can be formulated as follows:

Given a dataset $D = D^+ \cup D^-$, where $D^+ = \{X^+, Y^+\}$ and $D^- = \{X^-, Y^-\}$ are the positive and negative samples, respectively. HCL creates a rulebase R according to the dataset. The rulebase R is made up of positive and negative rules, i.e. $R = R^+ \cup R^-$

where $R^+ = \bigcup_{k=1}^{K^+} r_k$, $R^- = \bigcup_{k=1}^{K^-} r_k$, $k \in [1, K]$ is the index for the rules and K is the total number of rules, and $K = K^+ + K^-$. $R: \mathbf{A} \rightarrow \mathbf{B}$ where $\mathbf{A}$ and $\mathbf{B}$ are the input and output fuzzy sets that define the input and output linguistic terms. Each rule r_k is in the form shown in Equation 7.2.

$$r_k: \text{IF } x_1 \text{ is } \mathbf{A}_{1k}, \dots, x_i \text{ is } \mathbf{A}_{ik}, \dots, x_I \text{ is } \mathbf{A}_{Ik}, \text{ THEN } y_1 \text{ is } \mathbf{B}_{1k}, \dots, y_m \text{ is } \mathbf{B}_{mk}, \dots, y_M \text{ is } \mathbf{B}_{Mk} \quad (7.2)$$

where x_i is the input from the i^{th} dimension, y_m is the output from the m^{th} dimension, $\mathbf{A}_{ik}$ is the input linguistic term that links x_i to the k^{th} rule and $\mathbf{B}_{mk}$ is the output linguistic term that links the k^{th} rule to y_m . HCL uses trapezoidal fuzzy sets as its linguistic terms, which is defined by four parameters: Min , u , v and Max , where $Min < u < v < Max$. (Min, Max) are the supports of the fuzzy set. They are the maximum and minimum values of the input space. For example, for input linguistic term $\mathbf{A}_i$, its supports Min_i is preset to the minimum value of the i^{th} dimension (from training set), whereas Max_i is preset to the maximum value of the i^{th} dimension (from training set). That is, $Max_i = \max\{x_i^1, \dots, x_i', \dots, x_i^T\}$ and $Min_i = \min\{x_i^1, \dots, x_i', \dots, x_i^T\}$. (u, v) define the kernel of the trapezoidal fuzzy set where u and v are the lower and upper bounds respectively.

Every linguistic term (represented by fuzzy set) characterizes a concept. In this work, HCL has a hierarchy of two levels: a general (high level) and a specific (low level) fuzzy set, i.e. $\mathbf{A} = \overline{\mathbf{A}} \cup \underline{\mathbf{A}}$, where $\overline{\mathbf{A}}$ is the general concept, and $\underline{\mathbf{A}}$ is the specific concept. Since each linguistic term is defined by its membership function, it can be represented as in Equation 7.3.

$$\mu_{A_{ik}}(x_i) = \begin{cases} \mu_{\overline{A_{ik}}}(x_i) \\ \mu_{\underline{A_{ik}}}(x_i) \end{cases} \quad (7.3)$$

HCL inference is based on the difference between the firing strengths of positive and negative rules, as described in Equations 7.4 to 7.7.

$$Y = \begin{cases} Y^+, & \text{if } \mu_{R^+}(\mathbf{x}) > \mu_{R^-}(\mathbf{x}) \\ Y^-, & \text{if } \mu_{R^+}(\mathbf{x}) < \mu_{R^-}(\mathbf{x}) \end{cases} \quad (7.4)$$

where

$$\mu_{R^+}(\mathbf{x}) = \max_{k \in K^+} \left\{ \frac{1}{I} \sum_{i=1}^I \mu_{A_{ik}^+}(x_i) \right\} \text{ or } \frac{1}{K} \sum_{k=1}^{K^+} \left\{ \frac{1}{I} \sum_{i=1}^I \mu_{A_{ik}^+}(x_i) \right\} \quad (7.5)$$

$$\mu_{R^-}(\mathbf{x}) = \max_{k \in K^-} \left\{ \frac{1}{I} \sum_{i=1}^I \mu_{A_{ik}^-}(x_i) \right\} \text{ or } \frac{1}{K} \sum_{k=1}^{K^-} \left\{ \frac{1}{I} \sum_{i=1}^I \mu_{A_{ik}^-}(x_i) \right\} \quad (7.6)$$

When a sample $\mathbf{x} \in X^+$ is presented to HCL, the positive rules are fired more strongly than the negative rules, i.e. $\mu_{R^+}(\mathbf{x}) > \mu_{R^-}(\mathbf{x})$. This concurrently activates the positive rules and inhibits the negative rules, which results in a positive decision, $y \in Y^+$. Exploiting the lateral inhibition between positive and negative classes avoids possible confusion in the inference process.

HCL is made up of five-layer fuzzy neural network, with each layer corresponds to the elements of the fuzzy rules shown in Equation 7.2. The building block of HCL is shown in Figure 7.2.

The construction of the structure is autonomously executed. The overview of the HCL learning algorithm is as follows:

Given a dataset $D = \{(\mathbf{x}, \mathbf{d})^1, \dots, (\mathbf{x}, \mathbf{d})', \dots, (\mathbf{x}, \mathbf{d})^T\}$:

Step 1: Segregate the dataset into positive and negative samples:

$$D \rightarrow D^+ \cup D^-.$$

Figure 7.2: Building block of hierarchical complementary learning fuzzy neural network. Each layer corresponds to the elements of fuzzy rules: (1) input linguistic labels, X ; (2) input linguistic terms, A ; (3) rules, R ; (4) output linguistic terms, B ; (5) output linguistic labels, Y .

Step 2: Rank the features using some criteria, and arrange them in descending order: $F_{rank} = \{f_{topmost}, f_{topmost-1}, \dots, f_{bottommost}\}$.

Step 3: Select the topmost feature and remove it from F_{rank} . Put it into the HCL feature list F : $f_i \rightarrow f_{topmost}$, $F_{rank} = F_{rank} - f_{topmost}$, $F = F \cup f_i$, and for both positive and negative samples:

If there is no existing rule, create a rule based on the sample and go to Step 5. Otherwise, proceed to next step.

Step 4: Select the rule nearest to this sample and decide whether to expand the kernel to include it.

Step 5: Perform chunking and removal of redundant rules.

Step 6: Check the interpretability of the rules.

-
- Step 7: Assess the rules by discarding rules that are below the rule fitness threshold.
- Step 8: Check if the termination condition is met. If so, terminate learning, else, go back to Step 3.

Note that, Steps 3 to 7 are carried out separately and concurrently for positive and negative rules. The following sections provide the details of each step.

7.2.1 Step 1: Segregation of Dataset

This step segregates the dataset D into D^+ and D^- . This is to functionally model and facilitate the implementation of the segregation of positive and negative knowledge in human brain, where different brain areas are registered for recognitions of different objects. In other words, the training samples are divided and grouped into positive and negative samples. HCL then construct the positive and negative knowledge based on the positive samples alone, and likewise for the negative knowledge.

7.2.2 Step 2: Feature Ranking

This step ranks the features $f_i \in [1, I]$ and puts them into F_{rank} in descending order. This can be done using feature selection criteria such as *Chi-Square*, *entropy*, etc. In this work, *Augmented Variance Ratio* (AVR) [Zhang02] is used (see Equation 7.7).

$$AVR = \frac{S_b}{\frac{1}{M} \sum_{m=1}^M \frac{S_m}{\min_{m \neq m^*} |\mu_m - \mu_{m^*}|}} \quad (7.7)$$

where M is the number of classes, S_m is within-class variance for m^{th} class, S_b is between-class variance and μ_m is the mean for m^{th} class. The features are added into a list F_{rank} according to their ranking.

7.2.3 Step 3: Feature Selection and Rule Creation

This step selects a feature f_i with the highest AVR inside F_{rank} , and then removes it from F_{rank} . HCL classifies the data by maintaining a feature list. From the selected feature, HCL learns to classify the data. If there is no existing rules, create a rule from the current feature and go to Step 5. Otherwise, proceed to next step. Note that only specific fuzzy sets $\underline{A}_{ik}$ are created in new rule construction with their kernels initially centering on the input. The general fuzzy sets of the new rule can then be derived from

the union of these specific fuzzy sets, i.e. $\overline{A}_k = \bigcup_{i=1}^J \underline{A}_{ik}$.

7.2.4 Step 4: Rule Selection

This step selects a rule that is nearest to this sample. If there is a perfect match between the selected rule and the sample (i.e. no learning is required), go to Step 8 immediately. Otherwise, proceed to check if there is any sample belonging to other class in between the kernel of this nearest rule and the current sample. If there is, create a new rule based on this sample. Otherwise, expand the kernel of the fuzzy set to include this sample.

7.2.5 Step 5: Rule Combination

This step combines the rules by chunking the fuzzy sets to remove redundant rule [Gobet01]. The chunking operation can reduce the number of rules and hence improves the system interpretability. Equation 7.8 shows the chunking of two pieces of similar fuzzy sets into a single unit, and its scenarios are depicted in Figure 7.3.

$$[u_{ik}^+(t+1), v_{ik}^+(t+1)] = \begin{cases} [u_{ik}^+(t), v_{ik}^+(t)], & \text{if } u_{ik}^+(t) < u_{ik}^+(t), v_{ik}^+(t) > v_{ik}^+(t) \\ [u_{ik}^+(t), v_{ik}^+(t)], & \text{if } u_{ik}^+(t) > u_{ik}^+(t), v_{ik}^+(t) < v_{ik}^+(t) \\ [u_{ik}^+(t), v_{ik}^+(t)], & \text{if } u_{ik}^+(t) > u_{ik}^+(t) \cap \{(u_{ik}^-, v_{ik}^-) \cap [v_{ik}^+(t), u_{ik}^+(t)] = \emptyset\} \\ [u_{ik}^+(t), v_{ik}^+(t)], & \text{if } v_{ik}^+(t) < u_{ik}^+(t) \cap \{(u_{ik}^-, v_{ik}^-) \cap [v_{ik}^+(t), u_{ik}^+(t)] = \emptyset\} \\ [u_{ik}^+(t), v_{ik}^+(t)], & \text{if } u_{ik}^+(t) < u_{ik}^+(t), v_{ik}^+(t) < v_{ik}^+(t) \\ [u_{ik}^+(t), v_{ik}^+(t)], & \text{if } u_{ik}^+(t) > u_{ik}^+(t), v_{ik}^+(t) > v_{ik}^+(t) \end{cases} \quad (7.8)$$

where $[u_{ik}^+(t), v_{ik}^+(t)]$ is the kernel of the nearest positive rule, and $k^* \neq k, k' \neq k$.

Figure 7.3: The different scenarios of the chunking operation: (a) a fuzzy set is a subset of another. (b) two fuzzy sets are non-overlapping. (c) two fuzzy sets are overlapping.

7.2.6 Step 6: Tuning of Fuzzy Set

This step is to ensure the distinguishability among the rules by resetting the kernel and the support of the fuzzy sets. The strategy is that, given a fuzzy set, the system checks for the nearest two fuzzy sets (from the same class), and then tunes the support according to the kernel of the two nearest fuzzy sets. Equation 7.9 shows such tuning:

$$[Min_{ik}^+(t+1), Max_{ik}^+(t+1)] = \left[u_{ik}^+(t) - \frac{u_{ik}^+(t) - v_{ik_{Left}}^+(t)}{2}, v_{ik}^+(t) + \frac{u_{ik_{Right}}^+(t) - v_{ik}^+(t)}{2} \right] \quad (7.9)$$

where $(u_{ik_{Left}}^+(t), v_{ik_{Left}}^+(t))$ and $(u_{ik_{Right}}^+(t), v_{ik_{Right}}^+(t))$ are the kernels of the nearest rule smaller and greater than $(u_{ik}^+(t), v_{ik}^+(t))$, respectively, and $k^* \neq k$. Figure 7.4 shows an example of such tuning.

Figure 7.4: (a) The supports of the fuzzy sets are the same originally. (b) After tuning, the supports are adjusted such that they won't overlap with other's kernel. It is tuned to the midpoint between two neighboring kernels.

7.2.7 Step 7: Rule Assessment

This step is to assess the validity of the rules. It requires the definition of rule fitness. Rule fitness is set to the number of samples that the rule has correctly classified during the training process. If the rule fitness is lower than a pre-defined threshold, the rule is removed.

7.2.8 Step 8: Terminating Condition for the Learning Process

This step tests the terminating condition of the learning process. If the number of features, i.e. $|F|$, is greater than the maximum number of features f_{user} (a user-defined parameter), then terminate the learning process. Otherwise, go to Step 3. $|F|$ can be determined as follows:

If the current input feature gives a training accuracy more than the threshold, then add this feature to F , otherwise discard it (see Equation 7.10).

$$F(t+1) = \begin{cases} F(t) \cup f_i, & \text{if } accuracy_{f_i} > \rho \\ F(t), & \text{otherwise} \end{cases} \quad (7.10)$$

where $accuracy_{f_i}$ is the training accuracy attained by using the feature f_i alone.

7.3 HCL Inference

The inference process of HCL exploits the characteristics of positive and negative classes, as well as the lateral inhibition between the classes. It is summarized in Equations 7.11-7.16. Given a sample $\mathbf{x}$ and let $r_{k,f}^+$ be the k^{th} positive rule that is based on feature $f \in F$, infer

$$Y = \begin{cases} Y^+, & \text{if } \mu_{R^+}(\mathbf{x}) > \mu_{R^-}(\mathbf{x}) \\ Y^-, & \text{if } \mu_{R^+}(\mathbf{x}) < \mu_{R^-}(\mathbf{x}) \end{cases} \quad (7.11)$$

$$\begin{aligned} \mu_{R^+}(\mathbf{x}) &= \max_{k \in K^+} \{\mu_{r_k^+}(\mathbf{x})\} > \max_{k \in K^-} \{\mu_{r_k^-}(\mathbf{x})\} \\ &\cap \frac{1}{|F|} \sum_{f=1}^F \max_{k \in K^+, f \in F} \{\mu_{r_{k,f}^+}(\mathbf{x})\} > \frac{1}{|F|} \sum_{f=1}^F \max_{k \in K^-, f \in F} \{\mu_{r_{k,f}^-}(\mathbf{x})\} \end{aligned} \quad (7.12)$$

$$\begin{aligned} \mu_{R^-}(\mathbf{x}) &= \max_{k \in K^+} \{\mu_{r_k^+}(\mathbf{x})\} < \max_{k \in K^-} \{\mu_{r_k^-}(\mathbf{x})\} \\ &\cap \frac{1}{|F|} \sum_{f=1}^{|F|} \max_{k \in K^+, f \in F} \{\mu_{r_{k,f}^+}(\mathbf{x})\} < \frac{1}{|F|} \sum_{f=1}^{|F|} \max_{k \in K^-, f \in F} \{\mu_{r_{k,f}^-}(\mathbf{x})\} \end{aligned} \quad (7.13)$$

where

$$\mu_{r_k^+}(\mathbf{x}) = \frac{1}{I} \sum_{i=1}^I \mu_{\underline{A}_{ik}}(x_i) \quad (7.14)$$

$$\mu_{r_k^-}(\mathbf{x}) = \frac{1}{I} \sum_{i=1}^I \mu_{\underline{A}_{ik}}(x_i) \quad (7.15)$$

For cases when the decisions from maximum and average do not tally, the general fuzzy set can be considered (see Equation 7.16). This is equivalent to taking the average of the rules. For example, the maximum firing strength of positive is greater than the negative, but the average of maximum firing strength of positive is smaller than the negative, then infer using Equation 7.16. If $\max_{k \in K^+} \{\mu_{r_k^+}(\mathbf{x})\} > \max_{k \in K^-} \{\mu_{r_k^-}(\mathbf{x})\}$,

but $\frac{1}{|F|} \sum_{f=1}^{|F|} \max_{k \in K^+, f \in F} \{\mu_{r_{k,f}^+}(\mathbf{x})\} < \frac{1}{|F|} \sum_{f=1}^{|F|} \max_{k \in K^-, f \in F} \{\mu_{r_{k,f}^-}(\mathbf{x})\}$, then

$$Y = \begin{cases} Y^+, & \text{if } \max_{k \in K^+} \left\{ \frac{1}{I} \sum_{i=1}^I \mu_{A_{ik}^+}(x_i) \right\} > \max_{k \in K^-} \left\{ \frac{1}{I} \sum_{i=1}^I \mu_{A_{ik}^-}(x_i) \right\} \\ Y^-, & \text{Otherwise} \end{cases} \quad (7.16)$$

In Equation 7.16, $\max_{k \in K^+} \left\{ \frac{1}{I} \sum_{i=1}^I \mu_{A_{ik}^+}(x_i) \right\}$ can be approximated by $\frac{1}{K} \sum_{k=1}^{K^+} \left\{ \frac{1}{I} \sum_{i=1}^I \mu_{A_{ik}^+}(x_i) \right\}$,

and $\max_{k \in K^-} \left\{ \frac{1}{I} \sum_{i=1}^I \mu_{A_{ik}^-}(x_i) \right\}$ can be approximated by $\frac{1}{K} \sum_{k=1}^{K^-} \left\{ \frac{1}{I} \sum_{i=1}^I \mu_{A_{ik}^-}(x_i) \right\}$.

