

เว็บแอปพลิเคชันอัจฉริยะสำหรับการจำแนกกลุ่มเอกสารบนเว็บ
โดยใช้ตัวแทนอัจฉริยะ

INTELLIGENT WEB APPLICATION FOR WEB DOCUMENT
CLASSIFICATION USING INTELLIGENT AGENT

นางสาวภักทริยา	พิมานแมน
นางสาววาสนี	ห้องสวัสดิ์
นางสาวอริศรา	นพเก้า

โครงการพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรบัณฑิต
สาขาวิชาวิทยาการคอมพิวเตอร์
คณะวิทยาศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา ๒๕๕๘

เว็บแอปพลิเคชันอัจฉริยะสำหรับการจำแนกกลุ่มเอกสารบนเว็บ
โดยใช้ตัวแทนอัจฉริยะ

INTELLIGENT WEB APPLICATION FOR WEB DOCUMENT
CLASSIFICATION USING INTELLIGENT AGENT

นางสาวภัทรียา	พิมานแมน
นางสาววาสนี	ห้องสวัสดิ์
นางสาวอริสรา	นพเก้า

โครงการพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรวิทยาศาสตรบัณฑิต
สาขาวิชาวิทยาการคอมพิวเตอร์
คณะวิทยาศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2556

**INTELLIGENT WEB APPLICATION FOR WEB DOCUMENT
CLASSIFICATION USING INTELLIGENT AGENT**

MISS PATTREEYA	PIMARNMAN
MISS WASINEE	HONGSAWAT
MISS ARISSARA	NOPPAKAOW

**A SPECIAL PROJECT SUBMITTED IN PARTIAL FULFILLMENT
OF THE REQUIRMENT FOR THE DEGREE OF BACHELOR OF SCIENCE
IN COMPUTER SCIENCE
FACULTY OF SCIENCE
KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG
ACADEMIC YEAR 2013**

หัวข้อโครงการพิเศษ เว็บแอปพลิเคชันอัจฉริยะสำหรับการจำแนกกลุ่มเอกสารบนเว็บโดยใช้
 ตัวแทนอัจฉริยะ
 Intelligent Web Application for Web Document Classification Using
 Intelligent Agent

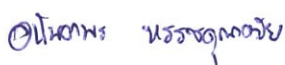


ชื่อนักศึกษา นางสาวภัทรียา พิมานแมน 53051046
 นางสาววาสิณี ห่องสวัสดิ์ 53052080
 นางสาวอริสรา นพเก้า 53051127

ปริญญา วิทยาศาสตรบัณฑิต

สาขาวิชา วิทยาการคอมพิวเตอร์

อาจารย์ที่ปรึกษา ดร.สายชล ใจเย็น

คณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง อนุมัติให้
 โครงการพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร วิทยาศาสตรบัณฑิต สาขาวิชาวิทยาการ
 คอมพิวเตอร์ ประจำปีการศึกษา 2556

คณะกรรมการสอบ	ลายมือชื่อ
ดร.อนันตพร หารรรษคุณาฒย ประธานกรรมการ	
รศ.ดร.จิรพร วีระพันธุ์ กรรมการ	
ดร.สายชล ใจเย็น กรรมการและอาจารย์ที่ปรึกษา	

ลิขสิทธิ์ของคณะวิทยาศาสตร์
 สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง

หัวข้อโครงการพิเศษ	เว็บแอปพลิเคชันอัจฉริยะสำหรับการจำแนกกลุ่มเอกสารบนเว็บโดยใช้ตัวแทนอัจฉริยะ		
ชื่อนักศึกษา	นางสาวภทริยา พิमानแมน	53051046	
	นางสาววาสนี ห่องสวัสดิ์	53051080	
	นางสาวอริสรา นพแก้ว	53051127	
ปริญญา	วิทยาศาสตร์บัณฑิต		
สาขาวิชา	วิทยาการคอมพิวเตอร์		
ปีการศึกษา	2556		
อาจารย์ที่ปรึกษา	ดร.สายชล ใจเย็น		

บทคัดย่อ

ในปัจจุบันนี้เป็นยุคของข้อมูลข่าวสาร ทำให้มีข้อมูลเอกสารทั้งที่เป็นประโยชน์และไม่เป็นประโยชน์ กระจายอยู่ตามเว็บไซต์เป็นจำนวนมาก ดังนั้นจึงมีเว็บไซต์ที่รวบรวมข้อมูลข่าวสารเกิดขึ้นเป็นจำนวนมาก ซึ่งโดยปกติแล้วเว็บไซต์เหล่านี้จะใช้คนเป็นผู้รวบรวมข่าวตามเว็บไซต์ต่างๆ และนำลิงค์ของแต่ละเว็บไซต์มาทำการจัดเรียงและสร้างดัชนีขึ้นมาเนื่องจากใช้คนเป็นผู้รวบรวมข่าว จึงมีข้อเสียคือถ้าเว็บไซต์มีการอัปเดตข่าวใหม่ จะต้องเข้าไปดูและรวบรวมข่าวในหน้าเว็บไซต์นั้นๆ ใหม่ ทำให้เกิดความล่าช้าในการอัปเดตข้อมูลข่าวใหม่ๆ และนอกจากนั้นยังมีค่าใช้จ่ายในการจัดจ้างอีกด้วย ดังนั้นวัตถุประสงค์ของการจัดทำปัญหาพิเศษนี้คือ การนำตัวแทนอัจฉริยะหรือซอฟต์แวร์ที่ทำหน้าที่เปรียบเสมือนตัวแทนในการเก็บรวบรวมข่าวที่เราต้องการมาทำการจำแนกข่าวออกเป็น 4 ประเภท คือ ข่าวไอที ข่าวบันเทิงข่าวการเมือง และข่าวกีฬา โดยการสอนตัวแทนอัจฉริยะนี้ให้สามารถแยกประเภทได้ว่าข่าวนั้นเป็นข่าวประเภทใด โดยใช้เทคนิคการเรียนรู้ของเครื่องและทำการสร้างเว็บแอปพลิเคชันเพื่อเก็บรวบรวมข่าวดังกล่าว

คำสำคัญ: ตัวแทนอัจฉริยะ, เว็บแอปพลิเคชัน, การเรียนรู้ของเครื่อง

Title	Intelligent Web Application for Web Document Classification Using Intelligent Agent		
Students	Miss Pattreeya Pimarnman		53051046
	Miss Wasinee Hongswat		53051080
	Miss Arissara Noppakaow		53051127
Degree	Bachelor of Science		
Major Program	Computer Science		
Academic Year	2013		
Advisor	Saichon Jaiyen, Ph.D.		

ABSTRACT

Currently this is the era of information with the number of documents both useful and useless scattered around the world. Consequently, there are many websites that collect information and news. Typically, these websites will use people to gather news on the website and put the link of each website to arrange and create indices. Because the people gather news, the disadvantage is when a website has updated news, we need to take a look and get the news on that website. As a result, sometimes cause delays to update new stories and also the cost of hiring. The purpose of this special problem is to develop the intelligent agent that can collect news and then classify them into 4 categories including IT, Entertainment, Politics and Sports. The intelligent agent is trained to be able to classify the type of news by using Machine Learning techniques and create a web application to collect such news.

Keywords: Intelligent Agent, Web Application, Machine Learning

กิตติกรรมประกาศ

ในการทำปัญหาพิเศษเรื่อง เว็บแอปพลิเคชันอัจฉริยะสำหรับการจำแนกกลุ่มเอกสารบนเว็บ โดยใช้ตัวแทนอัจฉริยะสามารถสำเร็จลุล่วงไปได้ด้วยดี คณะผู้จัดทำขอขอบพระคุณ ดร.สายชล ใจเย็น ผู้รับผิดชอบปัญหาพิเศษฉบับนี้ ที่กรุณาให้คำปรึกษาและคำแนะนำด้านการศึกษา ปัญหา การออกแบบระบบ และแนวทางการแก้ไขปัญหา ตลอดจนการตรวจสอบและแก้ไขการเขียนรายงานปัญหาพิเศษเล่มนี้และ ต้องขอขอบพระคุณ ดร.อนันตพร หารรัชคุณาชัย และ รศ.ดร.จิรพร วีระพันธุ์ ซึ่งเป็นประธานกรรมการและกรรมการ ที่กรุณาให้คำแนะนำในการทำปัญหาพิเศษฉบับนี้

นอกจากนี้คณะผู้จัดทำต้องขอขอบพระคุณ บิดา มารดา และผู้ปกครองที่ได้ให้คำปรึกษาในเรื่องต่างๆ รวมทั้งเป็นกำลังใจที่ดีเสมอมา

ขอบคุณเพื่อนๆทุกคน ที่คอยให้ความช่วยเหลือในด้านต่างๆ เกี่ยวกับปัญหาพิเศษนี้ จนทำให้ปัญหาพิเศษสำเร็จลุล่วงไปได้ด้วยดี

สุดท้ายนี้ต้องขอบคุณตัวเองที่สามารถอดทนต่ออุปสรรคต่างๆ จนกระทั่งปัญหาพิเศษนี้สามารถสำเร็จได้ด้วยดี

คณะผู้จัดทำ

มีนาคม 2557

สารบัญ

	หน้า
บทคัดย่อภาษาไทย	I
บทคัดย่อภาษาอังกฤษ	II
กิตติกรรมประกาศ	III
สารบัญ	IV
สารบัญตาราง	VIII
สารบัญรูปภาพ	IX
บทที่ 1 บทนำ	1
1.1 ที่มาและความสำคัญ	1
1.2 วัตถุประสงค์ของงานวิจัย	2
1.3 ขอบเขตของปัญหา	2
1.4 ประโยชน์ที่คาดว่าจะได้รับ	2
1.5 ขั้นตอนในการดำเนินงาน	2
บทที่ 2 ทฤษฎีและงานวิจัยที่เกี่ยวข้อง	3
2.1 ตัวแทนปัญญา (Intelligent Agent)	3
2.1.1 ทำความรู้จักกับตัวแทนปัญญา	3
2.1.2 ทำความรู้จักกับสภาพแวดล้อม	6
2.1.2.1 การกำหนดสภาพแวดล้อมของงาน	6
2.1.2.2 คุณสมบัติของสภาพแวดล้อม	7
2.1.3 ประเภทของตัวแทนปัญญา	10
2.1.3.1 Simple Reflex Agents	10
2.1.3.2 Model-based Reflex Agents	11
2.1.3.3 Goal-based Agent	11
2.1.3.4 Utility-based Agents	12
2.1.3.5 Learning Agents	12
2.1.4 Multi Agent System	13
2.2 การแทนความรู้ (Knowledge Representation)	14
2.2.1 ชนิดของความรู้	14
2.2.1.1 ความรู้ที่สะสมมาตั้งแต่ในอดีต (Long-term Knowledge)	14

สารบัญ (ต่อ)

	หน้า
2.2.1.2 ความรู้ที่เกิดขึ้นที่จากการรับรู้ (Short-term Knowledge)	14
2.2.1.3 ความรู้ใหม่ที่เกิดจากการใช้สติปัญญาของมนุษย์ (Inference Knowledge)	14
2.2.2 ระบบการแทนความรู้	15
2.3 การเรียนรู้ของเครื่อง (Machine Learning)	16
2.3.1 สถาปัตยกรรมการเรียนรู้ของเครื่อง	17
2.3.2 ขั้นตอนวิธีการเรียนรู้แบบมีผู้สอน (Supervised Learning)	17
2.3.2.1 โครงข่ายประสาทเทียม (Neural Network)	18
2.4 เหมืองข้อมูล (Data Mining)	24
2.4.1 แนวคิดและความหมายของเหมืองข้อมูล	24
2.4.2 สถาปัตยกรรมของการทำเหมืองข้อมูล	26
2.4.2.1 แหล่งข้อมูล	26
2.4.2.2 เซิร์ฟเวอร์	26
2.4.2.3 ฐานความรู้	26
2.4.2.4 ตัวประมวลผลเหมืองข้อมูล	26
2.4.2.5 ตัวประเมินคุณภาพของความรู้ที่ได้จากการทำเหมืองข้อมูล	26
2.4.2.6 ส่วนต่อประสานไปยังผู้ใช้	26
2.4.3 ขั้นตอนในการทำเหมืองข้อมูล	26
2.4.3.1 การทำความเข้าใจปัญหา	26
2.4.3.2 ทำความเข้าใจข้อมูล	27
2.4.3.3 การเตรียมข้อมูล	27
2.4.3.4 การสร้างแบบจำลอง	27
2.4.3.5 การประเมิน	28
2.4.4 การทำเหมืองข้อมูลเว็บ (Web Mining)	28
2.4.4.1 Web Content Mining	28
2.4.4.2 Web Structure Mining	28
2.4.4.3 Web Usage Mining	28

สารบัญ (ต่อ)

	หน้า
2.4.5 เทคนิคการทำเหมืองข้อมูลเว็บ	29
2.4.5.1 Statistical Analysis	29
2.4.5.2 Association Rule	29
2.4.5.3 Clustering	30
2.4.5.4 Classification	30
2.4.5.5 Sequential Pattern	30
2.4.5.6 Dependency Modeling	30
2.4.6 ขั้นตอนการทำเหมืองข้อมูลเว็บ	30
2.4.6.1 การเตรียมข้อมูล (Preprocessing)	30
2.4.6.2 การค้นหารูปแบบ (Pattern Discovery)	31
2.4.6.3 การวิเคราะห์รูปแบบ (Pattern Analysis)	31
2.5 เว็บแอปพลิเคชัน (Web Application)	31
2.5.1 ประเภทของเว็บแอปพลิเคชัน	32
2.5.2 ความหมายของเว็บแอปพลิเคชัน	32
2.5.3 ลักษณะการทำงานของเว็บแอปพลิเคชัน	32
2.5.4 ประโยชน์ของเว็บแอปพลิเคชัน	33
2.6 ASP.NET	34
2.6.1 ความหมายของ ASP.NET	34
2.6.2 ASP.NET MVC	34
2.6.3 ความแตกต่างระหว่าง ASP.NET Web Form และ ASP.NET MVC	35
2.6.4 ข้อดีของ MVC Pattern	36
2.7 ไวยากรณ์พื้นฐาน	37
2.7.1 ประกาศตัวแปร	37
2.7.1.1 การตั้งชื่อตัวแปร	37
2.7.1.2 ชนิดตัวแปร	38
2.7.2 การกำหนดเงื่อนไข	39
2.7.2.1 การใช้งาน if	39
2.7.2.2 การใช้งาน if-else	39

