

การประยุกต์ใช้งานโครงข่ายประสาทเทียมเชิงลำดับกับงานทางด้านการตัดคำ
ภาษาไทย

Applying Sequential Neural Networks to Thai Word Separation



H001801



รายงานนี้เป็นส่วนหนึ่งของวิชา โครงกรพัฒนาระบบงาน
หลักสูตรวิทยาศาสตรมหาบัณฑิต สาขาเทคโนโลยีสารสนเทศ
ภาคเรียนที่ 1 ปีการศึกษา 2544
คณะเทคโนโลยีสารสนเทศ
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ชื่อหัวข้อ การประยุกต์ใช้งาน โครงข่ายประสาทเทียมเชิงลำดับกับงานทางด้าน การตัดคำ ภาษาไทย

นักศึกษา นายวิทยา ตานี

อาจารย์ที่ปรึกษา ดร.ภัทรชัย สถิต โรจน์วงศ์

ระดับการศึกษา วิทยาศาสตร์มหาบัณฑิต สาขาเทคโนโลยีสารสนเทศ

แขนงวิชา วิทยาการสารสนเทศ

ปีการศึกษา 2544

บทคัดย่อ

ปัจจุบันนักวิจัยจำนวนมาก ได้ทำการวิจัยค้นคว้าเพื่อให้คอมพิวเตอร์มีความสามารถเข้าใจ ในภาษาธรรมชาติของมนุษย์ ภาษาไทยก็เป็นอีกภาษาหนึ่งที่นักวิจัยพยายามค้นคว้าหาวิธีการในการ ประมวลผลภาษานี้ แต่ด้วยข้อจำกัดทางภาษาที่ภาษาไทยไม่มีการเว้นวรรคระหว่างคำในประโยค ทำให้ไม่สามารถนำประโยคไปประมวลผลได้โดยทันที ดังนั้นระบบการตัดคำต่างๆ จึงถูกนำมาใช้ ในการแก้ปัญหานี้ ไม่ว่าจะเป็นการตัดคำโดยอาศัยกฎทางภาษา หรือการตัดคำโดยอาศัยพจนานุกรม เป็นต้น แต่กระนั้นการตัดคำก็ยังทำได้ไม่สมบูรณ์นัก จึงได้มีแนวคิดที่จะนำเอาวิธีการของระบบ โครงข่ายประสาทเทียมที่เรียนรู้ข้อมูลเชิงลำดับมาประยุกต์ใช้ เพื่อให้ประสิทธิภาพการตัดคำดียิ่งขึ้น

Title Applying Sequential Neural Networks to Thai Word Separation
Student Mr. Wittaya Tanee
Advisor Dr. Pattarachai Lalitrojwong
Level of Study Master of Science in Information Technology
Major Information Science
Academic Year 2001

ABSTRACT

Since many researchers have tried to find the way to make the computer understand natural languages. Thai language, is one that the researcher tries to analyze. However, Thai written language processing is the successive words without spaces in a sentence. Therefore, the word separation program used to solve this problem based on the language rules or a dictionary. Until now, most word separation programs cannot work perfectly. This project presents another approach to this task using sequential neural networks perform.

กิตติกรรมประกาศ

การพัฒนาระบบงานนี้จะ ไม่ประสบผลสำเร็จได้เลย ถ้าขาดบุคคลเหล่านี้ ผู้จัดทำจึงใคร่ขอ
ขอบพระคุณบุคคลดังต่อไปนี้

1. บิดา มารดา พี่และน้อง สำหรับกำลังใจที่ไม่เคยหมด
2. ดร. ภัทรชัย ลลิต โรจน์วงศ์ สำหรับคำแนะนำดีๆ ในด้านต่างๆ งานงานนี้สามารถทำได้
สำเร็จจุดดวงไปได้ด้วยดี
3. บริษัท โอจีเอ อินเตอร์เนชั่นแนล สำหรับโอกาสในการศึกษาครั้งนี้
4. เพื่อนร่วมงาน และเพื่อน IS8 สมทบ ทุกคน สำหรับความช่วยเหลือในด้านต่างๆเสมอ
มา

วิทยา ตานี



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

สารบัญ

	หน้า
บทคัดย่อภาษาไทย.....	I
บทคัดย่อภาษาอังกฤษ.....	II
กิตติกรรมประกาศ.....	III
สารบัญ.....	IV
สารบัญตาราง.....	V
สารบัญภาพ.....	VI
บทที่	
1. บทนำ.....	1
1.1 ความสำคัญและความเป็นมาของปัญหา.....	1
1.2 วัตถุประสงค์ของโครงการพัฒนาระบบงาน.....	1
1.3 แนวคิดที่ใช้ในโครงการพัฒนาระบบงาน.....	2
1.4 ขอบเขตของโครงการพัฒนาระบบงาน.....	2
2. ทฤษฎีเบื้องต้นของโครงข่ายประสาทเทียม.....	3
2.1 แนวคิดพื้นฐานของโครงข่ายประสาทเทียม.....	3
2.2 ส่วนประกอบที่สำคัญของโครงข่ายประสาทเทียม.....	4
2.3 แนวคิดพื้นฐานของการเรียนรู้ข้อมูลเชิงลำดับ.....	8
2.4 โครงข่ายประสาทเทียมแบบ Simple Recurrent Network.....	9
2.5 โครงข่ายประสาทเทียมแบบ Jordan 's Network.....	12
2.6 ตัวอย่างการประยุกต์ใช้งานโครงข่ายโครงข่ายประสาทเทียมสำหรับการเรียนรู้ข้อมูล เชิงลำดับ.....	14
3. การออกแบบและการทดลองใช้งานโครงข่ายประสาทเทียม.....	18
3.1 แนวคิดในการประยุกต์ใช้โครงข่ายประสาทเทียมสำหรับการเรียนรู้ข้อมูลเชิงลำดับ กับงานทางด้านการตัดคำภาษาไทย.....	18
3.2 การออกแบบและการพัฒนาโครงงาน.....	22
3.3 การทดลอง.....	35
3.4 สรุปผลการทดลอง.....	37

4. บทสรุปและข้อเสนอแนะ.....	38
4.1 บทสรุปของโครงการพัฒนาระบบงาน.....	38
4.2 ข้อเสนอแนะ.....	38
บรรณานุกรม.....	39
ประวัติผู้เขียน.....	40



สารบัญภาพ

หน้า

ภาพที่

2.1 การทำงานของเซลล์ประสาทในสมองมนุษย์.....	3
2.2 การทำงานของยูนิตเปรียบเทียบกับเซลล์ประสาทมนุษย์.....	4
2.3 การเชื่อมต่อแบบ Feedforward network.....	5
2.4 การเชื่อมต่อแบบ Interactive network.....	6
2.5 การเชื่อมต่อยูนิตแบบหลายระดับชั้น.....	6
2.6 รูปแบบการประมวลผลของโครงข่ายประสาทเทียมสำหรับการเรียนรู้ข้อมูลเชิงลำดับ.....	9
2.7 สถาปัตยกรรมโครงข่ายแบบ Simple Recurrent Network.....	10
2.8 สถาปัตยกรรมโครงข่ายแบบ Jordan's Network (ไม่ครบทุก Link).....	12
2.9 สถาปัตยกรรมโครงข่ายแบบ Jordan's Network.....	13
2.10 การทำงานของโครงข่ายประสาทเทียมสำหรับการเรียนรู้ข้อมูลเชิงลำดับในงานการถ่ายข้อความ.....	15
2.11 โครงข่ายประสาทเทียมสำหรับการเรียนรู้ข้อมูลเชิงลำดับในงานการแปลความหมายของคำไปเป็นการออกเสียงของคำ.....	16
2.12 โครงข่าย BA/DA/GA.....	17
3.1 SNNS-Manager.....	22
3.2 การกำหนดจำนวนยูนิตของโครงข่ายประสาทเทียม.....	24
3.3 โครงข่ายที่สร้างขึ้นจากเครื่องมือ SNNS.....	24
3.4 กรอบที่ใช้ในการควบคุมการเทรนหรือทดสอบโครงข่าย.....	27
3.5 กรอบการบันทึกโครงข่ายที่ผ่านการเทรนแล้ว.....	28
3.6 หน้าจอหลักของโปรแกรมการตัดคำภาษาไทย.....	29
3.7 การป้อนประโยคให้แก่โปรแกรมการตัดคำ.....	30
3.8 ตัวอย่างการสร้างโครงข่ายต้นไม้ของโปรแกรมการตัดคำ.....	31
3.9 หน้าจอการตรวจเช็ครูปประโยคกับแพลนในฐานข้อมูล.....	32
3.10 การทำนายของโครงข่ายประสาทเทียมเชิงลำดับ.....	33
3.11 หน้าจอขณะประมวลผลรูปแบบคำที่ตัดได้เป็นผลสำเร็จ.....	34
3.12 หน้าจอของการแสดงผลลัพธ์ของโปรแกรมการตัดคำ.....	35

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษเท่านั้น เมื่อนำไปเผยแพร่โดยไม่ได้รับอนุญาต

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.13 หน้าจอแสดงผลการตัดคำประโยคทดลองที่ 1..... 36

3.14 หน้าจอแสดงผลการตัดคำประโยคทดลองที่ 2.....36



บทที่ 1

บทนำ

1.1 ความสำคัญและความเป็นมาของปัญหา

ปัจจุบันศาสตร์ทางการประมวลผลภาษาธรรมชาติ (Natural Language Processing - NLP) ได้รับความสนใจอย่างมากจากนักวิจัย ในการพัฒนาเทคนิคค้นทฤษฎีหรือวิธีการใหม่ๆขึ้นมา เพื่อประมวลผลภาษาธรรมชาติที่มนุษย์ใช้ในการสื่อสารกัน โดยมีวัตถุประสงค์เพื่อให้คอมพิวเตอร์มีความเข้าใจในภาษาดังกล่าว เพื่อประโยชน์ในการควบคุมหรือสั่งงานคอมพิวเตอร์ให้มีความสะดวกง่ายดายยิ่งขึ้น โดยมนุษย์ไม่จำเป็นต้องเรียนรู้วิธีการหรือภาษาที่คอมพิวเตอร์เข้าใจ แต่ด้วยความหลากหลายของภาษาธรรมชาติ เป็นผลให้ศาสตร์ทางการประมวลผลภาษาธรรมชาติยังคงมีข้อบกพร่องอยู่บ้าง บางข้อจำกัดจะพบได้ในภาษาหนึ่งแต่อาจจะไม่พบในอีกภาษาหนึ่งก็เป็นได้ เช่นในระบบภาษาไทยรูปประโยคจะเกิดจากการนำคำมาเรียงกันอย่างต่อเนื่อง โดยไม่มีการเว้นช่องว่างระหว่างคำ ดังนั้นการที่จะให้คอมพิวเตอร์ประมวลผลระบบภาษาไทยได้นั้น จึงจำเป็นต้องมีการวิเคราะห์ประโยคเสียก่อน โดยตัดคำออกเป็นคำย่อยๆ โดยไม่ผิดไปจากความหมายเดิมที่ผู้สื่อสารต้องการสื่อ มิฉะนั้นผลลัพธ์ที่ได้จากการประมวลผลก็จะไม่ถูกต้อง ซึ่งปัญหาดังกล่าวจะไม่พบในระบบภาษาอังกฤษ เนื่องจากในภาษาอังกฤษจะมีการเว้นวรรคระหว่างคำอยู่แล้ว จากปัญหานี้ นักวิจัยจึงได้พยายามคิดค้นหาวิธีการต่างๆ เพื่อให้ได้ระบบการตัดคำที่ให้ผลดีที่สุด เช่น วิธีการตัดคำโดยการใช้กฎทางภาษา หรือการตัดคำโดยอาศัยพจนานุกรมต่างก็เป็นวิธีการที่ถูกนำมาใช้ แต่ผลลัพธ์ที่ได้ก็ยังคงมีความผิดพลาดอยู่บ้าง โครงการนี้จึงนำเสนออีกวิธีการหนึ่ง คือการใช้โครงข่ายประสาทเทียม

1.2 วัตถุประสงค์ของโครงการพัฒนาระบบงาน

1. เพื่อศึกษาการนำโครงข่ายประสาทเทียมที่เรียนรู้ข้อมูลเชิงลำดับมาประยุกต์ใช้ในการตัดคำภาษาไทย
2. เพื่อศึกษาคุณสมบัติและขั้นตอนการนำโครงข่ายประสาทเทียมสำหรับการเรียนรู้ข้อมูลเชิงลำดับมาประยุกต์ใช้งาน
3. เพื่อศึกษาการพัฒนาโปรแกรมโดยอาศัยเครื่องมือที่มีชื่อว่า SNNS (Stuttgart Neural

1.3 แนวคิดที่ใช้ในโครงการพัฒนาระบบงาน

แนวคิดที่นำมาใช้ในการพัฒนาระบบงานการประยุกต์ใช้โครงข่ายประสาทเทียมเชิงลำดับ ในการตัดคำภาษาไทย จะอาศัยโครงข่ายประสาทเทียมสำหรับการเรียนรู้ข้อมูลเชิงลำดับ แบบ Jordan's Network ที่ถูกสร้างขึ้นจากเครื่องมือที่เรียกว่า SNNS ร่วมกับโปรแกรมที่ถูกพัฒนาขึ้นมา จากโปรแกรมภาษา Visual Basic 6.0 โดยการสร้างโครงข่ายขึ้นมาและทำการฝึกสอนรูปแบบการเรียงตัวของหน้าที่คำในประโยคภาษาไทยเอาไว้ เมื่อนำมาโครงข่ายดังกล่าวมาทำงานร่วมกับ โปรแกรมที่พัฒนาขึ้น โครงข่ายจะมีหน้าที่ในการช่วยตัดสินใจตามรูปแบบที่ได้เรียนรู้มาว่าคำนั้น ควรตัดคำออกมาในลักษณะใด เมื่อพบคำที่สามารถตัดออกมาได้ในหลายรูปแบบ โดยโปรแกรมที่ถูกพัฒนาขึ้นมาจะต้องประกอบไปด้วยฐานข้อมูลคำที่ดีเพียงพอ การทำงานจึงจะมีประสิทธิภาพที่ดี

1.4 ขอบเขตของโครงการพัฒนาระบบงาน

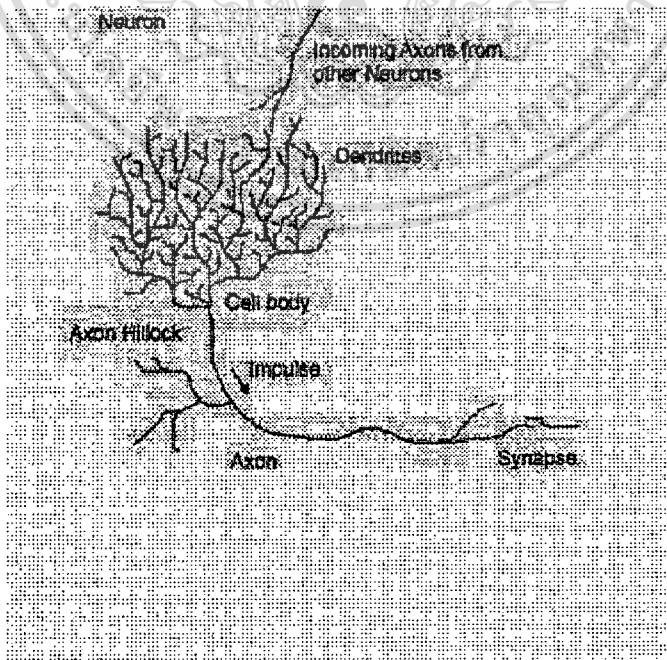
ขอบเขตของการพัฒนาระบบงานนี้จะเป็นการศึกษาถึงการนำโครงข่ายประสาทเทียม สำหรับการเรียนรู้ข้อมูลเชิงลำดับมาประยุกต์ใช้งานซึ่งในที่นี้ก็คือการตัดคำภาษาไทย โดยในการพัฒนาระบบงานนี้จะไม่คำนึงถึงคำที่ไม่สามารถค้นพบได้ในฐานข้อมูลคำ และจะตั้งข้อสมมติฐานว่าทุกคำมีอยู่ในฐานข้อมูลคำทั้งหมด เนื่องจากการประมวลผลคำที่ไม่สามารถค้นพบในฐานข้อมูล นั้นจะต้องอาศัยวิธีการอื่นมาช่วยในการวิเคราะห์คำการประมวลผลจึงจะมีความถูกต้อง ซึ่งจะไม่ขอก้าวถึงในการพัฒนาระบบงานนี้

บทที่ 2

ทฤษฎีเบื้องต้นของโครงข่ายประสาทเทียม

2.1 แนวคิดพื้นฐานของโครงข่ายประสาทเทียม

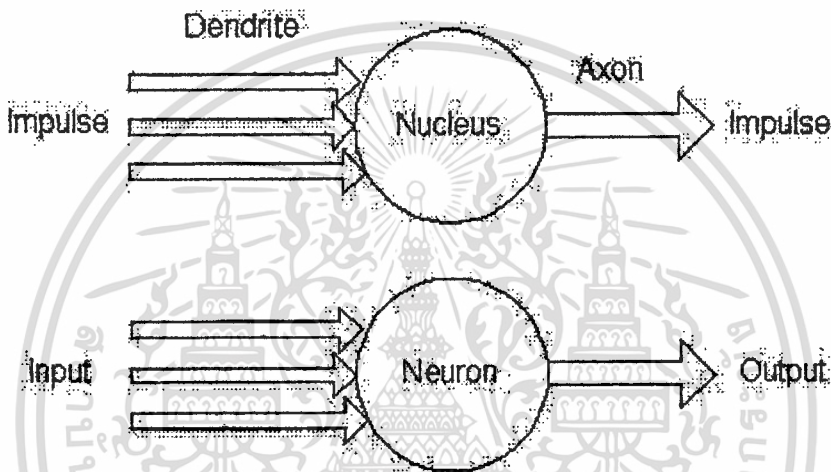
ในสมองมนุษย์เรานั้นประกอบไปด้วยเซลล์ประสาท (Neuron) จำนวนมากมายประมาณ 10^{11} เซลล์ ในแต่ละเซลล์จะมีเส้นเชื่อมโยงไปยังเซลล์อื่นๆอีกประมาณ 10^{14} เส้น เชื่อมโยงกันเป็นเครือข่ายขนาดใหญ่ดังแสดงในรูปที่ 2.1 (Cleeremans,1993) โดยภายในแต่ละเซลล์ประสาทจะประกอบไปด้วย นิวเคลียส (Nucleus) เด็นไดรท์ (Dendrite) และ แอ็กซอน (Axon) ทำหน้าที่ต่างๆกันไป เด็นไดรท์จะทำหน้าที่รับข้อมูลเข้ามาและส่งไปให้นิวเคลียสทำการประมวลผลแล้วจึงส่งผลลัพธ์ไปยังเซลล์ประสาทอื่นๆผ่านทางแอ็กซอน จากการทำงานของเซลล์ประสาทเหล่านี้เองจึงมีผู้พยายามคิดค้น การจำลองรูปแบบการทำงานของคอมพิวเตอร์ โดยเลียนแบบการทำงานของสมองมนุษย์ เพื่อใช้ในงานต่างๆ เช่น การสังเคราะห์เสียง การรู้จำตัวอักษร การจัดหมวดหมู่ การทำนายเหตุการณ์ เป็นต้น ดังนั้นจึงทำให้เกิดศาสตร์อีกแขนงหนึ่งทางด้านคอมพิวเตอร์ขึ้นมา เรียกว่า “Artificial Neural Network” หรือ โครงข่ายประสาทเทียมนั่นเอง