Alternatively, the positive and negative rules can be viewed as two separate systems and the complementary learning is viewed as an ensemble made up of these two subsystems. There are many ways of inferring the conclusion from these subsystems. In this work, for simplicity, the majority voting scheme is employed, i.e. take the decision of the majority. In cases where conflict arises, the HCL infers based on this voting scheme. Each positive rule makes positive decision if the similarity is high, otherwise, a negative decision is made (see Equations 7.17). Likewise for the negative rule (see Equation 7.18).

$$Decision_{r_k^+}(\mathbf{x}) = \begin{cases} +, & \text{if } \mu_{r_k^+}(\mathbf{x}) > \rho \\ -, & \text{otherwise} \end{cases} \quad (7.17)$$

$$Decision_{r_k^-}(\mathbf{x}) = \begin{cases} -, & \text{if } \mu_{r_k^-}(\mathbf{x}) > \rho \\ +, & \text{otherwise} \end{cases} \quad (7.18)$$

The overall decision of the positive and negative rule subsystems is taken as the majority votes from the rules within the subsystems respectively (see Equations 7.19 and 7.20). The final decision is made based on the output of these two systems: if both reach a consensus, then output the decision, else randomly choose a decision.

$$Decision_{R^+}(\mathbf{x}) = \begin{cases} +, & \text{if } |Decision_{r_k^+}(x) = +| > |Decision_{r_k^+}(x) = -| \\ -, & \text{otherwise} \end{cases} \quad (7.19)$$

$$Decision_{R^-}(\mathbf{x}) = \begin{cases} -, & \text{if } |Decision_{r_k^-}(x) = -| > |Decision_{r_k^-}(x) = +| \\ +, & \text{otherwise} \end{cases} \quad (7.20)$$

7.4 HCL ASSESSMENT

Pattern recognition includes clustering and classification. It is important in many areas, especially in knowledge extraction from data. Major approaches of pattern recognition come from *machine learning*, *statistics*, *database*, *data mining*, *artificial neural network*, and *computational intelligence* communities. With the rapid growth of data, there exists ultra-huge dataset with difficulties in handling large number of classes, dimensions, and samples. As a result, the amount of information in the world doubles every 20 months [Jensen02]. Thus, more effective and less computational expensive pattern recognition tools are needed to combat this growth. Many endeavors have been attempted. However, there is no panacea for the problem. Every algorithm possesses some bias that favors it in one domain, but not another [Sankar96]. This applies to HCL as well. Nonetheless, in order to assay the HCL as a pattern recognition tool, it is analyzed based on some desiderata derived from literatures. The desiderata of a pattern recognition tool are assessed in HCL in the following sections.

7.4.1 Good Classification Accuracy

How accurate a method can classify the dataset? This is inclusive of generalization and training accuracies. The higher the accuracy, the better is the system. As can be seen in Section 7.5, the generalization and recall accuracies of HCL are very competitive when it is benchmarked against other methods.

7.4.2 Effective treatment of high dimensionality

What are the strategies taken to handle the features/dimensions? Which features are to be considered collectively? These questions are important because it is meaningless to employ all the features for analysis. Besides, it may waste resources because the

density at a point in high dimensional data may be sparse [Agrawal98]. Moreover, some of the features may be noisy. In this work, HCL relies on a separate feature analysis to rank the features. HCL uses features from the most relevant to the least relevant to classify the data.

7.4.3 Good Interpretability

A good pattern recognition tool should offer means to describe its computations and results in such a way that it can be easily understood by the user. However, the provision of fuzzy rules does not necessarily warrant good interpretability. Steps are taken to ensure HCL satisfy a few established interpretability criteria [Penã-Reyes04][Casillas03] [Ishibushi01]:

1. *Complexity*: There are three facets to this criterion: 1) number of antecedents; 2) number of rules; 3) number of elements (number of linguistic terms per feature/dimension). The number of antecedents and number of elements are affected by f_{user} . Hence, the user has control over the number of antecedents a rule should have. HCL uses the chunking and removal of redundant rules (Equation 7.10) to reduce the number of rules as well as the number of elements. For HCL, the number of elements corresponds to the number of rules.
2. *Distinguishability*: This demands that the membership functions are easily differentiated so that a clear semantic meaning can be assigned. This criterion is satisfied through HCL learning because HCL considers samples from other class prior to its knowledge construction. Moreover, chunking and readjustment are done on the rules after learning to ensure there is minimal overlapping between fuzzy sets (see Equation 7.9).

-
3. *Normality*: At least one datum in the universe of discourse has to perfectly match with each fuzzy set constructed by the system. Since HCL is a data-driven and data-centered learning system, this criterion is satisfied without the need of additional strategy.
 4. *Coverage or completeness*: This is to ensure that for any possible input, there is at least one rule to describe it. This is satisfied through the data-centered learning and higher-level rule of HCL, provided that the training set is complete. Unfortunately, most of the time this is not the case. For input combination not seen in the training set, HCL does not interpolate. HCL infers based on the distance of the sample with its general rule, i.e. rule with general membership functions.

7.4.4 High Efficiency, scalability and usability

This refers to the memory requirement and computational load of the system. An efficient system has low memory requirement and computational load. The system should scale linearly with the dimensionality and size of the dataset, as well as the number of classes. In other words, neither the memory requirement should scale nonlinearly with input dimension nor the time complexity of the system should scale nonlinearly with the size of dataset. Many conventional approaches suffer from low scalability. For instance, the backpropagation learning and decision tree-based approach are computationally expensive, especially when it comes to non-separable function [Sankar96].

Given I = number of input dimension, T = number of samples, K = number of rules, and f_{user} = the user-defined maximum number of features, where $f_{user} \leq I$ ($f_{user} = I$, when it is the worst-case), the complexity analysis is done in terms of

7 Hierarchical Complementary Learning - T. Z. TAN

number of addition/subtraction and number of multiplication/division. The costs of comparison and assignment are assumed negligible. The complexity analysis of HCL is given in Table 7.1.

Table 7.1: Complexity analysis of HCL.

Description	Number of Addition/ subtraction	Number of Multiplication/ division	Remark
Rank all features according to AVR score.	$2I(T+M)$	$I(T+4)$	Compute the mean, variance, and AVR for all features.
Rule construction, learning, and chunking.	$f_{user} \{ T + \frac{K^+}{2} [K^+ - 1 + 1] + \frac{K^-}{2} [K^- - 1 + 1] \} = f_{user} \{ T + \frac{1}{2} ((K^+)^2 + (K^-)^2) \}$	0	In the worst case, $K^+ = T^+$, and $K^- = T^-$, i.e. generate rules for every sample.
Ensuring the distinguishability of the rule.	$f_{user} \{ 2(K^+ - 2) + 8K^+ - 2 + 2(K^- - 2) + 8K^- - 2 \} = f_{user} \{ 10(K^+ + K^-) - 12 \}$	$f_{user} \{ 4(K^+ + K^-) - 4 \}$	Getting the nearest neighbor and computing the new kernel.
Assess the rule.	$f_{user} \times T \times K$	$f_{user} \times T \times K$	

From Table 7.1, the computational complexity is $O(f_{user} \times T \times K)$. In the worst case, $f_{user} = I$, $K = T$. Thus, HCL is scalable to the number of dimension, but is less scalable to the number of samples. The memory requirement for HCL is $f_{user} \times 2K$, as one rule has only one linguistic term in HCL. Hence, in terms of memory, HCL is scalable to the number of dimension and number of samples.

On the other hand, although the rule of HCL is in conjunction/intersection configuration (Equation 7.2), its structures and learning algorithm allow easy disjunction/union rule configuration as well. This is because HCL can be viewed as a multiple single-feature complementary learning system. In this case, the rule generated by HCL is in the form described by Equation 7.21, which is a functional equivalent to Equation 7.2 (the propositional logic transformation between the two equations is given in Appendix B).

$$\begin{aligned}
 r_k: \{ & [\text{IF } x_1 \text{ is } \mathbf{A}_{1k} \text{ THEN } y_1 \text{ is } \mathbf{B}_{1k}, \dots, y_m \text{ is } \mathbf{B}_{mk}, \dots, y_M \text{ is } \mathbf{B}_{Mk}] \dots \text{OR} \\
 & [\text{IF } x_i \text{ is } \mathbf{A}_{ik} \text{ THEN } y_1 \text{ is } \mathbf{B}_{1k}, \dots, y_m \text{ is } \mathbf{B}_{mk}, \dots, y_M \text{ is } \mathbf{B}_{Mk}] \dots \text{OR} \\
 & [\text{IF } x_l \text{ is } \mathbf{A}_{lk} \text{ THEN } y_1 \text{ is } \mathbf{B}_{1k}, \dots, y_m \text{ is } \mathbf{B}_{mk}, \dots, y_M \text{ is } \mathbf{B}_{Mk}] \} \\
 (7.21)
 \end{aligned}$$

The union rule configuration can be employed if further complexity reduction is required. When a new input is added, union rule configuration yields only a linear increase in the computational time, as compared to intersection rule configuration that causes a combinatorial rule explosion [Combs98].

7.4.5 Robustness

A robust system is less susceptible to noise and outliers, insensitive to data presentation order, and does not assume the underlying data distribution [Agrawal98]. As uncertainty, noise, and outliers are inevitable, a robust pattern recognition tool should have outlier removal method or strategy to minimize the effect of outlier and noise. Unfortunately, there is no systematic way to determine whether a sample is outlier, extreme example, or special case. Thus, a better solution is to tolerate with these potential outliers/special cases, rather than deleting them.

Some statistical approaches assume or require the data to follow certain probability distribution in order to perform well. Moreover, some distance-based and projection approaches are highly susceptible to outliers. An outlier can distort not only the mean of samples, but also the pooled sample covariance matrix [Deogun97]. On the contrary, HCL is insensitive to data presentation order because of its learning. HCL, when forming a rule, takes into account the minimum and maximum values of that particular feature. When expanding the kernel of the rule, HCL first ensures that this expansion does not introduce overlaps between fuzzy sets, especially those from the opposite class, through the computation of the distance

between the rule and its neighbor. Thus, this alleviates the effect of outliers as well as the data presentation order. In addition, HCL does not assume the distribution of the underlying data.

7.4.6 Incremental data incorporation

It is highly desirable that the system can incorporate new data without having to execute the algorithm on the whole dataset again [Vijayakumar05]. Incremental learning has relatively low computational and memory requirements compared to non-incremental learning. Furthermore, incremental incorporation of new data into the system without having to retrain the system saves cost and time. HCL inherited these advantages as it is an incremental learning system.

7.4.7 Minimal Number of parameters

In general, the more parameters an algorithm has, the more likely it is able to learn a problem well. Thus, there arises the need for the search of optimal parameters settings. However, this often involves trial-and-error, or brute-force, which is time-consuming [Fernandez04]. Hence, it is highly desirable that an algorithm has lesser number of parameters. HCL requires only one parameter, i.e. the maximum number of features f_{user} .

7.4.8 Short response time

HCL system is able to make and explain its decision within a short time frame for new data. Thus, the computational and training time of the system should not take long. The time for HCL to output a response is generally fast.

7.4.9 One-pass of training data

It is costly to have multiple data scan of the dataset that is huge in number of classes, samples and dimensions. Thus, it is desirable to have only a single pass of the training data. HCL can fulfill this one-pass requirement by modifying its implementation. Instead of analyzing feature by feature, HCL is able to analyze a number of features in parallel and hence it requires only a one-pass of the training data.

7.5 Experiment

7.5.1 HCL and Proteomics

In order to assess the performance of HCL, five ultra-huge datasets are used. The dataset is divided into three mutually exclusive and stratified sets of cross-validation for training and testing. Training data is made up of 1/3 of the data, and the testing data is made up of the remaining unseen 2/3 of the dataset. The datasets used in the experiments, and their training/testing data distributions are given in Table 7.2. Note that the leukemia subtype dataset is used for subtype classification instead of differentiating cancerous and normal patients. This leukemia dataset consists of seven subtypes. The 7-class classification is broken down into seven 2-class classification problems (the details of the performance on 7-class classification are given in Table C1 of Appendix C).

The performance of HCL is benchmarked against other popular methods [Witten05]: *decision tree C4.5*, *Logistic Regression (LR)*, *Multilayer Perceptron (MLP)*, *Support Vector Machine (SVM)*, *Linear Discriminant Analysis (LDA)*, *Naïve Bayesian (NB)*, and *FALCON-AART* [Tan05b].

7 Hierarchical Complementary Learning - T. Z. TAN

Table 7.2: Dataset used in experiments and the distribution for training and testing set.

Dataset	Number of dimension	Training set		Testing set	
		Cancer (positive)	Normal (negative)	Cancer (positive)	Normal (negative)
1. DNA gene expression of prostate cancer [Singh02].	12600	26	20	51	38
2. Gene expression of Colorectal Carcinomas [Notterman01].	7465	6	6	12	12
3. Proteomics of ovarian cancer [Petricoin III02].	15154	39	34	77	66
5. Proteomics of Pancreas cancer [Hingorani03].	15154	26	32	54	75
6. Gene expression of Leukemia cancer subtypes [Yeoh02].	12558	215 (all are cancer cases)		112 (all are cancer cases)	

These systems were trained and tested on the reduced training/testing sets. In order to have a fair benchmarking, the top 20 features ranked by *Augmented Variance Ratio* (AVR) are extracted from the training/testing sets. Note that the performance of SVM is at its best performance when employing different kernels. Figure 7.5 shows the averaged performance of the systems for the five medical tasks, under different number of features.

In Figures 7.5 and 7.6, the HCL has a competent recognition performance over the other methods in the five medical tasks. Although each method has different optimal number of features in classifying the dataset, HCL outperforms them in most cases. Generally, complementary learning system has superior overall recognition performance, except for the task of prostate cancer diagnosis, where the FALCON-AART is affected by the presence of irrelevant features. As a result, FALCON-AART performance significantly deteriorates as these features are added into the system. On the contrary, the performance of HCL is consistent, without much oscillation in giving the diagnosis service. These irrelevant features can be viewed as noise to the system because they confuse the decision making.

Figure 7.5: Averaged recall accuracy of the five diagnosis tasks.

Figure 7.6: Averaged predict accuracy of the five diagnosis tasks.

Thus, from this perspective, HCL is less sensitive to the effect of noise. This is supported by the consistent performance throughout the five medical tasks. The best performance of the methods in these tasks, regardless of the number of features used, is summarized in Table 7.3. Note that the positive sample refers to the cancer example. Recall is the accuracy of the training set, whereas Predict is the accuracy of the testing set. The detailed performance of SVM using different kernels for the different tasks is given in Table 7.4.

7 Hierarchical Complementary Learning - T. Z. TAN

Table 7.3: Classification performance of the systems in the five medical tasks.