สารบัญ (ต่อ)

	หน้า
2.7.2.3 การใช้งาน if-else if	40
2.7.2.4 การใช้งาน switch/select case	40
2.7.3 การวนลูป(Loop)	41
2.7.3.1 การใช้งาน for	41
2.7.3.2 การใช้งาน while	42
2.7.3.3 การใช้งาน do while	42
2.7.3.4 การใช้งาน foreach	43
2.7.4 การสร้าง Array	43
2.8 การกำหนดStyle ด้วย CSS	44
2.8.1 ความหมายของ Cascading style Sheet (CSS)	44
2.8.2 พื้นฐานการใช้งาน CSS	44
2.8.2.1 Comment	44
2.8.2.2 Selectors	45
2.8.2.3 การกำหนดสี	47
2.8.2.4 การจัดการกับตัวอักษร	49
2.8.2.5 การกำหนดStyle ด้วย Background	50
2.9 การติดต่อฐานข้อมูล	51
2.9.1 ความหมายของฐานข้อมูล	51
2.9.2 คำสั่งในภาษา SQL	53
2.9.2.1 คำสั่ง Insert	54
2.9.2.2 คำสั่ง Update	54
2.9.2.3 คำสั่ง Delete	55
2.9.2.4 คำสั่ง Select	55
บทที่ 3 ขั้นตอนการดำเนินงาน	57
3.1การสร้างแบบจำลองสำหรับการจำแนกข่าวแต่ละประเภท	58
3.1.1 ขั้นตอนการรวบรวมข้อมูล (Collecting Data)	59
3.1.2 ขั้นตอนการเลือกคุณลักษณะ (Feature Extraction)	59
3.1.3 ขั้นตอนการสร้างดัชนี (Indexing)	60

สารบัญ (ต่อ)

	หน้า
3.1.4 ขั้นตอนการใช้อัลกอริทึมในการแบ่งประเภท (Classification Algorithm)	60
3.1.4.1 การฝึกสอน โครงข่าย (Training Process)	60
3.1.4.1 การทดสอบ โครงข่าย (Testing Process)	64
3.2 ขั้นตอนการออกแบบและพัฒนาเว็บแอปพลิเคชัน	65
3.2.1 การวิเคราะห์ระบบ	65
3.2.1.1 ความต้องการของระบบ	66
3.2.1.2 Use Case Diagram	66
3.2.1.3 คำอธิบาย Use Case	67
3.2.1.4 ตารางของระบบฐานข้อมูล	70
3.2.2 การออกแบบเว็บแอปพลิเคชัน	71
3.2.2.1 ส่วนของเว็บแอปพลิเคชัน	71
3.2.2.2 ส่วนติดต่อผู้ใช้	71
บทที่ 4 ผลการดำเนินงาน	72
4.1 ผลการดำเนินงานในส่วนของการสร้างแบบจำลอง	72
4.1.1 ผลการดำเนินงานจากการฝึกสอน โครงข่าย (Training Process)	72
4.1.2 ผลการดำเนินงานจากการทดสอบ โครงข่าย (Testing Process)	74
4.2 ผลการดำเนินงานในส่วนของการสร้างเว็บแอปพลิเคชัน	76
4.2.1 ความสามารถในการเข้าใช้ระบบของผู้ใช้	76
บทที่ 5 สรุปผลการดำเนินงานและข้อเสนอแนะ	80
5.1 สรุปผลการดำเนินงาน	80
5.1.1 สรุปผลการทำนายโดยใช้แบบจำลองนิเวศน์เน็ตเวิร์ค	80
5.1.2 สรุปผลการพัฒนาเว็บแอปพลิเคชัน	81
5.2 ปัญหาที่พบในการดำเนินงาน	81
5.3 ข้อเสนอแนะ	81
เอกสารอ้างอิง	83
ภาคผนวก ก. วิธีการติดตั้ง โปรแกรม Visual Studio Ultimate 2013	87
ภาคผนวก ข. วิธีการติดตั้ง โปรแกรม MATLAB	92
ภาคผนวก ค. วิธีการติดตั้ง โปรแกรม Copernic Summarizer	98

สารบัญตาราง

	หน้า
ตารางที่ 2.1 ชนิดของตัวแปรในภาษา C#	38
ตารางที่ 2.2 ตาราง Department	52
ตารางที่ 2.3 ตาราง Employee	53
ตารางที่ 3.1 ตัวอย่างบางส่วนของคำที่ไม่เกี่ยวข้องในการค้นหาหรือสืบค้น (Stop Word)	59
ตารางที่ 3.2 เวกเตอร์ตัวแทนเอกสาร	60
ตารางที่ 3.3 ผลการทดลองในการหาจำนวน โหนดในชั้นซ่อนที่ให้ค่าความผิดพลาดน้อยที่สุด	61
ตารางที่ 3.4 คำอธิบายของ U1: ดึงข่าวจากเว็บไซต์ภายนอก	67
ตารางที่ 3.5 คำอธิบายของ U2: แยกประเภทข่าว	67
ตารางที่ 3.6 คำอธิบายของ U3: เก็บรวบรวมข่าว	68
ตารางที่ 3.7 คำอธิบายของ U4: แสดงรายการข่าว	68
ตารางที่ 3.8 คำอธิบายของ U5: เชื่อมโยงไปยังเว็บไซต์ภายนอก	69
ตารางที่ 3.9 คำอธิบายของ U6: เลือกประเภทข่าว	69
ตารางที่ 3.10 คำอธิบายของ U7: เลือกรายการข่าว	70
ตารางที่ 3.11 ตาราง News	70
ตารางที่ 4.1 Confusion Matrix ของแบบจำลอง	75
ตารางที่ 4.2 การวัดประสิทธิภาพการทำงานของแบบจำลอง	75
ตารางที่ 5.1 Confusion Matrix ของเว็บแอปพลิเคชัน	81
ตารางที่ 5.2 การวัดประสิทธิภาพการทำงานของเว็บแอปพลิเคชัน	82

สารบัญรูปภาพ

	หน้า
รูปที่ 2.1 โครงสร้างเบื้องต้นของเอเจนต์	4
รูปที่ 2.2 โครงสร้างและการทำงานของ Simple Reflex Agents	10
รูปที่ 2.3 โครงสร้างและการทำงานของ Model-based Reflex Agents	11
รูปที่ 2.4 โครงสร้างและการทำงานของ Goal-based Agent	12
รูปที่ 2.5 โครงสร้างและการทำงานของ Utility-based Agent	12
รูปที่ 2.6 โครงสร้างและการทำงานของ Learning Agent	13
รูปที่ 2.7 ลักษณะโครงสร้างการทำงานภายในระบบ MAS	13
รูปที่ 2.8 องค์ประกอบหลักของระบบการแทนความรู้	15
รูปที่ 2.9 โครงสร้างของโครงข่ายประสาทเทียม	19
รูปที่ 2.10 โครงสร้างของ MLP	20
รูปที่ 2.11 วิวัฒนาการในการจัดเก็บและตีความหมายข้อมูล	25
รูปที่ 2.12 ขั้นตอนการทำ Web Usage Mining	31
รูปที่ 2.13 ขั้นตอนการทำงานของเว็บแอปพลิเคชัน	32
รูปที่ 2.14 ความแตกต่างระหว่าง ASP.NET Web Form กับ ASP.NET MVC	35
รูปที่ 3.1 แผนภาพแสดงภาพรวมของการพัฒนาเว็บแอปพลิเคชัน	57
รูปที่ 3.2 โครงสร้างของกระบวนการสร้างแบบจำลอง	58
รูปที่ 3.3 ข้อมูลอินพุตสำหรับการฝึกสอน	61
รูปที่ 3.4 ค่าความผิดพลาดเฉลี่ยเมื่อกำหนดให้จำนวนโหนดในชั้นซ่อนมี 5 โหนด	61
รูปที่ 3.5 ค่าความผิดพลาดเฉลี่ยเมื่อกำหนดให้จำนวนโหนดในชั้นซ่อนมี 10 โหนด	63
รูปที่ 3.6 ค่าความผิดพลาดเฉลี่ยเมื่อกำหนดให้จำนวนโหนดในชั้นซ่อนมี 15 โหนด	63
รูปที่ 3.7 ค่าเป้าหมายสำหรับการฝึกสอน	64
รูปที่ 3.8 การฝึกสอนโครงข่ายโดยใช้ Neural Network Toolbox ในโปรแกรม Matlab	64
รูปที่ 3.9 ข้อมูลอินพุตสำหรับการทดสอบ	65
รูปที่ 3.10 ค่าเป้าหมายสำหรับการทดสอบ	65
รูปที่ 3.11 Use Case Diagram ของระบบ	66
รูปที่ 3.12 โครงสร้างโดยรวมของเว็บแอปพลิเคชัน	71
รูปที่ 4.1 คำนวณน้ำหนักระหว่างชั้นซ่อนกับชั้นอินพุต	72
รูปที่ 4.2 คำนวณน้ำหนักระหว่างชั้นซ่อนกับชั้นเอาต์พุต	73

สารบัญรูปภาพ (ต่อ)

	หน้า
รูปที่ 4.3 ค่าไบแอสในชั้นซ่อน	73
รูปที่ 4.4 ค่าไบแอสในชั้นเอาต์พุต	73
รูปที่ 4.5 ความสัมพันธ์ระหว่างค่า MSE กับค่า Epochs	74
รูปที่ 4.6 หน้าแรกของเว็บแอปพลิเคชัน	77
รูปที่ 4.7 หน้าแรกของข่าวไอที	77
รูปที่ 4.8 หน้าหลักของข่าวบันเทิง	78
รูปที่ 4.9 หน้าหลักของข่าวการเมือง	78
รูปที่ 4.10 หน้าหลักของข่าวกีฬา	79
รูปที่ ก.1 หน้าเว็บ ไซตส์ที่ทำการดาวน์โหลด	87
รูปที่ ก.2 หน้ารายละเอียด	87
รูปที่ ก.3 คุณสมบัติของ Visual Studio ที่จะทำการติดตั้ง	88
รูปที่ ก.4 หน้าต่างแสดงการติดตั้ง	88
รูปที่ ก.5 หน้าต่างหลังจากการติดตั้งเสร็จเรียบร้อยแล้ว	89
รูปที่ ก.6 หน้าต่างสำหรับการเข้าสู่บัญชีไมโครซอฟท์	89
รูปที่ ก.7 รูปแบบของโปรแกรม	90
รูปที่ ก.8 หน้าเริ่มต้นของ Visual Studio 2013	90
รูปที่ ข.1 หน้าเริ่มต้นการติดตั้ง	91
รูปที่ ข.2 หน้ารายละเอียด	91
รูปที่ ข.3 หน้าต่างสำหรับการป้อน Installation Key	92
รูปที่ ข.4 รูปแบบการติดตั้ง	92
รูปที่ ข.5 หน้าต่างสำหรับการเลือกไอเดคทอรีพาทสำหรับการติดตั้ง	93
รูปที่ ข.6 หน้าต่างคุณสมบัติที่ต้องการติดตั้ง	93
รูปที่ ข.7 หน้าต่างการสำหรับการเลือก License.dat file	94
รูปที่ ข.8 หน้าต่าง Location	94
รูปที่ ข.9 หน้าต่างข้อมูลรายละเอียด	95
รูปที่ ข.10 ทำการติดตั้งเรียบร้อยแล้ว	95
รูปที่ ค.1 การตั้งค่าพารามิเตอร์	96
รูปที่ ค.2 ผลลัพธ์ที่ได้จากการรันโปรแกรม Copernic Summarizer	96

บทที่ 1

บทนำ

1.1 ที่มาและความสำคัญ

ในอดีตเราสามารถติดตามข่าวสารที่เกิดขึ้นในแต่ละวันได้โดยการอ่านจากสื่อสิ่งพิมพ์ต่างๆ ติดตามรายการข่าวทางโทรทัศน์ หรือฟังข่าวจากวิทยุ ซึ่งในแต่ละวันย่อมมีข่าวเกิดขึ้นหลากหลายประเภท เช่น ข่าวกีฬา ข่าวการเมือง ข่าวบันเทิง ข่าวไอที ข่าวอาชญากรรม ข่าวเศรษฐกิจ เป็นต้น การติดตามข่าวสารที่มากมายนี้ อาจทำให้ผู้รับข่าวสารมีโอกาสพลาดการติดตามข่าวที่ตนเองสนใจ เมื่อการใช้อินเทอร์เน็ตเริ่มเป็นที่แพร่หลายมากขึ้น ผู้คนติดต่อสื่อสารกันได้สะดวกและรวดเร็ว มีการนำเสนอข่าวสารผ่านทางเว็บไซต์เพิ่มขึ้นเป็นจำนวนมาก ทำให้ผู้ใช้นั้นได้รับข้อมูลข่าวสารที่ทันสมัยอยู่ตลอดเวลาซึ่งการนำเสนอข่าวในแต่ละเว็บไซต์ของสำนักข่าวนั้นจะมีแนวทางในการนำเสนอที่แตกต่างกันไป ทั้งเนื้อหาและรายละเอียดปลีกย่อยของข่าว ดังนั้นเมื่อผู้ใช้อต้องการอ่านข่าวจากเว็บไซต์ อาจเจอข่าวเรื่องเดียวกันในเว็บไซต์ต่างๆ แต่มีเนื้อหาข่าวต่างกัน หรือในบางครั้งผู้ใช้อาจต้องการอ่านข่าวแค่ประเภทใดประเภทหนึ่ง โดยเฉพาะ ทำให้ต้องสืบค้นหาข่าวประเภทนั้นๆ จากเว็บอื่นเพิ่มเติมอีก จึงเป็นการเสียเวลาในการสืบค้น อีกทั้งบางเว็บไซต์ยังมีโฆษณาแฝงต่างๆ มากมาย