รูปที่ 2.1 การทำงานของเซลล์ประสาทในสมองมนุษย์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Artificial Neural Network (ANN), Neural Networks, Neural Nets (NN) และ Connectionist Network เป็นศัพท์เทคนิคที่ถูกนำมาใช้ในการให้ความหมายถึงระบบโครงข่ายประสาทเทียม โดยแนวคิดของระบบโครงข่ายประสาทเทียมจะใช้ยูนิต (Units) ทำหน้าที่ในการประมวลผลข้อมูลเช่นเดียวกับนิวเคลียส และในแต่ละยูนิตจะมีส่วนอินพุตและเอาต์พุตทำหน้าที่เสมือนกับเส้นโคโรทและแอกซอนตามลำดับ เมื่อนำยูนิตแต่ละตัวมาต่อเข้าด้วยกันเป็นโครงข่ายก็จะได้การทำงานที่คล้ายกับเซลล์ประสาทมนุษย์นั่นเองดังรูปที่ 2.2



รูปที่ 2.2 การทำงานของยูนิตเปรียบเทียบกับเซลล์ประสาทมนุษย์

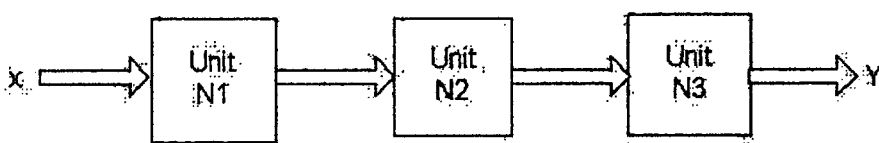
2.2 ส่วนประกอบที่สำคัญของโครงข่ายประสาทเทียม (Bechtel, 1997)

2.2.1 ยูนิต (Units)

ยูนิตถือได้ว่าเป็นองค์ประกอบหลักของสถาปัตยกรรมโครงข่ายประสาทเทียม เนื่องจากการประมวลผลของโครงข่ายจะเกิดขึ้นที่ยูนิตแต่ละตัว ไม่ว่าจะเป็นในเรื่องของการกระตุ้นยูนิต (Activation) การเชื่อมต่อระหว่างยูนิตแบบถ่วงน้ำหนัก (Weighted connections) วิธีการเรียนรู้ (Learning rules) ก็เกิดจากการทำงานที่ยูนิตนั่นเอง ในระบบโครงข่ายประสาทเทียม ยูนิตจะแสดงถึงสิ่งใดสิ่งหนึ่ง (Concepts) โดยหนึ่งความหมายอาจจะถูกแทนด้วยยูนิตหนึ่งยูนิตหรือหลายๆยูนิตก็เป็นได้ เช่น บุคคลใดบุคคลหนึ่ง อาจจะแทนด้วยยูนิตเพียงยูนิตเดียวหรือใช้กลุ่มของยูนิตในการแสดงบุคคลนั้นๆ ก็สามารถทำได้ จากการทำงานที่คล้ายกันของยูนิตนี้เอง เราสามารถจำแนกวิธีการแสดง ความหมายได้ 2 แบบคือ Local representation และ Distributed representation

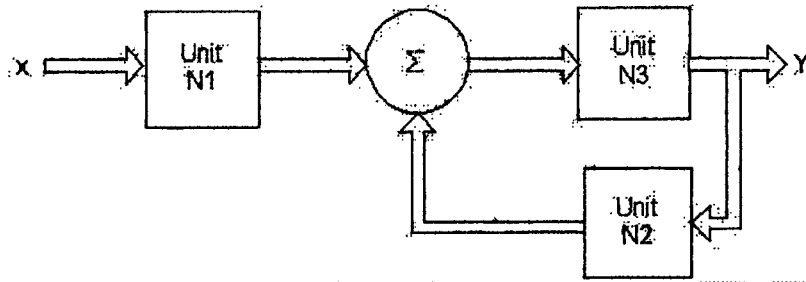
Local representation หรือเรียกอีกอย่างว่า Structured connectionist networks จะอาศัยหลักการที่ว่าให้ยูนิตแต่ละยูนิตในโครงข่าย แสดงถึงสิ่งใดสิ่งหนึ่งเลย ส่วน Distributed representation เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาดูเท่านั้น เมื่อนำมาใช้ให้ไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จะเป็นการใช้กลุ่มของยูนิต แสดงถึงสิ่งใดสิ่งหนึ่งเท่านั้น ข้อเสียของแบบ Local representation คือ เมื่อมียูนิตใดยูนิตหนึ่งได้รับอินพุตผิดพลาดหรือขาดหายไป จะทำให้สิ่งที่ยูนิตนั้นแสดงค่าแทนอยู่ ขาดหายไปจากการประมวลผลของโครงข่ายในครั้งนั้นทันที และเป็นผลให้เอาต์พุตที่ได้รับมีความคลาดเคลื่อนไปจากความต้องการ แต่ในกรณีของ Distributed representation นั้น ถ้าหากเกิดปัญหาเช่นเดียวกันขึ้นมา สิ่งทีกลุ่มยูนิตนั้นแสดงค่าแทนอยู่ก็จะไม่หายไป เนื่องจากยูนิตอื่นๆ ในกลุ่มก็ยังคงอยู่ ทำให้การประมวลผลของโครงข่ายยังคงประมวลผลได้แต่เปอร์เซ็นต์ความถูกต้องของเอาต์พุตอาจจะลดลงไปบ้างเท่านั้น ด้วยเหตุนี้มันจึงสามารถเรียนรู้สิ่งใหม่ๆ ได้โดยไม่สูญเสียสิ่งที่เรียนรู้อยู่แต่เดิม เพราะโดยปกติ การเรียนรู้หรือทดสอบสิ่งใหม่ทำให้เกิดการเปลี่ยนแปลงค่าการถ่วงน้ำหนักในแต่ละ link เพียงเล็กน้อยเท่านั้น และถึงแม้ว่าการเรียนรู้สิ่งใหม่จะทำให้ค่าการถ่วงน้ำหนักของ link บาง link เปลี่ยนแปลงไปจากเดิมมาก โครงข่ายก็ยังสามารถที่จะประมวลผลในสิ่งที่เรียนรู้อยู่แต่เดิมได้ เพราะค่าการถ่วงน้ำหนักของ link อื่นๆ ในกลุ่มยังคงอยู่ จะทำให้โครงข่ายยังคงสามารถประมวลผลในสิ่งที่เรียนรู้เดิมได้ เมื่อเปรียบเทียบกับแบบ Local representation แล้ว ถ้าค่าการถ่วงน้ำหนักใน link เปลี่ยนแปลงไปมากจากการที่เรียนรู้หรือทดสอบสิ่งใหม่ การกระตุ้นระหว่างยูนิตจะมีการเปลี่ยนแปลงไปด้วย ซึ่งยูนิตบางส่วนอาจจะไม่ได้รับการกระตุ้น ทำให้การประมวลผลของโครงข่ายขาดสิ่งที่ยูนิตนั้นแสดงไปในทันที ดังนั้นเอาต์พุตที่ได้จึงผิดพลาดไปจากความต้องการ เราจึงสามารถกล่าวได้ว่า Local representation ไม่สามารถเรียนรู้ได้ โดยทั่วไปถ้าพูดถึงระบบโครงข่ายประสาทเทียมแล้วจะหมายถึง Distributed representation นั่นเอง ในมุมมองของรูปแบบของการเชื่อมต่อระหว่างยูนิตแล้ว เราสามารถจำแนกโครงข่ายประสาทเทียมออกเป็น 2 กลุ่มด้วยกันคือ Feedforward network และ Interactive network หรือ recurrent network โดย Feedforward network จะเป็นโครงข่ายที่มีรูปแบบการเชื่อมต่อแบบทิศทางเดียว คือเมื่อมีการกระตุ้นที่อินพุต ยูนิตเหล่านั้นจะทำการแพร่กระจายการกระตุ้นไปยังยูนิต ที่อยู่ในระดับถัดไปเรื่อยๆ จนกระทั่งถึงระดับสุดท้ายหรือเอาต์พุตยูนิตดังแสดงในรูปที่ 2.3 โดยจะ ไม่มีการกระตุ้นย้อนกลับเลย ส่วนรูปแบบ Interactive network นั้นจะมีการนำการกระตุ้นที่ยูนิตระดับสุดท้ายย้อนกลับเข้ามากระตุ้นยูนิตในระดับที่ต่ำกว่าด้วย ดังนั้นรูปแบบนี้จึงถูกเรียกว่าเป็นการเชื่อมต่อแบบสองทิศทาง ดังรูปที่ 2.4



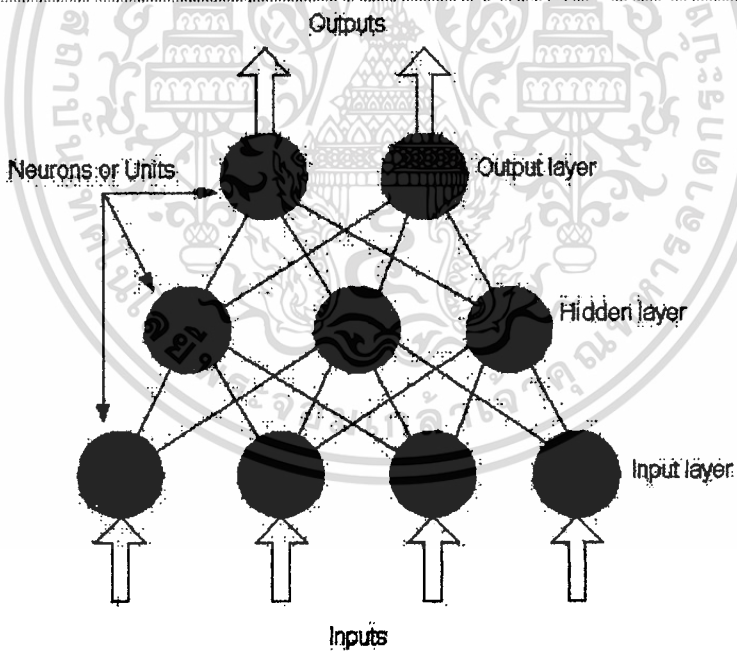
รูปที่ 2.3 การเชื่อมต่อแบบ Feedforward network

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.4 การเชื่อมต่อแบบ Interactive network

จากรูปแบบการเชื่อมต่อของยูนิตนี้เอง เราสามารถมองได้ว่า ยูนิตถูกแบ่งออกเป็นระดับชั้นต่างๆ (layer) โดยระดับชั้นแรกสุดที่รับอินพุต จะเรียกว่า Input layer และระดับชั้นสุดท้ายของยูนิต จะเรียกว่า Output layer ส่วนยูนิตที่อยู่ระหว่าง Input layer และ Output layer จะเรียกว่า Hidden layers ดังแสดงในรูปที่ 2.5



รูปที่ 2.5 การเชื่อมต่อยูนิตแบบหลายระดับชั้น

2.2.2 การกระตุ้น (Activation)

สิ่งที่ใช้ในการเชื่อมความสัมพันธ์ระหว่างยูนิตก็คือ ค่าของการกระตุ้น (Activation Value) โดยขณะเริ่มแรกแต่ละยูนิตจะถูกเซตค่าในการกระตุ้นเป็นค่าๆหนึ่ง ซึ่งเราเรียกว่า “Resting Activation” เมื่อระบบเริ่มทำงานค่าของการกระตุ้นจะแปรเปลี่ยนค่าโดยอัตโนมัติ ขึ้นอยู่กับอินพุต

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

จากภายนอกระบบ และการกระตุ้นจากยูนิตอื่นที่อยู่ภายในระบบ และค่าเหล่านี้จะสามารถเปลี่ยนแปลงได้เมื่อเวลาเปลี่ยนไป

2.2.3 link ที่เชื่อมต่อระหว่างยูนิต

หรืออาจจะเรียกว่า Weighted Connections เป็น link ที่เชื่อมต่อระหว่างยูนิต ถูกใช้ในการกำหนดความสามารถในการกระตุ้นของยูนิตใดยูนิตหนึ่ง เพื่อให้ยูนิตที่ต่ออยู่กับยูนิตนั้น เกิดการเปลี่ยนแปลง เมื่อมีการเปลี่ยนแปลงที่ตัวมันเกิดขึ้น โดยการเปลี่ยนแปลงที่เกิดจากการกระตุ้นของยูนิตนั้น อาจเพิ่มหรือลดค่าการกระตุ้นของอีกยูนิตหนึ่ง ขึ้นอยู่กับค่าน้ำหนักใน link นั้น ในระบบการเชื่อมต่อยูนิตแบบสองทิศทาง ยูนิต I จะเชื่อมต่อกับยูนิต U ด้วยค่าน้ำหนัก W_{IU} ในทางกลับกันยูนิต U จะเชื่อมต่อยูนิต I ด้วยค่าน้ำหนัก W_{UI} โดยทั้งสองค่านี้จะมีค่าเท่ากันหรือต่างกันก็ได้

2.2.4 กฎในการเรียนรู้ (Learning Rules)

ในระบบโครงข่ายประสาทเทียมนั้น เราสามารถจำแนกรูปแบบในการเรียนรู้ออกเป็น 2 กลุ่มใหญ่ๆด้วยกัน คือ การเรียนรู้แบบมีผู้สอน (Supervised Learning) และการเรียนรู้แบบไม่มีผู้สอน (Unsupervised Learning) ในการเรียนรู้แบบมีผู้สอนนั้น เราจะต้องป้อนอินพุตและเอาต์พุตให้แก่ระบบโครงข่ายประสาทเทียม โดยการป้อนจะต้องป้อนในลักษณะเป็นคู่ๆ เช่น ถ้าเราต้องการสอนให้มันบวกเลข เราจะต้องป้อนค่าสมการที่อินพุต และป้อนผลลัพธ์ที่เอาต์พุต แล้วระบบจึงจะเกิดการเรียนรู้ขึ้น ส่วนการเรียนรู้แบบไม่มีผู้สอนนั้น จะใช้กับปัญหาจำพวกที่ไม่มีคำตอบที่แน่นอนหรือเราไม่รู้คำตอบมาก่อนเลย และก็ไม่มีความชัดเจนที่แน่นอนในการแก้ปัญหาด้วย เช่น ปัญหาการจัดกลุ่มต่างๆ อินพุตคือให้ระบบโครงข่ายประสาทเทียม แบ่งกลุ่มคนจำนวนหนึ่งออกเป็นกลุ่มย่อยๆ จะเห็นได้ว่ามันไม่มีคำตอบที่แน่นอนว่าจะจัดกลุ่มออกมาในรูปแบบใด ขึ้นอยู่กับวิธีการที่ระบบโครงข่ายประสาทเทียมนำมาคิด

วิธีการเรียนรู้หรือกฎในการเรียนรู้ในระบบโครงข่ายประสาทเทียมได้มีผู้คิดค้นขึ้นมามากมายหลายวิธีด้วยกัน แต่ละวิธีก็มีข้อดีข้อเสียแตกต่างกันออกไป บางวิธีอาจจะเหมาะกับงานชนิดหนึ่ง แต่อาจจะไม่เหมาะกับงานอีกชนิดหนึ่งก็เป็นได้ ทั้งนี้วิธีการเรียนรู้ ยังขึ้นอยู่กับรูปแบบโครงสร้างของโครงข่ายประสาทเทียม และรูปแบบการทำงานที่ต้องการให้โครงข่ายประสาทเทียมทำ

2.3 แนวคิดพื้นฐานของการเรียนรู้ข้อมูลเชิงลำดับ

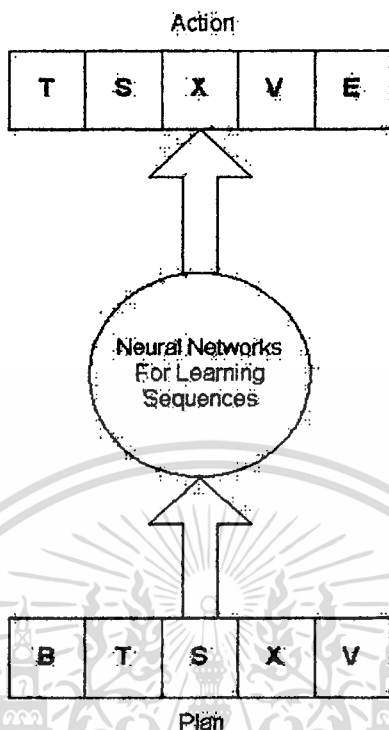
การเรียนรู้ข้อมูลเชิงลำดับ เป็นการพยายามให้โครงข่ายประสาทเทียมเลียนแบบพฤติกรรมเชิงลำดับของมนุษย์ เช่นการพูดอย่างต่อเนื่องของมนุษย์ จะเป็นในลักษณะที่ใช้คำพูดแต่ละคำเรียงกันอย่างเป็นลำดับ และพูดออกมาอย่างต่อเนื่อง (Cleeremans, 1993) ด้วยแนวคิดดังกล่าว จึงได้มีนักวิจัยจำนวนมาก ทำการค้นคว้าวิจัยและออกแบบสถาปัตยกรรมโครงข่ายประสาทเทียมใหม่ๆขึ้นมา เพื่อให้สามารถเรียนรู้ข้อมูลในรูปแบบที่เป็นเชิงลำดับได้

โดยปรกติข้อมูลเชิงลำดับที่มีการกำหนดลำดับข้อมูลไว้จะถูกเรียกว่า แพลน (Plan) เมื่อทำการป้อนลำดับข้อมูลเหล่านี้ให้แก่อินพุตของโครงข่ายประสาทเทียมที่มีการเรียนรู้ข้อมูลเชิงลำดับ โครงข่ายจะให้ผลลัพธ์ หรือ แอ็คชั่น (Action) ที่เอาต์พุตของโครงข่าย ซึ่งการตอบสนองของโครงข่ายต่อข้อมูลเชิงลำดับที่ป้อนเข้ามาแต่ละลำดับจะเรียกว่า สถานะ (State) โดยการเปลี่ยนแปลง สถานะจะมีผลกับแอ็คชั่นโดยตรง ดังนั้นเมื่อลำดับข้อมูลในแพลนเปลี่ยนไป เป็นผลให้สถานะมีการเปลี่ยนแปลงและเกี่ยวเนื่องไปถึงแอ็คชั่นมีการเปลี่ยนแปลงตามไปด้วย ผลลัพธ์ที่ได้จากเอาต์พุตของโครงข่ายคือ แอ็คชั่นที่เป็นเชิงลำดับที่สอดคล้องกับแพลนนั่นเอง

เราสามารถเขียนอธิบายความสัมพันธ์ของสิ่งต่างๆด้วยสมการ โดยกำหนดให้ X_1, X_2, \dots, X_n แทนลำดับของแอ็คชั่นที่เอาต์พุต "P" แทนแพลนที่ป้อนเข้าที่อินพุต และ S_n แทนสถานะปัจจุบันของโครงข่าย

$$S_{n+1} = g(S_n, P) \quad (1)$$

สมการ (1) เป็นสมการจากทฤษฎี Automata theory โดยฟังก์ชัน g ถูกใช้กำหนด สถานะถัดไป (Next-State) ของโครงข่าย ซึ่งขึ้นอยู่กับ สถานะปัจจุบันและแพลนหรือลำดับข้อมูลอินพุตที่เข้ามาขณะนั้น สมการนี้เองที่ถูกใช้เป็นพื้นฐานในการออกแบบโครงข่ายประสาทเทียมให้สามารถเรียนรู้ข้อมูลเชิงลำดับได้ โดยให้โครงข่ายสามารถประมวลผลได้ดังสมการดังกล่าว คือโครงข่ายประสาทเทียมสามารถให้ผลลัพธ์ทางเอาต์พุต เป็นแอ็คชั่นลำดับถัดไป (เป็นผลมาจากการสร้าง สถานะถัดไปนั่นเอง) ถึงแม้ว่า อินพุตจะเป็นลำดับข้อมูลปัจจุบันของแพลน เช่น แพลน TSXV เมื่อป้อนข้อมูลลำดับแรกให้แก่อินพุตของโครงข่ายคือ "T" เอาต์พุตที่ได้จากการประมวลผลของโครงข่ายคือ แอ็คชั่น "S" ในลำดับถัดมาเมื่อทำการป้อนข้อมูลลำดับที่สองของแพลนให้แก่โครงข่ายคือ "S" (จะเห็นได้ว่าเป็นค่าเดียวกันกับเอาต์พุตในลำดับก่อนหน้า) เอาต์พุตที่ได้จากการประมวลผลของโครงข่ายในลำดับนี้คือ "X" นั่นเอง ดังตัวอย่างในรูปที่ 2.6 ซึ่งการประมวลผลของโครงข่ายดังกล่าว เสมือนกับว่าโครงข่ายสามารถคาดการณ์อนาคตได้นั่นเอง