Method		Prostate	Colorectal	Ovarian	Pancreas	Leukemia
C4.5	Recall (%)	93.48	91.67	66.52	98.82	87.6
	Sensitivity (%)	92.1	58.3	95.7	96.53	8.14
	Specificity (%)	21.6	50	3.8	98.83	97
	Predict (%)	51.69	55.17	58.36	98.01	83.72
	Training Time (s)	0.02	0.02	0.02	0.02	0.02
LR	Recall (%)	100	100	91.38	100	81.55
	Sensitivity (%)	78.9	91.7	40.8	99.43	15.89
	Specificity (%)	41.2	50	49.1	98.53	89.66
	Predict (%)	58.3	70.83	45.19	98.21	78.37
	Training Time (s)	0.05	0.02	0.05	0.02	0.03
MLP	Recall (%)	98.83	100	76.86	100	88.49
	Sensitivity (%)	77.3	91.7	82.9	98.9	9.11
	Specificity (%)	41.2	50	18.9	99.4	98.07
	Predict (%)	57.18	70.83	57.59	99.2	85.09
	Training Time (s)	0.56	0.19	0.63	2.56	1.32
SVM	Recall (%)	96.65	91.67	68.97	99.21	86.08
	Sensitivity (%)	10.5	100	85.2	97.1	1.09
	Specificity (%)	98	100	18.9	99.4	98.84
	Predict (%)	60.67	100	58.36	98.21	86.19
	Training Time (s)	0.05	0.01	0.05	0.86	0.02
LDA	Recall (%)	73.91	83.33	75.14	99.21	86.97
	Sensitivity (%)	88.24	67.7	25.53	99.71	11.72
	Specificity (%)	16.79	75	75	98.94	96.4
	Predict (%)	58.3	70.83	55.26	99.44	82.88
	Training Time (s)	0.02	0.02	0.01	0.09	0.03
FALCON-AART	Recall (%)	66.22	91.67	76.86	100	82.69
	Sensitivity (%)	88.24	67.67	28.3	99.69	9.19
	Specificity (%)	10.53	100	86.71	98.76	92.38
	Predict (%)	56.06	83.33	62.31	99.0	80.17
	Training Time (s)	0.19	0.08	3	0.22	1.47
NB	Recall (%)	87.96	100	68.97	100	62.45
	Sensitivity (%)	66.8	100	90.8	95	38.19
	Specificity (%)	46.1	67.7	6.7	100	61.5
	Predict (%)	53.93	83.3	56.81	98.21	57.87
	Training Time (s)	0.02	0.02	0.02	0.02	0.02
HCL	Recall (%)	100	100	100	100	86.72
	Sensitivity (%)	75.51	100	50	96.08	6.25
	Specificity (%)	37.84	100	58.33	98.22	96.85
	Predict (%)	58.47	100	55.26	97.45	82.44
	Training Time (s)	76.02	8.5	8.45	0.86	8.39

Note: desirable values are bold. Detailed SVM results are shown in Table 7.4.

From Table 7.4, the HCL is good at both recall and generalization. HCL is able to attain 100% recall for most of the tasks, and it achieves relatively good performance for the testing results. HCL exhibits greater recall capability when compared against other methods (paired t-test: $\alpha = 0.05$, $t = 5.6$, $p < 0.001$). Moreover, HCL is also shown to have improved generalization performance (paired t-test: $\alpha = 0.05$, $t = 8.4$, $p < 0.001$). Both the prostate and pancreas cancers are comparatively difficult to diagnose, as they are considered rare diseases in contrast to ovarian and colorectal cancers. Thus, the systems do not perform well in those two tasks.

Table 7.4: Performance of SVM under different kernels.

Method		Prostate	Colorectal	Ovarian	Pancreas
SVM-Linear kernel	Recall (%)	73.91	91.67	98.82	68.97
	Sensitivity (%)	88.24	100	95	84.2
	Specificity (%)	15.79	100	100	18.9
	Predict (%)	57.3	100	98.21	57.36
	Training Time (s)	0.02	0.02	0.03	0.05
SVM-Polynomial Kernel	Recall (%)	71.74	83.33	99.21	77.59
	Sensitivity (%)	5.3	66.7	96.1	80.3
	Specificity (%)	88.2	75	99.4	17
	Predict (%)	52.81	70.83	98.21	54.26
	Training Time (s)	0.39	0.86	0.86	0.59
SVM-Radial Basis Function Kernel	Recall (%)	95.65	91.67	98.82	55.17
	Sensitivity (%)	10.5	100	91.7	100
	Specificity (%)	98	100	100	0
	Predict (%)	60.67	100	97.02	58.91
	Training Time (s)	0.05	0.01	0.03	0.03
SVM-Sigmoid Kernel	Recall (%)	56.52	91.67	100	55.17
	Sensitivity (%)	0	100	89.3	100
	Specificity (%)	100	100	98.1	0
	Predict (%)	57.3	100	91.76	58.91
	Training Time (s)	0.03	0.01	0.05	0.03

The training time of HCL is generally longer than most of the benchmarked techniques. This is because HCL loads the full set of data into the system. Another reason is that, HCL does not exploit parallel computing as it checks features in

sequential manner. The training time is affected by the number of features. The more features are used, the longer the training time.

One of the significant strengths HCL has is good system interpretability. HCL provides intuitive fuzzy rules to explain its operation. These rules can be used to validate and justify the system decision. When these rules are carefully validated by an expert, it can be used to guide inexperienced user. In contrast to black box CDSS such as MLP, a user can analyze HCL using fuzzy rules that they are familiar with. An example of the rules generated by HCL, FALCON-AART, and C4.5 are given in Table 7.5.

Table 7.5: Rules generated by HCL, FALCON-AART, and C4.5.

HCL (fuzzy rule)	FALCON-AART(fuzzy rule)	C4.5 (crisp rule)
IF gene1 is rather high, gene2 is very low, gene3 is medium, THEN Cancer.	IF gene1 is very high, gene2 is quite low, gene3 is medium, THEN Cancer.	IF gene3 > -16, THEN Cancer.
IF gene1 is quite high, gene2 is medium, gene3 is medium, THEN Normal.	IF gene1 is high, gene2 is very high, gene3 is very low, THEN Normal.	IF gene3 ≤ -16, THEN Normal.

Fuzzy rules have greater capacity than crisp rules because they can capture the uncertainty in the data. Apart from that, fuzzy rules allow the adoption of linguistic term, which encapsulate unnecessary details from the user. This facilitates the user in understanding the rules and system behavior. In addition, fuzzy rule has higher expressive power as it enables the employment of linguistic hedges such as “very”, and “rather”. Moreover, the application of linguistic terms allows user to analyze the system using familiar terms. It may seem that there is no difference between the rules generated by HCL and FALCON-AART. However, on closer examination, HCL has higher interpretability than FALCON-AART. The reasons are: (i) HCL employs strategies in ensuring the distinguishability of its antecedent (Figure 7.7). (ii) The

number of rules generated by HCL is generally lower than that of FALCON-AART, albeit it is still higher than C4.5. The number of rules generated by HCL, FALCON-AART and C4.5 are 10, 26, and 3 respectively.

As seen in Figure 7.7, HCL has greater interpretability than FALCON-AART without the interpretability checking system. It has clear and distinguishable rules antecedents. This eases the linguistic term assignment and reduces confusion during inference. On the contrary, it is difficult to assign proper linguistic term to FALCON-AART because of high degrees of overlapping. Thus, HCL has greater comprehensibility than FALCON-AART.

Figure 7.7: The antecedent fuzzy sets formed by HCL and FALCON-AART (dotted lines=negative rules).

7.5.2 HCL and Homeostasis

Abnormalities of hemostasis (stoppage of blood flow) and increased risk of *thrombosis* (presence of blood clot within blood vessel) have been reported in cancer patients [Gadducci97]. Thus, hemostatic parameters are potential indicators of disease free interval and long-term survival, as well as tumor growth and angiogenesis [Koh05]. This happens because tumor cells can activate systemic coagulation, induce tissue

factor production, and cause endothelial cells to express procoagulants. Consequently, tumor cells stimulate the extrinsic coagulation pathway [Koh05]. These hemostasis pathways are often altered in patient with different types of malignancies, especially for gynecologic cancer [Gadducci97].

Prognosis by means of hemostatic parameters is suitable for ovarian cancer because in advanced stage it shows hypercoagulation, increased platelets and improved fibrinolysis with further enhanced thrombin generation [Koh05]. Furthermore, ovarian tumor demonstrates pronounced aberration of the coagulation and fibrinolysis activation markers such as D-Dimer and fibrinogen levels

Hence, hemostasis is a promising prognostic tool to fight ovarian cancer. Unfortunately, high uncertainty exists when hemostatic assay is used alone. This is because blood samples have high risk of contamination, and the threshold values to determine whether one is normal or cancerous varies from individuals to individuals, and from populations to populations. In addition, some of the hemostatic assays are less specific [Gadducci97]. Realizing this, hemostatic assay is used with *computational intelligence* methods to provide accurate ovarian cancer prognosis.

The dataset is obtained from Singapore National University of Hospital (<http://www.nuh.edu.sg>), Department of Obstetrics and Gynecology. There are a total of 149 subjects divided into cancer and control groups. The control group includes the normal patients and patients with benign cyst (88 samples), whereas the cancer group contains patients of FIGO stages I-IV (61 samples).

The blood sampling was performed in the morning, where nine parts of blood from a clean venupuncture were mixed with a part of 0.129 mol/L trisodium citrate containing 0.21 mol/L hydroxylethylpiperazine-ethanesulfonic acid in cold plastic tube.

Subsequently, the blood was spun at 2000g for 15 minutes in a refrigerated centrifuge [Koh05]. The biological assays used are:

1. Plasma fibrinogen
2. D-Dimer
3. von Willebrand factor (vWF)
4. Antithrombin III complex (TAT)
5. Antithrombin III and plasminogen activities (ATIII)
6. Prothrombin Fragment 1+2 (F_{1+2})
7. Factor VII
8. Tissue factor pathway inhibitor (TFPI)

These assays were chosen because there are evidences that these assays are indicative of ovarian cancer or survival outcome [Koh05]. Evidence of tumor-associated activation of coagulation and fibrinolysis are indicated by plasma fibrin split products. One example of this is *D-Dimer* [Gadducci97]. D-Dimer is a good marker of ongoing fibrin formation and degradation in different disease and malignancies [Gadducci97].

Von Willebrand factor (vWF) is directly involved in the mechanism of platelet thrombus formation. Although it has no direct role in coagulation, it is required for normal hemostasis. Elevated levels of vWF may cause thrombotic complications [Koh05].

Both prothrombin fragment F_{1+2} and TAT are markers of *in vivo* hemostasis activation [Gadducci97]. TAT is generated by the reaction between thrombin and ATIII, whereas fragment F_{1+2} and the active thrombin are formed when prothrombin is cleaved — an important step in the coagulation cascade [Gadducci97].

Tissue factor that binds factor VII and its inhibitor TFPI are required to initialize blood coagulation by activating both factor X and factor IX. Hence, they are indicative of the hemostasis as well [Mackman06].

7 Hierarchical Complementary Learning - T. Z. TAN

The statistics of the data is summarized in Table 7.6. In Table 7.6, the difference between control and cancer groups is significant for most of the assays. For example, the age of the control group is significantly younger than the cancer group, and the fibrinogen level of the control group is significantly lower than that of the cancer group. In other words, performing prognosis using hemostasis can potentially achieve improved results.

Table 7.6: Statistics of hemostatic attributes.

Group		Age	Fi g/L	D-D ng/ml	vWF iu/ml	TAT μg/ml	ATIII iu/ml	F _{I+2} nmol/L	FVII %	TFPI ng/ml
Cancer	μ	51.6	5	214K	1.9	13.3	0.9	1.5	111.8	75.1
	σ	14.1	2	656K	1.1	13.9	0.2	0.9	33.9	48.4
	R	16-81	1.2-10.4	96K-486K	0.4-5.4	2-68	0.5-1.3	0.11-5	43-236	20-352
Benign	μ	41.7	3.8	788	1.3	6.8	1	1.1	103.2	65
	σ	14	1.3	1855	0.6	9.9	0.2	1.1	27.6	120
	R	16-74	1.7-7.3	50-153	0-3.5	1-48	0.5-1.3	0.3-6.6	33-166	0-999
p-value		<0.001	<0.001	<0.02	<0.001	<0.01	<0.01	0.02	0.1	>0.1

μ: Mean; σ: Standard deviation; R: Range; Fi: Fibrinogen; D-D: D-Dimer; FVII: Factor VII

The ten-fold cross-validation performance of ovarian cancer prognosis is summarized in Table 7.7. From Table 7.7, the HCL outperforms others in ovarian cancer prognosis, attaining high sensitivity and specificity. HCL demonstrates competent recall and generalization abilities. This is because HCL considers positive and negative knowledge, as well as the difference of positive and negative knowledge in its decision making. This also suggests that there are significant differences between positive and negative samples, affirming the information shown in Table 7.6. The training duration of HCL is longer than the others, because it analyzes the data feature by feature sequentially. This does not necessarily imply that the complexity of HCL is higher than the others.

7 Hierarchical Complementary Learning - T. Z. TAN

Table 7.7: Performance on ovarian cancer prognosis.

Method	Recall (%)	Sensitivity (%)	Specificity (%)	Predict (%)	Training time (s)
C4.5	88.89	79.4	9.1	62.22	0.05
MLP	91.11	88.2	45.5	77.78	0.44
Naïve Bayesian	75.56	67.6	54.5	64.44	0.01
LDA	75.56	80.55	10	65.22	0.03
HCL	99.6	90	65	77.5	2.6

One of HCL significant strengths is that, it provides fuzzy rules to explain its computation. In contrast to MLP, LDA, and Naïve Bayesian, HCL offers means for user to understand the system. These fuzzy rules allow user to validate the system using their familiar terms. In comparison to the crisp rule generated by decision tree (C4.5), fuzzy rule is able to capture uncertainties, and hence allows a more stable solution to be given. In other words, a small perturbation in data would not cause a totally different HCL. In this task, the averaged number of rules generated by HCL and C4.5 are five and four respectively, suggesting that the rulebase of both are compact. Table 7.8 displays the example of the positive and negative rules generated by HCL and C4.5.

Table 7.8: Rules generated for ovarian cancer prognosis.

	HCL (Fuzzy rule)	C4.5 (Crisp rule)
Positive	IF <i>ATIII</i> is low, <i>D-Dimer</i> is quite high, <i>Factor VII</i> is very low, THEN die within a year.	IF $vWF > 2.36$, <i>Factor VII</i> > 113 , <i>ATIII</i> ≤ 0.81 , THEN die within a year.
Negative	IF <i>ATIII</i> is medium, <i>D-Dimer</i> is medium, <i>Factor VII</i> is quite low, THEN survive.	IF $vWF > 2.36$, <i>Factor VII</i> > 113 , <i>ATIII</i> > 0.81 , THEN survive.

As shown in Table 7.8, the HCL rule is as intuitive as the rules provided by decision tree. Yet another merit of fuzzy rule is that it allows encapsulation of unnecessary details from the user. Apart from that, fuzzy rule generated by HCL is more expressive, as it allows the use of linguistic hedges such as “very”, “quite”, etc. These are described by the fuzzy sets autonomously constructed by HCL during

7 Hierarchical Complementary Learning - T. Z. TAN

learning. An example is given in Figure 7.8, illustrating the fact that the fuzzy sets generated by HCL are highly distinguishable. High distinguishability is important because it enables the unambiguous assignment of linguistic terms, which improves the interpretability of the system.

In addition, the rules generated by HCL can be used for concept validation. Table 7.8 shows that when ATIII decreases, D-Dimer increases, and Factor VII is low, the patient would likely not survive after the first 12 months. This supports the findings that, significant associations exist between preoperative hemostatic levels of ATIII, D-Dimer and survival outcome by 12 months [Koh05]. The rules also affirms that elevated D-Dimer levels were seen in those who died within 13 months from the disease, and a decreased ATIII level was seen in those patients who died at 24 months compared to patients who are still alive [Koh05].

Figure 7.8: Membership function of D-Dimer (Dotted lines are negative rules, solid lines are positive rules).

7.6 Summary

HCL functionally models the human neural mechanism of pattern recognition, as well as the hierarchical category learning. By implementing the complementary learning, the features of positive and negative classes, as well as the lateral inhibition between the two, have produced superior recognition performance. Furthermore, hierarchical learning is memory efficient as it allows better presentation of knowledge. It also employs the divide-and-conquer strategy. Since HCL is the confluence of the complementary and hierarchical category learning, HCL inherits their respective characteristics. It is also equipped with most of the desirable characteristics of a pattern recognition tool and these traits have been extensively verified through the experimental results.