ในปัจจุบันนี้เป็นยุคของข้อมูลข่าวสาร ทำให้มีข้อมูลเอกสารทั้งที่เป็นประโยชน์และไม่เป็นประโยชน์ กระจายอยู่ตามเว็บไซต์เป็นจำนวนมาก ดังนั้น เมื่อผู้ใช้อต้องการค้นหาเรื่องที่ตนเองสนใจ จะต้องเข้าเว็บไซต์ที่ให้บริการสืบค้น (Search Engine) เพื่อทำการค้นหาข้อมูล ผลลัพธ์ที่ได้ อาจมีทั้งที่เกี่ยวข้องและไม่เกี่ยวข้องกับสิ่งที่เราต้องการ จึงมีเว็บไซต์ที่ให้บริการรวบรวมข่าวสารเกิดขึ้น ซึ่งโดยปกติแล้วเว็บไซต์เหล่านี้จะใช้คนเป็นผู้รวบรวม โดยให้คนทำการค้นหาข่าวตามเว็บไซต์ต่างๆ และนำลิงค์ของแต่ละเว็บไซต์มาเรียงกันและสร้างดัชนีขึ้นมา ซึ่งมีข้อเสียคือ ทุกครั้งที่มีการอัปเดตข่าวใหม่ๆ จะต้องเข้าไปดูข้อมูลในเว็บไซต์นั้นๆ และนำมาทำการอัปเดตข่าว ซึ่งทำให้เสียเวลา และเนื่องจากใช้คนในการทำ จึงทำให้มีค่าใช้จ่ายสูง จึงได้จัดทำเว็บแอปพลิเคชันที่รวบรวมข่าวสารไว้ในที่เดียวกัน โดยจะทำการจำแนกข่าวออกเป็น 4 ประเภท คือ ข่าวกีฬา ข่าวการเมือง ข่าวไอที และข่าวบันเทิง โดยการนำตัวแทนปัญญา (Intelligent Agent) หรือซอฟต์แวร์ที่ทำหน้าที่เปรียบเสมือนตัวแทนในการเก็บรวบรวมข่าวที่เราต้องการ และสามารถตัดสินใจได้ว่าข่าวนี้เป็นข่าวประเภทใด โดยทำการสอนให้ตัวแทนปัญญานั้นฉลาดขึ้น ซึ่งทำได้โดยใช้เทคนิคการเรียนรู้ของเครื่อง (Machine Learning)

การทำงานของตัวแทนปัญญา (Intelligent Agent) นั้น เอเจนต์จะเข้าไปตามลิงค์ของเว็บไซต์และจะไปดูเนื้อหาในเว็บไซต์นั้นๆ ถ้าไม่ใช่ข้อมูลที่ต้องการก็จะไม่เก็บ จากนั้นจะดูว่าใน

เว็บไซต์นั้นมีลิงค์เชื่อมไปเว็บไซต์ได้อีกบ้างก็จะเข้าลิงค์ต่อๆ ไปอีก เมื่อเจอข้อมูลที่ต้องการก็จะทำการเก็บลงฐานข้อมูล แต่ในปัญหาพิเศษนี้ จะเป็นการนำตัวเอเจนต์ไปติดตามเว็บที่ได้เลือกไว้ และให้เอเจนต์ไปตรวจเนื้อหาข่าว จากนั้น จะทำการดึงลิงค์ข่าวมารวบรวมไว้ด้วยกัน ด้วยเหตุนี้ เพื่อแก้ปัญหาให้ผู้ใช้สามารถค้นหาข่าวที่ต้องการได้ตรงประเด็นและครอบคลุมที่สุด

1.2 วัตถุประสงค์

1. เพื่อจำแนกประเภทของข่าวว่าเป็น ข่าวการเมือง ข่าวกีฬา ข่าวบันเทิง หรือข่าวไอที โดยใช้ตัวแทนปัญญา
2. เพื่อรวบรวมข่าวแต่ละประเภทไว้ในเว็บแอปพลิเคชันเดียวกันเพื่อความสะดวกในการค้นหาข่าวแบบเฉพาะเจาะจง

1.3 ขอบเขตของปัญหา

เว็บแอปพลิเคชันนี้จะใช้ตัวแทนปัญญาในการจำแนกประเภทข่าวเฉพาะข่าวการเมือง ข่าวกีฬา ข่าวบันเทิง และข่าวไอที และสามารถทำงานได้โดยอิสระ ไม่ขึ้นกับระบบปฏิบัติการที่ใช้งานอยู่

1.4 ประโยชน์ที่คาดว่าจะได้รับ

ผู้ใช้สามารถค้นหาข่าวสารต่างๆ ได้รวดเร็วและตรงตามความต้องการ เนื่องจากมีการรวบรวมข่าวแต่ละประเภทไว้ในที่เดียวกัน

1.5 ขั้นตอนในการดำเนินงาน

1. ค้นหาข่าวการเมือง ข่าวกีฬา ข่าวบันเทิง และข่าวไอที จากเว็บไซต์ นำมาจัดเก็บเป็นคลังข้อมูล
2. คัดแยกคำในแต่ละข่าวที่จะใช้เป็นคำสำคัญ (Keyword) โดยใช้อัลกอริทึมตัดคำที่ไม่เกี่ยวข้อง (Stop Word) และคำซ้ำออกไป ซึ่งจะเหลือเฉพาะคำสำคัญเท่านั้น จากนั้นเก็บรวบรวมไว้ในไฟล์เดียวกัน
3. เขียนโปรแกรมเพื่อนับคำโดยนำคำสำคัญที่ได้นั้น มาทำการค้นหาว่าในแต่ละข่าวมีคำนั้นๆ อยู่กี่คำ
4. ใช้เทคนิคการเรียนรู้ของเครื่อง (Machine Learning) เรียนรู้ข้อมูลของคำที่ได้นับไว้เพื่อนำไปสร้าง ตัวแทนปัญญา (Intelligent Agent)
5. สร้างเว็บแอปพลิเคชันสำหรับรวบรวมข่าวสาร

บทที่ 2

ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1 ตัวแทนปัญญา (Intelligent Agent)

เทคโนโลยีทางด้านปัญญาประดิษฐ์มีอยู่มากมายที่ถูกนำมาประยุกต์ใช้งานร่วมกับงานต่างๆ ตามความต้องการของมนุษย์ “ตัวแทนปัญญา (Intelligent Agent: AI)” นับเป็นอีกหนึ่งเทคโนโลยีที่ได้รับการพัฒนาเพื่อรองรับความต้องการดังกล่าว โดยมีแนวโน้มว่าจะกลายเป็นเทคโนโลยีหรือเครื่องมือที่สำคัญของงานทางด้านสารสนเทศงานทางด้านวิทยาศาสตร์คอมพิวเตอร์และงานวิจัยทางด้านปัญญาประดิษฐ์ เป็นต้น ตัวแทนปัญญาสามารถทำงานได้ด้วยอัตโนมัติ และสามารถแก้ไข ปัญหาหรือตัดสินใจได้ด้วยตัวเองโดยปราศจากการควบคุมจากมนุษย์ ปัจจุบันตัวแทนปัญญาถูกนำมาใช้งานในชีวิตประจำวันมากขึ้น ยกตัวอย่างเช่น เว็บไซต์ Search Engine บางเว็บจะมีตัวแทนปัญญาทำหน้าที่ช่วยให้โปรแกรมสามารถเข้าถึงฐานข้อมูลที่มีขนาดใหญ่ หรือสามารถเสาะแสวงหาข้อมูลที่มีจำนวนมาก เพื่อเลือกเฉพาะข้อมูลที่ผู้ใช้ต้องการได้อย่างมีประสิทธิภาพ เป็นต้น

2.1.1 ทำความรู้จักกับตัวแทนปัญญา

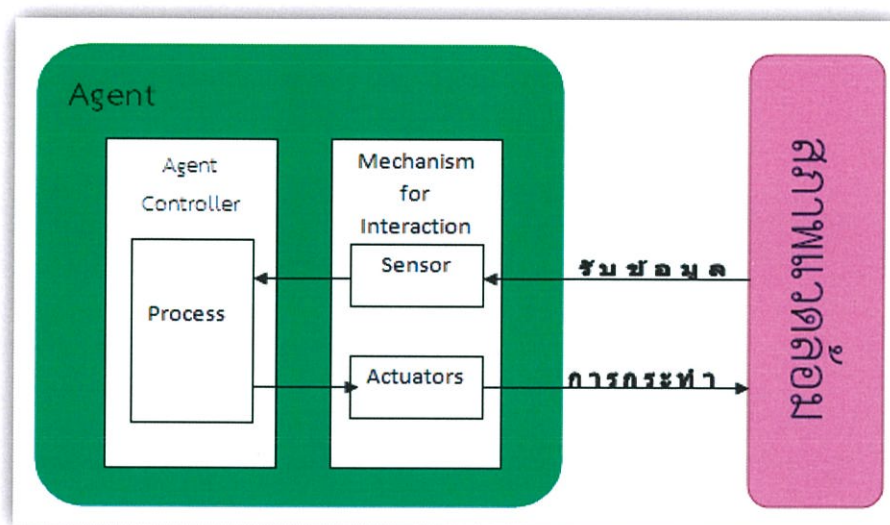
เอเจนต์ (Agent) หมายถึง ผู้ที่ได้มอบหมายให้ทำหน้าที่ตามที่มนุษย์ต้องการ หรือผู้ทำหน้าที่บางอย่างแทนมนุษย์ โดยเอเจนต์สามารถแบ่งได้เป็น 2 ลักษณะคือ

- Soft Agent เป็นเอเจนต์ที่มีลักษณะเป็นซอฟต์แวร์ ตัวอย่างเช่น ตัวแทนปัญญา และไวรัสคอมพิวเตอร์
- Hard Agent เป็นเอเจนต์ที่มีลักษณะทางกายภาพ สามารถจับต้องได้ ตัวอย่างเช่น สัตว์หรือหุ่นยนต์ เป็นต้น

โครงสร้างพื้นฐานของเอเจนต์จะต้องประกอบไปด้วย 2 ส่วนหลัก คือ

- 1) Mechanism for Interaction เป็นส่วนของกลไกการรับข้อมูลผ่าน Sensor และตอบสนองผ่าน Actuators กับสภาพแวดล้อมภายนอกได้ โดยส่วนนี้เป็นส่วนสำคัญที่เอเจนต์ควรมี
- 2) Agent Controller เป็นส่วนของตัวควบคุมกระบวนการทำงานของเอเจนต์ โดยลักษณะภายในส่วนนี้จะแตกต่างกันขึ้นอยู่กับเอเจนต์แต่ละประเภท

เอเจนต์ทำงานโดยรับข้อมูลต่างๆอย่างเป็นลำดับ (Percept Sequence) จากสภาพแวดล้อม (Environments) ภายนอกผ่านทางตัว Sensor ของเอเจนต์และสามารถตอบสนองกลับการกระทำไปยังสภาพแวดล้อมภายนอกผ่านทาง Actuators ของเอเจนต์ได้ แสดงโครงสร้างการทำงานเบื้องต้นของตัวเอเจนต์ ได้ดังรูปที่ 2.1



รูปที่ 2.1 โครงสร้างเบื้องต้นของเอเจนต์

หากพิจารณาตัวอย่างเอเจนต์ เช่น หุ่นยนต์ (Robotic Agent) จะพบว่ามีส่วน Sensor สามารถรับข้อมูลจากสิ่งแวดล้อมภายนอกได้ เช่น กล้องเครื่องตรวจจับรังสีอินฟราเรด Sensor วัดอุณหภูมิ ส่วนมอเตอร์ แขนหุ่นยนต์ ขาหุ่นยนต์ ทำหน้าที่เป็น Actuators สำหรับการกระทำเพื่อโต้ตอบกลับไปยังสภาพแวดล้อมภายนอก นอกจากนี้มนุษย์ยังสามารถเป็นเอเจนต์ได้เช่นกัน เนื่องจากมี หู ตา หรืออวัยวะอื่นๆที่สามารถตอบสนองกลับได้ เป็นต้น

จากการที่ได้ศึกษาแนวคิดของเอเจนต์แล้ว พบว่าเอเจนต์จะทำงานตามกระบวนการที่ได้รับข้อมูลต่างๆจากสภาพแวดล้อม โดยยังไม่มีกระบวนการคิดและการนำกระบวนการความฉลาดมาใช้ ดังนั้น นักพัฒนาจึงได้สร้างกลไกความฉลาดขึ้นภายในเอเจนต์ซึ่งเรียกว่า “ตัวแทนปัญญา (Intelligent Agent)”

ตัวแทนปัญญา (Intelligent Agent : IA) หมายถึงหน่วยของซอฟต์แวร์ที่สามารถดำเนินการหรือประมวลผลบางอย่างให้กับผู้ใช้หรือให้กับโปรแกรมอื่นได้อย่างรวดเร็วและอัตโนมัติ การดำเนินการบางอย่างจะต้องใช้องค์ความรู้เข้ามาพิจารณา โดยองค์ความรู้สามารถได้รับจากการเขียนโปรแกรมจากผู้พัฒนา หรือรับรู้มาจากข้อมูลที่รับมาจากสภาพแวดล้อม เพื่อนำเอาองค์ความรู้เหล่านั้นมาสรุปความ เพื่อให้ได้เป็นการกระทำตามความต้องการของผู้ใช้หรือ โปรแกรมอื่นได้ วัตถุประสงค์หลักของตัวแทนปัญญา คือ การทำหน้าที่บางอย่างแทนมนุษย์ ซึ่งเปรียบเสมือน

ตัวแทนของมนุษย์ โดยลักษณะการดำเนินงานจะเป็นแบบอัตโนมัติ และผลลัพธ์ที่ได้จะต้องมีประสิทธิภาพมากกว่าหรือเทียบเท่ากับมนุษย์