รูปที่ 2.6 รูปแบบการประมวลผลของโครงข่ายประสาทเทียมสำหรับการเรียนรู้ข้อมูลเชิงลำดับ

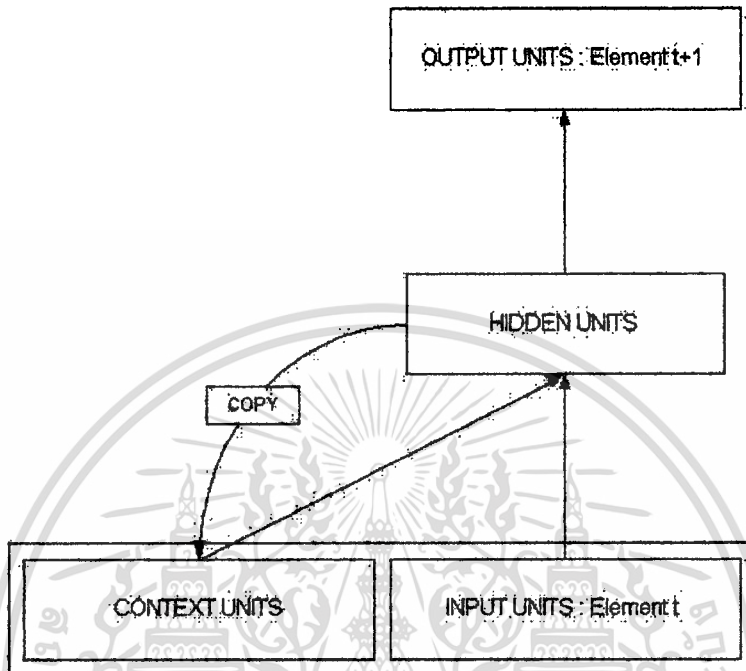
ปัจจุบัน ได้มีนักวิจัยจำนวนมากทำการออกแบบโครงข่ายประสาทเทียมสำหรับการเรียนรู้ข้อมูลเชิงลำดับบนพื้นฐานหลักการดังที่กล่าวมา เช่น สถาปัตยกรรมโครงข่ายแบบ Simple Recurrent Network และ Jordan's Network เป็นต้น โดยสถาปัตยกรรมทั้งสองแบบต่างก็ใช้หลักการของ Back propagation learning rule เป็นวิธีในการเรียนรู้ ซึ่งจะอาศัยหลักการคำนวณค่าความผิดพลาดที่เกิดขึ้นที่เอาต์พุต โดยการเปรียบเทียบกับที่ได้จริงจากเอาต์พุตกับค่าที่ต้องการ ซึ่งค่าความผิดพลาดที่ได้นี้จะถูกใช้ในการปรับค่าการถ่วงน้ำหนักของ link ในการเชื่อมต่อ ด้วยการแพร่กระจายค่าความผิดพลาดกลับเข้ามายังโครงข่าย ทำหน้าที่เสมือนมียูนิตเพิ่มเข้ามาในโครงข่าย โดยทำหน้าที่กระตุ้นให้ยูนิตหลักในโครงข่ายเกิดการปรับค่าการถ่วงน้ำหนักในการเชื่อมต่อ ให้ได้ค่าที่เอาต์พุตตรงกับความต้องการมากที่สุด

2.4 โครงข่ายประสาทเทียมแบบ Simple Recurrent Network

ในปี 1988 Elman ได้ทำการนำเสนอสถาปัตยกรรมโครงข่ายประสาทเทียมที่เขาได้คิดค้นขึ้นมา และเรียกมันว่า Simple Recurrent Network (SRN) (Cleeremans, 1993) โดยสถาปัตยกรรมดังกล่าวมีคุณสมบัติในการเรียนรู้ข้อมูลเชิงลำดับได้เป็นอย่างดี จึงทำให้สถาปัตยกรรมโครงข่ายนี้เป็น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ที่นิยมในหมู่นักวิจัย ที่จะนำเอาไปประยุกต์ใช้ในงานด้านการเรียนรู้ข้อมูลเชิงลำดับ ซึ่งเป็นผลให้เกิดงานวิจัยอื่นๆตามมาอีกจำนวนมาก



รูปที่ 2.7 ถึงสถาปัตยกรรมโครงข่ายแบบ Simple Recurrent Network

จากรูปที่ 2.7 แสดงถึงสถาปัตยกรรมของ Simple Recurrent Network จะประกอบไปด้วย 3 ระดับชั้นด้วยกันคือ Input layer, Output layer และ Hidden layer โดยในส่วนของ Input layer จะประกอบไปด้วยกลุ่มของยูนิต 2 กลุ่มด้วยกัน คือกลุ่มของ Input units ทำหน้าที่รับข้อมูลเชิงลำดับ หรือแปลนข้อมูลจากภายนอกโครงข่าย และกลุ่มของ Context units ทำหน้าที่นำค่าการกระตุ้นจาก Hidden units ที่กระทำต่อ Output units ในช่วงเวลาก่อนข้อมูลอินพุตลำดับปัจจุบัน (ซึ่งก็คือค่าการกระตุ้นที่ทำให้ Output units ให้ผลลัพธ์เท่ากับข้อมูลปัจจุบัน เนื่องจากการประมวลผลที่เสมือนการคาดการณของ Hidden units ในเวลาก่อนหน้านั้นนั่นเอง) มาป้อนให้กับ Hidden units เป็นผลให้สถาปัตยกรรมโครงข่ายแบบนี้สามารถประมวลผลเลียนแบบสมการ Automata theory ได้ กล่าวคือ ข้อมูลที่เข้ามาทาง Input units เทียบได้กับลำดับข้อมูลในแปลน ส่วนการกระตุ้นที่ได้จาก Context units เทียบได้กับสเตตปัจจุบัน (การตอบสนองของโครงข่ายต่อข้อมูลอินพุตลำดับก่อนหน้าข้อมูลปัจจุบันที่ก่อให้เกิดผลลัพธ์เท่ากับข้อมูลปัจจุบัน หรือการตอบสนองของโครงข่ายที่ก่อให้เกิดผลลัพธ์เท่ากับข้อมูลปัจจุบันนั่นเอง) ซึ่งจะเห็นได้ว่าครบตามองค์ประกอบของฟังก์ชัน g ในสมการ (1) และเนื่องจากโครงข่ายถูกฝึก (Train) ให้เกิดการสร้างการกระตุ้นที่เป็นสเตตถัดไป ทำให้

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Hidden units สามารถสร้างการกระตุ้นที่เป็นสเตทถัดไปขึ้นมาแล้ว ส่งค่าการกระตุ้นไปยัง Output units ผลลัพธ์ที่ได้ทางเอาต์พุตก็คือ แอ็คชันลำดับถัดไปนั่นเอง

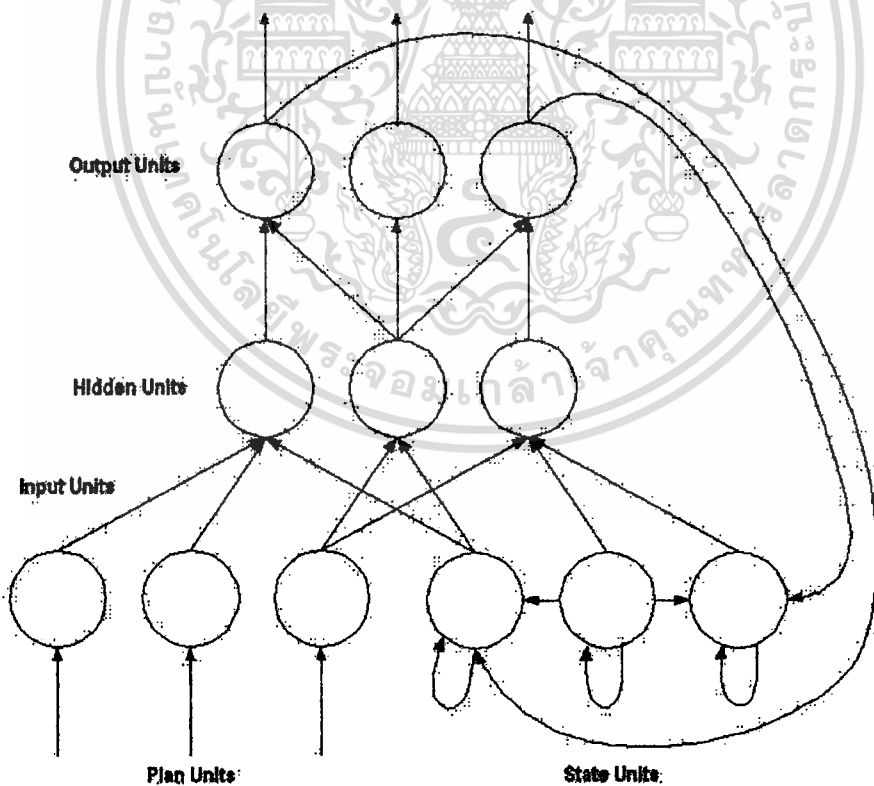
โดยปรกติแล้ว เราสามารถใช้สถาปัตยกรรมโครงข่ายแบบทั่วไป เช่น Feedforward Network ทำการฝึกหัดให้มันสามารถสร้างการกระตุ้นที่เป็นสเตทถัดไปได้ และสามารถประมวลผลได้เสมือนว่าคาดการณ์อนาคตได้เช่นกัน แต่สิ่งหนึ่งที่โครงข่ายแบบธรรมดาทำไม่ได้คือมันไม่สามารถจำแนกข้อมูลเชิงลำดับที่เหมือนกันแต่ถูกป้อนให้โครงข่ายต่างเวลากันได้ เช่นข้อมูลเชิงลำดับที่ป้อนให้โครงข่ายมีแพทเทิร์นนี้ PVXXTXS ถ้าเป็นโครงข่ายธรรมดา เมื่อป้อนข้อมูลถึงลำดับที่สามคือ PVX ผลลัพธ์ที่ได้จากเอาต์พุต คือ PVXX ซึ่งถูกต้องตามลำดับที่ต้องการ เมื่อข้อมูลลำดับถัดมาถูกป้อนให้แก่โครงข่าย PVXX ผลลัพธ์ที่ได้จากโครงข่ายแบบธรรมดาคือ PVXXX ซึ่งผิดจากลำดับที่ต้องการ เนื่องจากโครงข่ายแบบธรรมดาไม่สามารถฝึกหัดให้ได้ผลลัพธ์ต่างกันออกไปเมื่ออินพุตที่เหมือนเดิมถูกป้อนเข้ามาในระบบ เพราะโครงข่ายแบบธรรมดาให้ผลลัพธ์ทางเอาต์พุตโดยไม่คำนึงถึงข้อมูลลำดับก่อนหน้านั้น แต่ถ้าเป็นในกรณีของโครงข่ายแบบที่สามารถเรียนรู้ข้อมูลเชิงลำดับได้นั้น เอาต์พุตที่ได้จะตรงกับความต้องการ เนื่องจากว่าผลลัพธ์ของเอาต์พุตจะขึ้นอยู่กับข้อมูลที่อยู่ลำดับก่อนหน้านั้นด้วย สังเกตได้จากสมการ (1) คือ สเตทถัดไปจะขึ้นอยู่กับข้อมูลอินพุตและสเตทปัจจุบัน ดังนั้นถึงแม้ว่าข้อมูลทางอินพุตจะเหมือนเดิม แต่ผลลัพธ์ที่ได้ทางเอาต์พุต ก็อาจจะไม่เหมือนเดิมก็ได้ เพราะสเตทปัจจุบันของแต่ละช่วงเวลาจะต่างกันออกไป จึงทำให้โครงข่ายสามารถจำแนกได้ว่า ได้ทำการประมวลผลข้อมูลถึงลำดับใดแล้ว

ข้อเสียของ โครงข่ายแบบ Simple Recurrent Network คือ โครงข่ายแบบนี้แยกความแตกต่างระหว่างข้อมูลเชิงลำดับที่มีข้อมูลบางส่วนที่เหมือนกันและต่อเนื่องกันได้ไม่ดีนัก เช่นแพทเทิร์นข้อมูล PSSSP และ TSSST เนื่องจากข้อมูลลำดับสุดท้ายจะขึ้นอยู่กับข้อมูลลำดับแรกสุดเป็นตัวจำแนกว่าเอาต์พุตสุดท้ายจะเป็น P หรือ T ซึ่งเสมือนว่าต้องการให้โครงข่ายคาดการณ์ระยะไกล แต่โดยคุณสมบัติของโครงข่ายแบบ Simple Recurrent Network แล้วสามารถคาดการณ์ได้ ในระยะใกล้เท่านั้น เพราะว่าการตัดสินใจว่าสเตทใดคือสเตทถัดไปขึ้นอยู่กับข้อมูลอินพุต และสเตทปัจจุบัน ซึ่งในที่นี้ข้อมูลอินพุตลำดับรองสุดท้ายของแพทเทิร์นทั้งสองคือ S เหมือนกัน และสเตทปัจจุบันก็มีค่าที่เหมือนกัน เนื่องจากข้อมูล S ที่เกิดขึ้นหลายๆครั้ง ดังนั้นจึงทำให้โครงข่ายแบบ Simple Recurrent Network ไม่สามารถจำแนกความแตกต่างได้ แต่ถ้ามีการเพิ่มจำนวน Hidden units แล้ว เป็นผลให้สเตทปัจจุบันอาจจะไม่เหมือนกันก็ได้ และจะเป็นผลให้โครงข่ายสามารถจำแนกความแตกต่างได้ แต่ถ้าข้อมูลมีจำนวนซ้ำกันมากขึ้น สเตทปัจจุบันอาจจะกลับมาเหมือนกันอีกก็เป็นได้ ดังนั้นจึงกล่าวได้ว่าโครงข่ายแบบนี้ไม่เหมาะสมกับข้อมูลในลักษณะนี้ (Bechtel, 1997)

2.5 โครงข่ายประสาทเทียมแบบ Jordan 's Network

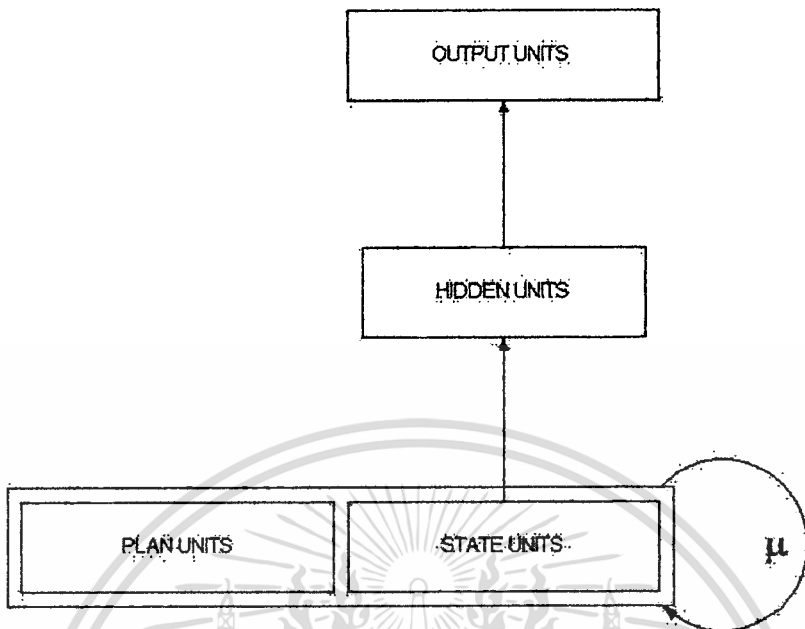
สถาปัตยกรรมโครงข่ายแบบ Jordan's Network (Jordan, 1986) ถูกนำเสนอในปี 1986 โดย Jordan ซึ่งโครงข่ายแบบนี้ถูกพัฒนาขึ้นมาบนพื้นฐานของทฤษฎี Automata theory เช่นเดียวกับ Simple Recurrent Network และได้รับความนิยมในการนำไปประยุกต์ใช้งานไม่แพ้กับแบบ Simple Recurrent Network เลย

จากรูปที่ 2.8 และ 2.9 แสดงถึงสถาปัตยกรรมโครงข่ายแบบ Jordan's Network จะแสดงให้เห็นระดับชั้นของ Input layer, Output layer และ Hidden layer โดยในส่วนของ Input layer จะประกอบไปด้วยกลุ่มของยูนิต 2 กลุ่มด้วยกันคือ Plan units ทำหน้าที่รับข้อมูลเชิงลำดับจากภายนอกโครงข่าย และ State units ทำหน้าที่รับการกระตุ้นย้อนกลับจากเอาต์พุต และการย้อนกลับภายในตัวมันเอง แล้วจึงส่งการกระตุ้นไปยัง Hidden units ส่วน Output units ทำหน้าที่ให้ผลลัพธ์หรือแอ็คชันที่เกิดจากการประมวลผลของโครงข่าย โดยปรกติแล้วเอาต์พุตของโครงข่ายจะไม่เป็นในรูปเชิงเส้น ดังนั้น โครงข่ายจึงจำเป็นต้องมี Hidden layer อีกชั้นหนึ่ง



รูปที่ 2.8 สถาปัตยกรรมโครงข่ายแบบ Jordan's Network (ไม่ครบทุก Link)

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 2.9 สถาปัตยกรรม โครงข่ายแบบ Jordan's Network

สมการเอาต์พุตฟังก์ชันของโครงข่ายแบบ Jordan's Network คือ $X_n = f(S_n, P)$ เมื่อ " X_n " คือแเอ็คชันเมื่อเวลา n " P " คือแพลนหรือลำดับข้อมูลที่ป้อนเข้ามาทางอินพุต และ S_n คือ สเตทปัจจุบัน และจากการป้อนกลับจาก Output units มายัง State units และการป้อนกลับจาก State units เข้ามายังตัวมันเอง ทำให้สามารถสร้างสเตทถัดไปได้ โดยอ้างจากสมการ (1) เมื่อการป้อนกลับจาก Output units เป็นเสมือนข้อมูลอินพุตที่เข้ามา และการป้อนกลับตัวมันเอง เป็นเสมือนสเตทปัจจุบัน เป็นผลให้ State units สามารถประมวลผลสร้างการกระตุ้นที่เป็นสเตทถัดไปขึ้นมาส่งไปยัง Hidden units เมื่อรวมกับการกระตุ้นจาก Plan units ทำให้ Hidden units สามารถทำการประมวลผลสร้างการกระตุ้นที่ทำให้เกิดแเอ็คชันในลำดับถัดไปได้ เนื่องจาก X_n ขึ้นอยู่กับสเตทปัจจุบัน และ แพลนแต่ละขณะนี้สเตทปัจจุบันที่กระตุ้นจาก State units เป็นสเตทถัดไป ดังนั้นแเอ็คชันที่ได้จึงเป็นแเอ็คชันในลำดับถัดไป นั่นก็คือระบบโครงข่ายแบบนี้สามารถประมวลผลเสมือนกับการคาดการณ์อนาคตได้เช่นกัน