Part IV:

Conclusions

and

Future Work

Chapter 8 Conclusions and Future Work

8.1 Conclusions

Computational Intelligence (CI) system is a promising research area as it brings forth fruitful result in many applications. In general, the application of CI can be categorized into automation and decision support of certain tasks. However, the adoption of CI system into practice is few, especially for *Clinical Decision Support System* (CDSS). This arises because of a few issues, but the main reason lies in the psychology. The poor acceptance is because most systems lack human-like reasoning and easily understandable operation. Moreover, some systems provide no means for the user to validate and to understand their computations. Consequently, users are reluctant to trust the decision suggested by these systems. Thus, in order to improve the user acceptance, a system must provide high interpretability and tractability, and at the same time possesses inference process that is similar to that of the user. Besides, the CI systems have to be accurate in the tasks that they are supporting. Moreover, as these CI systems generate their knowledge based on the target class (positive class) only, the ignorance of the contribution from the non-target class (negative class) may cause poor performance. On the other hand, the advancement in high-throughput microbiology technology such as *DNA microarray* and *proteomics* posted the high-dimensionality challenge to these CI systems. Since microarray and proteomics show increasing promise in various tasks, a competent CI tool should offer support in these new technologies as well.

Thus, *Complementary Learning Fuzzy Neural Network* (CLFNN) [Tan06a][Tan05a] is proposed. CLFNN is a biological brain-inspired FNN that functionally models the pattern recognition mechanism in human brain. Since many

8 Conclusions and Future Work - T. Z. TAN

tasks require pattern recognition, and the fact that human is effective in recognition, modeling the complementary learning mechanism may offer superior performance [Kasabov07]. In contrast to common CI system, which considers only the characteristics of positive/negative class, CLFNN learns the features of both positive and negative classes, as well as the lateral inhibition between the two. Since CLFNN exploits the discriminating relationship between the positive and negative samples, CLFNN could potentially gives better performance. Viewing from the other end, CLFNN gives a closer representation to the problem because CLFNN constructs positive and negative rules explicitly to describe the problem, and no assumption is made to the problem space.

Three CLFNNs have been developed based on the functional model of complementary learning at different levels:

- *Fuzzy Adaptive Learning Control with Another Adaptive Resonance Theory* (FALCON-AART) [Tan05b] functionally models the interleaved complementary learning in hippocampus.
- *Genetic Complementary Learning* (GCL) [Tan06e][Tan05c] mimics the complementary gene selection process.
- *Hierarchical Complementary Learning* (HCL) [Tan07a] imitates the human hierarchical category learning process.

In general, CLFNN has good recognition and generalization. In most of the medical and bioinformatics tasks in this thesis, CLFNN demonstrates good pattern recognition capability, being interpretable, and possesses human-like reasoning. Comparatively, CLFNN outperforms positive/negative and neutral systems. Apart from that, CLFNN is equipped not only with rule provision, but also rule system that

ensures the interpretability criteria such as distinguishability, normality, coverage, and complexity.

Unfortunately, the CLFNN is not without shortcomings. Since CLFNN specifically generates positive and negative rules, in general, it has relatively larger rulebase. This shortcoming can be alleviated using the strategy of ensuring interpretability to minimize the number of rules. Likewise, each of the three CLFNNs proposed has its unique advantages and limitations. For instance, FALCON-AART is biologically less plausible, requires additional supervisory system to ensure interpretability. GCL is stochastic, and does not guarantee optimal solution [Kasabov07]. HCL does not take into account of the relation between the features, and hence, the information between the relationships of features is lost. Yet, with these limitations, the CLFNN satisfies the requirements of high accuracy and interpretability.

8.2 Summary of Contributions

The solutions proposed for the research issue described are summarized in Table 8.1.

Table 8.1: The issues and their corresponding proposed solutions.

Research Issue		Solution Proposed	Chapter
Accuracy	Pattern Recognition	FALCON-AART (FA)	Chapter 5
	Randomness, probability	Genetic Complementary Learning	Chapter 6
	Imbalanced data distribution	Complementary Learning	Chapter 4
Transparency	Interpretability	Rule Supervisory system or integrated hierarchical learning.	Chapters 5, 6, 7
	Tractability	Mapping of Fuzzy inference scheme, human-like reasoning, tractability criteria proposed, and biological plausible.	Chapters 3, 5, 6, 7
High Dimensionality		Hierarchical Complementary Learning	Chapter 7

8.2.1 Major Contribution

8.2.1.1 Accuracy

Most of the CI tools belong to either positive/negative learning or neutral learning. Complementary learning learns from positive and negative samples, as well as the lateral inhibition between the two. As complementary learning considers the information conveyed by the negative class and the lateral inhibiting relationship between the positive and negative classes, improved performance may be offered. Furthermore, Complementary learning has the architecture that separates the positive and negative knowledge, which minimizes the interference and eases the inference process.

Complementary learning is a functional model of human pattern recognition. The biological support of complementary learning is established and formalized (for details, please refer to Chapter 4). The CLFNN is then formalized such that no ambiguity is found in the description, and such that distinction can be made between this novel class of system and the other CI techniques. The psychological or biological category complementary learning paradigm belongs to is given in a form of taxonomy in Chapter 4. Its characteristics and properties are analyzed in order to demonstrate the capability of complementary learning. The enhanced performance of complementary learning on imbalanced dataset is also given in Chapter 4, as one of the desirable characteristics of CLFNN.

8.2.1.2 Transparency

The conventional CI tool does not cater to the psychological needs of the user. As discussed in Chapter 3, most of the systems do not provide the facility of explanation, and lack the reasoning process that the user can relate with. Furthermore, they are

based on mathematical logic and probabilistic inference, which can be difficult to understand. In response to these, a functional model of human pattern recognitions using complementary learning is proposed. Functional modeling of pattern recognition not only gives the credibility and biological plausibility to the system, but also gives a reasoning process that the user can relate with. As such, more common ground between the user and the system is built. Consequently, user may find it easier to trust the system. The support and the description of the functional model are given in Chapter 4.

The explicit generation of positive and negative knowledge gives better modeling of the problem. As shown in Chapter 4, complementary learning gives a closer representation of the problem. It also exhibits greater scalability to include new class categorization in its rulebase.

In order to improve the tractability of the system, tractability criteria are proposed in Chapter 3. Fuzzy inference scheme is mapped onto the CLFNN so as to formalize the operation of the system, and to produce human-like reasoning. As such, the user can analyze the system in his or her familiar term, using his or her own behavior as reference. The formalized inference process facilitates the tracking of the system as well.

8.2.1.3 High Dimensionality

Human being is complementary in nature, and the complementary learning can be observed at different levels. The existence of multiple forms of complementary learning is probably for handling different tasks and information. Hence, different forms of complementary learning systems are developed for data of different nature. From chapter 5, FALCON-AART is developed based on human hippocampal learning. In chapter 6, a complementary learning based on the complementary gene selection

process, GCL is developed. GCL inherits the randomness from the gene selection process. Thus, it is stochastic in nature, and has the potential of offering optimal solution. The hierarchical complementary learning that draws inspiration from human hierarchical category learning is discussed in Chapter 7. The hierarchical nature of the system allows it to handle high dimensional data.

8.2.2 Minor Contribution

Since interpretability is a very important factor determining user acceptance, the interpretability and tractability of the proposed systems are ensured. The interpretability is ensured by implementing a supervisory system to check the knowledge constructed. This can be seen in FALCON-AART and GCL.

8.3 Future Work

The performance of the CI system depends on the data structure and distribution. There is no panacea that works for all kinds of data. Perhaps that is the reason why there are various forms of complementary learning in human. For example, the complementary hippocampal learning is at the brain structure level, the complementary gene selection is at the genetic level, and the hierarchical category learning is viewed from the psychological level. These complementary learnings have common traits of positive and negative knowledge segregation, inference based on the features of positive and negative samples, as well as the lateral inhibition of positive and negative samples. There are also differences among the different complementary learnings. The differences are to cater to different kinds of data or situation. Each of the complementary learnings has its strengths and shortcomings. Thus, exploring complementary learning at human body systems and cellular levels, may boost the system accuracy in pattern recognition further.

One of the complementary learning systems to be investigated is the immune system. The human immune system uses complementary learning in recognizing antigens that invade the human body. The antibody (T cell) decides to bind to the antigen or not based on the *self* and *non-self recognition* [Leng02], which can be viewed as a positive (non-self) and negative (self) recognition. It is found that the self and non-self recognition are fuzzy. Hence, FNN is suitable to model this mechanism. Apart from that, human immune system is known to exhibit *double plasticity* [Bersini99]. The immune network is a network of antibody where each existing unit can alter its concentration (parameter plasticity), as well as simply appear or disappear (structural plasticity). Thus, the operation is quite similar to the structural and parameter learnings of FALCON-AART. When antigen enters the human body, antibody is recruited to bind the antigen. New set of antibody is produced to cater to new set of antigen. This happens autonomously and alters the immune network structure. Antibody responsible to bind to an antigen does not bind to other types of antigen. This is analogous to the lateral inhibition of positive and negative knowledge, where “positive” antibody responds to “positive” antigen only, and vice versa. These exclusive characteristics of complementary immune system will be researched in future.

On the other hand, one reason why human is effective in problem solving, apart from the capability to learn, is the ability to make decision. Thus, functional model of the human decision making has the potential of yielding superior performance. The decision making process of human being is complementary. Similar to complementary learning, complementary decision making [Tan07d] process can be observed at psychological and biological levels. In complementary decision making, positive and negative decision making are the decision that leads to positive and

negative outcomes. The evidence of psychological complementary decision making can be seen from the psychology study of the affective effect, where the people maintain a positive and negative pool of *markers* [Slovic05]. The decision making is based on the signals of these markers. On the other hand, the biological evidence is found from the neuroscience studies, particularly the *visual-saccadic decision making* studies [Glimcher03]. These studies show that in various discrimination tasks, there are positive and negative neurons that respond favorably to positive and negative stimuli (reward and punishment) respectively. Whenever a positive stimulus is presented, positive neurons are activated, and negative neurons are inhibited concurrently. Complementary decision making therefore, exhibits the traits of complementary learning. By functionally modeling the complementary decision making, promising result may be offered.

Alternatively, the complementary learning in emotion will be investigated. Emotion is complementary. This can be seen from the neuroscience findings that the different human emotions excite distinct brain activation patterns [Damasio00]. In other words, when “happy” is supposed to be felt, the brain activation patterns of happiness is activated, whereas the brain activation patterns of sadness, anger, etc is deactivated. In addition, the amygdala, a structure often implicated in emotion, has dedicated neurons for reward (positive) and punishment (negative) signals. These findings indicate that human emotional learning is complementary as well. Since emotional learning is often associated with survival, functionally model this type of complementary learning may produce superior performance.

Different forms of complementary learning have their own strengths and weaknesses. This inspires the development of a complementary learning ensemble system. The ensemble incorporates all the different complementary learning systems

proposed. The major concern in ensemble development is the heterogeneity between the models, and the voting scheme. However, as each of the complementary learning system is based on different functional models, the heterogeneity of the ensemble is ensured. As for voting scheme, it can either use simple majority voting scheme, or it can be built based on the biological interaction between these complementary learning systems. The later approach is preferred as it gives a greater credibility.

Aside from that, the future work of CLFNN includes the assessment or modification of CLFNN as a tool for chance discovery. This is particularly useful for the application in medicine because of its high uncertainties and high level of risk. Since CLFNN has the capacity of good recognition performance and being interpretable, CLFNN can be a promising chance discovery tool [Tan07b]. Besides, chance can be categorized as opportunity (positive) and risk (negative). Thus, CLFNN has the advantage of giving interpretations that are closer to the context of chance discovery. Moreover, the fuzzy rules autonomously constructed can be used to identify potential chances. On the other hand, the application of CLFNN with chance discovery gives the system an explanation in terms of probability and chances, which could be useful for improving the transparency of the system. This will be investigated in future.

Another way of improving CLFNN includes using local entropy measure to ensure the distinguishability of the rules [Zhou06b]. The CLFNN of this work makes use of similarity measure to ensure the distinguishability of the rules, which requires high computational load. The local entropy measure can be integrated with CLFNN learning, and hence avoids the need of supervisory system to ensure the distinguishability of the rules. This enhances the system efficiency.

In summary, CLFNN shows promising role in CDSS and other application areas. It is found to be a competent business decision support system as well [Tan06e][Tan05c]. The present system however has relatively larger rulebase than its counterparts. Thus, one future work is to improve the system from this perspective, on top of building the new CLFNN.

References

- [Abel98] T. Abel, E. Kandel, "Positive and negative regulatory mechanisms that mediate long-term memory storage", *Brain Research Reviews*, **26**, 360-378, 1998.
- [ACS07] American Cancer Society, *Cancer facts & figures 2007*. Atlanta: American Cancer Society, Inc. 2007.
- [Adankon07] M. M. Adankon, M. Cheriet, "Optimizing resources in model selection for support vector machine", *Pattern Recognition*, **40**, 953-963, 2007.
- [Akgöbek06] Ö. Akgöbek, Y. S. Aydin, E. Öztemel, M. S. Aksoy, "A new algorithm for automatic knowledge acquisition in inductive learning", *Knowledge-based Systems*, **19**(6), 388-395, 2006.
- [Agrawal98] R. Agrawal, J. Gehrke, D. Gunopulos, P. Raghavan, "Automatic subspace clustering of high dimensional data for data mining application", *Proceedings of SIGMOD'98*, 94-105, 1998.
- [Aizenstein00] H. J. Aizenstein, A. W. MacDonald, V. A. Stenger, R. D. Nebes, J. K. Larson, S. Ursu, C. S. Carter, "Complementary category learning systems identified using event-related functional MRI", *Journal of Cognitive Neuroscience*, **12**(6), 977-987, 2000.
- [Alain01] C. Alain, S. Arnott, S. Hevenor, S. Graham, C. Grady, "What and where in the human auditory system", *Proceedings of the National Academy of Sciences*, **98**(21), 12301-12306, 2001.
- [Amalu03] W. C. Amalu, "A review of breast thermography", International Academy of Clinical Thermology, 2003. Available: <http://www.iact-org.org/articles/articles-review-btherm.html>.
- [An03] A. An, "Learning classification rules from data", *Computers and Mathematics with Applications*, **45**, 737-748, 2003.
- [Anderson07] J. G. Anderson, "Social, ethical and legal barriers to E-health", *International Journal of Medical Informatics*, in press, 2007.
- [Angelini08] L. Angelini, D. Marinazzo, M. Pellicoro, S. Stramaglia, "Semi-supervised learning by search of optimal target vector", *Pattern Recognition Letters*, **29**, 34-39, 2008.
- [Baak03] P. A. Baak, F. R. C. Path, M. A. J. A. Hermsen, G. Meijer, J. Schmidt, E. A. M. Janssen, "Genomics and proteomics in cancer", *European Journal of Cancer*, **39**(9), 1199-1215, 2003.
- [Bagirov01] A. Bagirov, A. Rubinov, J. Yearwood, A. Stranieri, "A global optimization approach to classification in medical diagnosis and prognosis", *Proceedings of the 34th Annual Hawaii International Conference on System Sciences*, 1-9, 2001.
- [Balas01] C. Balas, "A novel optical imaging method for the early detection, quantitative grading, and mapping of cancerous and precancerous

- lesions of cervix", *IEEE Transactions on Biomedical Engineering*, **48**(1), 96-104, 2001.
- [Banich00] M. T. Banich, M. P. Milham, R. Atchley, N. J. Cohen, A. Webb, T. Wszalek, A. F. Kramer, Z. P. Liang, A. Wright, J. Shenker, R. Magin, "fMRI studies of Stroop tasks reveal unique roles of anterior and posterior brain systems in attentional selection", *Journal of Cognitive Neuroscience*, **12**(6), 988-1000, 2000.
- [Baraldi02] A. Baraldi, E. Alpaydin, "Constructive feedforward ART clustering networks-part 2", *IEEE Transactions on Neural Networks*, **13**, 622-677, 2002.
- [Baseler02] H. A. Baseler, A. A. Brewer, L. T. Sharpe, A. B. Morland, H. Jägle, B. A. Wandell, "Reorganization of human cortical maps caused by inherited photoreceptor abnormalities", *Nature Neuroscience*, **5**(4), 364-370, 2002.
- [Bayer83] S. A. Bayer, "3H-thymidine-radiographic studies of neurogenesis in the rat olfactory bulb", *Experimental Brain Research*, **50**, 329-340, 1983.
- [Beason-Held05] L. L. Beason-Held, S. Golski, M. A. Kraut, G. Esposito, S. M. Resnick, "Brain activation during encoding and recognition of verbal and figural information in older adult", *Neurobiology of Aging*, **26**(2), 237-250, 2005.
- [Berlin06] A. Berlin, M. Sorani, I. Sim, "A taxonomic description of computer-based clinical decision support systems", *Journal of Biomedical Informatics*, **39**, 656-667, 2006.
- [Bersini99] H. Bersini, "The endogenous double plasticity of the immune network and the inspiration to be drawn for engineering artifacts", in D. Dasgupta (Eds.), *Artificial immune systems and their applications*, Heidelberg: Springer-Verlag, 1999.
- [Bhanu02] B. Bhanu, A. Dong, "Concepts learning with fuzzy clustering and relevance feedback", *Engineering Applications of Artificial Intelligence*, **15**, 123-138, 2002.
- [Bhaskar06] J. Bhaskar, D. C. Hoyle, Sameer Singh, "Machine learning in bioinformatics: a brief survey and recommendations for practitioners", *Computers in Biology and Medicine*, **36**, 1104-1125, 2006.
- [Blake98] C. L. Blake, C. J. Merz, UCI Repository of machine learning databases. Irvine, CA: University of California, Department of Information and Computer Science, 1998.
- Available: <http://www.ics.uci.edu/~mllearn/MLRepository.html>.
- [Blonder04] L. X. Blonder, C. D. Smith, C. E. Davis, M. L. Kesler/West, T. F. Garrity, M. J. Alvison, A. H. Andersen, "Regional brain response to faces of humans and dogs", *Cognitive Brain Research*, **20**, 384-394, 2004.