คุณสมบัติของตัวแทนปัญญา

- 1) Autonomous ตัวแทนปัญญาจะต้องมีความเป็นอิสระ สามารถดำเนินการด้วยตนเองได้อย่างอัตโนมัติ และปราศจากการควบคุมจากมนุษย์ โดยมีวัตถุประสงค์เพื่อบรรลุเป้าหมายที่ผู้ใช้ต้องการ
- 2) Rationale ผลกระทำของตัวแทนปัญญาจะต้องตอบสนองได้อย่างมีเหตุผล และสอดคล้องกับสภาพแวดล้อมปัจจุบันให้มากที่สุด
- 3) Dedicate & Automated การดำเนินงานบางอย่างอาจต้องการกระทำซ้ำๆกันทุกวัน ทำให้จำเป็นต้องนำตัวแทนปัญญามาช่วยดำเนินงานเหล่านั้น โดยปกติตัวแทนปัญญาได้รับการออกแบบมาเพื่อทำงานเฉพาะด้านและทำงานอย่างอัตโนมัติ ดังนั้นหากองค์กรมีงานหลายด้าน จึงจำเป็นต้องพัฒนาระบบตัวแทนปัญญาหลายตัว (Multi Agent System) เพื่อรองรับการทำงานที่หลากหลาย
- 4) Friendly ตัวแทนปัญญาจะต้องใช้งานง่ายและน่าเชื่อถือ
- 5) Able to learn ตัวแทนปัญญาสามารถเรียนรู้พฤติกรรมของผู้ใช้ เพื่อทำนายพฤติกรรมที่จะเกิดขึ้นต่อไปของผู้ใช้ได้โดยคุณสมบัตินี้ไม่จำเป็นต้องมีอยู่ในตัวแทนปัญญาทุกตัวด้วย
- 6) Communicative ตัวแทนปัญญาสามารถติดต่อสื่อสารกับตัวแทนปัญญาตัวอื่นเพื่อส่งคำสั่งหรือแลกเปลี่ยนข้อมูลระหว่างกันได้ (หากเป็นระบบ Multi Agent System ที่ต้องอาศัยการทำงานของตัวแทนปัญญาหลายตัวร่วมกัน) และตัวแทนปัญญาจะต้องสามารถโต้ตอบผู้ใช้ได้ด้วย
- 7) Cooperative ในระบบ Multi Agent System ตัวแทนปัญญาแต่ละตัว จะสามารถทำงานร่วมกับตัวแทนปัญญาตัวอื่นได้ เพื่อช่วยให้งานบรรลุล่งสำเร็จได้
- 8) Mobile ตัวแทนปัญญาสามารถเคลื่อนที่ข้ามเครือข่ายอินเทอร์เน็ตไปยัง Host ต่างๆ ที่ต้องการติดต่อสื่อสารได้
- 9) Persistent ตัวแทนปัญญาจะต้องทำงานได้อย่างต่อเนื่อง แม้ว่าจะต้องย้ายที่อยู่ไปยังระบบหรือเครือข่ายต่างชนิดกัน

10) Adaptive ตัวแทนปัญญาสามารถรองรับการเปลี่ยนแปลงของสภาพแวดล้อมในรูปแบบต่างๆ ได้ โดยจะต้องมีกระบวนการคิดตัดแปลงในการแก้ปัญหา กับแต่ละสถานการณ์ที่พบในขณะนั้น

เอเจนต์ในทางปัญญาประดิษฐ์จะต้องเป็น Rational Agent ซึ่งมีการรับข้อมูลผ่าน Sensor เข้ามาแล้วมีกลไกในการประมวลผลโดยใช้เหตุผลจากองค์ความรู้ที่มีอยู่ภายในเอเจนต์นำมาตัดสินใจเลือกผลการกระทำที่คาดว่าจะเหมาะสมที่สุดเพื่อให้ได้ผลลัพธ์ที่มีประสิทธิภาพ ดังนั้นในหัวข้อถัดไปจะกล่าวถึงหลักการสร้างหรือออกแบบเอเจนต์ โดยจำเป็นต้องมีความเข้าใจถึงเรื่องสภาพแวดล้อมของงาน เพื่อนำไปกำหนดขอบเขต (Scope) การทำงานให้กับตัวเอเจนต์ และนำไปวิเคราะห์คุณสมบัติสภาพแวดล้อมของงานว่าเอเจนต์ที่ถูกสร้างขึ้นนั้นประกอบไปด้วยข้อจำกัดใดบ้าง ก่อนนำไปออกแบบพัฒนาโครงสร้างภายในเอเจนต์ เพื่อรองรับงานที่ต้องการได้

2.1.2 ทำความรู้จักกับสภาพแวดล้อม

ก่อนสร้างหรือออกแบบเอเจนต์ จะต้องมีความเข้าใจถึงสภาพแวดล้อมให้ชัดเจน เพื่อให้เอเจนต์สามารถรับรู้ข้อมูลและตอบสนองได้ ดังนั้น อันดับแรกต้องมีการกำหนดสภาพแวดล้อมที่เกี่ยวข้องกับงานของเอเจนต์ (Task Environments) ซึ่งเป็นการกำหนดวัตถุประสงค์ ขอบเขตการทำงานของเอเจนต์ รวมถึงกำหนดสิ่งที่เอเจนต์จำเป็นต้องมี หลังจากกำหนดสภาพแวดล้อมของงานสำเร็จแล้ว ต่อมาจะต้องศึกษาคุณสมบัติสภาพของงาน (Properties of task environments) เพื่อทำความเข้าใจและนำมาวิเคราะห์เอเจนต์

2.1.2.1 การกำหนดสภาพแวดล้อมของงาน

การกำหนดสภาพแวดล้อมจะต้องเกี่ยวข้องกับงานของเอเจนต์ รวมทั้งตัวที่ทำหน้าที่รับข้อมูลหรือกระทำด้วย ดังนั้นสิ่งที่ต้องกำหนดจึงได้แก่ Performance, Environments, Actuators และ Sensors สภาพแวดล้อมดังกล่าวมาทั้งหมดรวมเรียกว่า “หลักการ PEAS (Performance, Environments, Actuators, Sensors)” มีความหมายดังนี้

- Performance** กำหนดสถานะของเอเจนต์ หรือกำหนดวัตถุประสงค์ความต้องการให้ชัดเจน เพื่อนำมาเป็นตัววัดสมรรถภาพการทำงานของเอเจนต์ได้
- Environments** กำหนดขอบเขตสภาพแวดล้อมที่เป็นไปได้ ซึ่งต้องมีความสอดคล้องและสนับสนุนการทำงานของเอเจนต์

Actuators	กำหนดส่วนประกอบของเอเจนต์ ที่สามารถกระทำการโต้ตอบกับสภาพแวดล้อมได้และกำหนดเฉพาะเท่าที่จำเป็น
Sensors	กำหนดส่วนของเอเจนต์ ที่สามารถรับข้อมูลกับสภาพแวดล้อมได้และกำหนดเท่าที่จำเป็น

2.1.2.2 คุณสมบัติของสภาพแวดล้อม

คุณสมบัติของสภาพแวดล้อมจะต้องสนับสนุนงานของเอเจนต์ ได้แก่ Fully Observable, Deterministic, Episodic, Static, Discrete และ Single Agent โดยคุณสมบัติล้วนเป็นคุณสมบัติเชิงอุดมคติที่เอเจนต์สามารถทำงานได้เป็นอย่างดี แต่เมื่อนำไปประยุกต์ใช้งานจริงคุณสมบัติของเอเจนต์อาจมีการเปลี่ยนแปลงได้ ทำให้ต้องนำคุณสมบัติต่างๆมาวิเคราะห์ เพื่อหาข้อจำกัดว่า คุณสมบัติของสภาพแวดล้อมรูปแบบใดที่เหมาะสมกับเอเจนต์นั้น

1. Fully Observable

เป็นคุณสมบัติของสภาพแวดล้อมที่สามารถเข้าถึงและมองเห็นได้ทั้งหมด ทำให้ตัว Sensors ของเอเจนต์สามารถรับรู้ข้อมูลหรือสถานะต่างๆจากสภาพแวดล้อมได้โดยง่าย เช่น เกมหมากรุก (Chess) ซึ่งเอเจนต์สามารถมองเห็นตำแหน่งของหมากรุกแต่ละตัวทั้งของฝ่ายเอเจนต์เองและฝ่ายตรงข้ามได้อย่างชัดเจน

แต่ในความเป็นจริงบางสภาพแวดล้อมของงานไม่สามารถเข้าถึงและมองเห็นได้ หรือที่เรียกว่า “Partially Observable” เช่น เกมไพ่โป๊กเกอร์ โดยเอเจนต์จะไม่สามารถทราบได้ว่าผู้เล่นฝั่งตรงข้ามถือไพ่อะไรอยู่บ้าง และปัญหาหุ่นยนต์เก็บของหากสถานการณ์ขณะนั้นมีเสียงดัง มีหมอกควัน จะทำให้ตัว Sensors ของเอเจนต์รับข้อมูลจากสภาพแวดล้อมไม่ชัดเจน เป็นต้น

2. Deterministic

เอเจนต์สามารถคาดเดาสถานการณ์ล่วงหน้าได้จากการทำงานของเอเจนต์เอง โดยพิจารณาสถานการณ์ปัจจุบัน เพื่อช่วยวิเคราะห์ว่าผลกระทบต่อสภาพแวดล้อมหลังจากที่มีการกระทำของเอเจนต์นั้นเป็นอย่างไร โดยเอเจนต์สามารถทำงานได้ตามเป้าหมาย หากสภาพแวดล้อมไม่มีความซับซ้อนและสามารถเข้าถึงได้ แต่หากกรณีที่สถานการณ์เป็นการแข่งขันที่มีผู้เล่นมากกว่าหนึ่งคนขึ้นไป จะเรียกคุณสมบัตินี้ว่า “Strategic” เช่น เกมหมากรุก และเกมไพ่โป๊กเกอร์ เป็นต้น

แต่ในความเป็นจริงสภาพแวดล้อมของงานบางประเภทอาจมีความซับซ้อนมาก ทำให้เอเจนต์ไม่สามารถคาดเดาสถานการณ์ล่วงหน้าได้ คุณสมบัตินี้เรียกว่า “Stochastic” เช่น ระบบขับแท็กซี่อัตโนมัติ เอเจนต์ไม่สามารถคาดการณ์เหตุการณ์ข้างหน้าว่าจะมีอะไรเกิดขึ้นได้ (เช่น คนข้ามถนน หรือรถติด เป็นต้น) ทำให้ยากต่อการวิเคราะห์คาดการณ์ล่วงหน้าได้

3. Episodic

ประสบการณ์ของเอเจนต์ที่ได้รับมาจากสภาพแวดล้อมนั้น ถูกแบ่งออกมาเป็นฉาก (Episode) ดังนั้นการกระทำของเอเจนต์จะตอบสนองโดยขึ้นอยู่กับฉากนั้นๆ ไม่ขึ้นอยู่กับฉากก่อนหน้า หรือกล่าวอย่างง่ายได้ว่า การกระทำจะเกิดขึ้นในช่วงเวลาหนึ่งและจบไป โดยไม่มีการกระทบหรือเชื่อมต่อไปอีก เช่น ระบบวิเคราะห์ภาพถ่ายดาวเทียม ซึ่งเอเจนต์จะรับข้อมูลภาพถ่ายจากดาวเทียมมาทีละฉาก เพื่อนำมาวิเคราะห์เป็นข้อมูลต่างๆ

แต่ในความเป็นจริงสภาพแวดล้อมของงานบางประเภทไม่สามารถแบ่งออกเป็นฉากได้ เนื่องจากเกิดผลกระทบต่อเนื่อง เช่น เกมหมากรุก การเดินหมาแต่ละครั้ง จะทำให้สถานการณ์ขณะนั้นเกิดผลกระทบได้ คุณสมบัติของสภาพแวดล้อมดังกล่าวเรียกว่า “Sequential”

4. Static

เอเจนต์จะพิจารณาปัญหาได้ดี เมื่ออยู่ในสภาพแวดล้อมที่ไม่มีการเปลี่ยนแปลง (Static) ทำให้ผลจากการกระทำของเอเจนต์เป็นไปตามเป้าหมาย เช่น เกมปริศนาอักษรไขว้ เนื่องจากสภาพแวดล้อมของเกมจะคงที่ตลอด ไม่มีการเปลี่ยนแปลง

หากกรณีที่สภาพแวดล้อมไม่มีการเปลี่ยนแปลง แต่มีเงื่อนไขมาเกี่ยวข้อง เช่น เวลาจะส่งผลกระทบต่อสมรรถนะการทำงานของเอเจนต์ได้ คุณสมบัติของสภาพแวดล้อมประเภทนี้เรียกว่า “Semidynamic” เช่น เกมปริศนาอักษรไขว้ที่มีเงื่อนไขการจับเวลา ซึ่งถ้าผู้เล่นใช้เวลาคิดเกินเวลาที่กำหนดจะถูกหักคะแนน หรือเกมหมากรุกที่มีการจับเวลา เป็นต้น

สภาพแวดล้อมของงานบางประเภท สามารถเกิดการเปลี่ยนแปลงได้หรือที่เรียกว่า “Dynamic” ทำให้เอเจนต์ต้องมีความรอบคอบในเรื่องเวลาและการสังเกตอยู่เสมอ เนื่องจากสภาพแวดล้อมนั้นเปลี่ยนแปลงตลอดเวลา เช่น สภาพแวดล้อมของ “ระบบขับแท็กซี่อัตโนมัติ” ในขณะที่แท็กซี่วิ่งอยู่นั้นสภาพแวดล้อมจะมีการเปลี่ยนแปลงอยู่เสมอ เป็นต้น