จากการป้อนกลับตัวมันเองของ State units ด้วยค่าการถ่วงน้ำหนักเท่ากับ μ และการป้อนกลับจาก Output units ด้วยค่าการถ่วงน้ำหนักเท่ากับ 1 ทำให้ได้สมการอินพุตของสเตทที่เวลา n ดังนี้ $S_n = \mu S_{n-1} + X_{n-1}$ จะเห็นได้ว่าอินพุตของสเตทปัจจุบันจะเกิดจากการรวมกันระหว่างเอาต์พุตของสเตทในเวลาก่อนหน้านี้นี้คูณกับค่าการถ่วงน้ำหนัก รวมกับค่าแเอ็คชันก่อนหน้านี้นี้ ด้วยเหตุนี้จึงเสมือนว่าสเตทปัจจุบันเป็นการผสมผสาน (modulate) กันของข้อมูลในอดีต ดังนั้นเมื่อทำการป้อนข้อมูลเชิงลำดับที่มีลักษณะข้อมูลในลำดับบางส่วนเหมือนกันและต่อเนื่องกันเช่นเดียวกับที่ Simple

เอกซามเปิลการคำนวณค่า

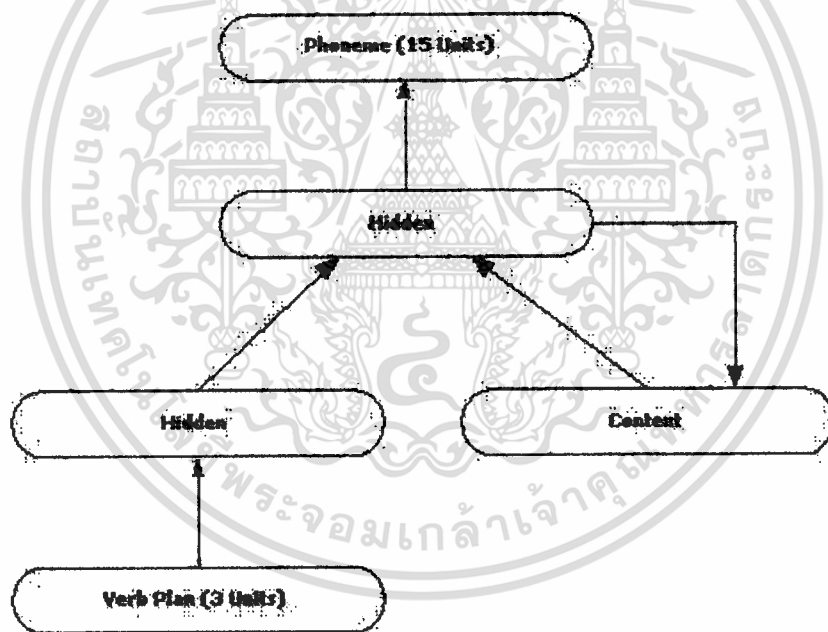
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

Recurrent Network พบปัญหา จะเห็นได้ว่าโครงข่ายแบบ Jordan's Network สามารถจำแนกความแตกต่างระหว่างข้อมูลเชิงลำดับสองชุดได้อย่างถูกต้อง เนื่องจากข้อมูลในอดีตที่ผสมผสานกันทำให้สเตตปัจจุบันของทั้งสองชุดไม่เหมือนกัน (เพราะข้อมูลเริ่มต้นของทั้งสองชุด ต่างกัน) ดังนั้นโครงข่ายจึงสามารถประมวลผลได้ผลลัพธ์ที่เอาต์พุตอย่างถูกต้อง

2.6 ตัวอย่างการประยุกต์ใช้งานโครงข่ายโครงข่ายประสาทเทียมสำหรับการเรียนรู้ข้อมูลเชิงลำดับ

ในปี 1998 Miikkulainen และ Dyer (Miikkulainen and Dyer, 1998:49-56) ได้ประยุกต์ใช้โครงข่ายประสาทเทียมสำหรับการเรียนรู้ข้อมูลเชิงลำดับในงานการถ่ายข้อความ จากบทความเรื่องราวต่างๆเรื่องหนึ่งไปเป็นอีกบทความหนึ่ง ที่มีรายละเอียดมากขึ้น เช่น บทความขนาดสั้น "John went to Loen's. John asked the waiter for lobster. John gave a large tip" เมื่อผ่านการประมวลผลของโครงข่ายประสาทเทียมในงานการถ่ายข้อความแล้ว จะได้บทความดังนี้ "John went to Leone's. The waiter seated John. John asked the waiter for lobster. The waiter brought John the lobster. John ate the lobster. The lobster tasted good. John paid the waiter. John gave a large tip. John left Leone's." เป็นต้น โดยเรื่องราวที่ระบบโครงข่ายประสาทเทียมนี้จะสามารถถ่ายข้อความได้ เป็นเรื่องราวต่างๆที่มีแบบแผนที่ไม่ซับซ้อน และพบอยู่โดยทั่วไป เช่น การไปพบแพทย์ การเดินทาง โดยเครื่องบิน การเข้าการประชุม การไปร้านอาหาร เป็นต้น ระบบโครงข่ายจะมองเรื่องราวที่เกิดขึ้นในรูปของเหตุการณ์ที่เกิดขึ้นอย่างต่อเนื่องเป็นลำดับ และสามารถอนุมานเหตุการณ์ที่มีอยู่ในเรื่องราว สามารถเพิ่มเติมเหตุการณ์ที่เกี่ยวข้องลงไปในเรื่องราวได้ รวมถึงสามารถเข้าใจถึงสรรพนามที่ปรากฏอยู่ในรูปประโยคของเรื่องราวได้ โครงสร้างของระบบนี้เกิดจากการนำโครงข่ายประสาทเทียมเชิงลำดับสี่โครงข่ายมาเชื่อมต่อเข้าด้วยกันดังแสดงในรูปที่ 2.10 เพื่อทำหน้าที่ในการประมวลผลต่างๆกันไป การแทนค่าของคำต่างๆในระบบนี้จะถูกเก็บไว้ในโครงข่ายฐานข้อมูล คำศัพท์ภายนอก การทำงานของระบบนี้เริ่มจากประโยคแรกของเรื่องราวถูกดึงจากโครงข่ายฐานข้อมูลคำศัพท์คำต่อคำไปยังอินพุตของโครงข่ายในส่วนของการวิเคราะห์ประโยค เอาต์พุตที่ได้จากโครงข่ายนี้จะอยู่ในรูปของประโยคที่ผ่านการวิเคราะห์แล้ว หลังจากนั้นประโยคดังกล่าวจะถูกนำไปป้อนเป็นอินพุตของโครงข่ายที่สองเพื่อทำการวิเคราะห์เหตุการณ์ รวมถึงการเพิ่มเติมรายละเอียดที่เกี่ยวข้องเข้าไปในเรื่องราวด้วย เอาต์พุตที่ได้จากโครงข่ายนี้ จะถูกนำไปป้อนยังโครงข่ายที่สาม เพื่อทำการสร้างเหตุการณ์ของเรื่องราวจากรายละเอียดต่างๆที่ถูกเพิ่มเติมเข้าไป และเหตุการณ์ที่ถูกสร้างขึ้นนี้จะถูกป้อนเข้าไปเป็นอินพุตของโครงข่ายสุดท้าย ที่ทำหน้าที่สร้างคำของเรื่องราวที่ผ่านการถ่ายข้อความแล้วนั่นเอง

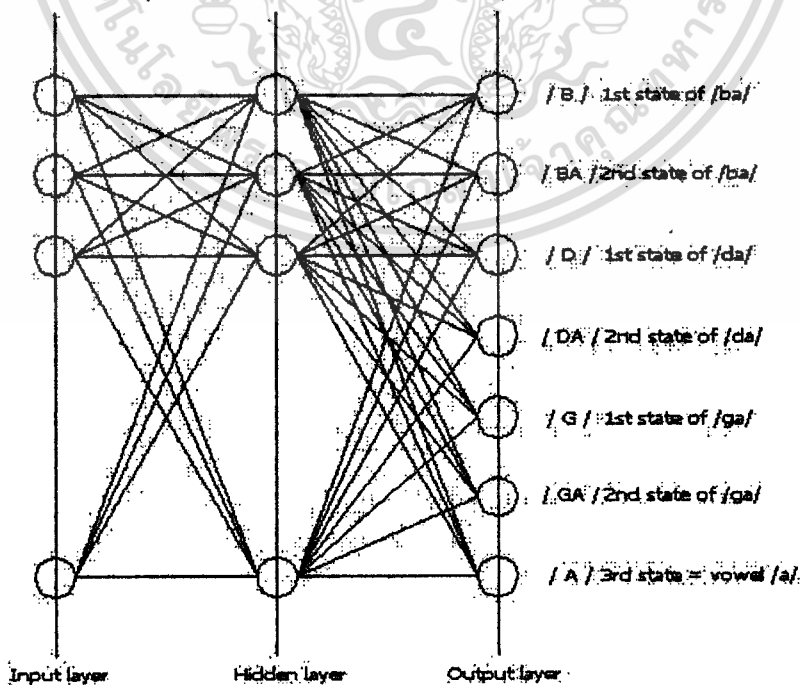
Simple Recurrent Networks ที่นำมาประยุกต์ใช้ในงานนี้ ในส่วนเอาต์พุตของโครงข่ายจะประกอบไปด้วยยูนิตจำนวน 15 ยูนิตซึ่งใช้แสดงลำดับข้อมูลของเสียงขนาด 15 บิต ตามผลของข้อมูลที่แสดงถึงคำกริยาปัจจุบัน หรือคำกริยาที่แสดงถึงเหตุการณ์ในอดีตที่เข้ามาทางอินพุตขนาด 2 บิต และอีก 1 บิต สำหรับใช้ระบุถึงคำกริยานั้นๆว่าเป็นคำกริยาที่ระบุถึงเหตุการณ์ปัจจุบัน หรือระบุถึงเหตุการณ์ในอดีต นั่นคือที่ Input layer ของโครงข่ายนี้จะประกอบไปด้วย ยูนิตจำนวน 3 ยูนิต และยังมีการป้อนกลับจาก hidden units มายัง context units เพื่อให้ context units ทำหน้าที่เสมือนข้อมูลอินพุตของโครงข่ายตามรูปแบบโครงสร้างของโครงข่ายประสาทเทียมแบบ Simple Recurrent Networks ซึ่งผลที่ได้จากการวิจัยครั้งนี้ เขาได้พบว่าโครงข่ายนี้ให้ผลลัพธ์ของการแปลงคำกริยาแสดงเหตุการณ์ในปัจจุบัน ไปเป็นคำกริยาที่แสดงเหตุการณ์ในอดีตได้ไม่ดีเท่ากับการแปลงคำกริยาแสดงเหตุการณ์ในอดีต ไปเป็นคำกริยาแสดงเหตุการณ์ปัจจุบัน และโครงข่ายนี้ยังทำการเรียนรู้ชุดของตัวอักษรที่มีความยาวไม่สม่ำเสมอได้ไม่ดีนัก



รูปที่ 2.11 โครงข่ายประสาทเทียมสำหรับการเรียนรู้ข้อมูลเชิงลำดับในงานการแปลความหมายของคำไปเป็นการออกเสียงของคำ

ในปี 1991 Luz, Kobayashi และ (Niimi Luz et. al, 1991:140-145) ได้ทำการประยุกต์ใช้โครงข่ายประสาทเทียมสำหรับการเรียนรู้ข้อมูลเชิงลำดับในงานทางด้านความรู้จำเสียงพูด โดยเน้นไปทางการรู้จำศัพท์พยางค์เดียว (monosyllable) ในภาษาญี่ปุ่น โดยพื้นฐานการประยุกต์นี้เกิดจากความต้องการการควบคุมคำพูดที่พูดออกมาอย่างเป็นลำดับ ซึ่งโครงข่ายประสาทเทียมแบบธรรมดาสามารถประมวลผลงานการรู้จำเสียงแบบที่เสียงถูกแบ่งออกเป็นช่วงๆได้เป็นอย่างดี แต่ไม่สามารถเอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาค้นคว้าเท่านั้น เมื่อนุญญาติให้ไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ประมวลผลได้ดีนัก ถ้าเสียงที่ต้องการจำแนกเป็นเสียงในลักษณะต่อเนื่องกัน ตัวอย่างของปัญหานี้ จะพบได้ในงาน เช่น การจำแนกคำศัพท์พยางค์เดียวเป็นต้น ดังนั้นจึงได้มีแนวคิดที่จะนำโครงข่ายประสาทเทียมแบบ Multi-Layer Perceptron (MLP) ร่วมกับวิธีการแบบ dynamic programming algorithm มาใช้ในการประมวลผลข้อมูลเสียงในลักษณะของโครงข่ายแบบเชิงลำดับ เพื่อแก้ปัญหา ดังที่ได้กล่าวมาข้างต้น ในงานวิจัยนี้ได้กำหนดขอบเขตความรู้จำไว้เพียงแค่งาน BDG task เท่านั้น นั่นคือการจำแนกเฉพาะหน่วยเสียง (phonemes) /b/, /d/, และ /g/ ซึ่งในหน่วยของเสียงนี้จะประกอบไปด้วยศัพท์พยางค์เดียว 13 รูปแบบ และถูกเรียกว่า /ba/, /da/, /ga/, /be/, /de/, /ge/, /bi/, /gi/, /bo/, /do/, /go/, /bw และ /gw/ โดยในการประมวลผลงานนี้จะต้องใช้โครงข่ายประสาทเทียม 5 โครงข่ายตามจำนวนของสระทั้ง 5 ที่ปรากฏอยู่ในศัพท์พยางค์เดียว โดยทั่วไปแล้วเอาต์พุตโครงข่ายจะถูกแยกแยะระหว่าง /ba/, /da/ และ /ga/ ดังตัวอย่างรูปที่ 2.12 จะแสดงให้เห็นว่า โครงข่ายจะประกอบไปด้วยกัน 7 เซต โดย 3 เซตแสดงถึงพยัญชนะ /b/, /d/ และ /g/ อีก 1 เซตแสดงถึงสระ และอีก 3 เซตแสดงการเปลี่ยนแปลงจากพยัญชนะไปเป็นสระ ซึ่งแต่ละเซตจะแสดงถึงผลลัพธ์ที่ได้จากการประมวลผลของโครงข่าย โดยจะมีเพียง 1 เอาต์พุตยูนิตเท่านั้นที่จะให้ผลลัพธ์ออกมา ในส่วนของ Input layer จะประกอบไปด้วยยูนิตจำนวน 28 ยูนิต รับข้อมูลเสียงที่ถูกแบ่งออกเป็นช่วงๆ ได้ 2 ช่วง โดยหนึ่งช่วงของเสียงจะต้องใช้จำนวนยูนิตเท่ากับ 14 ยูนิต ทำหน้าที่รับข้อมูลเสียงที่เข้ามาอย่างเป็นลำดับและต่อเนื่องกันเข้ามาในโครงข่ายเพื่อให้โครงข่ายทำการประมวลผล ซึ่งผลที่ได้จากการทดลองพบว่า งานประยุกต์นี้สามารถให้ผลลัพธ์ในการรู้จำเสียงได้เป็นอย่างดี



รูปที่ 2.12 โครงข่าย BA/DA/GA

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆ ทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

บทที่ 3

การออกแบบและการทดลองใช้งานโครงข่ายประสาทเทียมเชิงลำดับ

3.1 แนวคิดในการประยุกต์ใช้โครงข่ายประสาทเทียมสำหรับการเรียนรู้ข้อมูลเชิงลำดับกับงานทางด้าน การตัดคำภาษาไทย

การที่มนุษย์เราจะตั้งงานหรือควบคุมให้คอมพิวเตอร์ทำงานอะไรได้นั้น มนุษย์จะต้องศึกษาทำความเข้าใจในภาษาที่คอมพิวเตอร์เข้าใจ เพื่อเป็นสื่อให้มนุษย์และคอมพิวเตอร์สื่อสารกันได้ ซึ่งก่อให้เกิดความยุ่งยากอย่างยิ่งที่มนุษย์จะต้องจดจำรูปแบบของภาษาที่คอมพิวเตอร์เข้าใจ จึงได้เกิดแนวคิดที่จะทำให้คอมพิวเตอร์มีความเข้าใจภาษาธรรมชาติของมนุษย์ ซึ่งเป็นผลก่อให้เกิดศาสตร์ที่เรียกว่าการประมวลผลภาษาธรรมชาติ หรือ Natural Language Processing (NLP) ขึ้นมา โดยประโยชน์ที่ได้รับจะส่งผลให้การใช้งานคอมพิวเตอร์เป็นไปอย่างสะดวกมากขึ้น คอมพิวเตอร์จะสามารถเข้ามาช่วยในงานด้านต่างๆ ที่เกี่ยวข้องกับภาษาได้มากขึ้น เช่น เป็นเครื่องแปลภาษามนุษย์จากภาษาหนึ่งไปเป็นอีกภาษาหนึ่ง สามารถนำคอมพิวเตอร์มาช่วยตรวจและวิเคราะห์เอกสารต่างๆ ที่มีว่าเกี่ยวข้องกับเรื่องใด สามารถให้คอมพิวเตอร์ช่วยในการค้นคืนข้อมูลตามความต้องการของผู้ใช้ได้ หรือสามารถให้คอมพิวเตอร์ช่วยสรุปสาระและประเด็นสำคัญที่ปรากฏในเอกสารนั้น เป็นต้น

โดยพื้นฐานของการประมวลผลภาษาธรรมชาติแล้วการ วิเคราะห์รูปประโยคเป็นสิ่งสำคัญอย่างยิ่ง เนื่องจากการที่คอมพิวเตอร์จะสามารถเข้าใจในภาษาธรรมชาติได้นั้น คอมพิวเตอร์ต้องสามารถตีความสิ่งที่มนุษย์จะสื่อสารกับคอมพิวเตอร์ให้ได้เสียก่อน แล้วจึงจะสามารถปฏิบัติหรือทำตามคำสั่งนั้นได้ ซึ่งรูปประโยคที่คอมพิวเตอร์จะสามารถตีความได้นั้น จะต้องผ่านการตัดคำออกเป็นคำย่อยๆ อย่างถูกต้องเสียก่อน โดยเฉพาะอย่างยิ่งในรูปประโยคภาษาไทย การตัดคำเป็นสิ่งสำคัญอย่างมาก เนื่องจากรูปประโยคในภาษาไทยนั้นจะไม่มีการเว้นวรรคระหว่างคำเช่นรูปประโยคในภาษาอังกฤษ การตัดคำในภาษาไทยจึงมีความยุ่งยากกว่ามากที่จะตัดคำให้ออกมาตรงกับ ความหมายที่มนุษย์ต้องการสื่อสารจริงๆ เช่นประโยค “ฉันชอบบอกว่าเธอตากลม” เมื่อผ่านการตัดคำแล้วควรได้ผลลัพธ์ดังนี้ “ฉัน-ขอ-บอก-ว่า-เธอ-ตา-กลม” ซึ่งจะเห็นได้ว่าการตัดคำมีโอกาสที่ผิดพลาดได้สูงมาก เช่นอาจจะตัดออกมาเป็นแบบนี้ก็เป็นได้ “ฉัน-ขอ-บอก-ว่า-เธอ-ตา-กลม” โดยทั่วไปแล้ววิธีการตัดคำที่เป็นที่นิยมจะมีอยู่ด้วยกัน 2 วิธี คือการตัดคำโดยอาศัยกฎทางภาษา และการตัดคำโดยอาศัยพจนานุกรม แต่ถึงกระนั้นการตัดคำก็ยังคงทำได้ไม่สมบูรณ์นัก จึงต้องมีการนำทฤษฎี

อื่นๆ มาประยุกต์ใช้เพื่อให้ระบบการตัดคำมีประสิทธิภาพที่ดียิ่งขึ้น รวมถึงวิธีการของระบบ โครงข่ายประสาทเทียมเชิงลำดับ ก็เป็นอีกวิธีการหนึ่งที่ถูกนำมาใช้ในงานนี้ เพื่อช่วยในการพิจารณาการตัดคำที่มีความคลุมเครือ โดยอาศัยลำดับของหน้าที่ของคำในประโยคเป็นตัวพิจารณา