-
- [Bodegård01] A. Bodegård, S. Geyer, C. Grefkes, K. Zilles, P. E. Roland, "Hierarchical processing of tactile shape in the human brain", *Neuron*, **31**, 317-328, 2001.
- [Bukach04] C. M. Bukach, D. N. Bub, M. E. J. Masson, D. S. Lindsay, "Category specificity in normal episodic learning: applications to object recognition and category-specific agnosia", *Cognitive Psychology*, **48**(1), 1-46, 2004.
- [Buntine91] W. Buntine, R. Caruana, "Introduction to induction and recursive partitioning", Technical Report FIA-91-28, NASA, Ames Research Center, 1991.
- [Byriel99] J. Byriel, "Neuro-fuzzy classification of cells in cervical smears", Master of Science Thesis, Department of Automation, Technical University of Denmark, 1999.
- [Call03] J. Call, "Beyond learning fixed rules and social cues: abstraction in the social arena", *Philosophical Transactions of the Royal Society of London. Section B: Biological Science*, **358**(1435), 1189-1196, 2003.
- [Camastra05] F. Camastra, A. Verri, "A novel kernel method for clustering", *IEEE Transactions on Pattern Analysis and Machine Intelligence*, **27**(5), 801-805, 2005.
- [Carter05] A. J. R. Carter, J. Hermisson, T. F. Hansen, "The role of epistatic gene interactions in the response to selection and the evolution of evolvability", *Theoretical Population Biology*, **68**, 179-196, 2005.
- [Casillas03] J. Casillas, O. Cordon, F. Herrera, "Interpretability improvements to find the balance interpretability - accuracy in fuzzy modeling: an overview", in J. Casillas, O. Cordon, F. Herrera, L. Magdalena (Eds.), *Accuracy improvements in linguistic fuzzy modeling*, Studies in Fuzziness and Soft Computing, Heidelberg: Springer-Verlag, 27-45, 2003.
- [Cendrowska87] J. Cendrowska, "PRISM: An algorithm for inducing modular rules", *International Journal of Man-Machine Studies*, **27**, 349-370, 1987.
- [Cestnik87] G. Cestnik, I. Kononenko, I. Bratko, "Assistant-86: a knowledge-elicitation tool for sophisticated users", in I. Bratko and N. Lavrac (Eds.), *Progress in Machine Learning*, Sigma Press, 31-45, 1987.
- [Chang04] X. Chang, J. H. Lilly, "Evolutionary design of a fuzzy classifier from data", *IEEE Transactions on Systems, Man, and Cybernetics. Part B: Cybernetics*, **34**(4), 1894-1906, 2004.
- [Chatterjee04] A. Chatterjee, A. Rakshit, "Influential rule search scheme (IRSS)—a new fuzzy pattern classifier", *IEEE Transactions on Knowledge and Data Engineering*, **16**(8), 881-893, 2004.
- [Cherkassky06] V. Cherkassky, V. Krasnopolsky, D. P. Solomatine, J. Valdes, "Computational intelligence in earth sciences and environmental applications: issues and challenges", *Neural Networks*, **19**, 113-121, 2006.

-
- [Chialvo99] D. R. Chialvo, P. Bak, "Learning from mistake", *Neuroscience*, **90**(4), 1137–1148, 1999.
 - [Chu03] S. –C. Chu, J. –W. Teng, T. –C. Hsiao, Z. –Y. Chen, J. –C. Yuan, H. K. Chiang, "Implementation of autofluorescent excitation-emission matrices for differentiation of cervical tissue", *The 5th Pacific Rim Conference on Lasers and Electro-Optics*, **2**, 591, 2003.
 - [CIPME07] Committee on Identifying and Preventing Medication Error, P. Aspden, J. Wolcott, J. L. Bootman, L. R. Cronenwett (Eds.), *Preventing medication errors: quality chasm series*, US: National Academies Press, 2007. ISBN: 0-309-10147-6.
 - [Clark89] P. Clark, T. Niblett, "The CN2 induction algorithm", *Machine Learning*, **3**, 261-283, 1989.
 - [Clark04] L. Clark, R. Cools, T. W. Robbins, "The neuropsychology of ventral prefrontal cortex: decision-making and reversal learning", *Brain and Cognition*, **55**, 41-53, 2004.
 - [Collobert06] R. Collobert, F. Sinz, J. Weston, L. Bottou, "Large scale transductive SVMs", *Journal of Machine Learning Research*, **7**, 1687-1712, 2006.
 - [Combs98] W. E. Combs, J. E. Andrews, "Combinatorial rule explosion eliminated by a fuzzy rule configuration", *IEEE Transactions on Fuzzy Systems*, **6**(1), 1-11, 1998.
 - [Conrad04] C. D. Conrad, D. D. MacMillan, S. Tsekhanov, R. L. Wright, S. E. Baran, R. A. Fuchs, "Influence of chronic corticosterone and glucocorticoid receptor antagonism in the Amygdala on fear conditioning", *Neurobiology of Learning and Memory*, **81**, 185–199, 2004.
 - [Cooper95] L. N. Cooper (Eds), *How we learn: how we remember: towards an understanding of brain and neural systems: selected papers of Leon N. Cooper*, Singapore: World Scientific, 1995.
 - [Courtney98] S. M. Courtney, L. Petit, J. M. Maisog, L. G. Ungerleider, J. V. Haxby, "An area specialized for spatial working memory in human frontal cortex", *Science*, **279**, 1347-1351, 1998.
 - [Damasio00] A. R. Damasio, T. J. Grabowksi, A. Bechara, H. Damasio, L. L. B. Ponto, H. Parvizi, R. D. Hichwa, "Subcortical and cortical brain activity during the feeling of self-generated emotions", *Nature Neuroscience*, **3**(10), 1049-1057, 2000.
 - [Davis86] D. D. Davis, "The uniqueness of human learning", *Medical Hypotheses*, **21**(2), 153-164, 1986.
 - [De Lillo04] C. De Lillo, "Imposing structure on a Corsi-type task: evidence for hierarchical organization based on spatial proximity in serial-spatial memory", *Brain and Cognition*, **55**, 415-426, 2004.

-
- [De Jong93] K. A. De Jong, W. M. Spears, D. F. Gordon, "Using genetic algorithms for concept learning", *Machine Learning*, **16**, 161-188, 1993.
- [Deogun97] J. S. Deogun, V. V. Raghavan, A. Sarkar, H. Sever, "Data mining: trends in research and development", in T. Y. Lin, N. Cercone (Eds), *Rough sets and data mining, analysis of imprecise data*, Massachusetts: Kluwer Academic Publishers, 9-45, 1997.
- [Destrebecqz03] A. Destrebecqz, P. Peigneux, S. Laureys, C. Degueldre, G. Del Fiore, J. Aerts, A. Luxen, M. van der Linden, A. Cleeremans, P. Maquet, "Cerebral correlates of explicit sequence learning", *Cognitive Brain Research*, **16**, 391-398, 2003.
- [Deutsch04] T. Deutsch, T. Gergely, V. Trunov, "A computer system for interpreting blood glucose data", *Computer Methods and Programs in Biomedicine*, **76**, 41-51, 2004.
- [Dinka06] D. Dinka, J. M. Nyce, T. Timpka, "The need for transparency and rationale in automated systems", *Interacting with Computers*, **18**, 1070-1083, 2006.
- [Duchaine01] B. Duchaine, L. Cosmides, J. Tooby, "Evolutionary psychology and the brain", *Current Opinion in Neurobiology*, **11**, 225-230, 2001.
- [Duchenne06] O. Duchenne, J. -Y. Audibert, R. Keriven, "Transductive segmentation", CERTIS Research Report 06-27, Willow Technical Report, 2006.
- [Duval85] L. D. Duval, R. A. Wagner, Y. Han, D. W. Loveland, "Finding test-and-treatment procedures using parallel computation", Technical paper, ADA162141, Duke University Durham NC Department of Computer Science, 1985.
- [Economou00] G. P. K. Economou, "GEORGIA: an overview", in P. J. G. Lisboa, E. C. Ifeachor, P. S. Szczepaniak (Eds.), *Artificial neural networks in biomedicine*, London: Springer-Verlag, 2000.
- [Eichenbaum00] H. Eichenbaum, "A cortical-hippocampal system for declarative memory", *Nature Review Neuroscience*, **1**, 41-50, 2000.
- [Eistein95] G. O. Einstein, M. A. McDaniel, S. L. Richardson, M. J. Guynn, A. R. Cunfer, "Aging and prospective memory: examining the influences of self-initiated retrieval processes", *Journal of Experimental Psychology: Learning, Memory, and Cognition*, **21**(4), 996-1007, 1995.
- [El-Yaniv05] R. El-Yaniv, L. Gerzon, "Effective transductive learning via objective model selection", *Pattern Recognition Letters*, **26**, 2104-2115, 2005.
- [Evans05] C. Evans, "Clinical decision making theories: patient assessment in A & E", *Emergency Nurse*, **13**(5), 16-19, 2005.
- [Ezziane06] Z. Ezziane, "Applications of artificial intelligence in bioinformatics: a review", *Expert Systems with Applications*, **30**(1), 2-10, 2006.

-
- [Farukhi04] F. Farukhi, "Clinical decision support systems", in D. Neuhauser, *Public Health Management & Policy*, Case Western Reserve University, 2004.
- [Fay01] J. C. Fay, G. J. Wychkoff, C. -I. Wu, "Positive and negative selection on the human genome", *Genetics*, **158**, 1227-1234, 2001.
- [Fernandez04] F. Fernandez, P. Isasi, "Evolutionary design of nearest prototype classifiers", *Journal of Heuristics*, **10**(4), 431-454, 2004.
- [Fieschi03] M. Fieschi, J. C. Dufour, P. Staccini, J. Gouvernet, O. Bouhaddou, "Medical decision support systems: old dilemmas, and new paradigms?" *Methods of Information in Medicine*, **42**(3), 190-198, 2003.
- [Fisher36] R. A. Fisher, "The use of multiple measurements in taxonomic problem", *Annals of Eugenics*, **7**(2), 179-188, 1936.
- [Fletcher93] S. W. Fletcher, W. Black, R. Harris, B. K. Rimer, S. Shapiro, "Report of the international workshop on screening for breast cancer", *Journal of the National Cancer Institute*, **85**, 1644-1656, 1993.
- [Flood98] J. F. Flood, J. E. Morley, "Learning and memory in the SAMP Mouse", *Neuroscience and Biobehavioral Reviews*, **22**(1), 1-20, 1998.
- [Fountain95] S. B. Fountain, J. D. Rowan, "Coding of hierarchical versus linear pattern structure in rats and humans", *Journal of Experimental Psychology: Animal Behaviour Processes*, **21**(3), 187-202, 1995.
- [Fry05] A. J. Fry, J. J. Wernegreen, "The roles of positive and negative selection in the molecular evolution of insect endosymbionts", *Gene*, **355**, 1-10, 2005.
- [Furuhashi97] T. Furuhashi, "Development of if-then rules with the use of DNA coding", in T. Furuhashi, *Fuzzy Evolutionary Computation*, 107-125, 1997.
- [Furuhashi02] T. Furuhashi, "An interpretability of fuzzy models", *Advances in Soft Computing, AFSS 2002*, Berlin: Springer, **2275**, 205-212, 2002.
- [Gadducci97] A. Gadducci, R. Marrai, U. Baicchi, M. Ferdeghini, A. Fanucchi, C. Weiss, A. R. Genazzani, "Preoperative D-Dimer plasma assay is not a predictor of clinical outcome for patients with advanced ovarian cancer", *Gynecologic Oncology*, **66**, 85-88, 1997.
- [Gan03] W. B. Gan, E. Kwan, G. Feng, J. R. Sanes, J. W. Lichtman, "Synaptic dynamism measure over minutes to months: age-dependent decline in an autonomic ganglion", *Nature Neuroscience*, **6**(9), 956-961, 2003.
- [Gauthier00] I. Gauthier, P. Skudlarski, J. C. Gore, A. W. Anderson, "Expertise for cars and birds recruits brain areas involved in face recognition", *Nature Neuroscience*, **3**(2), 191-197, 2000.
- [Gazzaniga00] M. S. Gazzaniga (Eds), *The new cognitive neurosciences*, 2nd Ed, Massachusetts Institute of Technology, 2000.

-
- [George05] D. George, J. Hawkins, "A hierarchical Bayesian model of invariant pattern recognition in the visual cortex", *Proceedings of International Joint Conference on Neural Network*, 2005.
- [Giard92] R. W. M. Giard, J. Hermans, "The value of aspiration cytologic examination of the breast. A statistical review of the medical literature", *Cancer*, **69**, 2104-2110, 1992.
- [Glimcher03] P. W. Glimcher, "The neurobiology of visual-saccadic decision making", *Annual Review of Neuroscience*, **26**, 133-179, 2003.
- [Gobet01] F. Gobet, P. C. R. Lane, S. Croker, P. C. -H. Cheng, G. Jones, L. Oliver, J. M. Pine, "Chunking mechanism in human learning", *Trends in Cognitive Science*, **5**(6), 236-243, 2001.
- [Gould99a] E. Gould, A. J. Reeves, M. S. A. Graziano, C. G. Gross, "Neurogenesis in the neocortex of adult primates", *Science*, **286**, 548-552, 1999.
- [Gould99b] E. Gould, A. J. Reeves, M. Fallah, P. Tanapat, C. G. Gross, E. Fuchs, "Hippocampal neurogenesis in old world primates", *Proceedings of the National Academy of Sciences of the United States of America*, **96**(9), 5263-5267, 1999.
- [Gross00] C. G. Gross, "Neurogenesis in the adult brain: death of dogma", *Nature Reviews*, **1**, 67-73, 2000.
- [Grossberg01] S. Grossberg, "The complementary brain", in G. Dorffner, H. Bischof, K. Hornik (Eds.), *ICANN*, LNCS 2130, Heidelberg: Springer-Verlag, 3-4, 2001.
- [Haney01] J. Haney, K. Lukowiak, "Context learning and the effect of context on memory retrieval in *Lymnaea*", *Learning and Memory*, **8**(1), 35-43, 2001.
- [Hart83] L. A. Hart, *Human brain and human learning*, New York: Longman, 1983.
- [Hester04] R. L. Hester, K. Murphy, J. J. Foxe, D. M. Foxe, D. C. Javitt, H. Garavan, "Predicting success: patterns of cortical activation and deactivation prior to response inhibition", *Journal of Cognitive Science*, **16**(5), 776-785, 2004.
- [Hingorani03] S. R. Hingorani, E. F. Petricoin III, A. Maitra, V. Rajapakse, C. King, M. A. Jocabetz, S. Ross, T. P. Conrads, T. D. Veenstra, B. A. Hitt, Y. Kawaguchi, D. Johann, L. A. Liotta, H. C. Crawford, M. E. Putt, T. Jacks, C. V. E. Wright, R. H. Hruban, A. M. Lowy, D. A. Tuveson, "Preinvasive and invasive ductal pancreatic cancer and its early detection in the mouse", *Cancer Cell*, **4**(6), 437-450, 2003.
- [Hng03] L. Hng, H. Inoue, K. Miyasaka, M. Tsukamoto, "Automatic generation of fuzzy classification systems using hyper-cone membership functions", *Proceedings of 2003 IEEE International Symposium on Computational Intelligence in Robotics and Automation*, 658-663, 2003.