5. Discrete

สถานะของสภาพแวดล้อมที่มีคุณสมบัติเป็น Discrete (Discrete-State Environment) จะมีลักษณะที่เป็นอิสระจากกัน ไม่ต่อเนื่องกัน โดยเอเจนต์จะสามารถรับข้อมูลแต่ละสถานะของสภาพแวดล้อมอย่างชัดเจน เช่น เกมหมากรุก จะมีสถานะที่แตกต่างกันอย่างชัดเจน (ตัวหมากรุกเดินครั้งละหนึ่ง จะเท่ากับหนึ่งสถานะ)

แต่ในความเป็นจริงสถานะของสภาพแวดล้อมอาจมีคุณสมบัติต่อเนื่องหรือที่เรียกว่า “Continuous” (Continuous-State Environment) ซึ่งเอเจนต์ต้องรับรู้และโต้ตอบกับสถานะของสภาพแวดล้อมอย่างต่อเนื่องได้ จึงเป็นเรื่องยากสำหรับการจัดการกับสภาพแวดล้อมที่มีคุณสมบัติเช่นนี้ เช่น ระบบขับแท็กซี่อัตโนมัติ หากขณะที่รถกำลังแล่น เอเจนต์จะต้องรับข้อมูลสถานะจากสภาพแวดล้อมตลอดเวลา ขณะเดียวกันเอเจนต์จะต้องตอบสนองกับสภาพแวดล้อมตามความเหมาะสมด้วย จึงทำให้เอเจนต์ต้องมีการจัดการที่ดีเพื่อรับมือกับสภาพแวดล้อมดังกล่าว

6. Single Agent

เป็นสภาพแวดล้อมที่มีเอเจนต์ทำงานเพียงตัวเดียว และสามารถจัดการกับงานได้ด้วยตัวเอง ซึ่งเหมาะสมกับงานเฉพาะด้านทั่วไป เช่น สภาพแวดล้อมของเกมปริศนาอักษรไขว้ จะมีเอเจนต์หนึ่งตัวคอยจัดการแก้ปัญหาอักษรให้เสร็จสมบูรณ์โดยใช้เอเจนต์ตัวเดียวก็เพียงพอ

แต่ในความเป็นจริงสภาพแวดล้อมของงานบางงานอาจมีเอเจนต์ได้มากกว่าหนึ่งตัวได้ หรือที่เรียกกันว่า “Multi Agent” ซึ่งแบ่งวัตถุประสงค์ได้เป็น 2 ประเภท คือ Multi Agent สำหรับทำงานร่วมกันและ Multi Agent ที่ทำงานแข่งขันกัน

6.1 Multi Agent สำหรับทำงานร่วมกัน (Multi Agent for Cooperation) สภาพแวดล้อมของงานบางงานอาจมีความซับซ้อนมาก จึงจำเป็นต้องมีเอเจนต์มากกว่าหนึ่งตัว คอยทำงานร่วมกันเป็นระบบ เพื่อช่วยเพิ่มประสิทธิภาพการทำงานให้สูงขึ้น

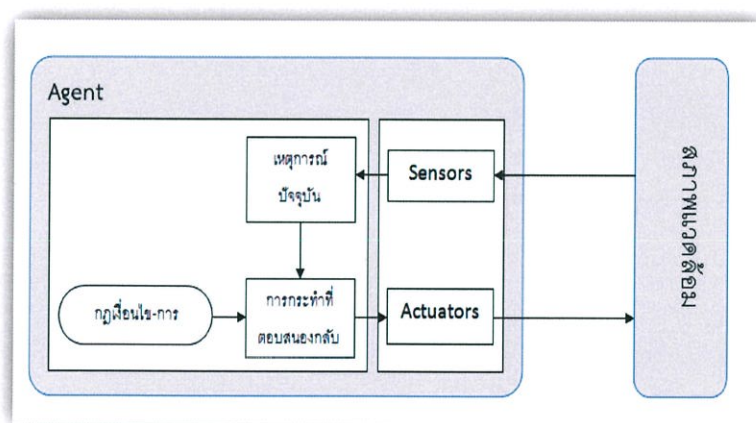
6.2 Multi Agent ที่ทำงานแข่งขันกัน (Multi Agent for Competition) หากมีเอเจนต์มากกว่าหนึ่งตัวแต่ทำหน้าที่เดียวกัน จะเหมาะสมกับงานประเภทแข่งขันระหว่างกัน เนื่องจากเอเจนต์แต่ละตัวต่างมีความคิดเป็นของตัวเอง

2.1.3 ประเภทของตัวแทนปัญญา

เอเจนต์แต่ละประเภท จะมีส่วนของ Mechanism for Interaction ที่เหมือนกัน ซึ่งประกอบด้วย Sensors ตอรับข้อมูลและ Actuators คอยตอบสนองกับสภาพแวดล้อมภายนอก แต่กระบวนการภายใน Agent Controller จะแตกต่างกันขึ้นอยู่กับเอเจนต์แต่ละประเภทว่าจะพิจารณาปัจจัยใดบ้าง เพื่อให้เอเจนต์ทำงานเป็นไปตามวัตถุประสงค์ที่ต้องการ ดังนั้นจึงแบ่งประเภทของเอเจนต์ตามโครงสร้าง ออกเป็น 5 ประเภท ได้แก่ Simple Reflex Agents, Model-based Reflex Agents, Goal-based Agents, Utility-based Agent และ Learning Agent

2.1.3.1 Simple Reflex Agents

Simple Reflex Agents เป็นประเภทของเอเจนต์ที่มีรูปแบบโครงสร้างอย่างง่าย หลักการทำงานของเอเจนต์ประเภทนี้ จะทำหน้าที่ตอบสนองกลับด้วยการกระทำจากข้อมูลที่ได้รับจาก Sensors ณ เวลาปัจจุบันเท่านั้น โดยไม่สนใจข้อมูลต่างๆที่ได้รับมาก่อนหน้านี้ ซึ่งกระบวนการดังกล่าวอาจเรียกอีกอย่างหนึ่งว่า “Mapping” โดยภายในโปรแกรมจะมีตัวควบคุมด้วยกฎเงื่อนไข-การกระทำ (Condition-Action Rules) โครงสร้างและการทำงานของ Simple Reflex Agents มีลักษณะดังรูปที่ 2.2

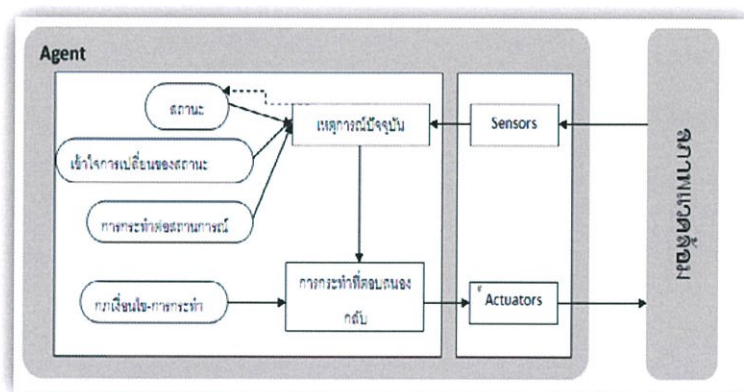


รูปที่ 2.2 โครงสร้างและการทำงานของ Simple Reflex Agents

ตัวอย่างการทำงานของเอเจนต์ประเภทนี้ เช่น เครื่องปรับอากาศที่มีลักษณะเป็น Simple Reflex Agents หากกรณีตัว Sensors ตรวจพบว่าขณะนี้อากาศร้อนเครื่องปรับอากาศก็จะทำงานหรือหากกรณีที่ตัว Sensors ตรวจพบว่าขณะนี้อากาศหนาวเครื่องปรับอากาศก็จะหยุดทำงาน เป็นต้น

2.1.3.2 Model-based Reflex Agents

Model-based Reflex Agents เป็นเอเจนต์ที่ถูกพัฒนามาจาก Simple Reflex Agents เพื่อเพิ่มประสิทธิภาพการทำงาน โดยเอเจนต์ประเภท Simple Reflex Agents จะอาศัยข้อมูลที่ได้รับมาจาก Sensors ในขณะเวลาหนึ่งๆเท่านั้น แต่ Model-based Reflex Agents จะคำนึงถึงสถานะของข้อมูลที่ได้รับมา จึงจำเป็นต้องกำหนดหน่วยความจำ

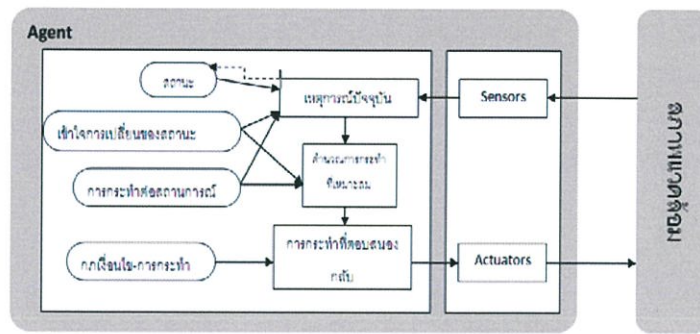


รูปที่ 2.3 โครงสร้างและการทำงานของ Model-based Reflex Agents

สำหรับตัวอย่างการทำงานของเครื่องปรับอากาศที่มีลักษณะเป็น Model-based Reflex Agents คือ หากกรณีตัว Sensors ตรวจพบว่าขณะนี้ห้องมีอุณหภูมิที่ห้องสูง เครื่องปรับอากาศก็จะปรับอุณหภูมิให้เหมาะสมกับห้องนั้นๆ

2.1.3.3 Goal-based Agent

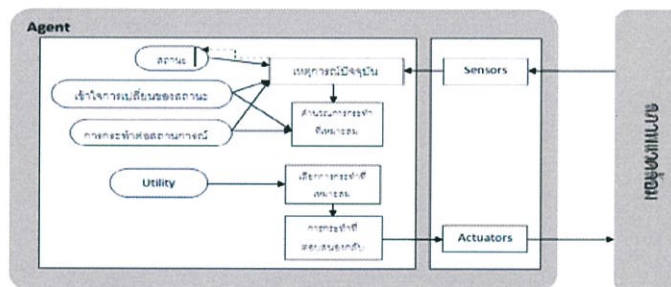
การทราบสถานะของสภาพแวดล้อมปัจจุบันตลอดเวลาเพียงอย่างเดียวนั้น ไม่เพียงพอต่อการตัดสินใจและทำให้ผลลัพธ์ที่ออกมาไม่ประสบความสำเร็จได้ ดังนั้นจึงได้กำหนดเป้าหมายให้ชัดเจนเพื่อแก้ปัญหาที่ตั้งกล่าวข้างต้นได้ โดยประเภทของเอเจนต์ที่ใช้แนวคิดดังกล่าวนี้เรียกว่า “Goal-based Agent” นอกจากนี้เอเจนต์ยังสามารถเข้าใจในสิ่งที่ตัวเอเจนต์กระทำเอง ว่าสามารถทำให้สภาพแวดล้อมเปลี่ยนแปลงอะไร และเป็นไปในแนวทางใดได้บ้าง ดังนั้นจึงต้องมีการคำนวณเมื่อมีการกระทำที่เปลี่ยนแปลงต่อสภาพแวดล้อมเพื่อให้ได้การกระทำที่เหมาะสม Goal-based Agent ถูกนำมาใช้ในงานปัญญาประดิษฐ์เป็นส่วนใหญ่ เนื่องจากเอเจนต์ประเภทนี้เริ่มมีความฉลาดมากกว่าสองประเภทที่ได้กล่าวไว้ก่อนหน้านี้ โครงสร้างและการทำงานของ Goal-based Agent มีลักษณะดังรูปที่ 2.4



รูปที่ 2.4 โครงสร้างและการทำงานของ Goal-based Agent

2.1.3.4 Utility-based Agent

Utility-based Agent เป็นการพัฒนาต่อมาจาก Goal-based Agent ซึ่งการกำหนดเป้าหมายเพียงอย่างเดียวจะทำให้ประสบความสำเร็จได้ แต่จะมีวิธีและขั้นตอนการเลือกการกระทำที่เหมาะสมที่สุด โดยมีฟังก์ชันของ Utility มาทำหน้าที่วิเคราะห์การกระทำแต่ละวิธีว่ามีประสิทธิภาพและเป็นไปตามเป้าหมายมากน้อยเพียงใด โดยพบว่าเอเจนต์ประเภทนี้มีคุณภาพสูงกว่าเอเจนต์ประเภทอื่นๆ ที่กล่าวมาก่อนหน้านี้ แต่เนื่องจากเทคโนโลยีในปัจจุบันไม่ได้หยุดนิ่งอยู่กับที่ ทำให้วิวัฒนาการของโครงสร้างเอเจนต์ สามารถพัฒนาไปได้อีกในอนาคต โครงสร้างและการทำงานของ Utility-based Agent มีลักษณะดังรูปที่ 2.5

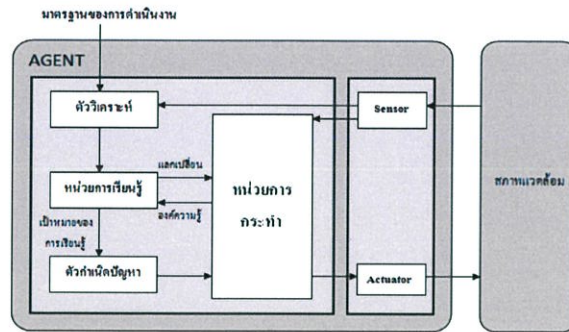


รูปที่ 2.5 โครงสร้างและการทำงานของ Utility-based Agent

2.1.3.5 Learning Agent

Learning Agent เป็นเอเจนต์ที่มีกระบวนการสร้างการเรียนรู้ (Learning) โดยมีตัววิเคราะห์เพื่อทำการวิเคราะห์ข้อมูลที่ได้รับมา และมีการนำมาตรฐานของการดำเนินงานที่อาจกำหนดโดยผู้เชี่ยวชาญเข้ามาช่วยพิจารณา แล้วนำข้อมูลที่ถูกระบุวิเคราะห์ป้อนลงในหน่วยการเรียนรู้ (Learning Elements) ซึ่งบางครั้งมีการแลกเปลี่ยนตอบสนองระหว่างกันกับหน่วยของการกระทำ (Performance Element) เพื่อนำไปพัฒนาผลลัพธ์ของการกระทำและสร้างองค์ความรู้ใหม่ให้กับหน่วยการเรียนรู้ของเอเจนต์ได้ หน่วยของการเรียนรู้สามารถส่งข้อมูลที่เป็นเป้าหมายของการเรียนรู้ (Learning Goals) ไปยังตัวกำเนิดปัญหา (Problem Generator) เพื่อทำหน้าที่ติดต่อกับหน่วยการ