คำและหน้าที่ของคำในภาษาไทยสามารถจำแนกออกได้เป็น 7 ชนิดด้วยกัน โดยแต่ละชนิดยังสามารถจำแนกย่อยได้อีกดังนี้

1. คำนาม คือ คำที่หมายถึงบุคคล สัตว์ วัตถุ สิ่งของ สภาพ อาการ ลักษณะ ทั้งที่เป็นรูปธรรม และนามธรรม
 - สามานยนาม คือ คำที่เป็นชื่อของคน สัตว์ สิ่งของ และกิริยาอาการทั่วไป เช่น เด็ก เก้าอี้ เงิน โจร ลิง ท่าทาง นิสัย ปัญญา เป็นต้น
 - วิสามานยนาม คือ คำนามที่เป็นชื่อเฉพาะ เช่น สุชาติ เมื่อน้ำเจ้าพระยา สมหญิง เป็นต้น
 - ลักษณะนาม คือ คำนามที่ทำหน้าที่ประกอบนามอื่น เพื่อแสดงรูปลักษณะ ขนาด เช่น ตำ เล่ม คัน ผูก แปลง เป็นต้น
 - สมุหนาม คือ คำนามที่ทำหน้าที่แสดงหมวดหมู่ของสามานยนามและวิสามานยนาม หรือเป็นชื่อของสถานที่และองค์กรต่างๆ เช่น โขลง ศาล วัด โรงเรียน เป็นต้น
 - อาการนาม คือ คำนามซึ่งเกิดจากคำกริยา หรือคำวิเศษณ์ที่มีคำว่า “การ” หรือ “ความ” นำหน้า เช่น การยืน การเดิน ความรู้สึก ความรู้ เป็นต้น
2. คำสรรพนาม คือ คำที่ใช้แทนคำนามที่ผู้พูดหรือผู้เขียนได้กล่าวแล้ว หรือเป็นที่เข้าใจระหว่างกัน
 - บุรุษสรรพนาม คือ สรรพนามที่ใช้แทนชื่อผู้พูด ผู้ที่พูดด้วย หรือผู้ที่พูดถึง เช่น ฉัน ผม คุณ เขา มัน เป็นต้น
 - ประพันธสรรพนาม คือ สรรพนามที่ใช้แทนนามหรือแทนสรรพนามที่อยู่ติดต่อกันข้างหน้าทำหน้าที่เชื่อมประโยค เช่น ผู้ที่ ซึ่ง อัน เป็นต้น
 - วิภาคสรรพนาม คือ สรรพนามที่ใช้แทนนาม หรือสรรพนามที่กล่าวโดยซ้ำอีกครั้งหนึ่ง อาจแสดงความหมายรวม หรือแสดงความหมายแยก เช่น ต่าง บ้าง กัน เป็นต้น
 - นิยมสรรพนาม คือ สรรพนามที่ใช้ชี้ระยะที่กำหนดให้รู้ว่าใกล้หรือไกล เช่น นี่ นั่น โนน นั้น โนน เป็นต้น
 - ปฤจฉาสรรพนาม คือ คำสรรพนามที่มีความหมายเป็นคำถาม เช่น ใคร อะไร ผู้ใด สิ่งใด ไฉน ไหน เป็นต้น
 - อนิยมสรรพนาม คือ คำสรรพนามบอกความไม่เจาะจง ใช้แทนนามซึ่งไม่กำหนดแน่นอนลงไปว่าเป็นสิ่งนั้น เช่น ใคร อะไร ผู้ใด สิ่งใด เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- สรรพนามใช้เน้นนามตามความรู้สึของผู้พูด คือ แก มั่น ท่าน เธอ เป็นต้น

3. คำกริยา คือคำที่มีความหมายว่ากระทำ หรือมีอาการ หรืออยู่ในสภาพใดสภาพหนึ่ง

- สหกรรมกริยา คือ กริยาที่ต้องมีกรรมมารองรับจึงจะให้ความสมบูรณ์ เช่น จับ ขุด กัด เปิด หยิบ เป็นต้น
- อกรรมกริยา คือ กริยาที่มีความสมบูรณ์ เช่น ว่ายน้ำ ต่อย ฟัน งอก เป็นต้น
- วิกตรกริยา คือ กริยาที่ต้องมีคำนาม สรรพนาม หรือคำวิเศษณ์มาขยายจึงจะให้ความ เช่น เป็น เหมือน คล้าย เท่า เป็นต้น
- กริยานุเคราะห์ คือ กริยาที่ทำหน้าที่ช่วยกริยาชนิดอื่น เช่น จะ ถูก กำลัง แล้ว เป็นต้น
- กริยาสภาวะมาลา คือ กริยาที่ทำหน้าที่คล้ายนาม อาจเป็นประธาน เป็นกรรม หรือเป็นบทขยายส่วนใดส่วนหนึ่งของประโยค เช่น เป็น แต่ง ตัว เก่ง ซื่อ เลื่อ กิฬา

4. คำวิเศษณ์ คือ คำที่ทำหน้าที่ประกอบคำนาม คำสรรพนาม คำกริยา และคำวิเศษณ์ด้วยกันให้ ได้รับความชัดเจนยิ่งขึ้น

- ลักษณะวิเศษณ์ คือ คำวิเศษณ์ที่ประกอบเพื่อบอกลักษณะต่างๆ เช่น ดี ชั่ว ดำ ขาว แบน ใหญ่ เล็ก เป็นต้น
- กาลวิเศษณ์ คือ คำวิเศษณ์ที่ประกอบเพื่อบอกเวลา เช่น เช้า สาย บ่าย เย็น เป็นต้น
- สถานที่วิเศษณ์ คือ คำวิเศษณ์ที่ประกอบเพื่อบอกสถานที่ เช่น บน ล่าง เหนือ เป็นต้น
- ปริมาณวิเศษณ์ คือ คำวิเศษณ์ที่ประกอบเพื่อบอกจำนวน เช่น มาก น้อย บ้าง บาง หนึ่ง สอง สาม เป็นต้น
- นิยมวิเศษณ์ คือ คำวิเศษณ์ที่ประกอบเพื่อบอกความชี้เฉพาะ เช่น นี้ นั้น โนน นี้ เป็นต้น
- อนิยมวิเศษณ์ คือ คำวิเศษณ์ที่ประกอบเพื่อบอกความไม่ชี้เฉพาะ เช่น ไค อะไร ใคร ไหน เป็นต้น
- ปฤจฉาววิเศษณ์ คือ คำวิเศษณ์ที่ประกอบเพื่อบอกเนื้อความเป็นคำถาม เช่น ทำไม อย่างไร เท่าไร เป็นต้น
- ประดิษฐวิเศษณ์ คือ คำวิเศษณ์ที่ประกอบเพื่อบอกเสียงร้องหรือเสียงขานรับ เช่น จำ ขอรับ ไวย เป็นต้น
- ประดิษริวิเศษณ์ คือ คำวิเศษณ์ที่ประกอบเพื่อบอกความปฏิเสธหรือไม่ยอมรับ เช่น ไม่ มิใช่ มิได้ เป็นต้น

5. คำบุพบท คำที่แสดงความสัมพันธ์ระหว่างคำ หรือกลุ่มคำที่อยู่หลังคำบุพบทนั้น เพื่อบอก สถานที่ เวลา แสดงอาการ หรือแสดงความเป็นเจ้าของ

- บอกสถานที่ เช่น ใน ได้ บน เป็นต้น

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

- บอกเวลา เช่น จน โดย เป็นต้น
 - แสดงอาการ เช่น สำหรับ ด้วย ตาม กับ ต่อ เป็นต้น
 - แสดงความเป็นเจ้าของ เช่น สำหรับ ของ เป็นต้น
6. คำสันธาน คือ คำที่เชื่อมประโยคกับประโยค หรือคำกับกลุ่มคำ
- เชื่อมคำกับคำ เช่น และ หรือ เป็นต้น
 - เชื่อมประโยคกับประโยค เช่น เมื่อ แต่ กว่า ก็ เป็นต้น

7. คำอุทาน คือ คำที่เปล่งออกมาเพื่อแสดงอารมณ์ หรือความรู้สึก เช่น โอ๊ย ดายละวา เป็นต้น

ในประโยคภาษาไทย คำแต่ละคำจะเรียงกันอย่างต่อเนื่องเป็นลำดับ เมื่อแทนคำเหล่านั้นด้วยหน้าทีของคำที่มันทำหน้าที่นั้นอยู่ เราจะได้แบบแผนที่เป็นเชิงลำดับออกมา เช่น ประโยค “ฉันชอบบอกว่าเธอตากลม” เมื่อแทนหน้าที่ของคำแต่ละคำด้วยตัวอักษรจะได้แบบแผนข้อมูลเชิงลำดับออกมามี “SVVASNA” (ในที่นี้สมมติให้ “S” แทนคำสรรพนาม “V” แทนคำกริยา “A” แทนคำบุพบท “N” แทนคำนาม) หรือ “เด็กนั่งตากลม” จะได้แบบแผนข้อมูลเชิงลำดับเป็น “NVVN” เป็นต้น เมื่อนำแบบแผนข้อมูลเชิงลำดับเหล่านี้ไปประยุกต์ใช้กับโครงข่ายประสาทเทียมสำหรับการเรียนรู้ข้อมูลเชิงลำดับ โดยให้โครงข่ายเรียนรู้แบบแผนเหล่านี้ไว้ เมื่อข้อมูลลำดับแรกของประโยค “ฉันชอบบอกว่าเธอตากลม” ถูกป้อนให้แก่โครงข่ายคือ “S” เอาต์พุตที่ได้ก็คือ “V” ดังนั้นการตัดสินใจว่าคำต่อไปควรตัดออกมาเป็นรูปแบบไหน จะใช้ผลลัพธ์ที่ได้จากโครงข่ายเป็นตัวพิจารณา ถ้าคำต่อไปสามารถตัดได้มากกว่า 1 รูปแบบ เช่นอาจจะตัดออกมาได้ทั้งคำที่เป็นคำกริยาและคำนาม คำที่จะถูกตัดออกมาจริงๆก็คือ คำกริยาตามผลลัพธ์ที่ได้จากโครงข่าย ซึ่งข้อดีของวิธีการนี้ก็ยังสามารถช่วยการตัดสินใจในกรณีที่เกิดความคลุมเครือขึ้น และถึงแม้ทั้งสองประโยคจะเป็นคำที่ต่อเนื่องกันในลักษณะเดียวกันทั้งสองประโยคการตัดคำระหว่างสองประโยคอาจจะไม่เหมือนกันก็เป็นได้ เช่นประโยค “ฉันชอบบอกว่าเธอตากลม” และ “เด็กนั่งตากลม” การตัดคำว่า “ตากลม” ที่ได้จากประโยคแรกจะเป็น “ตา-กลม” แต่ในประโยคหลังจะเป็น “ตา-ล-ม” เนื่องจากแบบแผนหน้าที่ของคำที่โครงข่ายได้เรียนรู้มาไม่เหมือนกัน ในประโยคแรกจะเป็น “SVVASNA” เมื่อหน้าที่คำของประโยคแรกถูกป้อนเข้าไปในโครงข่ายเรื่อยๆ จนถึงคำว่าเธอ ซึ่งถูกแทนด้วย “S” เอาต์พุตที่ได้จากโครงข่ายในขณะนี้ก็คือ “N” ดังนั้นการตัดคำว่า “ตากลม” จะต้องตัดออกมาในรูปของคำนาม ไม่ใช่ตัดออกมาในรูปของคำกริยา ผลลัพธ์ที่ได้จึงเป็นคำว่า “ตา” แต่ในกรณีของประโยคหลัง ลำดับของหน้าที่คำจะเป็น “NVVN” ดังนั้นเมื่อหน้าที่ของคำถูกป้อนเข้าไปในโครงข่ายเรื่อยๆ จนถึงคำว่า “นั่ง” ซึ่งถูกแทนด้วย “V” ดังนั้นผลลัพธ์ที่ได้จากโครงข่ายคือ “V” การตัดคำถัดไปจึงต้องตัดออกมาในรูปของคำกริยา จึงได้คำว่า “ตา” เป็นผลลัพธ์

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้าไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.2 การออกแบบและการพัฒนาโครงการ

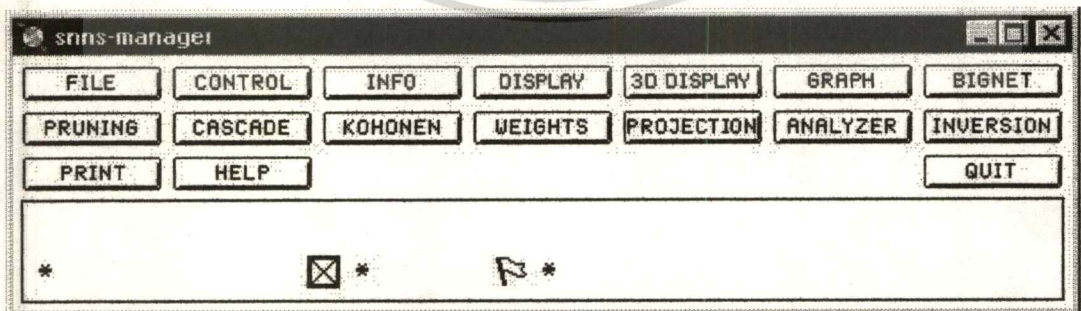
3.2.1 การกำหนดตัวแปรที่ใช้แทนหน้าที่ของคำในประโยค

ในการพัฒนาโครงการนี้จะกำหนดตัวแปรสำหรับแทนหน้าที่ของคำในประโยคไว้ด้วยกัน 8 ตัวแปร โดยจะให้แต่ละตัวแปรแทนหน้าที่คำต่างๆดังนี้

- n. ใช้แทนคำนาม (Noun)
- vi. ใช้แทนคำกริยา (Intransitive verb)
- vt. ใช้แทนคำสรรพนาม (Transitive verb)
- adj. ใช้แทนคำคุณศัพท์ (Adjective)
- adv. ใช้แทนคำกริยาวิเศษณ์ (Adverb)
- pre. ใช้แทนคำบุพบท (Preposition)
- pro. ใช้แทนคำสรรพนาม (Pronoun)
- con. ใช้แทนคำสันธาน (Conjunction)

3.2.2 การสร้างโครงข่ายประสาทเทียมเชิงลำดับด้วยเครื่องมือ SNNS

การสร้างโครงข่ายประสาทเทียมด้วยเครื่องมือ SNNS สามารถทำได้สองวิธีการด้วยกัน คือ การสร้างในโหมดกราฟฟิกและในโหมดข้อความ ในที่นี้ได้เลือกวิธีการสร้างในโหมดกราฟฟิก ซึ่งก่อนที่จะเรียกเครื่องมือ SNNS มาใช้งานได้นั้น จะต้องทำการเรียกโปรแกรม X-Win32 ของบริษัท StarNet Communication Corp. Version 3.4.1 ขึ้นมาเสียก่อนเพื่อจำลองให้เครื่องคอมพิวเตอร์เป็นเสมือน X-terminal แล้วจึงจะเรียกเครื่องมือ SNNS มาใช้งานได้จะปรากฏหน้าจอดังรูปที่ 3.1



รูปที่ 3.1 SNNS-Manager

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

ในส่วนนี้จะเรียกว่า SNNS-Manager ทำหน้าที่ควบคุมในการใช้งานโปรแกรม SNNS ทั้งหมดไม่ว่าจะเป็นการสร้าง การเทรน การทดสอบ โครงข่าย หรือการประเมินผลเองก็ตาม ซึ่ง SNNS เองรองรับโครงข่ายมาตรฐาน โครงข่ายประสาทเทียมในหลายรูปแบบด้วยกัน Jordan's Networks ก็เป็นหนึ่งในมาตรฐานของโครงข่ายประสาทเทียมที่ SNNS รองรับ ในการพัฒนาโครงการนี้ได้เลือกใช้โครงข่ายแบบนี้เอง โดยทำการกำหนดยูนิตที่ระดับชั้นต่างๆดังนี้

- ในชั้น Input Layer กำหนดให้มียูนิตทั้งหมด 9 ยูนิตด้วยกัน โดยให้แต่ละยูนิตทำหน้าที่รับรูปแบบข้อมูลที่จะเข้ามาทางอินพุต ซึ่งข้อมูลที่รับเข้ามาทางอินพุตจะเป็นในลักษณะการนำหน้าที่ของคำมาเข้ารหัสจะมีอยู่ด้วยกัน 8 รูปแบบ ส่วนอีกหนึ่งรูปแบบจะเป็นรหัสที่ใช้แทนสิ่งที่บ่งบอกถึงการสิ้นสุดลำดับของข้อมูล โดยมีรูปแบบดังนี้

vi. แทนด้วยรหัส 000000010

vt. แทนด้วยรหัส 000000100

adv. แทนด้วยรหัส 000001000

adj. แทนด้วยรหัส 000010000

n. แทนด้วยรหัส 000100000

pre. แทนด้วยรหัส 001000000

con. แทนด้วยรหัส 010000000

pro. แทนด้วยรหัส 100000000

end. แทนด้วยรหัส 000000001 (รหัสการสิ้นสุดลำดับของข้อมูล)

- ในชั้น Hidden Layer และชั้น Context Layer กำหนดให้มียูนิตทั้งหมด 9 ยูนิต
- ในชั้น Output Layer กำหนดให้มียูนิตทั้งหมด 9 ยูนิตด้วยกันเช่นเดียวกับในชั้นของ Input Layer โดยให้แต่ละยูนิตแทนหน้าที่ของคำทั้ง 8 และแทนการสิ้นสุดลำดับของข้อมูลอีก 1 ยูนิต

ดังนั้นจากรายการ Bignet ใน SNNS-Manager จึงเลือกโครงข่ายที่ต้องการเป็น Jordan แล้วจึงกำหนดค่าจำนวนยูนิตต่างๆลงไปตามต้องการดังแสดงในรูปที่ 3.2

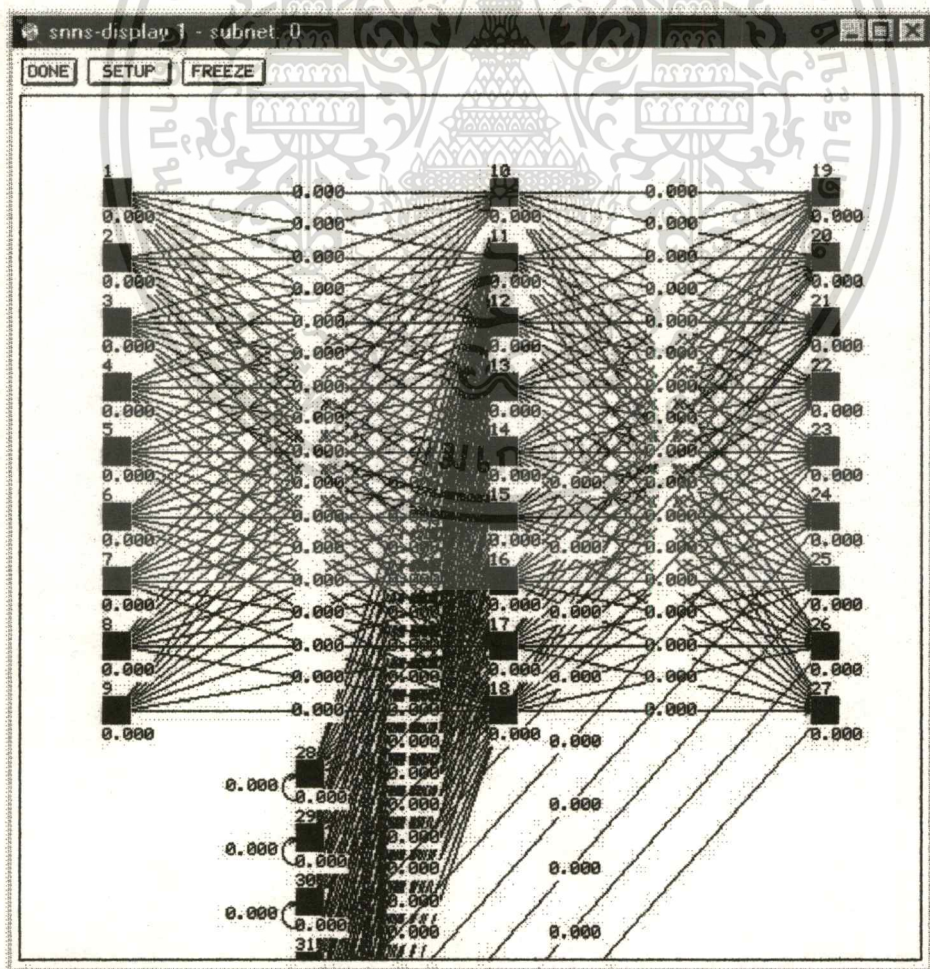
BigNet (Jordan)