-
- [Ho04] S. H. Ho, S. H. Jee, J. E. Lee, J. S. Park, "Analysis on risk factors for cervical cancer using induction technique", *Expert Systems with Applications*, **27**, 97-105, 2004.
- [Hoyer03] W. J. Hoyer, J. Cerella, S. V. Onyper, "Item learning in cognitive skill training: effects of item difficulty", *Memory and Cognition*, **31**(8), 1260-270, 2003.
- [Hruschka06] E. R. Hruschka, N. F. F. Ebecken, "Extracting rules from multilayer preceptrons in classification problems: a clustering-based approach", *Neurocomputing*, **70**, 384-397, 2006.
- [Huang05] K. Huang, H. Yang, I. King, M. R. Lyu, "Local learning vs. global learning: an introduction to maxi-min margin machine", in L. Wang (Ed.), *Support vector machines: theory and applications*, Studies in Fuzziness and Soft Computing, Heidelberg: Springer Berlin, **177**, 113-131, 2005.
- [Huwendiek99] O. Huwendiek, W. Brookmann, "Function approximation with decomposed fuzzy systems", *Fuzzy Sets and Systems*, **101**, 273-286, 1999.
- [Isasi04] P. Isasi, J. C. Hernandez, "Introduction to the applications of evolutionary computation in computer security and cryptography", *Computational Intelligence*, **20**(3), 445-449, 2004.
- [Ishibushi01] H. Ishibushi, T. Nakashima, "Effect of rule weights in fuzzy rule-based classification systems," *IEEE Transactions on Fuzzy Systems*, **9**(4), 506-515, 2001.
- [Jensen02] R. Jensen, Qiang Shen, "Fuzzy-rough sets for descriptive dimensionality reduction", *Proceedings of the 2002 IEEE International Conference on Fuzzy Systems*, **1**, 29-34, 2002.
- [Joseph96] R. Joseph, "Hippocampus" in R. Joseph, *Neuropsychiatry, Neuropsychology, Clinical Neuroscience*, (2nd Ed.), US: Lippincott Williams & Wilkins, 1996.
- [Joo05] M. G. Joo, J. S. Lee, "A class of hierarchical fuzzy system with constraints on the fuzzy rules", *IEEE Transactions on Fuzzy Systems*, **13**(2), 194-203, 2005.
- [Joy05] J. E. Joy, E. E. Penhoel, D. B. Petitti (Eds), *Saving women's lives: strategies for improving breast cancer detection and diagnosis*, Washington, D.C: The National Academies Press, 2005.
- [Kaplan77] M. S. Kaplan, J. W. Hinds, "Neurogenesis in the adult rat: electron microscopic analysis of light radioautographs", *Science*, **197**, 1092-1094, 1977.
- [Karazinov04] D. M. Karazinov, R. A. Boakes, "Learning about cues that prevent an outcome: conditioned inhibition and differential inhibition in human predictive learning", *Quarterly Journal of Experimental Psychology Section B: Comparative and Physiological Psychology*, **57**(2), 153-178, 2004.

-
- [Kasabov97] N. K. Kasabov, M. J. watts, "Genetic algorithms for structural optimization, dynamic adaptation and automated design of fuzzy neural networks", *International Conference on Neural Networks*, 4, 2546-2549, 1997.
- [Kasabov04a] N. Kasabov, "Knowledge-based neural networks for gene expression data analysis, modelling and profile discovery", *Biosilico*, 2(6), 253-261, 2004.
- [Kasabov04b] N. Kasabov, S. Pang, "Transductive support vector machines and applications in bioinformatics for promoter recognition", *Neural Information Processing*, 3(2), 31-38, 2004.
- [Kasabov07] N. Kasabov, "Global, local and personalized modelling and pattern discovery in bioinformatics: an integrated approach", *Pattern Recognition Letters*, 28, 673-685, 2007.
- [Kononenko84] I. Kononenko, I. Bratko, and E. Roskar. Experiments in automatic learning of medical diagnostic rule. Technical Report, Faculty of Electrical Engineering E. Kardelj, University of Ljubljana, 1984.
- [King03] R. King, B. Herr, M. Fekete, M. Lingen, "Use of proteomics for identifying molecular markers of oral cancer", *Oral Surgery, Oral Medicine, Oral Pathology, Oral Radiology & Endodontics*, 96 (3), 301, 2003.
- [Kleinsorge04] T. Kleinsorge, "Hierarchical switching with two types of judgment and two stimulus dimensions", *Experimental Psychology*, 51(2), 145-149, 2004.
- [Koh05] S. C. L. Koh, R. Khalil, F. -K. Lim, A. Ilancheran, M. Choolani, "The association between fibrinogen, von Willebrand Factor, Antithrombin III, and D-Dimer levels and survival outcome by 36 months from ovarian cancer", *Clinical Application of Thrombosis/Hemostasis*, 11(3), T1-T6, 2005.
- [Kornack01] D. R. Kornack, P. Rakic, "Cell proliferation without neurogenesis in adult primate neocortex", *Science*, 294, 2127-2130, 2001.
- [Kubota04] R. Kubota, T. Yamakawa, K. Horio, "Reproduction strategy based on self-organizing map for real-coded genetic algorithms", *Neural Information Processing*, 5(2), 28-32, 2004.
- [Kubota07] R. Kubota, K. Horio, T. Yamakawa, "Binary Self-Organizing Map with modified updating rule and its application to reproduction of genetic algorithm", *IEICE – Transactions on Information and Systems*, E90-D (1), 382-383, 2007.
- [Kwok00] J. T. -Y. Kwok, "The evidence framework applied to support vector machines", *IEEE Transactions on Neural Networks*, 11(5), 1162-1173, 2000.
- [Lavrač 99] N. Lavrač, "Machine learning for data mining in medicine", in W. Horn et al. (Eds.), *AIMDM'99*, LNAI 1620, Heidelberg: Springer-Verlag, 47-62, 1999.

-
- [Lee03] M. -L. Lee, H. -Y. Chung, F. -M. Yu, "Modeling of hierarchical fuzzy systems", *Fuzzy Sets and Systems*, **138**, 343-361, 2003.
 - [Leng02] Q. Leng, Z. Bentwich, "Beyond self and nonself: fuzzy recognition of the immune system", *Scandinavian Journal of Immunology*, **56**(3), 224-232, 2002.
 - [Levy02] B. J. Levy, M. C. Anderson, "Inhibitory processes and the control of memory retrieval", *Trends in Cognitive Sciences*, **6**(7), 299-305, 2002.
 - [Li04] L. Li, H. Tang, Z. Wu, J. Gong, M. Gruidl, J. Zou, M. Tockman, R. A. Clark, "Data mining techniques for cancer detection using serum proteomic profiling", *Artificial Intelligence in Medicine*, **32**, 71-83, 2004.
 - [Li05a] G. Li, T. -Y. Leong, L. Zhang, "Translation initiation sites prediction with mixture Gaussian models in human cDNA sequences", *IEEE Transactions on Knowledge and Data Engineering*, **17**(8), 1152-1160, 2005.
 - [Li05b] H. Li, T. Jiang, "A class of edit kernels for SVMs to predict translation initiation sites in eukaryotic mRNAs", *Journal of Computational Biology*, **12**(6), 702-718, 2005.
 - [Li07] Q. Li, L. Jiao, Y. Hao, "Adaptive simplification of solution for support vector machine", *Pattern Recognition*, **40**, 972-980, 2007.
 - [Lin96] C. H. Lin, C. S. Lee, *Neural fuzzy systems: a neuro-fuzzy synergism on intelligent system*, Englewood Cliffs, NJ: Prentice Hall.
 - [Lisboa02] P. J. G. Lisboa, "A review of evidence of health benefit from artificial neural networks in medical intervention", *Neural Networks*, **15**, 11-39, 2002.
 - [Lisboa06] P. J. Lisboa, A. F. G. Taktak, "The use of artificial neural networks in decision support in cancer: a systematic review", *Neural Networks*, **19**, 408-415, 2006.
 - [Lisman05] J. E. Lisman, A. A. Grace, "The hippocampal-VTA loop: controlling the entry of information into long-term memory", *Neuron*, **46**(5), 703-713, 2005.
 - [Liu92] N. Liu, K. J. Cios, "Learning rules by integer linear programming", *Proceedings of ISIE'92*, 246-250, 1992.
 - [Liu02] H. Liu, J. Li, L. Wong, "A comparative study on feature selection and classification methods using gene expression profiles and proteomic patterns", *Genomic Informatics*, **13**, 51-60, 2002.
 - [Liu03] H. Liu, L. Wong, "Data mining tools for biological sequences", *Journal of Bioinformatics and Computational Biology*, **1**, 139-160, 2003.
 - [Liu04] G. Liu, "Local structural balance and functional interaction of excitatory and inhibitory synapses in hippocampal dendrites", *Nature Neuroscience*, **7**, 373-379, 2004.

-
- [Liu05] P. Liu, H. Li, "Hierarchical TS fuzzy system and its universal approximation", *Information Sciences*, **169**, 279-303, 2005.
- [Long01] W. J. Long, "Medical informatics: reasoning methods", *Artificial Intelligence in Medicine*, **23**, 71-87, 2001.
- [Love02] B. C. Love, "Concept learning", in *The Encyclopedia of Cognitive Science*, Nature Publishing Group, 2002.
- [Lu06] K. Lu, J. Zhao, D. Cai, "An algorithm for semi-supervised learning in image retrieval", *Pattern Recognition*, **39**, 717-720, 2006.
- [Lupien98] S. J. Lupien, M. de Leon, S. de Santi, A. Convit, C. Tarshish, N. P. V. Nair, M. Tharkur, B. S. McEwen, R. L. Hauger, M. J. Meaney, "Cortisol levels during human aging predict hippocampal atrophy and memory deficits", *Nature Neuroscience*, **1**(1), 69-73, 1998.
- [Ma05] T. M. Ma, Q. Song, M. R. Marshall, N. Kasabov, "TWNFC – Transductive neural-fuzzy classifier with weighted data normalization and its application in medicine", *International Conference on Intelligent Agents, Control and Automation, Computational Intelligence for modeling and International conference on Web Technologies and Internet Commerce*, **1**, 479-484, 2005.
- [Mackman06] N. Mackman, "Role of tissue factor in hemostasis and thrombosis", *Blood Cells, Molecules, and Diseases*, **36**(2), 104-107, 2006.
- [Mac Namee02] B. Mac Namee, P. Cunningham, S. Byrne, O. I. Corrigan, "The problem of bias in training data in regression problems in medical decision support", *Artificial Intelligence in Medicine*, **24**, 51-70, 2002.
- [Maddox04] W. T. Maddox, F. G. Ashby, "Dissociating explicit and procedural-learning based systems of perceptual category learning", *Behavioural Processes*, **66**, 309-332, 2004.
- [Malone06] J. Malone, K. McGarry, C. Bowerman, "Automated trend analysis of proteomics data using an intelligent data mining architecture", *Expert Systems with Applications*, **30**, 24-33, 2006.
- [Mangasarian95] O. L. Mangasarian, W. N. Street, W. H. Wolberg, "Breast cancer diagnosis and prognosis via linear programming", *Operations Research*, **43**(4), 570-577, 1995.
- [Mangasarian90] O. L. Mangasarian, W. N. Street, W. H. Wolberg, "Pattern recognition via linear programming: theory and application", *Medical Diagnosis*, 22-31, 1990.
- [Meesad03] P. Meesad, G. G. Yen, "Combined numerical and linguistic knowledge representation and its application to medical diagnosis", *IEEE Transactions on Systems, Man, and Cybernetics: Part A*, **33** (2), 206-222, 2003.
- [Michalski86] R. Michalski, I. Mozetic, J. Hong, N. Lavrac, "The multi-purpose incremental learning system AQ15 and its testing applications to three medical domains", *Proceedings of the Fifth National*

-
- Conference on Artificial Intelligence*, Philadelphia, PA: Morgan Kaufmann, 1041-1045, 1986.
- [Mons04] N. Mons, L. Segu, X. Nogues, M. C. Buhot, "Effects of age and spatial learning on adenylyl cyclase mRNA expression in the mouse hippocampus", *Neurobiology of Aging*, **25**(8), 1095-1106, 2004.
- [Montironi03] R. Montironi, R. Mazzucchelli, M. Scarpelli, "Molecular techniques and prostate cancer diagnostic", *European Urology*, **44**, 390-400, 2003.
- [Mullins06] I. M. Mullins, M. S. Siadat, J. Lyman, K. Scully, C. T. Garrett, W. G. Miller, R. Muller, B. Robson, C. Apte, S. Weiss, I. Rigoutsos, D. Platt, S. Cohen, W. A. Knaus, "Data mining and clinical data repositories: insights from a 667,000 patient dataset", *Computers in Biology and Medicine*, **36**, 1351-1377, 2006.
- [NCI01] National Cancer Institute, Clinical Proteomics Program Databank, 2001.
- [NEDST03] National Electronic Decision Support Taskforce, *Electronic decision support for Australia's health sector*, Commonwealth of Australia, 2003.
- [Notterman01] D. A. Notterman, U. Alon, A. J. Sierk, A. J. Levine, "Transcriptional gene expression profiles of colorectal adenoma, adenocarcinoma and normal tissue examined by oligonucleotide arrays", *Cancer Research*, **61**, 3124-3130, 2001.
- [O'Kane04] G. O'Kane, E. A. Kensinger, S. Corkin, "Evidence for semantic learning in profound amnesia: an investigation with patient H. M.", *Hippocampus*, **14**(4), 411-412, 2004.
- [O'Leary02] T. J. O'Leary, "Proficiency testing in cervical cytopathology", *Accreditation and Quality Assurance*, **7**, 357-361, 2002.
- [O'Reilly02] R. C. O' Reilly, K. A. Norman, "Hippocampal and neocortical contributions to memory: advances in the complementary learning system framework", *Trends in Cognitive Sciences*, **6**(12), 505-510, 2002.
- [Ohlsson04] M. Ohlsson, "WeAidU—a decision support system for myocardial perfusion images using artificial neural networks", *Artificial Intelligence in Medicine*, **30**, 49-60, 2004.
- [Olaru03] C. Olaru, L. Wehenkel, "A complete fuzzy decision tree technique", *Fuzzy Sets and Systems*, **138**, 221-254, 2003.
- [OpenClinical02] OpenClinical, *Quality and safety of clinical decision support systems: a draft protocol for discussion*, OpenClinical 2002.
- [Pang04] S. Pang, N. Kasabov, "Inductive vs transductive inference, global vs local models: SVM, TSVM, and SVMT for gene expression classification problems", *Proceedings of IEEE International Joint Conference on Neural Networks*, **2**, 1197-1202, 2004.
- [Pedersen97] A. G. Pedersen, H. Nielsen, "Neural network prediction of translation initiation sites in eukaryotes: perspectives for EST and

- genome analysis", *Proceedings of 5th International Conference on Intelligent Systems for Molecular Biology*, 226-233, 1997.
- [Pedrycz96] W. Pedrycz, J. Valente de Oliveira, "Optimization of fuzzy models", *IEEE Transactions on Systems, Man and Cybernetics, Part B: Cybernetics*, **26**(4), 627-636.
- [Penã-Reyes99] C. A., Penã-Reyes, M. Sipper, "Designing breast cancer diagnostic systems via a hybrid fuzzy-genetic methodology", *Proceedings of the 8th IEEE International Conference Fuzzy System*, **1**, 135-139, 1999.
- [Penã-Reyes04] C. A. Penã-Reyes, "Coevolutionary fuzzy modeling", *Lecture Notes in Computer Sciences*, **3204**, Heidelberg: Springer-Verlag, 2004.
- [Pendharkar06] P. C. Pendharkar, "A data mining-constraint satisfaction optimization problem for cost effective classification", *Computers and Operations Research*, **33**(11), 3124-3135, 2006.
- [Petricoin III02] E. F. Petricoin III, A. M. Ardekani, B. A. Hitt, P. J. Levine, V. A. Fusaro, S. M. Steinberg, G. B. Mills, C. Simone, D. A. Fishman, E. C. Kohn, L. A. Liotta, "Use of proteomic patterns in serum to identify ovarian cancer", *The Lancet*, **359**, 572-577, 2002.
- [Petricoin04] E. F. Petricoin, L. A. Liotta, "Proteomic approaches in cancer risk and response assessment", *Trends in Molecular Medicine*, **10**(2), 59-65, 2004.
- [Pomeroy02] S. L. Pomeroy, P. Tamayo, M. Gaasenbeek, L. M. Sturla, M. Angello, M. E. McLaughlin, J. Y. Kim, L. C. Goumnerova, et al., "Prediction of central nervous system embryonal tumor outcome based on gene expression", *Nature*, **415**, 436-442, 2002.
- [Poole99] D. Poole, A. Mackworth, R. Goebel, *Computational Intelligence: A Logical Approach*, New York: Oxford University Press, 1999.
- [Quah07] K. H. Quah, C. Quek, "MCES: A novel Monte Carlo evaluative selection approach for objective feature selections", *IEEE Transactions on Neural Networks*, in press, 2007.
- [Quek00] C. Quek, W. L. Tung, "A novel approach to the derivation of fuzzy membership functions using FALCON-MART architecture", *Pattern Recognition Letters*, **22**(9), 941-958, 2000.
- [Quek05] C. Quek, A. Singh, "POP-Yager: a novel self-organizing fuzzy neural network based on the Yager inference", *Expert Systems with Applications*, **29**(1), 229-242, 2005.
- [Quek06] C. Quek, R. W. Zhou, "Structure and learning algorithms of a nonsingleton input fuzzy neural network based on the approximate analogical reasoning schema", *Fuzzy Sets and Systems*, **157**(13), 1814-1831, 2006.
- [Rajapakse05] J. C. Rajapakse, L. S. Ho, "Markov encoding for detecting signals in genomic sequences", *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, **2**(2), 1-12, 2005.