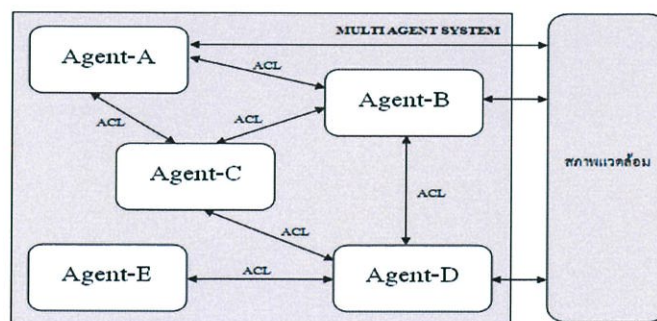
กระทำเพื่อให้ได้องค์ความรู้หรือประสบการณ์ใหม่ และช่วยให้ผลลัพธ์ของการกระทำตรงตามเป้าหมายการเรียนรู้ได้ โครงสร้างและการทำงานของ Learning Agent มีลักษณะดังรูปที่ 2.6



รูปที่ 2.6 โครงสร้างและการทำงานของ Learning Agent

2.1.4 Multi Agent System

ปัจจุบันเอเจนต์ไม่ได้ทำงานอยู่เพียงลำพัง แต่สามารถทำงานร่วมกันระหว่างเอเจนต์ได้แต่เพราะเอเจนต์ถูกออกแบบขึ้นมาเพื่อรองรับกับงานเฉพาะด้านเท่านั้น ดังนั้น บางองค์กรที่มีงานเฉพาะด้านที่หลากหลาย จึงจำเป็นต้องออกแบบให้เอเจนต์หลายตัวมาทำงานร่วมกันเป็นระบบ เพื่อรองรับงานเฉพาะด้านที่มีภายในองค์กรนั้นได้ และช่วยเพิ่มประสิทธิภาพการทำงานให้สูงขึ้น ระบบการทำงานร่วมกันระหว่างเอเจนต์หลายตัว จะเรียกว่า “Multi Agent System (MAS)” แต่ละเอเจนต์ภายในระบบ MAS จะทำงานตามหน้าที่ในส่วนรับผิดชอบอย่างอิสระและไม่ได้ถูกควบคุมจากศูนย์กลาง บางครั้งงานของเอเจนต์ที่เสร็จแล้วสามารถส่งมอบงานไปยังเอเจนต์อื่นที่เกี่ยวข้องให้นำไปทำงานต่อได้ นอกจากสามารถทำงานร่วมกันได้แล้ว เอเจนต์ยังสามารถโต้ตอบสื่อสารระหว่างกันได้ โดยใช้ภาษาสื่อสารของเอเจนต์โดยเฉพาะ หรือที่เรียกว่า “Agent Communication Language (ACL)” เป็นภาษามาตรฐานที่ใช้สื่อสาร และแลกเปลี่ยนข้อมูลเพื่อป้องกันการแย่งกันทำงานของเอเจนต์ที่ทำงานอย่างอิสระ เป็นต้น ระบบ MAS สามารถรับข้อมูลและตอบสนองกับสภาพแวดล้อมภายนอกได้เช่นกัน



รูปที่ 2.7 ลักษณะโครงสร้างการทำงานภายในระบบ MAS

2.2 การแทนความรู้ (Knowledge Representation)

คอมพิวเตอร์สามารถเก็บความรู้และคิดหาเหตุผลได้ ทำให้เราสามารถสร้างโปรแกรมประยุกต์ที่มีทักษะกระบวนการคิดและหาข้อสรุปได้ หรือที่เรียกว่าการอนุมาน (Inference) โปรแกรมประยุกต์พวกนี้เรียกว่า ระบบผู้เชี่ยวชาญ (Expert System) ซึ่งนำมาใช้ในการจำแนกข้อมูล เช่น โปรแกรมจำแนกประเภทเว็บเพจอย่างอัตโนมัติสำหรับ Search Engine ซอฟต์แวร์การจำแนกประเภทภาพถ่ายดาวเทียมเพื่อตรวจจับการรั่วไหลของน้ำมัน เป็นต้น

ดังนั้น การสร้างระบบผู้เชี่ยวชาญจำเป็นที่จะต้องมีความรู้ในเรื่องโครงสร้างข้อมูลที่ใช้ในการเก็บความรู้ วิธีการดึงความรู้จากฐานความรู้และวิธีการอนุมานตามหลักเหตุผล ตามลำดับ

2.2.1 ชนิดของความรู้

การสร้างระบบผู้เชี่ยวชาญที่ดีนั้น เราควรทำความรู้จักกับชนิดความรู้ในมุมมองของมนุษย์ ซึ่งความรู้ที่สะสมในมนุษย์แต่ละคน แบ่งได้ 3 ชนิด คือ

2.2.1.1 ความรู้ที่สะสมมาตั้งแต่ในอดีต (Long-term Knowledge)

ความรู้ชนิดนี้เกิดจากการเรียนรู้ของมนุษย์ตั้งแต่แรกเกิดที่อาจได้รับโดยตรงจากการศึกษาเล่าเรียนและจากประสบการณ์ชีวิต ดังนั้น การสร้างโปรแกรมประยุกต์จึงจำเป็นต้องมีเนื้อหาที่รองรับเพื่อเก็บความรู้ในรูปแบบนี้ เรียกว่า ฐานความรู้ (Knowledge Base)

2.2.1.2 ความรู้ที่เกิดทันทีจากการรับรู้ (Short-term Knowledge)

ความรู้ในลักษณะนี้ เกิดขึ้นจากการรับรู้หรือรู้จำ ซึ่งเกิดเมื่อมนุษย์รับข้อมูลจากประสาทสัมผัสทั้งห้าแล้ว สามารถรู้ว่าสิ่งที่ได้รับรู้คืออะไร เช่น เมื่อเรามองเห็นภาพใบหน้าของคนที่เรา รู้จัก สมองจะประมวลผลแล้วรู้ว่าบุคคลในภาพนั้นคือผู้ใด

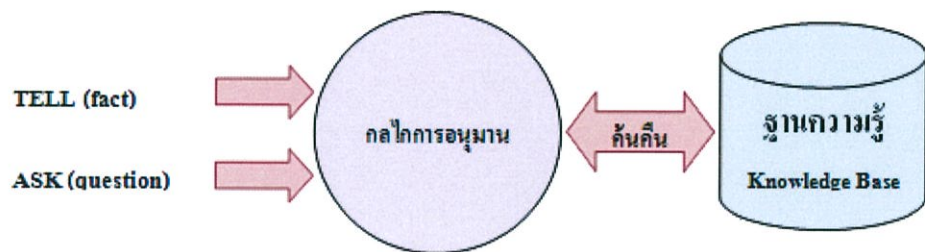
2.2.1.3 ความรู้ใหม่ที่เกิดจากการใช้สติปัญญาของมนุษย์ (Inference Knowledge)

ความรู้ใหม่เกิดจากกลไกการอนุมาน โดยมีกระบวนการคิดวิเคราะห์จากความรู้ในฐานความรู้ และมนุษย์ใช้กลไกดังกล่าวเมื่อต้องเผชิญกับสถานการณ์ใหม่ๆ

2.2.2 ระบบการแทนความรู้

การสร้างโปรแกรมประยุกต์ที่มีความสามารถในการคิดสรุปหาเหตุผลได้เหมือนมนุษย์ ต้องจัดเตรียมระบบที่มีความสามารถดังนี้

- มีเนื้อหาที่รองรับการเก็บความรู้
- มีฟังก์ชันการค้นคืนความรู้จากฐานความรู้
- มีฟังก์ชันสำหรับการป้อนความรู้ใหม่
- มีกลไกการคิดหาเหตุผล
- มีฟังก์ชันการนำปัญหาไปถามระบบ



รูปที่ 2.8 องค์ประกอบหลักของระบบการแทนความรู้

ฟังก์ชัน TELL (fact) หมายถึง กระบวนการเพิ่มความรู้ในรูปข้อเท็จจริงให้แก่ระบบ โดยที่ระบบอาจสร้างส่วนต่อประสานสำหรับผู้ใช้ (User Interface) ที่กำหนดรูปแบบการนำข้อเท็จจริงเข้าสู่ระบบ เช่น ให้ผู้ใช้ใส่ข้อมูลลงในช่องว่างที่ระบบกำหนดไว้ หลังจากนั้นระบบจะทำการแปลงข้อมูลที่ผู้ใช้ใส่เข้าไป โดยแปลงให้สอดคล้องกับรูปแบบการแทนความรู้ที่ระบบนั้นใช้

ความรู้ในรูปของข้อเท็จจริงที่ผู้ใช้ใส่เข้าสู่ระบบจะถูกนำไปตรวจสอบในฐานความรู้เพื่อตรวจสอบว่ามีข้อเท็จจริงดังกล่าวอยู่ในนั้นแล้วหรือไม่ ถ้ายังไม่พบข้อเท็จจริงดังกล่าวกลไกการอนุมานจะทำการคิดหาเหตุผลเพื่อให้ได้ข้อสรุปใหม่ หรือความรู้ใหม่โดยใช้ขั้นตอนวิธี Forward Chaining สำหรับฟังก์ชัน ASK (question) จะถูกเรียกใช้เมื่อผู้ใช้ใส่คำถามเข้าสู่ระบบโดยคำถามอาจเกิดจากผู้ใช้งานที่ต้องการคำตอบจากระบบทำให้เกิดกลไกการอนุมานแบบ Backward Chaining เพื่อหาคำตอบให้กับผู้ใช้

2.3 การเรียนรู้ของเครื่อง (Machine Learning)

การเรียนรู้ของเครื่อง (Machine Learning) คือศาสตร์หนึ่งของสาขาปัญญาประดิษฐ์ เนื่องจากเป็นเรื่องของการพยายามทำให้เครื่องคอมพิวเตอร์หรือเครื่องจักร มีความสามารถในการทำงานที่ใช้ความฉลาด ด้วยการสร้างความสามารถในการเรียนรู้จากข้อมูลเข้า (Input Data) ซึ่งเป็นข้อมูลที่ผู้ใช้จัดเตรียมให้กับเครื่องเหล่านั้น เช่น วิทยุติดรถยนต์บางรุ่นสามารถรู้จำภาษาพูดของมนุษย์ (Speech Recognition) ผู้ขับรถยนต์สามารถสั่งงานด้วยเสียงเพื่อให้วิทยุปรับจูนคลื่นวิทยุไปยังสถานีต่างๆ ได้อย่างอัตโนมัติ การเรียนรู้ของวิทยุเกิดขึ้นในตอนใช้งานครั้งแรก โดยผู้ใช้จะต้องทำการฝึกให้โปรแกรมที่ติดตั้งในวิทยุ ให้รู้จำเสียงพูดโดยการสั่งงานด้วยคำสั่งเดิมซ้ำกันหลายๆ ครั้ง เพื่อให้วิทยุติดรถยนต์รู้จำเสียงของผู้ขับ

ตัวอย่างที่เห็นได้ทั่วไปอีกกรณี คือกล้องถ่ายรูปที่สามารถตรวจจับภาพใบหน้ามนุษย์ได้ (Face Detection) เพื่อทำการปรับความคมชัดที่เหมาะสม ซึ่งก่อนที่จะผลิตกล้องถ่ายรูปที่ตรวจจับภาพใบหน้าคนได้จะต้องผ่านการวิจัยค้นคว้าในห้องปฏิบัติการ โดยใช้ขั้นตอนวิธีการเรียนรู้ของเครื่องที่สามารถเรียนรู้ข้อมูลรูปภาพได้เป็นอย่างดี

การคิดค้นโปรแกรมที่สามารถเรียนรู้ได้อย่างได้ผลนั้นจำเป็นจะต้องมีการทดลองบนชุดข้อมูลหลายชุดจนมั่นใจในประสิทธิภาพในแง่ความแม่นยำของการประมวลผล ซึ่งกระบวนการทดลองเพื่อปรับปรุงประสิทธิภาพของขั้นตอนวิธีการเรียนรู้จะต้องใช้เวลานานในห้องปฏิบัติการวิจัย จนกว่าจะได้ผลเป็นที่น่าพอใจแล้วจึงผลิตในเชิงการค้าได้

ประเด็นถัดมาคือเมื่อนำซอฟต์แวร์ที่มีกลไกการเรียนรู้ไปใช้งานแล้วผู้ใช้งานจำเป็นต้องเข้าใจในข้อเท็จจริงที่ว่า ซอฟต์แวร์จำพวกนี้อาจทำงานได้ไม่ถูกต้องร้อยเปอร์เซ็นต์ เช่น ซอฟต์แวร์รู้จำตัวอักษร หรือ โอซีอาร์ (Optical Character Recognition) ซึ่งใช้กับเครื่องสแกนเนอร์เพื่อแปลงข้อความในเอกสารให้เป็นเท็กซ์ไฟล์ เพื่อให้ผู้ใช้สามารถเปลี่ยนแปลง แก้ไขเอกสารต้นฉบับได้ ในความเป็นจริงนั้นพบว่า ซอฟต์แวร์ดังกล่าวอาจทำการแปลงตัวอักษรได้ไม่ถูกต้องทุกตัว ซึ่งอาจเกิดจากปัญหาความคมชัดของเอกสารต้นฉบับ เช่น ตัวอักษร “ก” อาจถูกแปลงเป็น “ท” เป็นต้น ซึ่งผู้ใช้จะต้องทำการตรวจสอบข้อผิดพลาดแล้วทำการแก้ไขให้ถูกต้องด้วยตนเอง เช่นเดียวกับวิทยุติดรถยนต์ที่สั่งงานด้วยเสียง ในบางครั้งผู้ใช้อาจพบว่า เมื่อสั่งงานด้วยเสียงพูดแล้ววิทยุกลับทำงานผิดพลาด อย่างไรก็ตามความผิดพลาดของซอฟต์แวร์ที่ผลิตในเชิงการค้าจะต้องมีความผิดพลาดที่น้อยที่สุดเท่าที่จะทำได้ ซึ่งเป็นสาเหตุให้ซอฟต์แวร์ประเภทนี้มีไม่มากนักในท้องตลาด เนื่องจากยังต้องการค้นคว้าวิจัยในห้องปฏิบัติการอีกมาก