	No. of Units	No. of Col.
Input Layer:	9	1
Hidden Layer:	9	1
Output Layer:	9	1

CREATE NET DONE

รูปที่ 3.2 การกำหนดจำนวนยูนิตของโครงข่ายประสาทเทียม

เมื่อทำการกำหนดค่าจำนวนยูนิตแล้ว จึงเลือกรายการสร้างโครงข่ายขึ้นมา เป็นผลให้ได้โครงข่ายแบบ Jordan's Networks ดังรูปที่ 3.3



รูปที่ 3.3 โครงข่ายที่สร้างขึ้นจากเครื่องมือ SNNS

เอกสารนี้เป็นเอกสารที่สงวนลิขสิทธิ์สำหรับการใช้งานเพื่อการศึกษาเท่านั้น เมื่อนำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

หลังจากที่มีการสร้างโครงข่ายขึ้นมาแล้ว จะต้องทำการเทรนโครงข่ายเพื่อให้เรียนรู้รูป

แบบลำดับหน้าที่ของคำในประโยคภาษาไทยดังตัวอย่างเช่น

- \bar{n} -vi. (เสื่อ-กັด)
- n.-vt.-n. (เสื่อ-กັด-เค็ก)
- n.-vt.-n.-n. (ครู-เป็น-ข้าราชการ-กระทรวงศึกษาธิการ)
- n.-vi.-n. (ลิง-ชอบ-ผลไม้)
- n.-vi.-vt.-n. (ลิง-ชอบ-กิน-ผล ไม้)
- n.-vt.-n.-n.-adj. (สมร-ส่ง-ขนม-ร้านค้า-มาก)
- n.-vi.-vt.-pre.-n. (เด็ก-นั่ง-อยู่-บน-เก้าอี้)
- n.-adj.-vi. (นก-น้อย-บิน)
- n.-adj.-vt.-n. (เสื่อ-คู-กັด-เค็ก)
- n.-adj.-vt.-n.-n. (เสื่อ-คู-กັด-เจ้าหน้าที่-สวนสัตว์)
- n.-n.-adj.-vi. (ขบวน-ทัศนาร-ใหญ่-เดินทาง)
- n.-n.-adj.-vt.-n. (ขบวน-ทัศนาร-ใหญ่-ไป-รถไฟ)
- pro.-vi. (เขา-เดิน)
- pro.-vt.-n. (เขา-จับ-แมว)
- pro.-vt.-n.-n. (ฉัน-เป็น-เจ้าหน้าที่-ห้องสมุด)
- pro.-vi.-n. (คุณ-ชอบ-กระต่าย)
- pro.-vi.-vt.-n. (เธอ-ชอบ-อ่าน-หนังสือ)
- pro.-vt.-n.-n.-adj. (ฉัน-นับ-เหรียญ-เงิน-เก่า)
- pro.-vi.-vt.-pre.-n. (ท่าน-อาศัย-อยู่-ใน-ถ้ำ)
- pro.-adj.-vi. (เจ้า-ฮ้วน-วิ่ง)
- pro.-adj.-vt.-n. (เจ้า-ฮ้วน-กิน-ขนม)
- pro.-adj.-vt.-n.-n. (เจ้า-ฮ้วน-กิน-ขนม-ไทย)
- vt.-n. (ผู้-ศึก)
- vt.-n.-adj. (ผู้-ศึก-ใหญ่)
- vi.-n. (จับ-ตัว)
- vi.-adv. (นั่ง-นาน)

เอกสารนี้เป็น \bar{n} -adj.-vi.-n.-n. (กว้าง-เป็น-คำ-วิเศษณ์) เอกสารศึกษานี้เป็นการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

และจะต้องมีการจัดเตรียมรูปแบบข้อมูลที่ต้องการเทรนเสียก่อน ซึ่งโปรแกรม SNNS ต้องการเพิ่มข้อมูลแบบข้อความตามรูปแบบที่โปรแกรมกำหนด ดังตัวอย่างต่อไปนี้

SNNS pattern definition file V3.2

generated at Fri Apr 22 16:56:45 1994

No. of patterns : 150

No. of input units : 9

No. of output units : 9

Input pattern 1:

000100000

Output pattern 1:

000000010

Input pattern 2:

000000010

Output pattern 2:

000000001

Input pattern 3:

000000001

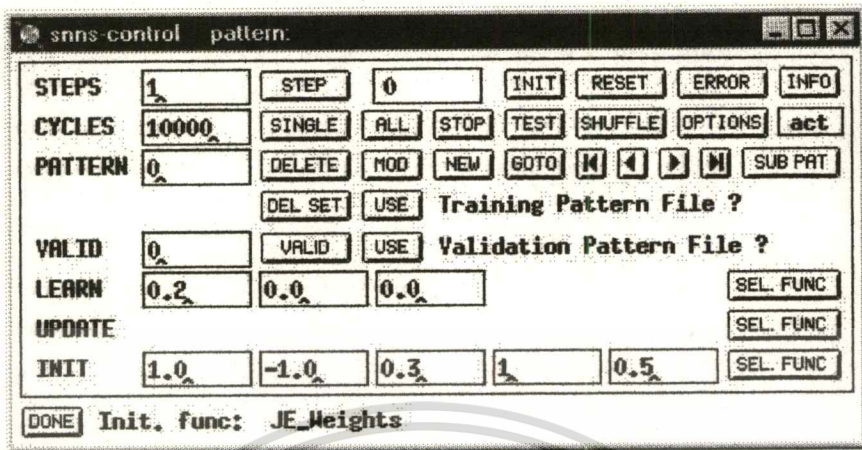
Output pattern 3:

000100000

จากตัวอย่างเพิ่มข้อมูลในการเทรนโครงข่าย จะประกอบไปด้วยส่วนหัวของเพิ่มข้อมูล และจะมีการกำหนดจำนวนรูปแบบที่ต้องการเทรนในที่นี้กำหนดไว้ 150 รูปแบบ หลังจากนั้นจะเป็นการกำหนดจำนวนอินพุตยูนิตและเอาต์พุตยูนิต ซึ่งต้องเป็นไปตามที่เราได้ออกแบบโครงข่ายไว้ แล้วจึงตามด้วยรูปแบบข้อมูลที่ต้องการเมื่อมีอินพุตแบบนี้เข้ามา เอาต์พุตที่ได้จะเป็นแบบใด ซึ่งจะควบคุมกลับไปจนครบตามจำนวนรูปแบบที่เราได้กำหนดไว้

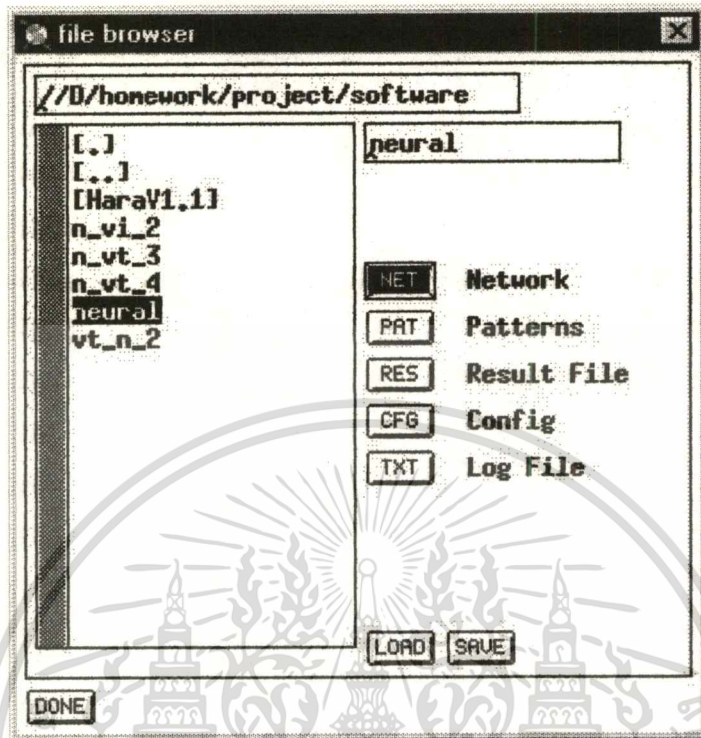
เมื่อจัดเตรียมเพิ่มข้อมูลที่ต้องการเทรนเรียบร้อยแล้ว ก็ทำการโหลดเพิ่มข้อมูลในการเทรนเข้าไปในโปรแกรม โดยเลือกรายการ File จาก SNNS-Manager จากนั้นจึงทำการเทรนโครงข่ายได้โดยการเลือกรายการ Control จาก SNNS-Manager จะปรากฏหน้าจอดังรูปที่ 3.4

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.4 กรอบที่ใช้ในการควบคุมการเทรนหรือทดสอบโครงข่าย

ในส่วนนี้จะเป็นการควบคุมในการเทรนโครงข่ายหรือทดสอบโครงข่าย สิ่งที่ต้องกำหนดก่อนการเทรนก็คือการเลือก Learning function, Update function และ Initial function ในที่นี้เลือกเป็น JE_BP, JE_Spatial และ JE_Weight ตามลำดับ และจะต้องกำหนดค่าพารามิเตอร์สำหรับแต่ละ Function ด้วย โดยกำหนดค่า Learning Rate เป็น 0.2 ส่วน Initial Function กำหนดให้เป็นค่า 1.0,-1.0,0.3,1 และ 0.5 แล้วจึงกำหนดค่า Cycles ว่าต้องการมีการเทรนข้อมูลในแฟ้มข้อมูลกี่รอบ ซึ่งในที่นี้กำหนดไว้ 10000 รอบ เมื่อมีการกำหนดค่าต่างๆแล้ว รายการ Reset เพื่อเป็นการ Reset โครงข่าย ก่อนที่จะเลือกรายการ Init เพื่อเป็นการกำหนดค่าการถ่วงน้ำหนักเริ่มต้นให้แก่แต่ละ Link แล้วจึงเลือกรายการ All เพื่อเป็นการเริ่มเทรนโครงข่ายทุกรูปแบบตามที่ได้กำหนดไว้แล้ว เมื่อการเทรนเสร็จสิ้น เราสามารถทดสอบโครงข่ายว่าเป็นไปตามที่ต้องการหรือไม่โดยการเลือกรายการ Test จะเป็นการทดสอบแต่ละรูปแบบที่มีอยู่ในแฟ้มข้อมูลเทรน โดยเอาที่พหุคูณที่มีค่ามากที่สุดจะถูกแสดงเป็นสีที่ต่างจากเอาที่พหุคูณอื่นๆ และเมื่อได้โครงข่ายที่สามารถทำงานได้ตามลำดับข้อมูลที่ต้องการแล้วก็ทำการบันทึกโครงข่ายนี้ไว้เพื่อใช้กับโปรแกรมที่พัฒนาขึ้นมา โดยวิธีการเรียกใช้โครงข่ายนี้จะกระทำในโหมดข้อความหรือที่เรียกว่า Batch Mode ซึ่งการบันทึกโครงข่ายสามารถทำได้โดยการเลือกรายการ File จาก SNNS-Manager จากนั้นต้องระบุชื่อโครงข่ายแล้วจึงเลือกรายการ Save ดังหน้าจอในรูปที่ 3.5



รูปที่ 3.5 กรอบการบันทึกโครงข่ายที่ผ่านการเทรนแล้ว

3.2.3 การพัฒนาโปรแกรมการตัดคำภาษาไทยโดยอาศัยโครงข่ายประสาทเทียมเชิงลำดับ

การพัฒนาโปรแกรมสำหรับการตัดคำภาษาไทย โดยอาศัยโครงข่ายประสาทเทียมเชิงลำดับ ในครั้งนี้จะใช้โปรแกรมภาษา Visual Basic Version 6.0 เป็นเครื่องมือในการพัฒนา และใช้ฐานข้อมูลที่สร้างขึ้นจาก โปรแกรม Microsoft Access 97 จะประกอบไปด้วยตารางต่างๆดังนี้

Lexicon Table ใช้ในการเก็บฐานข้อมูลคำศัพท์และหน้าที่ของคำในภาษาไทย

- Word เป็น Attribute แบบ text ใช้เก็บคำศัพท์
- Attribute เป็น Attribute แบบ text ใช้เก็บหน้าที่ของคำ

Encode Table ใช้ในการเก็บข้อมูลการเข้ารหัสจากหน้าที่คำเป็นรูปแบบสำหรับโครงข่าย

- Attribute เป็น Attribute แบบ text ใช้เก็บหน้าที่ของคำ
- Encode เป็น Attribute แบบ text ใช้เก็บรูปแบบสำหรับโครงข่าย

Temp Table ใช้ในการเก็บโครงข่ายต้นไม้ (Tree) ที่สร้างขึ้นสำหรับการตัดคำ

- Number เป็น Attribute แบบ text ใช้เก็บหมายเลขของโหนด
- Word เป็น Attribute แบบ text ใช้เก็บคำที่อยู่ภายในโหนดนั้น
- Link เป็น Attribute แบบ text ใช้เก็บหมายเลขโหนดที่โหนดนี้ขึ้นอยู่กับ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

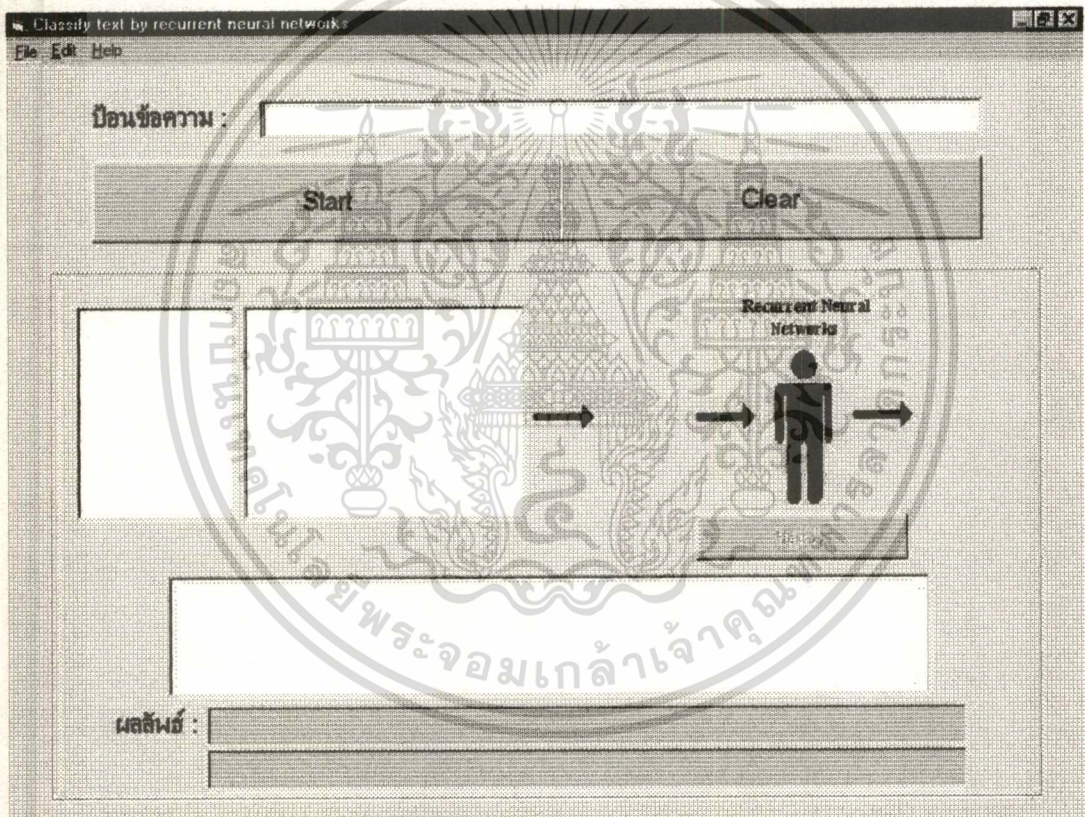
Plan Table ใช้ในการเก็บชื่อเพลงลำดับข้อมูลที่สามารถตัดคำได้

- Plan เป็น Attribute แบบ text ใช้เก็บชื่อเพลงลำดับข้อมูล
- Length เป็น Attribute แบบ text ใช้เก็บความยาวของเพลง

Plantemp Table ใช้ในการเก็บชื่อเพลงที่ประโยชน์สามารถตัดคำได้

- Plan เป็น Attribute แบบ text ใช้เก็บชื่อเพลง

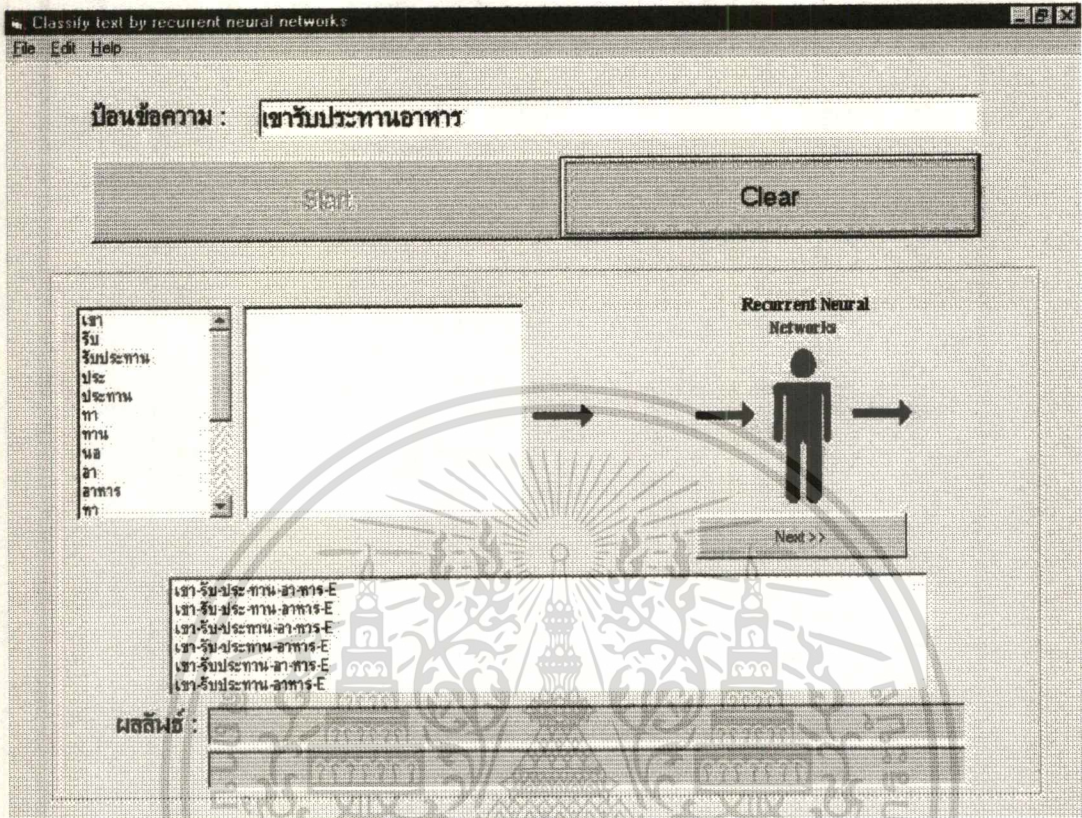
เมื่อเรียกใช้โปรแกรมสำหรับการตัดคำภาษาไทยโดยอาศัยโครงข่ายประสาทเทียมเชิงลำดับ จะปรากฏหน้าจอ ดังรูปที่ 3.6