-
- [Riesenhuber99] M. Riesenhuber, T. Poggio, "Hierarchical models of object recognition in cortex", *Nature Neuroscience*, **2**(11), 1019-1025, 1999.
- [Röcken04] C. Röcken, M. P. A. Ebert, A. Roessner, "Proteomics in pathology, research and practice", *Pathology—Research and Practice*, **200**, 69-82, 2004.
- [Rodland04] K. D. Rodland, "Proteomics and cancer diagnosis: the potential of mass spectrometry", *Clinical Biochemistry*, **37**, 579-583, 2004.
- [Rudman04] W. J. Rudman, "The impact of a web-based reporting system on medication error occurrence reports part 1: identifying needs and solutions", *Towards an Electronic Patient Record (TEPR) Conference*, 2004.
- [Sabatini02] S. Sabatini, F. Solari, G. Andreani, C. Bartolozzi, G. M. Bisio, "A hierarchical model of complex cells in visual cortex for the binocular perception of motion-in-depth", *Proceedings of Advances in Neural Information Processing Systems Conference*, 1271-1278, 2002.
- [Sala03] J. B. Sala, P. Rämä, S. M. Courtney, "Functional topography of a distributed neural system for spatial and nonspatial information maintenance in working memory", *Neuropsychologia*, **41**(3), 341-356, 2003.
- [Salamov98] A. A. Salamov, T. Nishikawa, M. B. Swindells, "Assessing protein coding region integrity in cDNA sequencing projects", *Bioinformatics*, **14**, 384-390, 1998.
- [Sankar96] K. P. Sankar, P. P. Wong (Eds), *Genetic algorithms for pattern recognition*, Boca Raton: CRC Press, Inc., 1996.
- [Santos07] A. A. P. Santos, N. C. Affonso da Costa Jr., L. dos Santos Coelho, "Computational intelligence approaches and linear models in case studies of forecasting exchange rates", *Expert Systems with Applications*, in press, 2007.
- [Schmidtke96] K. Schmidtke, R. Handschu, H. Vollmer, "Cognitive procedural learning in Amnesia", *Brain and Cognition*, **32**, 441-467, 1996.
- [Schwaighofer03] A. Schwaighofer, V. Tresp., "Transductive and inductive methods for approximate Gaussian process regression", in S. Becker, S. Thrun, K. Obermayer (Eds.), *Advances in Neural Information Processing Systems*, US: MIT Press, **15**, 953-960, 2003.
- [Semolini03] R. Semolini, F. J. Von Zuben, "Transductive support vector machines for classification of microarray gene expression data", *Proceedings of the International Joint Conference on Neural Networks*, **4**, 2946-2951, 2003.
- [Setiono96] R. Setiono, H. Liu, "Symbolic representation of neural networks", *Computer*, **29**(3), 71-77, 1996.
- [Setiono00] R. Setiono, "Generating concise and accurate classification rules for breast cancer diagnosis", *Artificial Intelligence in Medicine*, **18**(3), 205-219, 2000.

-
- [Sewards03] T. V. Sowards, M. A. Sowards, "Representations of motivational drives in mesial cortex, medial thalamus, hypothalamus and midbrain", *Brain Research Bulletin*, **61**, 25-49, 2003.
 - [Shin06] H. Shin, M. K. Markey, "A machine learning perspective on the development of clinical decision support systems utilizing mass spectra of blood samples", *Journal of Biomedical Informatics*, **39**(2), 227-248, 2006.
 - [Sickles84] E. A. Sickles, "Mammographic features of early breast cancer", *AM J Roentgenol*, **143**, 461, 1984.
 - [Silva03] V. de Silva, J. B. Tenenbaum, "Global versus local methods in nonlinear dimensionality reduction", in S. Becker, S. Thrun, K. Obermayer, *Advances in Neural Information Processing Systems*, Cambridge: MIT Press, **15**, 705-712, 2003.
 - [Singh02] D. Singh, P. G. Febbo, K. Ross, D. G. Jackson, J. Manola, C. Ladd, P. Tamayo, A. A. Renshaw, A. V. D'Amico, J. P. Richie, E. S. Lander, Massimo Loda, P. W. Kantoff, T. R. Golub, W. R. Sellers, "Gene expression correlates of clinical prostate cancer behavior", *Cancer Cell*, **1**, 203-209, 2002.
 - [Sitter04] B. Sitter, T. Bathen, B. Hagen, C. Arentz, F. E. Skjeldestad, I. S. Gribbestad, "Cervical cancer tissue characterized by high-resolution magic angle spinning MR spectroscopy", *MAGMA*, **16**, 174-181, 2004.
 - [Slovic05] P. Slovic, E. Peters, M. L. Finucane, D. G. MacGregor, "Affect, risk, and decision making", *Health Psychology*, **24**(4, suppl.), S35-S40, 2005.
 - [Smith02] J. W. Smith, L. A. Fetsko, R. Xu, Y. Wang, "Dopamine D2L receptor knockout mice display deficits in positive and negative reinforcing properties of morphine and in avoidance learning", *Neuroscience*, **113**(4), 755-765, 2002.
 - [Song05] Q. Song, N. K. Kasabov, "NFI: a neuron-fuzzy inference method for transductive reasoning", *IEEE Transaction on Fuzzy Systems*, **13**(6), 799-808, 2005.
 - [Taha97] I. Taha, J. Ghosh, "Evaluation and ordering rules extracted from feedforward networks", *Proceedings of the IEEE International Conference on Neural Networks*, 221-226, 1997.
 - [Tan05a] T. Z. Tan, C. Quek, G. S. Ng, "Ovarian cancer diagnosis by hippocampus and neocortex-inspired learning memory structures", *Neural Networks*, **18**(5-6), 818-825, 2005.
 - [Tan05b] T. Z. Tan, C. Quek, G. S. Ng, "FALCON-AART: an improved version of complementary learning fuzzy neural network", *Proceedings of 12th International Conference on Neural Information Processing*, 628-633, 2005.
 - [Tan05c] T. Z. Tan, C. Quek, G. S. Ng, "Brain-inspired genetic complementary learning for stock market prediction", *Proceedings of IEEE Congress of Evolutionary Computing*, **3**, 2653-2660, 2005.

-
- [Tan06a] T. Z. Tan, C. Quek, G. S. Ng, "an alternative learning model: complementary learning", *under preparation*, 2006.
 - [Tan06b] T. Z. Tan, C. Quek, G. S. Ng, "Genetic complementary learning decision support system for cervical cancer", *under preparation*, 2006.
 - [Tan06c] T. Z. Tan, C. Quek, G. S. Ng, "A decision support system based on complementary learning paradigm: a comparative study based on breast cancer", *under preparation*, 2006.
 - [Tan06e] T. Z. Tan, C. Quek, G. S. Ng, "Biological brain-inspired genetic complementary learning for stock market and bank failure prediction", *Computational Intelligence*, in press, 2006.
 - [Tan07a] T. Z. Tan, C. Quek, G. S. Ng, "Hierarchical complementary learning", *IEEE Transaction Computational Biology and Bioinformatics*, in press, 2007.
 - [Tan07b] T. Z. Tan, C. Quek, G. S. Ng, "Chance discovery using complementary learning fuzzy neural network", in Y. Ohsawa, S. Tsumoto (Eds.), *Chance discoveries in real world decision-making*, Data-based Interaction of Human Intelligence and Artificial Intelligence Series: Studies in Computational Intelligence, **30**, Springer-Verlag, 2007. ISBN: 3-540-34352-0.
 - [Tan07c] T. Z. Tan, G. S. Ng, C. Quek, "Complementary learning fuzzy neural network—an approach to imbalanced dataset", *Proceedings of International Joint Conference on Neural Networks (IJCNN)*, accepted for publication, 2007.
 - [Tan07d] T. Z. Tan, C. Quek, G. S. Ng, "Complementary decision making system", submitted to *Artificial Intelligence in Medicine*, May, 2007.
 - [Tang05] L. Tang, H. Liu, "Bias analysis in text classification for highly skewed data", *Proceedings of the Fifth IEEE International Conference on Data Mining*, 781-784, 2005.
 - [Tenenbaum06] J. B. Tenenbaum, T. L. Griffiths, C. Kemp, "Theory-based Bayesian models of inductive learning and reasoning", *Trends in Cognitive Sciences*, **10**(7), 309-318, 2006.
 - [Thompson-Schill03] S. L. Thompson-Schill, "Neuroimaging studies of semantic memory: inferring 'how' from 'where'", *Neuropsychologia*, **4**, 280-292, 2003.
 - [Thompson87] R. F. Thompson (Eds). *Learning and memory, selected and with an introduction by Richard F. Thompson*, Boston: Birkhäuser Inc, 1987.
 - [Thomson03] A. M. Thomson, A. P. Bannister, "Interlaminar connections in the neocortex", *Cerebral Cortex*, **13**, 5-14, 2003.
 - [Toda02] T. Toda, M. Taoka, "Hierarchical somesthetic processing of tongue inputs in the post-central somatosensory cortex of conscious macaque monkeys", *Experimental Brain Research*, **147**, 243-251, 2002.

-
- [Turksen90] I. B. Turksen, Z. Zhong, "An approximate analogical reasoning schema based on similarity measures and interval-valued fuzzy sets", *Fuzzy Sets and System*, **34**, 223-346, 1990.
- [Tutz05] G. Tutz, H. Binder, "Localized classification", *Statistics and Computing*, **15**, 155-166, 2005.
- [Valente de Oliveira99] J. Valente de Oliveira, "Semantic constraint for membership function optimization", *IEEE Transactions on Systems, Man, and Cybernetics. Part A: Systems and Humans*, **29**(1), 128-138, 1999.
- [van't Veer02] L. J. van't Veer, H. Dai, M. J. van de Vijver, Y. D. He, A. A. M. Hart, M. Mao, H. L. Peterse, K. van der Kooy, M. J. Marton, A. T. Witteveen, G. J. Schreiber, R. M. Kerkhoven, C. Roberts, P. S. Linsley, R. Bernards, S. H. Friend, "Gene expression profiling predicts clinical outcome of breast cancer", *Nature*, **415**, 530-536, 2002.
- [Vijayakumar05] S. Vijayakumar, A. D'Souza, "Incremental online learning in high dimensions", *Neural Computation*, **17**, 2602-2634, 2005.
- [Voss96] M. Voss, "Fuzzy Billedbehandling", Masters Thesis, Department of Automation, Technical University of Denmark, 1996.
- [Wang97] W. -J. Wang, "New similarity measures on fuzzy sets and on elements", *Fuzzy Sets and Systems*, **85**, 305-309, 1997.
- [Wang99] L. -X. Wang, "Analysis and design of hierarchical fuzzy systems", *IEEE Transactions on Fuzzy Systems*, **7**(5), 617-624, 1999.
- [Wei00] C. Wei, L. -X. Wang, "A note on universal approximation by hierarchical fuzzy systems", *Information Sciences*, **123**, 241-248, 2000.
- [White04] C. N. White, D. W. Chan, Z. Zhang, "Bioinformatics strategies for proteomic profiling", *Clinical Biochemistry*, **37**, 636-641, 2004.
- [Willingham01] D. B. Willingham, K. Goedert, "The role of taxonomies in the study of human memory", *Cognitive, Affective, & Behavioral Neuroscience*, **1**(3), 250-265, 2001.
- [Witten05] I. Witten, E. Frank, *Data mining: practical machine learning tools and techniques*, 2 Ed., San Francisco: Morgan Kaufmann, 2005.
- [Wolberg90] W. H. Wolberg, O. L. Mangasarian, "Multisurface method of pattern separation for medical diagnosis applied to breast cytology", *Proceedings of the National Academy of Sciences*, 9193-9196, 1990.
- [Wong03] Y. Wong, J. Liu, T. Zhao, Q. Ji, "Recognizing translation initiation sites of eukaryotic genes based on the cooperatively scanning model", *Bioinformatics*, **19**(15), 1972-1977, 2003.
- [Wright96] W. G. Wright, E. F. McCance, T. J. Carew, "Developmental emergence of long-term memory for sensitization in *Aplysia*", *Neurobiology of Learning and Memory*, **65**, 261-268, 1996.
- [Wu07] C. -H. Wu, G. -H. Tzeng, Y. -J. Goo, W. -C. Fang, "A real-valued genetic algorithm to optimize the parameters of support vector

- machine for predicting bankruptcy”, *Expert System with Applications*, **32**, 397-408, 2007.
- [Xiao05] Z. Xiao, D. Prieto, T. P. Conrads, T. D. Veenstra, H. J. Issaq, “Proteomic patterns: their potential for disease diagnosis”, *Molecular and Cellular Endocrinology*, **230**, 95-106, 2005.
- [Yager05] R. R. Yager, “An extension of the Naïve Bayesian classifier”, *Information Sciences*, **176**, 577-588, 2006.
- [Ye07] J. Ye, J. Qiao, M. Li, X. Ruan, “A tabu based neural network learning algorithm”, *Neurocomputing*, **70**, 875-882, 2007.
- [Yeoh02] E. -J. Yeoh, M. E. Ross, S. A. Shurtleff, W. K. Williams, D. Patel, R. Mahfouz, F. G. Behm, S. C. Raimondi, M. V. Relling, A. Patel, C. Cheng, D. Campana, D. Wilkins, X. Zhou, J. Li, H. Liu, C. -H. Pui, W. E. Evans, C. Naeve, L. Wong, J. R. Downing, “Classification, subtype discovery and prediction of outcome in pediatric acute lymphoblastic leukemia by gene expression profiling”, *Cancer Cell*, **1**, 133-143, 2002.
- [Yoshida95] K. Yoshida, H. Motoda, “CLIP: concept learning from inference patterns”, *Artificial Intelligence*, **75**, 63-92, 1995.
- [Yu06] C. Yu, M. T. Manry, J. Li, P. L. Narasimha, “An efficient hidden layer training method for the multilayer perceptron”, *Neurocomputing*, **70**, 525-535, 2006.
- [Zeng02] F. Zeng, R. H. C. Yap, L. Wong, “Using feature generation and feature selection for accurate prediction of translation initiation sites”, *Genome Informatics*, **13**, 192-200, 2002.
- [Zhang92] J. Zhang, “Selecting typical instances in instance-based learning”, *Proceedings of the Ninth International Machine Learning Conference*, 470-479, Aberdeen, Scotland: Morgan Kaufmann, 1992.
- [Zhang94] J. Zhang, D. A. Norman, “Representations in distributed cognitive tasks”, *Cognitive Science*, **18**, 87-122, 1994.
- [Zhang97] T. Zhang, R. Ramakrishnan, M. Livny, “BIRCH: a new data clustering algorithm and its application”, *Data Mining and Knowledge Discovery*, **1**(2), 141-182, 1997.
- [Zhang00] Z. Zhang, H. Zhang, “ProstAsure index — a serum-based neural network-derived composite index for early detection of prostate cancer”, in P. J. G. Lisboa, E. C. Ifeachor, P. S. Szczepaniak (Eds.), *Artificial neural networks in biomedicine*, London: Springer-Verlag, 2000.
- [Zhang02] Z. Zhang, Y. Liu, T. Zhao, “SVM based feature screening applied to hierarchical cervical cancer detection”, *International Conference on Diagnostic Imaging and Analysis*, August, 2002.
- [Zhou96] R. W. Zhou, C. Quek, “POPFNN: a pseudo outer-product based fuzzy neural network”, *Neural Networks*, **9**(9), 1569-1581, 1996.