2.3.1 สถาปัตยกรรมการเรียนรู้ของเครื่อง

โปรแกรมที่มีความสามารถในการเรียนรู้ มีลักษณะหรือรูปแบบในการเรียนรู้อย่างไรนั้น สามารถเปรียบเทียบได้จากการเรียนรู้ของมนุษย์ดังต่อไปนี้

- มนุษย์สามารถเรียนรู้ได้จากคำสอนของครูหรือผู้รู้ (Supervised Learning)
- มนุษย์สามารถเรียนรู้ได้ด้วยตนเองจากประสบการณ์ในอดีต (Learning from Experience)
- มนุษย์สามารถเรียนรู้ได้จากการจดจำ (Rote Learning)
- มนุษย์สามารถเรียนรู้ได้จากการได้รับคำสั่งและการฝึกฝน (Learning by Doing)

ดังนั้น เราสามารถสร้างขั้นตอนวิธีที่มีความสามารถในการเรียนรู้ โดยอาศัยหลักการข้างต้น ได้เช่นเดียวกัน กล่าวคือ โปรแกรมจะรับข้อมูลเข้าที่มีจำนวนที่มากพอที่จะทำให้โปรแกรมสามารถสร้างแนวคิด (Concept) หรือสร้างรูปแบบ (Pattern) ของชุดข้อมูลนั้น เราเรียกชุดข้อมูลที่ใช้สำหรับการเรียนรู้ว่า ข้อมูลฝึก (Training Data) ซึ่งผู้พัฒนาโปรแกรมมีหน้าที่ในการนำเข้าข้อมูลฝึกสู่โปรแกรม

ขั้นตอนถัดมาคือการประเมินคุณภาพของแนวคิดที่ได้จากการเรียนรู้ว่าสามารถนำไปใช้งานจริงได้หรือไม่ โดยการนำข้อมูลอีกชุดมาทำการทดสอบ (Test Data) ซึ่งเป็นชุดข้อมูลคนละชุดกับข้อมูลฝึกเพื่อทำการทดสอบประสิทธิภาพว่าสามารถประมวลผลข้อมูลทดสอบได้ถูกต้องกี่เปอร์เซ็นต์

สาเหตุที่เราสามารถประเมินประสิทธิภาพในแง่ความถูกต้องได้ เนื่องจากชุดข้อมูลทดสอบนี้มีผลเฉลย (Labeled Data) สำหรับข้อมูลทุกเรคคอร์ด ซึ่งค่าเฉลยนี้ได้มาจากผู้เชี่ยวชาญในข้อมูลนั้นเป็นผู้จัดเตรียมไว้ล่วงหน้า เช่น ถ้าเป็นข้อมูลอาการป่วยของคนไข้ ผู้ให้ค่าเฉลยคือแพทย์ผู้เชี่ยวชาญ หากเป็นข้อมูลพันธุ์พืช ผู้ให้ค่าเฉลยคือนักพฤกษศาสตร์ เป็นต้น

2.3.2 ขั้นตอนวิธีการเรียนรู้แบบมีผู้สอน (Supervised Learning)

การเรียนรู้ประเภทนี้เปรียบเทียบกับกับการเรียนรู้แบบมีครูฝึกที่ทำหน้าที่ให้ค่าเฉลย ดังนั้นจะต้องจัดเตรียมข้อมูลฝึกให้มีผลเฉลยสำหรับขั้นตอนวิธีเพื่อใช้ในการค้นหาและสร้างแนวคิดของข้อมูลฝึกแต่ละประเภท (Class)

ขั้นตอนวิธีการเรียนรู้แบบมีผู้สอนนี้จะให้ประสิทธิภาพสูงในแง่ความถูกต้องในการจำแนกข้อมูลแต่ละประเภท เมื่อการเรียนรู้บนชุดข้อมูลฝึกเสร็จสิ้นลง ผู้พัฒนาโปรแกรมจะต้องทำการ

ทดสอบเพื่อวัดประสิทธิภาพของความรู้หรือรูปแบบที่เป็นผลลัพธ์จากการเรียนรู้ โดยให้ขั้นตอนวิธีการจำแนกข้อมูลบนข้อมูลทดสอบ เมื่อได้ประสิทธิภาพที่น่าพอใจแล้วจึงนำไปใช้งานจริง ข้อจำกัดของการเลือกให้ขั้นตอนวิธีการเรียนรู้แบบมีผลเฉลยคือ ชุดข้อมูลฝึกบางชุดไม่มีผลเฉลย เนื่องจากสาเหตุต่างๆดังนี้

- ข้อมูลฝึกมีปริมาณมากแต่การให้คำเฉลยสำหรับแต่ละตัวอย่างต้องใช้เวลาานาน เช่น ปัญหาการจำแนกประเภทของเว็บเพจ พบว่ามีจำนวนเว็บเพจนับล้านหน้า แต่การให้คำเฉลยต้องใช้เวลาานานในการอ่านข้อความในเว็บเพจก่อนจะทำการระบุประเภทให้แก่เว็บเพจหน้านั้น
- ข้อมูลฝึกบางชุดมีปริมาณมากแต่ขาดผู้เชี่ยวชาญที่จะมาให้ผลเฉลยได้ เช่น ระบบตรวจจับการรั่วของน้ำมันในมหาสมุทร ต้องใช้ดาวเทียมถ่ายภาพโดยใช้สัญญาณเรดาร์ ซึ่งภาพถ่ายมีจำนวนมากและการวิเคราะห์ว่าภาพถ่ายดังกล่าวนั้นเป็นรอยปนเปื้อนของน้ำมันหรือเป็นเพียงมฆวตอากาศนั้นทำได้ยากมาก

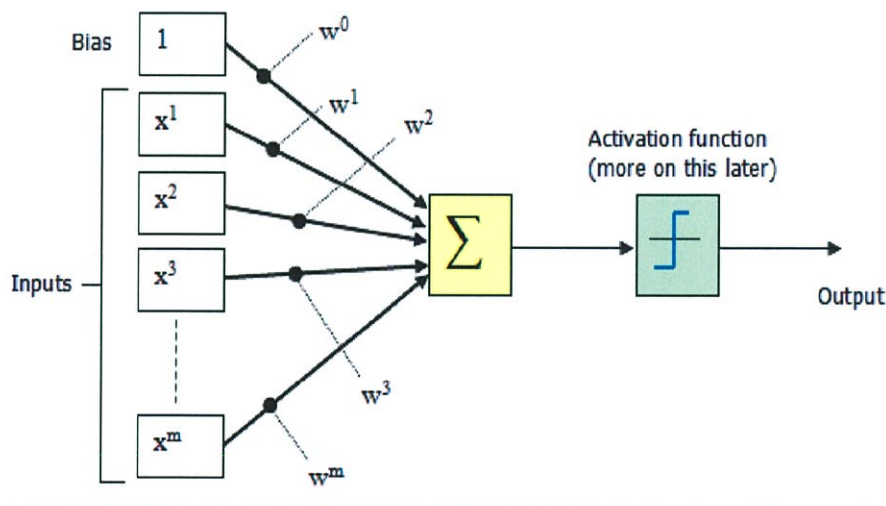
ขั้นตอนการเรียนรู้ที่จัดอยู่ในกลุ่มการเรียนรู้แบบมีผู้สอน ได้แก่การเรียนรู้ด้วยต้นไม้ตัดสินใจ (Decision Tree) การเรียนรู้แบบเบย์ส (Bayesian Learning) การเรียนรู้ด้วยโครงข่ายประสาทเทียม (Neural Networks)

2.3.2.1 โครงข่ายประสาทเทียม (Neural Network)

โครงข่ายประสาทเทียม (neural network) หรือที่มักจะเรียกสั้นๆว่าโครงข่ายประสาท (neural network หรือ neural net) คือโมเดลทางคณิตศาสตร์สำหรับประมวลผลสารสนเทศด้วยการคำนวณแบบคอนเนกชันนิสต์ (connectionist) เพื่อจำลองการทำงานของโครงข่ายประสาทในสมองมนุษย์ด้วยวัตถุประสงคที่จะสร้างเครื่องมือซึ่งมีความสามารถในการเรียนรู้การจดจำแบบรูป (Pattern Recognition) และการอุปมาความรู้เช่นเดียวกับความสามารถที่มีในสมองมนุษย์แนวคิดเริ่มต้นของเทคนิคนี้ได้มาจากการศึกษาข่ายงานไฟฟ้าชีวภาพ (bioelectric network) ในสมองซึ่งประกอบด้วยเซลล์ประสาทหรือ “นิวรอน” (neurons) และจุดประสานประสาท (synapses) แต่ละเซลล์ประสาทประกอบด้วยปลายในการรับกระแสประสาทเรียกว่า “เดนไดรท์” (Dendrite) ซึ่งเป็นอินพุตและปลายในการส่งกระแสประสาทเรียกว่า “แอกซอน” (Axon) ซึ่งเป็นเหมือนเอาท์พุทของเซลล์ เซลล์เหล่านี้ทำงานด้วยปฏิกิริยาไฟฟ้าเคมีเมื่อมีการกระตุ้นด้วยสิ่งเร้าภายนอกหรือกระตุ้นด้วยเซลล์ด้วยกันกระแสประสาทจะวิ่งผ่านเดนไดรท์เข้าสู่นิวเคลียสซึ่งจะเป็นตัวตัดสินใจว่าต้องกระตุ้น

เซลล์อื่นๆต่อหรือไม่ถ้ากระแสประสาทแรงพอนิวเคลียสก็จะกระตุ้นเซลล์อื่นๆต่อไปผ่านทางแอกซอน

โครงสร้างของโครงข่ายประสาทเทียมมีโครงสร้างแตกต่างจากโครงข่ายในสมองแต่ก็ยิ่งเหมือนสมองในแง่ที่ว่าโครงข่ายประสาทเทียมคือการรวมกลุ่มแบบขนานของหน่วยประมวลผลย่อยๆและการเชื่อมต่อนี้เป็นส่วนสำคัญที่ทำให้เกิดสติปัญญาของโครงข่ายเมื่อพิจารณาขนาดแล้วสมองมีขนาดใหญ่กว่าโครงข่ายประสาทเทียมอย่างมากรวมทั้งเซลล์ประสาทยังมีความซับซ้อนกว่าหน่วยย่อยของโครงข่ายอย่างไรก็ดีหน้าที่สำคัญของสมองเช่นการเรียนรู้ยังคงสามารถถูกจำลองขึ้นอย่างง่ายด้วยโครงข่ายประสาทนี้สำหรับในคอมพิวเตอร์นิวรอนประกอบด้วยอินพุตและเอาต์พุตเหมือนกัน โดยจำลองให้อินพุตแต่ละอันมีค่าน้ำหนัก(weight) เป็นตัวกำหนดน้ำหนักของอินพุต โดยโครงข่ายแต่ละหน่วยจะมีค่า threshold เป็นตัวกำหนดว่าน้ำหนักรวมของอินพุตว่าต้องมีมากเพียงใดจึงจะสามารถส่งเอาต์พุตไปยังโครงข่ายตัวอื่นได้เมื่อนำโครงข่ายแต่ละหน่วยมาต่อกันให้ทำงานร่วมกันการทำงานนี้ในทางตรรกะแล้วจะเหมือนกับปฏิกิริยาเคมีที่เกิดในสมองเพียงแต่ในคอมพิวเตอร์ทุกอย่างเป็นตัวเลข

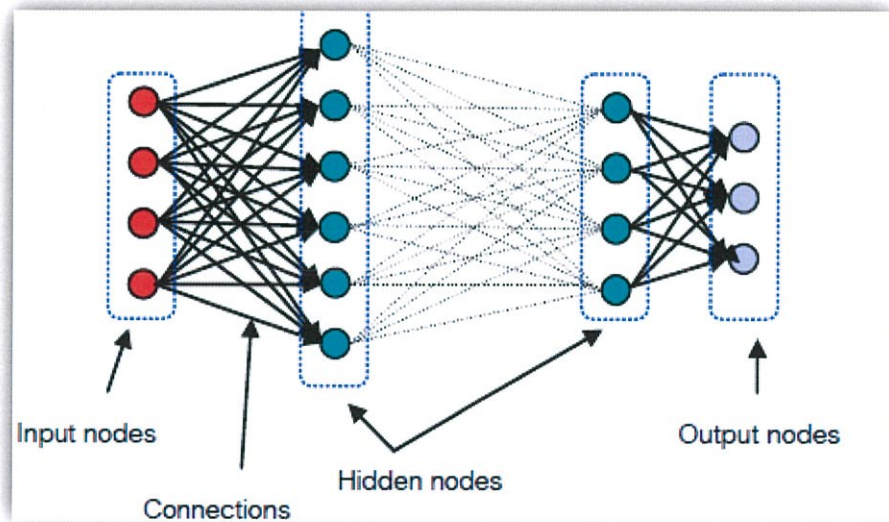


รูปที่ 2.9 โครงสร้างของโครงข่ายประสาทเทียม

โครงข่ายประสาทเทียมแบบหลายชั้น (MLP)