รูปที่ 3.6 หน้าจอหลักของโปรแกรมการตัดคำภาษาไทย

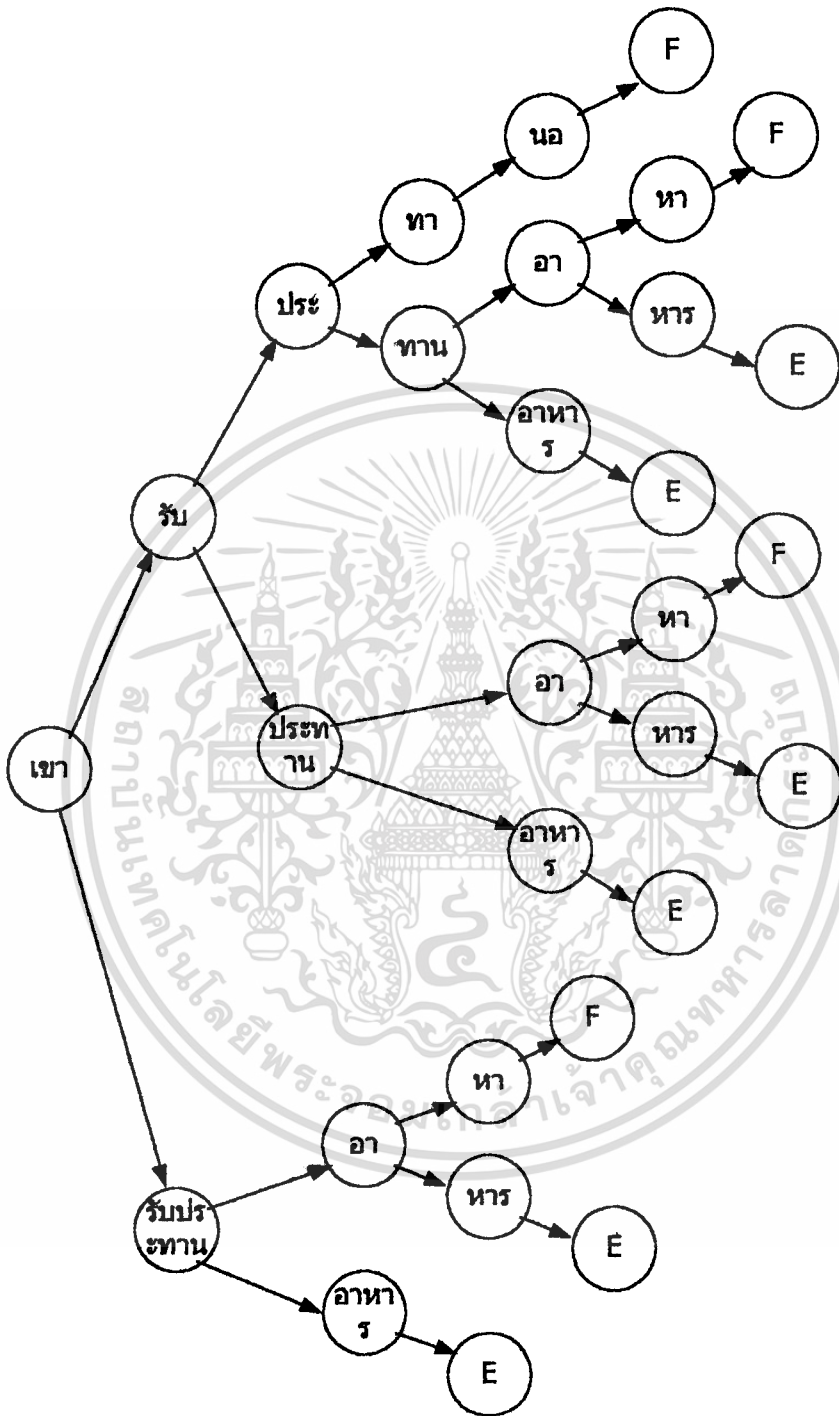
หน้าจอนี้จะเป็นส่วนหน้าจอหลักของโปรแกรมที่ใช้ดำเนินการตัดคำ เมื่อป้อนประโยคที่ต้องการให้มีการตัดคำผู้ใช้สามารถทำได้โดยการ ทิมป์ประโยคลงในช่อง “ป้อนข้อความ” หลังจากนั้นจึงทำการคลิกปุ่ม “Start” เพื่อเริ่มดำเนินการ ผลลัพธ์ที่จะ ได้จะเป็นดังหน้าจอในรูปที่ 3.7

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.7 การป้อนประโยคให้แก่โปรแกรมการตัดคำ

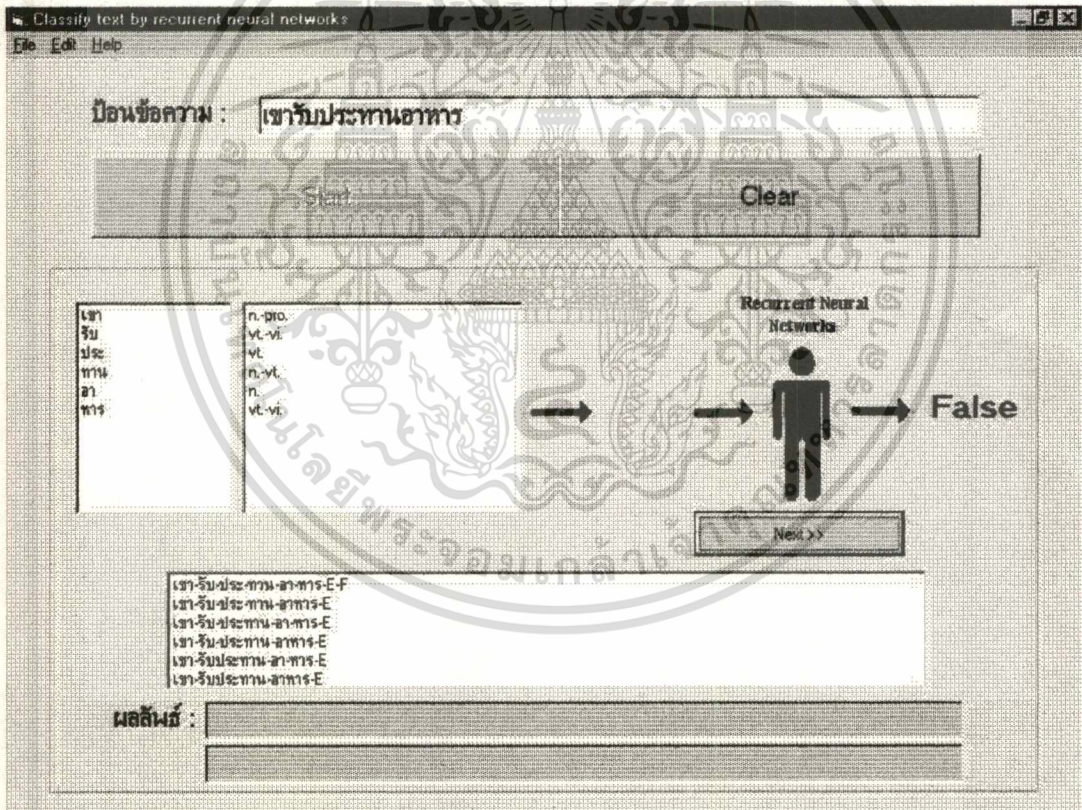
เมื่อผู้ใช้ทำการป้อนประโยคที่ต้องการตัดคำแล้วคลิกปุ่ม “Start” โปรแกรมจะทำการตัดคำในทุกรูปแบบเท่าที่เป็นไปได้ โดยใช้วิธีการเขียนโปรแกรมแบบเรียกตัวเอง (Recursive) เพื่อสร้างโครงข่ายต้นไม้ขึ้นมา แล้วจึงทำการท่องไปในโครงข่ายต้นไม้เพื่อหาประโยคทั้งหมดที่เป็นไปได้ออกมา ซึ่งการสร้างโหนดในโครงข่ายต้นไม้จะอาศัยวิธีการตัดคำโดยตรวจสอบกับฐานข้อมูลว่ามีคำดังกล่าวอยู่ในฐานข้อมูลหรือไม่ ถ้าพบก็สร้างโหนดขึ้นมา แล้วจึงตัดในคำต่อไป แต่ถ้าหากไม่พบคำดังกล่าวในฐานข้อมูลจะถือว่า การตัดคำในรูปแบบนี้เป็นไปไม่ได้ โปรแกรมจะสร้างโหนดที่มีค่าเป็น “F” ขึ้นมา เพื่อเป็นการบอกว่าการตัดในรูปแบบนี้เป็นไปไม่ได้ ส่วนการสร้างโหนดแบบใดทำได้จนถึงสิ้นสุดประโยค โปรแกรมก็จะทำการสร้างโหนด “E” เพื่อเป็นการบอกให้ทราบว่ากันตัดรูปแบบนี้สามารถเป็นไปได้อย่างไร จากประโยคตัวอย่าง “เขารับประทานอาหาร” สามารถสร้างโครงข่ายต้นไม้ได้ดังรูปที่ 3.8



รูปที่ 3.8 ตัวอย่างการสร้างโครงข่ายต้นไม้ของโปรแกรมการตัดคำ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

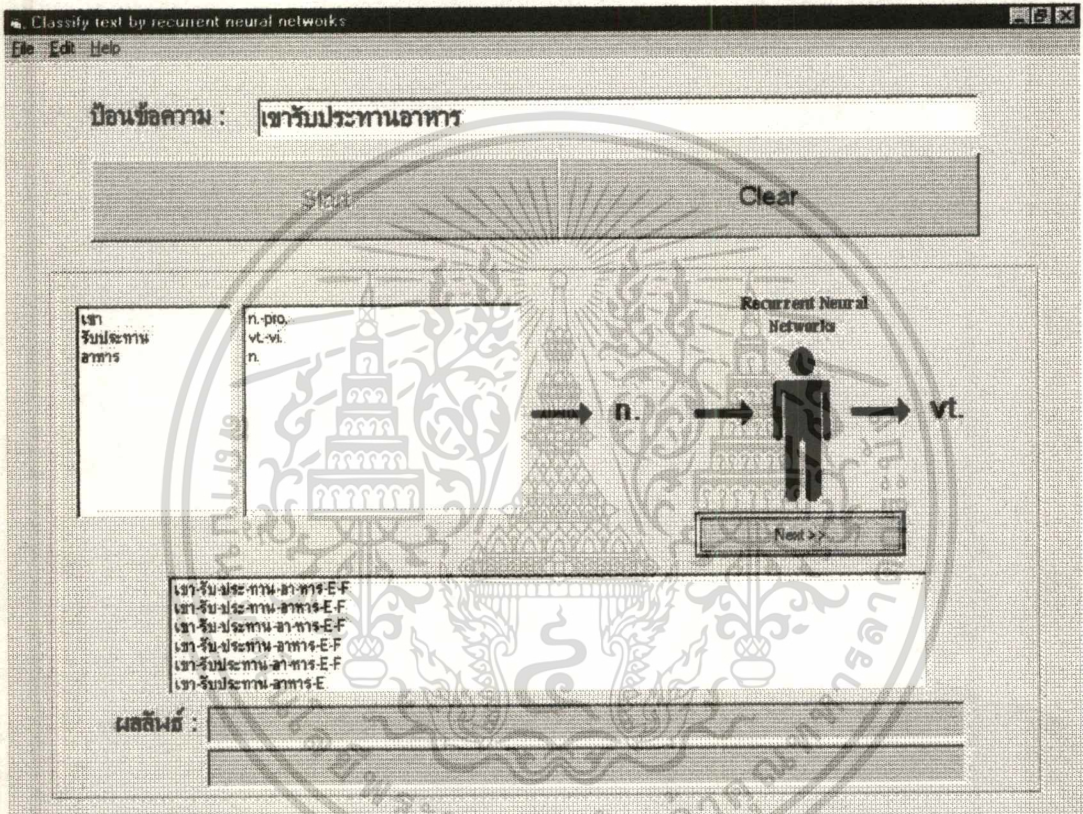
จากโครงข่ายต้นไม้ที่สร้างขึ้นมา เราสามารถทอ้งไปโครงข่ายต้นไม้แล้วได้ผลลัพธ์ออกมาว่ารูปแบบประโยคที่สามารถตัดได้มีอยู่ด้วยกัน 6 รูปแบบด้วยกัน ในขั้นตอนถัดมาเมื่อผู้ใช้คลิกปุ่ม “Next” โปรแกรมจะนำประโยคแรกสุดจากทั้งหมด ไปแยกออกเป็นคำย่อยๆ พร้อมทั้งแสดงหน้าที่ของคำนั้นๆทั้งหมดเท่าที่จะเป็นไปได้ในรูปแบบประโยคภาษาไทย พร้อมทั้งทำการตรวจสอบกับฐานข้อมูลแพลนว่าประโยคดังกล่าวสามารถตรวจเช็คได้กับแพลนไคบัง โดยพิจารณาจากจำนวนคำในประโยคและหน้าที่ของคำสองคำแรกในประโยคเป็นตัวพิจารณา ถ้าหากรูปแบบประโยคดังกล่าวไม่ตรงกับเงื่อนไขของแพลนใดๆเลย โปรแกรมจะถือว่าประโยคดังกล่าวไม่ถูกต้องและจะทำการแสดงผลว่า “False” พร้อมทั้งแสดงตัวอักษร “F” ต่อท้ายประโยคดังกล่าวในกรอบข้อมูลที่แสดงรูปแบบประโยคต่างๆ ดังแสดงในรูปที่ 3.9



รูปที่ 3.9 หน้าจอการตรวจเช็ครูปแบบประโยคกับแพลนในฐานข้อมูล

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อผู้ใช้ทำการคลิก “Next” อีกครั้งโปรแกรมจะทำการนำรูปแบบการตัดคำแบบถัดๆ ไป ตรวจสอบเช็คกับฐานข้อมูลพจนานุกรมไปเรื่อยๆ ถ้าหากประโยคดังกล่าวตรงตามเงื่อนไขของพจนานุกรมจะทำการนำหน้าที่ของคำคำแรกในรูปแบบนั้น เข้าไปประมวลผลยังโครงข่ายประสาทเทียมที่ได้จัดเตรียมไว้ เพื่อให้โครงข่ายบ่งบอกถึงหน้าที่ของคำถัดไปว่าควรจะเป็นคำที่ทำหน้าที่อะไร ดังแสดงในรูปที่ 3.10



รูปที่ 3.10 การทำนายของโครงข่ายประสาทเทียมเชิงลำดับ

ในขั้นตอนนี้ โปรแกรมจะทำการสร้างเพิ่มข้อมูลข้อความตามรูปแบบที่โปรแกรม SNNS ต้องการ พร้อมทั้งเรียกใช้งาน โปรแกรม SNNS ในโหมดข้อความ (Batch Mode) ดังนี้

```
c:\batchman -f neural.bat
```

ซึ่งข้อมูลที่อยู่ในแฟ้มข้อมูล “neural.bat” จะเป็นคำสั่งที่กระทำกับโครงข่าย เพื่อให้โครงข่ายทำนายหน้าที่คำถัดไปที่ควรจะเป็น โดยจะมีรูปแบบดังนี้

```
loadNet("d:\homework\project\software\neural.net")
```

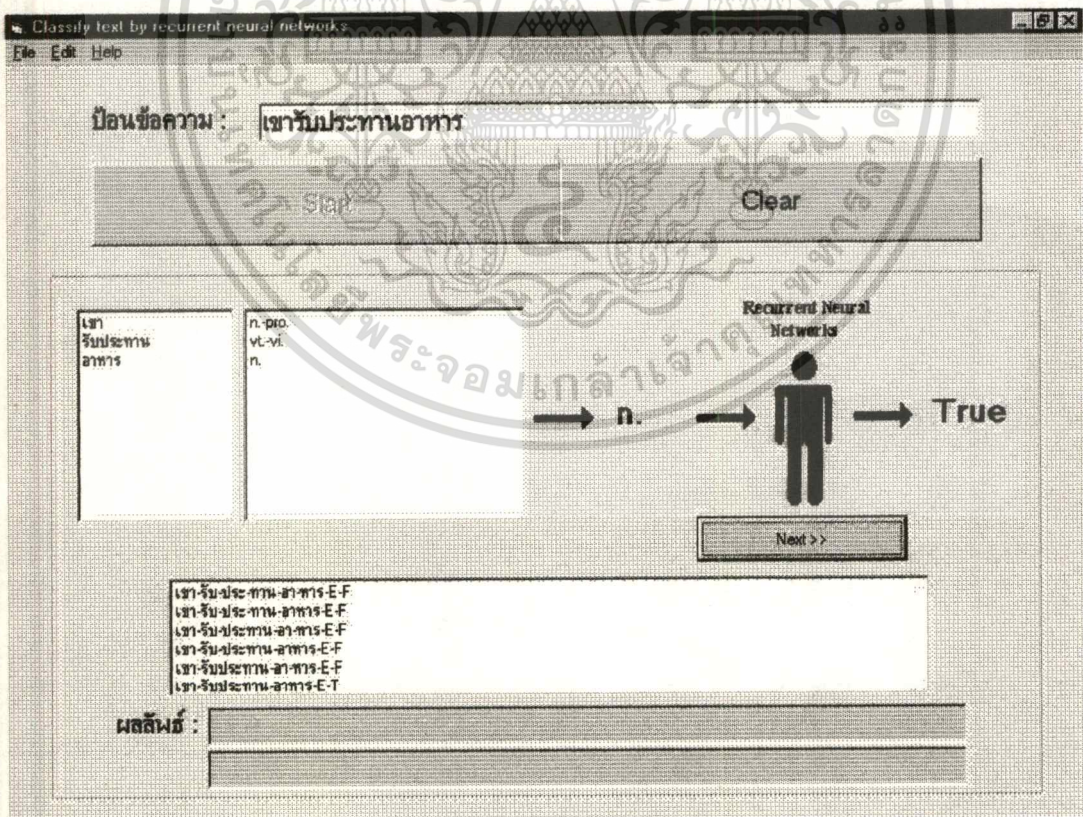
```
loadPattern("d:\homework\project\software\input.pat")
```

```
saveResult("d:\homework\project\software\output.res", 1, PAT, TRUE, FALSE, "Create")
```