-
- [Zhou99] R. W. Zhou, C. Quek, "POPFNN-AARS(S): A pseudo outer-product based fuzzy neural network", *IEEE Transactions on Systems, Man, and Cybernetics*, **29**(6), 859-870, 1999.
 - [Zhou02] W. Zhou, R. Coggins, "Computational models of the amygdala and the orbitofrontal cortex: a hierarchical reinforcement learning system for robotic control", in R. I. McKay, J. Slaney (Eds.), *AI 2002*, LNAI 2557, Heidelberg: Springer-Verlag, 419-430, 2002.
 - [Zhou03] Z. -H. Zhou, Y. Jiang, "Medical diagnosis with C4.5 rule preceded by artificial neural network ensemble", *IEEE Transactions on Information Technology in Biomedicine*, **7**(1), 37-42, 2003.
 - [Zhou06a] Z. -H. Zhou, X. -Y. Liu, "Training cost-sensitive neural networks with methods addressing the class imbalance problem", *IEEE Transactions on Knowledge and Data Engineering*, **18**(1), 63-77, 2006.
 - [Zhou06b] S. -M. Zhou, J. Q. Gan, "Constructing accurate and parsimonious fuzzy models with distinguishable fuzzy sets based on an entropy measure", *Fuzzy Sets and Systems*, **157**(8), 1057-1074, 2006.
 - [Zhou07] S. -M. Zhou, J. Q. Gan, "Constructing L2-SVM-based fuzzy classifiers in high-dimensional space with automatic model selection and fuzzy rule ranking", *IEEE Transactions on Fuzzy Systems*, in press, 2007.
 - [Zien00] A. Zien, G. Rätsch, S. Mika, B. Schölkopf, T. Lengauer, K. -R. Müller, "Engineering support vector machine kernels that recognize translation initiation sites", *Bioinformatics*, **16**(9), 799-807, 2000.

Appendix A- Publication Arises from the Research Work:

Book Chapter

1. T. Z. Tan, C. Quek, G. S. Ng, "Chance discovery using complementary learning fuzzy neural network", in Y. Ohsawa, S. Tsumoto (Eds.), *Chance Discoveries in Real World Decision-Making*, Data-based Interaction of Human Intelligence and Artificial Intelligence Series: Studies in Computational Intelligence, **30**, 117-134, Springer-Verlag, 2006. ISBN: 3-540-34352-0.

Journal Paper

Published

2. T. Z. Tan, C. Quek, G. S. Ng, E. Y. K. Ng, "A novel cognitive interpretation of breast cancer thermography with complementary learning fuzzy neural memory structure", *Expert System with Applications*, **33**(3), 652-666, 2007.
3. T. Z. Tan, C. Quek, G. S. Ng, "Biological brain-inspired genetic complementary learning for stock market and bank failure prediction", *Computational Intelligence*, **23**(2), 236-261, 2007.
4. T. Z. Tan, C. Quek, G. S. Ng, "Ovarian cancer diagnosis by hippocampus and neocortex-inspired learning memory structures", *Neural Networks*, **18**(5-6), 818-825, 2005.
5. T. Z. Tan, G. S. Ng, C. Quek, "A biologically and psychologically inspired fuzzy decision support system: hierarchical complementary learning", *IEEE Transactions on Computational Biology and Bioinformatics*, **5**(1), 67-79, 2008.
6. T. Z. Tan, G. S. Ng, C. Quek, K. Razvi, "Ovarian Cancer diagnosis using complementary learning fuzzy neural network", *Artificial Intelligence in Medicine*, in press, 2008.

Under Review

7. T. Z. Tan, G. S. Ng, C. Quek, "Proteomics and complementary learning fuzzy neural network", submitted to *Bioinformatics*, under second review, 2008.
8. T. Z. Tan, G. S. Ng, C. Quek, "Complementary decision making system", submitted to *Artificial Intelligence in Medicine*, under second review, 2008.
9. T. Z. Tan, G. S. Ng, C. Quek, S. C. L. Koh, "Hemostasis and complementary Learning decision support system for ovarian cancer", *Submitted to Neurocomputing*, 2008.

Under Preparation

10. T. Z. Tan, G. S. Ng, C. Quek, "A Pattern Recognition Tool based on Improved Complementary Learning Fuzzy Neural Network", *under preparation*, 2008.
11. T. Z. Tan, G. S. Ng, C. Quek, "A decision support system based on complementary learning paradigm: a comparative study based on breast cancer", *under preparation*, 2008.

Appendix A: Publications Arises from this Research Work – T. Z. TAN

12. T. Z. Tan, G. S. Ng, C. Quek, "Genetic complementary learning decision support system for cervical cancer", under preparation, 2008.
13. T. Z. Tan, G. S. Ng, C. Quek, "An alternative learning model: complementary learning", *under preparation*, 2008.

Conference Paper

Year 2008

14. T. Z. Tan, G. S. Ng, C. Quek, "Improving tractability of clinical decision support system", *World Congress of Computational Intelligence, International Joint Conference on Neural Networks*, in press, 2008.

Year 2007

15. T. Z. Tan, G. S. Ng, C. Quek, "Complementary learning fuzzy neural network—an approach to imbalanced dataset", *Proceedings of International Joint Conference on Neural Networks (IJCNN)*, in press, 2007.
16. T. Z. Tan, G. S. Ng, C. Quek, "Complementary decision making system for aiding medical image diagnosis", *Proceedings of International Joint Conference on Neural Networks (IJCNN)*, in press, 2007.
17. T. Z. Tan, G. S. Ng, C. Quek, "A computational intelligence system based on memory consolidation model", *Proceedings of IEEE Congress on Evolutionary Computing*, 2630-2635, 2007.
18. T. Z. Tan, G. S. Ng, C. Quek, "PACL: a novel class of complementary learning fuzzy neural network", *IEEE Congress on Evolutionary Computing*, in press, 2007.

Year 2006

19. T. Z. Tan, G. S. Ng, C. Quek, "Motion estimation by a novel brain cellular fuzzy neural network", *Proceedings of IEEE International Workshop on Digital Infotainment and Visualization*, Singapore, November 27, 2006.
20. T. Z. Tan, G. S. Ng, A. Wahab, "Complementary learning for image segmentation", *Proceedings of IEEE International Workshop on Digital Infotainment and Visualization*, Singapore, November 27, 2006.
21. T. Z. Tan, G. S. Ng, A. Wahab, "Complementing SPECT diagnosis with Genetic Complementary Learning Fuzzy Neural Network", *Proceedings of IEEE International Workshop on Digital Infotainment and Visualization*, Singapore, November 27, 2006.
22. T. Z. Tan, G. S. Ng, C. Quek, "Clinical diagnosis using proteomics and complementary learning", *Proceedings of World Congress of Computational Intelligence, IJCNN*, 5172-5179, 2006.

Appendix A: Publications Arises from this Research Work – T. Z. TAN

23. T. Z. Tan, G. S. Ng, C. Quek, "Genetic complementary learning for translation initialization start sites prediction", *Proceedings of World Congress of Computational Intelligence*, CEC, 963-970, 2006.
24. T. Z. Tan, G. S. Ng, C. Quek, S. C. L. Koh, "Ovarian cancer prognosis by hemostasis and complementary learning", in I. King, J. Wang, L. Chan, D. Wang, (Eds), *Neural Information System*, LNCS, **4234**, 145-154, Springer Verlag, 2006. ISBN: 3-540-46484-0.
25. T. Z. Tan, G. S. Ng, S. S. Erdogan, "A neuropsychology-inspired learning system for human uncertainty monitoring", *Proceedings of 9th International Conference on Automation, Robotics, and Computer Vision* (ICARCV), 1317-1324, 2006.
26. T. Z. Tan, G. S. Ng, C. Quek, S. C. L. Koh, "A prognosis tool based on Hemostasis and genetic complementary learning", *Proceedings of International Conference on Biomedical and Pharmaceutical Engineering* (ICBPE), 362-367, 2006.

Year 2005

27. T. Z. Tan, C. Quek, G. S. Ng, "T2CLFNN: A Neuropsychology-Inspired Fuzzy Neural Network that models human uncertainty monitoring", *Proceedings of 12th International Conference on Neural Information Processing* (ICONIP), 699-704, 2005.
28. T. Z. Tan, C. Quek, G. S. Ng, "FALCON-AART: an improved version of Complementary Learning fuzzy neural network", *Proceedings of 12th International Conference on Neural Information Processing* (ICONIP), 628-633, 2005.
29. T. Z. Tan, C. Quek, G. S. Ng, "Brain-inspired genetic complementary learning for stock market prediction", *Proceedings of IEEE Congress of Evolutionary Computing* (CEC), **3**, 2653-2660, 2005.
30. T. Z. Tan, C. Quek, G. S. Ng, "Ovarian Cancer diagnosis using complementary learning fuzzy neural network", *Proceedings of International Joint Conference on Neural Networks* (IJCNN), 3034-3039, 2005.

Year 2004

31. T. Z. Tan, C. Quek, G. S. Ng, E. Y. K. Ng, "Breast cancer diagnosis using thermography and complementary learning fuzzy neural network", *Proceedings of the First International BioEngineering Conference* (IBEC), 65-68, 2004.

Technical Report and Report

32. T. Z. Tan, "Complementary learning fuzzy neural network for medical domain and bioinformatics domain applications", Ph. D first year report, Department of Computer Science, School of Computer Engineering, Nanyang Technological University, Singapore, 2005.
33. T. Z. Tan, C. Quek, G. S. Ng, "FALCON-AART: an improved version of FALCON-ART", Technical Report, C2i-TR-04/002, School of Computer Engineering, Nanyang Technological University, Singapore, 2004.

APPENDIX B: Propositional Logic Transformation

For a single-output- two-input conjunction rule:

$$\text{IF } x_1 \text{ is } \mathbf{A}_1 \text{ AND } x_2 \text{ is } \mathbf{A}_2 \text{ THEN } y \text{ is } \mathbf{B} \quad (\text{B1})$$

It is represented as propositional logic:

$$x_1 \text{ is } \mathbf{A}_1 = p$$

$$x_2 \text{ is } \mathbf{A}_2 = q$$

$$y \text{ is } \mathbf{B} = r$$

The initial construct of the rule in propositional logic form:

$$(p \wedge q) \Rightarrow r$$

By material implication

$$\sim (p \wedge q) \vee r$$

By DeMorgan's Law

$$(\sim p \wedge \sim q) \vee r$$

By association

$$\sim p \vee (\sim q \vee r)$$

By Commutation

$$(\sim q \vee r) \vee \sim p$$

By material implication

$$(q \Rightarrow r) \vee \sim p$$

By addition

$$((q \Rightarrow r) \vee \sim p) \vee r$$

By association

$$(q \Rightarrow r) \vee (\sim p \vee r)$$

By material implication

$$(q \Rightarrow r) \vee (p \Rightarrow r)$$

By commutation

$$(p \Rightarrow r) \vee (q \Rightarrow r)$$

Thus, Equation B1 is functionally equivalent to Equation B2.

$$\{[\text{IF } x_1 \text{ is } \mathbf{A}_1 \text{ THEN } y \text{ is } \mathbf{B}] \text{ OR } [\text{IF } x_2 \text{ is } \mathbf{A}_2 \text{ THEN } y \text{ is } \mathbf{B}]\} \quad (\text{B2})$$

APPENDIX C: Detailed Experimental Result

HCL result

Table C1: Performance on Leukaemia subtype classification.

Method		Recall (%)	Sensitivity (%)	Specificity (%)	Predict (%)
C4.5	BCR	94.69	0	100	96.26
	E2A	96.46	50	82.7	79.97
	MLL	93.81	0	100	93.93
	Hyper	80.53	0	100	80.37
	Other	76.1	0	100	75.7
	T-ALL	86.73	0	100	86.92
	TEL-AML	77.88	0	96.3	72.9
LR	BCR	54.87	62.5	49	49.53
	E2A	97.35	27.8	92.9	87.38
	MLL	94.69	0	98.5	92.52
	Hyper	83.19	2.4	98.3	79.44
	Other	74.34	9.6	96.3	75.23
	T-ALL	86.73	0	100	86.92
	TEL-AML	79.65	1.9	92.6	70.56
MLP	BCR	95.58	0	97.1	93.46
	E2A	96.46	55.6	89.3	86.45
	MLL	94.69	0	98.5	92.34
	Hyper	81.42	2.4	98.3	79.44
	Other	76.1	0	100	75.7
	T-ALL	86.73	0	100	86.92
	TEL-AML	81.42	5.8	96.3	74.3
SVM-Sigmoid (other kernels cannot classify using only 1/3 data for training)	BCR	94.69	0	100	96.26
	E2A	92.92	0	100	91.59
	MLL	94.69	0	100	93.93
	Hyper	80.53	0	100	80.37
	Other	75.22	3.8	97.5	74.77
	T-ALL	86.73	0	100	86.92
	TEL-AML	70.8	3.8	94.4	72.5
LDA	BCR	94.69	0	100	96.26
	E2A	97.35	60	90.95	87.38
	MLL	94.69	0	99.48	90.19
	Hyper	75.22	4.76	96.51	78.5
	Other	76.99	15.38	91.98	73.36
	T-ALL	86.73	0	100	86.92
	TEL-AML	76.11	1.9	88.89	67.57
FA	BCR	76.11	25	76.69	74.77
	E2A	92.92	5.56	99.49	91.59
	MLL	93.02	7.69	95.02	89.72
	Hyper	78.76	7.14	88.37	72.43
	Other	75.22	13.46	92.59	73.36
	T-ALL	87.61	3.57	99.46	86.92
	TEL-AML	75.22	1.92	95.06	72.43
NB	BCR	54.87	62.5	49	49.53
	E2A	80.53	33.33	76.5	72.9
	MLL	51.33	38.5	56.2	55.14
	Hyper	54.87	40.5	51.7	49.53
	Other	73.45	15.4	92	73.36
	T-ALL	50.44	39.3	48.9	47.66
	TEL-AML	71.68	30.8	56.2	50
HCL	BCR	90.27	25	91.26	88.79
	E2A	92.92	5.56	100	92.06
	MLL	88.5	0	95.52	89.72
	Hyper	82.31	2.38	98.26	79.44
	Other	79.65	1.92	98.77	75.23
	T-ALL	84.11	0	87.76	76.11
	TEL-AML	82.3	1.92	99.38	75.7

Abbreviations: BCR: B lineage with t(9,22); E2A: t(1,19); MLL: chromosome 11 rearrangement; T-ALL: T-lineage with hyperdiploid > 50-hyper; TEL-AML: t(12,21).

APPENDIX D: Syntactic Dataset

x_1	x_2	Class
0.15	0.42	+
0.13	0.45	+
0.2	0.5	+
0.2442	0.4	+
0.2869	0.39	+
0.13	0.51	+
0.2094	0.4	+
0.18	0.4	+
0.2477	0.52	+
0.2	0.43	+
0.2442	0.45	+
0.5	0.3	–
0.47	0.33	–
0.558	0.39	–
0.3	0.37	–
0.536	0.32	–
0.472	0.41	–
0.41	0.38	–
0.442	0.43	–
0.538	0.4	–
0.5	0.487	–
0.52	0.36	–