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า

ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

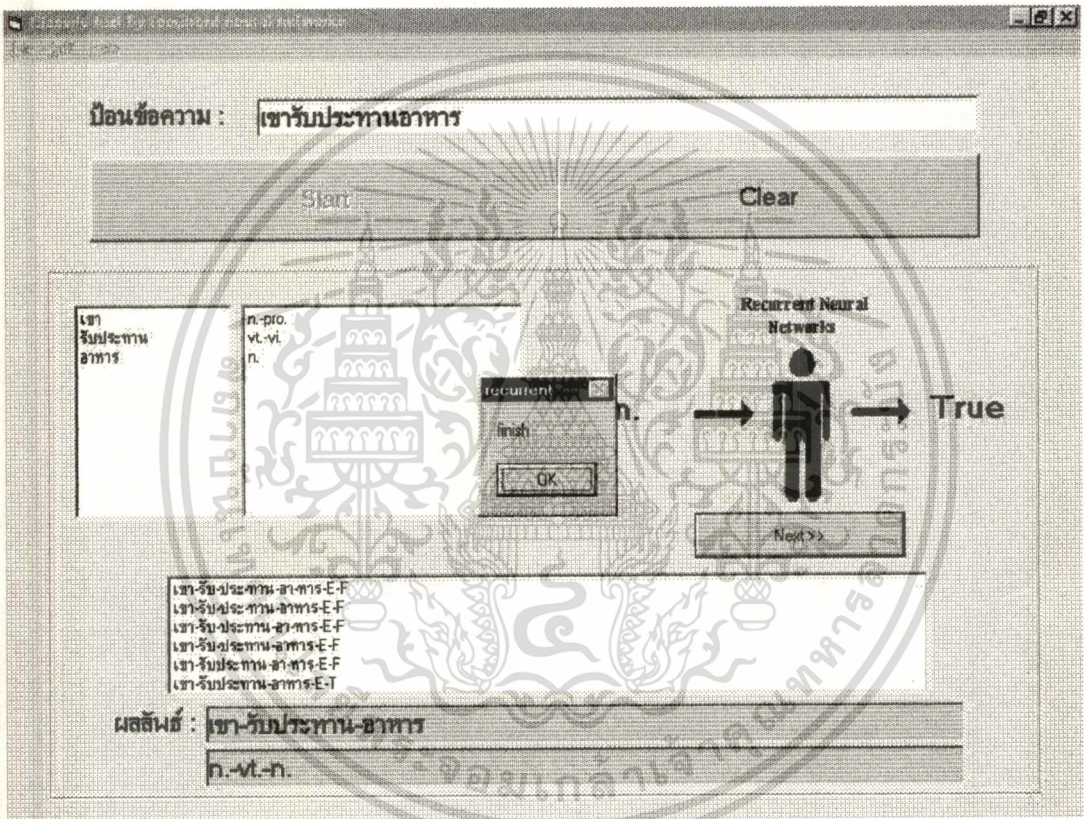
จะเป็นคำสั่งในการโหลดโครงข่ายที่ได้เทรนไว้แล้วขึ้นมาใช้งาน พร้อมทั้งโหลดเพิ่มข้อมูลที่ต้องการทดสอบกับโครงข่ายประสาทเทียม หลังจากนั้นจึงสั่งให้โครงข่ายทำการประมวลผลแล้วบันทึกลงเพิ่มข้อมูลซึ่งชื่อว่า “output.res” เมื่อได้ผลลัพธ์โปรแกรมจะทำการแสดงค่าที่ได้จากโครงข่ายบนหน้าจอ เมื่อผู้ใช้ทำการคลิกปุ่ม “Next” อีกครั้งหนึ่ง โปรแกรมจะทำการนำผลลัพธ์ที่ได้จากโครงข่ายไปเปรียบเทียบกับค่าถัดไปในลำดับค่าที่ได้แสดงไว้ ว่าค่าดังกล่าวสามารถทำหน้าที่ได้ตามที่โครงข่ายทำนายไว้หรือไม่ ถ้าค่าดังกล่าวสามารถทำหน้าที่ดังที่โครงข่ายทำนายไว้ โปรแกรมจะนำหน้าที่ค่านั้นมาเป็นอินพุตให้โครงข่ายอีกครั้งเพื่อทำนายหน้าที่ของค่าในลำดับถัดๆ ไป แต่ถ้าหากค่านั้นไม่สามารถทำหน้าที่ได้ดังที่โครงข่ายทำนาย โปรแกรมจะถือว่ารูปแบบดังกล่าวไม่ถูกต้อง แล้วแสดงผลลัพธ์ทางเอาต์พุตว่า “False” พร้อมทั้งแสดงตัวอักษร “F” ต่อท้ายรูปแบบนี้ในกรอบของการแสดงรูปแบบทั้งหมดที่เป็นไปได้ในการตัดคำ หลังจากนั้นจึงนำรูปแบบถัดไปเข้าไปประมวลผลอีกครั้ง แต่ถ้ารูปแบบดังกล่าวสามารถเป็นไปได้ตามรูปแบบที่โครงข่ายประสาทเทียมได้แนะนำไว้จนจบเพลน โครงข่ายจะแสดงผลลัพธ์ทางเอาต์พุตว่า “True” และจะแสดงตัวอักษร “T” ต่อท้ายรูปแบบนั้นๆ ดังแสดงในรูปที่ 3.11



รูปที่ 3.11 หน้าจอขณะประมวลผลรูปแบบคำที่ตัดได้เป็นผลสำเร็จ

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

เมื่อผู้ใช้คลิกปุ่ม “Next” อีกครั้งโปรแกรมจะแสดงผลที่ได้ลงในกรอบผลลัพธ์ พร้อมทั้งแสดงการเรียงตัวของหน้าที่ของคำในรูปแบบนั้นให้ผู้ใช้ได้ทราบอีกด้วย แต่ถ้าหากการทดสอบครบทั้งหมดแล้วไม่มีรูปแบบใดที่โครงข่ายบอกว่าถูกต้องเลย โปรแกรมจะเลือกรูปแบบที่มีจำนวนคำน้อยที่สุดเป็นผลลัพธ์โดยอัตโนมัติ แต่จะไม่มีผลลัพธ์ของหน้าที่คำในประโยคแสดง หรือถ้าหากมีรูปแบบที่ถูกต้องมากกว่าหนึ่งประโยค โปรแกรมก็จะเลือกเอาประโยคที่ถูกต้องที่มีจำนวนคำน้อยที่สุดทำการแสดงผลเช่นกัน ดังหน้าจอที่แสดงในรูปที่ 3.12

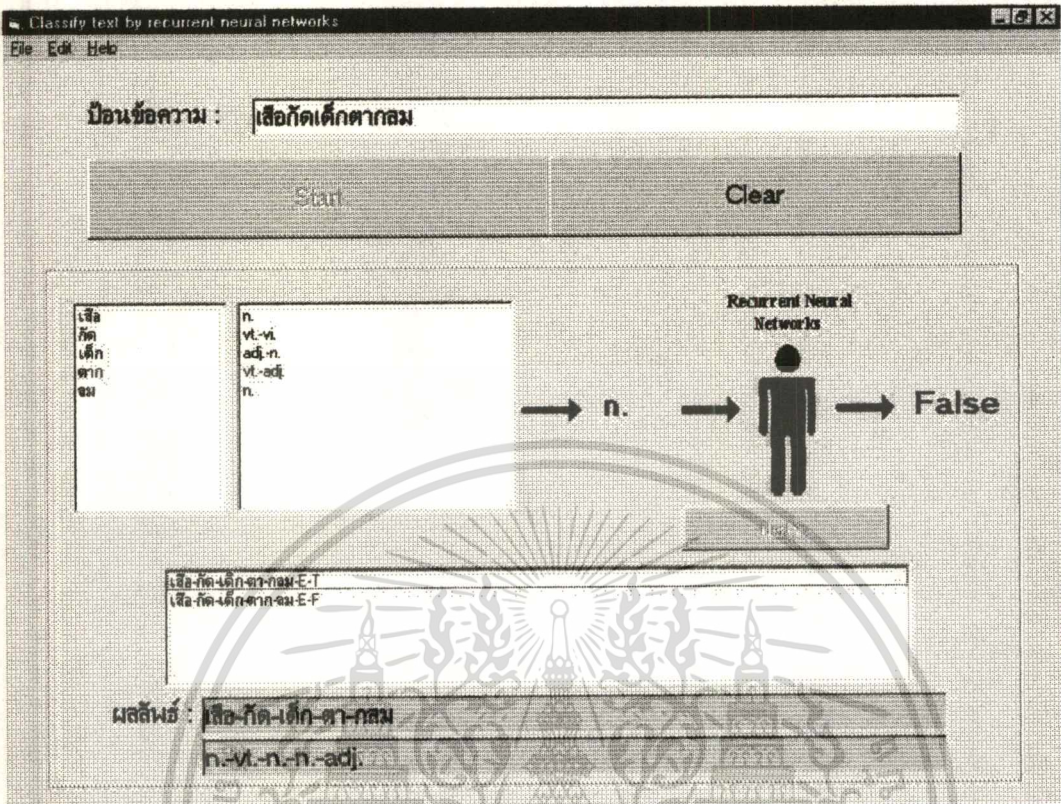


รูปที่ 3.12 หน้าจอของการแสดงผลของโปรแกรมการตัดคำ

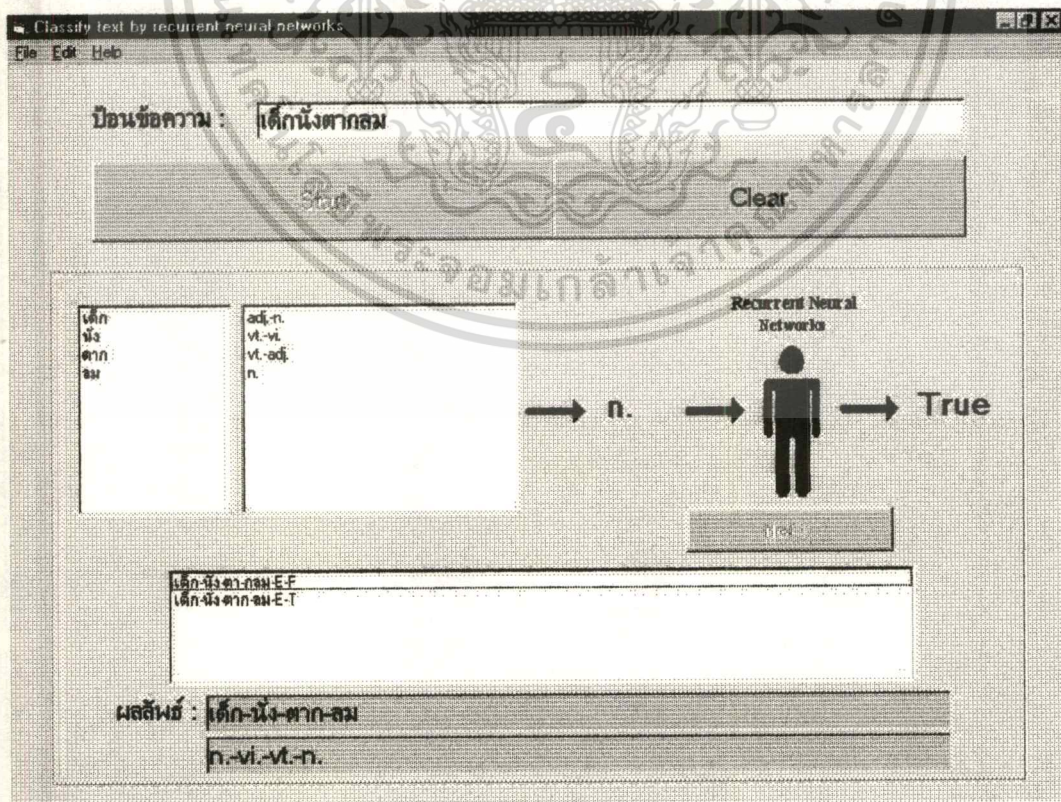
3.3 การทดลอง

ในการทดลองใช้โปรแกรมการตัดคำ จะเลือกใช้ประโยคที่มีความคลุมเครือเป็นเครื่องทดสอบ เพื่อแสดงให้เห็นถึงประสิทธิภาพของการใช้โครงข่ายประสาทเทียมเชิงลำดับ มาช่วยในการตัดสินใจ ในที่นี้จึงเลือกประโยค “เสือกักเด็กตากลม” และ “เด็กนั่งตากลม” ผลที่ได้จะเป็นดังรูปที่ 3.13 และ 3.14

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า ไม่ว่าจะกรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้



รูปที่ 3.13 หน้าจอแสดงผลการตัดคำประโยคทดลองที่ 1



รูปที่ 3.14 หน้าจอแสดงผลการตัดคำประโยคทดลองที่ 2

เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ดัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้

3.5 สรุปผลการทดลอง

จากการทดลองจะเห็นได้ว่า ถึงแม้ว่าทั้งสองประโยคมีคำว่า “ตากลม” เหมือนกัน ซึ่งเป็นคำที่มีลักษณะคลุมเครือว่าควรจะตัดออกมาในรูปของ “ตา-กลม” หรือ “ตาก-ลม” ก็ไม่จำเป็นว่าผลลัพธ์ที่ได้จากโปรแกรมในการตัดคำจากประโยคทั้งสองจะต้องเหมือนกัน ขึ้นอยู่กับรูปประโยคจะเป็นตัวตัดสินว่าควรจะตัดออกมาในรูปแบบใด เช่นในประโยคแรก “เสือกักเด็กตากลม” จะถูกตัดตามแผนທີ່โครงข่ายได้เรียนรู้ไว้คือแผน “n.-vt.-n.-n.-adj.” นั่นคือหลังคำนามตัวที่สอง โครงข่ายแนะนำว่าควรจะตัดคำถัดไปออกมาในรูปของคำนามดังนั้นผลลัพธ์ที่ได้จึงเป็นคำว่า “ตา” แทนที่จะเป็นคำว่า “ตาก” ซึ่งเป็นคำกริยา หรือในประโยคที่สอง “เด็กนั่งตากลม” ประโยคนี้จะถูกตัดตามแผน “n.-vi.-vt.-n.” จะเห็นได้ว่าหลังคำว่านั่งซึ่งเป็น “vi.” โครงข่ายจะแนะนำว่าควรตัดคำถัดไป ในรูปของกรรมกริยา ดังนั้นผลลัพธ์ที่ได้จึงเป็นคำว่า “ตาก” แทนที่จะเป็นคำว่า “ตา” นั่นเอง

และจากการทดลองนำประโยคต่างๆมาทดลองตัดคำ จะพบว่าโครงข่ายสามารถให้คำแนะนำได้อย่างถูกต้อง เฉพาะประโยคที่เป็นไปตามรูปแบบแผนที่โครงข่ายได้เรียนรู้ไว้แล้วเท่านั้น ส่วนรูปประโยคที่โครงข่ายไม่ได้เรียนรู้ไว้จะไม่สามารถให้คำแนะนำได้ และจะแจ้งว่าประโยคนั้นมีความผิดพลาด หลังจากนั้นจึงเลือกประโยคที่เป็นไปได้ในการตัดคำประโยคนั้นออกมา โดยจะพิจารณาเลือกประโยคที่มีจำนวนคำน้อยที่สุดเป็นประโยคผลลัพธ์ของโปรแกรม

บทที่ 4

บทสรุปและข้อเสนอแนะ

4.1 บทสรุปของโครงการพัฒนาระบบงาน

จากการพัฒนาโครงการในครั้งนี้ เราจะเห็นได้ว่าโครงข่ายประสาทเทียมเชิงลำดับ สามารถที่จะเรียนรู้ข้อมูลที่เป็นแบบแผนและเป็นเชิงลำดับได้เป็นอย่างดี สามารถนำไปประยุกต์ใช้ในงานต่างๆ ได้อย่างมากมาย ดังตัวอย่างในการพัฒนาโครงการนี้จะเป็นการนำโครงข่ายประสาทเทียมเชิงลำดับมาประยุกต์ใช้ในงานการตัดคำภาษาไทย เพื่อช่วยในการตัดสินใจกรณีที่เกิดความคลุมเครือว่าจะตัดคำออกมาในรูปแบบใด โดยทำหน้าที่เป็นเสมือนผู้ตรวจสอบและตัดสินใจว่าการตัดรูปแบบใดจึงจะถูกต้องตามรูปแบบที่โครงข่ายได้เรียนรู้มา และผลพลอยได้จากการใช้วิธีการนี้ตัดคำก็คือได้หน้าที่ของคำในประโยคนั้นด้วย ซึ่งจะเป็นประโยชน์อย่างยิ่งถ้าหากนำไปประยุกต์ใช้ในงานทางด้านการประมวลผลภาษาธรรมชาติ แต่อย่างไรก็ดีเนื่องจากรูปแบบการเรียงตัวของหน้าที่ของคำมีความหลากหลายมาก จึงเป็นไปได้ที่จะรวบรวมรูปแบบการเรียงตัวทุกรูปแบบมาเพื่อสอนให้โครงข่ายเรียนรู้ได้เรียนรู้ รวมถึงรูปประโยคที่มีลักษณะการเรียงตัวไม่เป็นปกติหรือเป็นส่วนหรือแสดง ต่างก็เป็นผลให้โครงข่ายไม่สามารถตัดสินใจได้อย่างถูกต้อง

4.2 ข้อเสนอแนะ

การพัฒนาโครงการนี้ยังมีข้อดีโดยตรงเรื่องความเร็วในการประมวลผล ถ้าหากต้องการนำไปประยุกต์ใช้กับงานอื่นๆ ต้องมีการปรับปรุงเรื่องนี้เสียก่อน อาจจะต้องเปลี่ยนแปลงวิธีการตัดคำทุกรูปแบบที่เป็นไปได้ด้วยวิธีการอื่นซึ่งมีความเร็วมากกว่า รวมถึงการเรียกใช้งานโปรแกรม SNNS ในโหมดของ DOS ทำได้ค่อนข้างช้าและมีการเรียกใช้งานหลายๆ ครั้งในการตัดคำเพียงหนึ่งประโยคเป็นผลทำให้การประมวลผลช้าลงยิ่งขึ้น วิธีการแก้ไขอาจจะต้องพัฒนาโปรแกรมในการสร้างและใช้งานโครงข่ายประสาทเทียมเชิงลำดับเอง จะทำให้ลดเวลาในการเรียกข้ามโหมดระหว่าง Windows และ DOS ไป และโครงการนี้ยังมีข้อจำกัดตรงที่ จะสามารถตัดคำได้เฉพาะที่มีอยู่ในฐานข้อมูลเท่านั้น ดังนั้นหากต้องการนำไปใช้งานจริงอาจจะต้องหาวิธีการอื่นๆ มาเสริมจุดบกพร่องตรงนี้ต่อไป

บรรณานุกรม

- Bechtel, W. and Abrahamsen, A. 1997. *Connectionism and the Mind*. Blackwell Publisher.
- Cleeremans, A. 1993. *Mechanism of Implicit Learning*. MIT Press.
- Garrison W. C, and Kim P. 1991. "Learning the past tense in a recurrent network acquiring the mapping from meaning to sounds". *Proceedings of the Thirteenth Annual Conference of the Cognitive Science Society*, 328-333
- Jordan, M.I. 1986. "Attractor dynamics and parallelism in a connectionist sequential machine." *Proceeding of the Eighth Annual Conference of the Cognitive Science Society*. Amherst, MA : 531-546
- Luz, W. et. al. 1991. "Speech recognition using a sequential neural network." *IEEE International Joint Conference on Neural Networks*. Vol. 1, 140-145.
- Miikkulainen, R., and Dyer, M.G. 1998. "A modular neural network architecture for sequential paraphrasing of script-based stories." *International Joint Conference on Neural Networks (IJCNN)*. Vol. 2, 49-56.
- SNN. 1995. "SNNS Stuttgart Neural Network Simulator User Manual, Version 4.1", Report No. 6/95, University of Stuttgart, URL: <http://www-ra.informatik.uni-tuebingen.de/SNNS/>.

ประวัติผู้เขียน

ชื่อผู้เขียน	นายวิทยา ตานี
สถานที่เกิด	เชียงราย
วุฒิการศึกษา	ค.อ.บ คณะครุศาสตร์อุตสาหกรรม สถาบันเทคโนโลยีพระจอมเกล้าเจ้า คุณทหารลาดกระบัง
ตำแหน่งหน้าที่ สถานที่ทำงาน	System Support บริษัท OGA International



เอกสารนี้เป็นเอกสารที่สงวนไว้สำหรับการใช้งานเพื่อการศึกษาเท่านั้น ไม่อนุญาตให้นำไปใช้ประโยชน์ด้านการค้า
ไม่ว่ากรณีใดๆทั้งสิ้น อีกทั้งห้ามมิให้ตัดแปลงเนื้อหา และต้องอ้างอิงถึงเจ้าของเอกสารทุกครั้งที่มีการนำไปใช้