โครงข่ายประสาทเทียมแบบหลายชั้น (MLP) สามารถแก้ปัญหาที่ซับซ้อนมากๆ ได้ดีกว่าโครงข่ายประสาทเทียมแบบชั้นเดียว แต่การฝึกสอนอาจจะยากกว่า ซึ่งโครงข่ายประสาทเทียมแบบหลายชั้นที่มีหนึ่งชั้นซ่อนหรือมากกว่านั้น การส่งข้อมูลระหว่างอินพุตและเอาต์พุต จะถูกส่งไปในทิศทางเดียวคือ จากอินพุตไปเอาต์พุต ไม่มีการส่งข้อมูลจากเอาต์พุตกลับไปยังอินพุต แต่มีการ

เรียนรู้รูปแบบหนึ่งที่มีการส่งข้อมูลจากเอาต์พุตกลับไปยังอินพุตเรียกว่า การเรียนรู้แบบแพร่ย้อนกลับ (Backpropagation Learning) เป็นการเรียนรู้แบบควบคุม คือ จะมีเซตของข้อมูลอินพุตและเอาต์พุตเป้าหมาย โดยทำการฝึกสอนให้กับโครงข่ายประสาทเทียมรอบแรก และเปรียบเทียบเอาต์พุตที่ได้จากระบบกับเอาต์พุตเป้าหมาย ค่าที่ได้ก็คือ ค่าผิดพลาดของชั้นเอาต์พุต หลังจากนั้นส่งค่าผิดพลาดของชั้นเอาต์พุตนี้กลับไปเพื่อใช้ในการคำนวณค่าผิดพลาดของชั้นซ่อนและนำค่าผิดพลาดที่ได้จากการคำนวณทั้งหมดไปปรับค่าถ่วงน้ำหนัก(Update Weight) ในทุกๆชั้นของโครงข่าย



รูปที่ 2.10 โครงสร้างของ MLP

ขั้นตอนการเรียนรู้ของโครงข่ายประสาทเทียม

- ขั้นตอนที่ 1 ออกแบบโครงสร้างของระบบทำการออกแบบระบบ โดยต้องกำหนดจำนวนชั้นฟังก์ชันกระตุ้นและเงื่อนไขการหยุดฝึกสอน
- ขั้นตอนที่ 2 กำหนดค่าถ่วงน้ำหนัก (w) และไบแอส (b) เริ่มต้นโดยทำการสุ่มค่าถ่วงน้ำหนักในการเชื่อมต่อและค่าไบแอสทั้งหมดในโครงข่ายโดยกำหนดให้มีค่าน้อยๆ
- ขั้นตอนที่ 3 การฝึกสอน เป็นขั้นตอนการฝึกสอนเพื่อให้โครงข่ายประสาทเทียมได้เรียนรู้ ซึ่งประกอบด้วย 3 ส่วนหลัก คือ ขั้นตอน Feedforward (ขั้นตอนที่ 4 ถึง 7) ขั้นตอน Backpropagation of Error (ขั้นตอนที่ 8 ถึง 9) และสุดท้ายเป็นขั้นตอนการปรับปรุ้ค่าถ่วงน้ำหนักและไบแอส (ขั้นตอนที่ 10)
- ขั้นตอนที่ 4 ป้อนคู่ของอินพุตและเอาต์พุตป้อนคู่ของการฝึกสอน คือ อินพุต (p) และเอาต์พุตเป้าหมาย (t) เข้าสู่โครงข่ายประสาทเทียม

- ขั้นตอนที่ 5 รับสัญญาณอินพุต เป็นการรับสัญญาณอินพุตเข้ามาในแต่ละโหนดของอินพุต และแพร่กระจายไปสู่ทุกๆ โหนดในชั้นถัดไป
- ขั้นตอนที่ 6 คำนวณผลรวมค่าถ่วงน้ำหนักและอินพุตที่รับเข้ามาในทุกโหนดของโครงข่ายเป็นการคำนวณหาผลรวมของสัญญาณอินพุตที่ป้อนเข้ามาตามเส้นทางของค่าถ่วงน้ำหนักต่างๆ โดย s_m คือค่าผลรวมของชั้นซ่อน ดังสมการที่ 2.1

$$s_m = \sum_{n=1}^N x_n W_{nm} + b_m \quad (2.1)$$

- โดยที่ x_n คือ สัญญาณอินพุต
- W_{nm} คือ ค่าถ่วงน้ำหนักประจำตัวของแต่ละโหนดในชั้นซ่อน
- b_m คือ ค่าไบแอสประจำตัวของแต่ละโหนดในชั้นซ่อน

จากนั้นทำการปรับสัญญาณที่ออกจากชั้นนี้ด้วยฟังก์ชันกระตุ้นแบบซิกมอยด์ (Sigmoid function) ดังสมการที่ 2.2 แล้วทำการส่งต่อไปยังชั้นต่อไป

$$f(s_m) = \frac{1}{1 + e^{(-s_m)}} \quad (2.2)$$

ทำการคำนวณตามขั้นตอนที่ 6 ให้กับแต่ละโหนดในทุกชั้นซ่อนของโครงข่ายประสาทเทียม

- ขั้นตอนที่ 7 คำนวณผลรวมค่าถ่วงน้ำหนักและอินพุตที่รับเข้ามาในโหนดเอาต์พุต เป็นการคำนวณหาผลรวมของสัญญาณอินพุตที่ป้อนเข้ามาตามเส้นทางของค่าถ่วงน้ำหนักต่างๆ โดย v_j คือค่าผลรวมของชั้นเอาต์พุต สมการ 2.3 ดังนี้

$$v_j = \sum_{m=1}^M y_m W_{mj} + b_j \quad (2.3)$$

- โดยที่ y_m คือ สัญญาณเอาต์พุตจากชั้นที่สอง ซึ่งเป็นอินพุตของชั้นเอาต์พุต
- W_{mj} คือ ค่าถ่วงน้ำหนักประจำตัวของแต่ละโหนดในชั้นเอาต์พุต
- b_j คือ ค่าไบแอสประจำตัวของแต่ละโหนดในชั้นเอาต์พุต

จากนั้นทำการปรับสัญญาณที่ออกจากชั้นนี้ด้วยฟังก์ชันกระตุ้นดังสมการที่ 2.4 ซึ่งเป็นเอาต์พุตของโครงข่าย

$$z_j = f(v_j) \quad (2.4)$$

ขั้นตอนที่ 8 คำนวณค่าผิดพลาด เป็นการหาค่าความคลาดเคลื่อนระหว่างเอาต์พุตที่ได้จริงกับเอาต์พุตเป้าหมาย

$$e_j = z_j(1 - z_j)(t_j - z_j) \quad (2.5)$$

โดยที่ t_j คือ เอาต์พุตเป้าหมาย

ขั้นตอนที่ 9 หาคอนุพันธ์ของฟังก์ชันในทุกชั้น

ขั้นตอนที่ 10 ปรับปรุงค่าถ่วงน้ำหนักและค่าไบแอส ทำการปรับปรุงค่าถ่วงน้ำหนักในทุกๆชั้นของโครงข่ายดังสมการที่ 2.6

$$w_{mj} = w_{mj} + \eta e_j z_j \quad (2.6)$$

และปรับปรุงค่าไบแอสในทุกๆชั้น ดังสมการที่ 2.7

$$b_{mj} = b_{mj} + \eta e_j \quad (2.7)$$

โดยที่ η คือ อัตราการเรียนรู้

ขั้นตอนที่ 11 วนไปทำขั้นตอนที่ 4 ถึงขั้นตอนที่ 9 จนข้อมูลถูกฝึกสอนให้กับโครงข่ายประสาทเทียมครบทุกชุด

ขั้นตอนที่ 12 การหยุดฝึกสอนโครงข่ายประสาทเทียมซึ่งมีเงื่อนไขดังนี้

1. ค่าความผิดพลาด โดยทำการเปรียบเทียบค่าผิดพลาด (E) ที่คำนวณได้จากขั้นตอนที่ 8 กับค่าความผิดพลาดสูงสุด (E_k) ที่ได้กำหนดไว้ ถ้าค่าความผิดพลาดที่ได้มากกว่าค่าความผิดพลาดสูงสุด ($E > E_k$) จะต้องกลับไปทำขั้นตอนที่ 4 และเริ่มจากข้อมูลชุดแรกใหม่ จนกระทั่งค่าความผิดพลาดที่ได้น้อยกว่าค่าความผิดพลาดสูงสุด ($E < E_k$) จึงหยุดทำการฝึกสอนให้กับโครงข่าย คำนวณหาค่าความผิดพลาด ดังสมการที่ 2.8

$$MSE = \frac{\sum_{i=1}^I \sum_{j=1}^J (t_{ij} - z_{ij})^2}{ij} \quad (2.8)$$

2. จำนวนรอบของการฝึกสอน หยุดทำการฝึกสอนให้กับโครงข่าย เมื่อจำนวนรอบการฝึกสอนให้กับโครงข่ายถึงจำนวนรอบที่ตั้งไว้

การวัดประสิทธิภาพของแบบจำลอง

ค่าที่นิยมใช้ในการวัดประสิทธิภาพมี 4 ค่า ได้แก่

1) ค่าความแม่นยำ (Accuracy) คือ ค่าที่บอกว่าโปรแกรมสามารถทำนายได้แม่นยำมากน้อยเท่าใด หาได้จาก

$$\text{Accuracy} = \frac{(TP+TN)}{(TP+TN+FP+FN)} \quad (2.9)$$

2) ค่าความระลึก (Recall) คือ ค่าที่บอกว่าโปรแกรมทำนายว่าจริง เป็นอัตราส่วนเท่าใดของจริงทั้งหมด หาได้จาก

$$\text{Recall} = \frac{TP}{(TP+FN)} \quad (2.10)$$

3) ค่าความเที่ยง (Precision) คือ ค่าที่บอกว่าโปรแกรมทำนายว่าจริง ถูกต้องเท่าใด หาได้จาก

$$\text{Precision} = \frac{TP}{(TP+FP)} \quad (2.11)$$

4) ค่าถ่วงดุล(Tradeoff) ระหว่างค่าความแม่นยำและค่าความเที่ยงเรียกว่า “F-Measure” โดยจะพิจารณาจากค่าความแม่นยำและค่าความเที่ยงแล้วนำมาหาค่าเฉลี่ยระหว่างค่าทั้งสอง

$$\text{F-Measure} = \frac{2 \times \text{TP}}{(2 \times \text{TP}) + \text{FP} + \text{FN}} \quad (2.12)$$

โดยที่

True Positive (TP) คือ จำนวนเอกสารที่ได้ผ่านการจำแนกคลาสและมีค่าเป็น Positive เมื่อทำนายด้วยแบบจำลองแล้วมีผลเป็น Positive

True Negative (TN) คือ จำนวนเอกสารที่ได้ผ่านการจำแนกคลาสและมีค่าเป็น Negative เมื่อทำนายด้วยแบบจำลองแล้วมีผลเป็น Positive

False Positive (FP) คือ จำนวนเอกสารที่ได้ผ่านการจำแนกคลาสและมีค่าเป็น Negative เมื่อทำนายด้วยแบบจำลองแล้วมีผลเป็น Positive

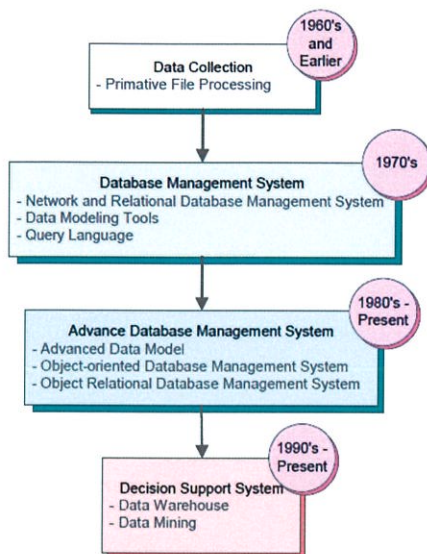
False Negative (FN) คือ จำนวนเอกสารที่ได้ผ่านการจำแนกคลาสและมีค่าเป็น Positive เมื่อทำนายด้วยแบบจำลองแล้วมีผลเป็น Negative

2.4เหมืองข้อมูล (Data Mining)

2.4.1 แนวคิดและความหมายของเหมืองข้อมูล

Data Mining คือการค้นหาค่าความสัมพันธ์และรูปแบบ (Pattern) ทั้งหมดซึ่งมีอยู่จริงในฐานข้อมูลแต่ได้ถูกซ่อนไว้ภายในข้อมูลจำนวนมาก Data Mining จะทำการสำรวจและวิเคราะห์อย่างอัตโนมัติหรือกึ่งอัตโนมัติในปริมาณข้อมูลจำนวนมากให้อยู่ในรูปแบบที่เต็มไปด้วยความหมายและอยู่ในรูปของกฎ (Rule) โดยความสัมพันธ์เหล่านี้แสดงให้เห็นถึงความรู้ต่างๆที่มีประโยชน์ในฐานข้อมูลในปัจจุบัน การทำเหมืองข้อมูลได้ถูกนำไปประยุกต์ใช้ในงานหลายประเภททั้งในด้านธุรกิจที่ช่วยในการตัดสินใจของผู้บริหาร ในด้านวิทยาศาสตร์และการแพทย์ รวมทั้งในด้านเศรษฐกิจและสังคม

ดังแสดงในรูปที่ 2.11 การทำเหมืองข้อมูลเปรียบเสมือนอีกวิวัฒนาการหนึ่งในการจัดเก็บและตีความหมายข้อมูล จากเดิมที่มีการจัดเก็บข้อมูลอย่างง่าย มาสู่การจัดเก็บในฐานข้อมูลที่ สามารถดึงค่าสารสนเทศของข้อมูลมาใช้ จนถึงการทำเหมืองข้อมูลที่สามารถค้นพบความรู้ที่ซ่อนอยู่ในข้อมูล



รูปที่ 2.11 วิวัฒนาการในการจัดเก็บและตีความหมายข้อมูล (ที่มา: <http://km.cpc.ac.th/jintana>)

ในการทำเหมืองข้อมูลนั้น ต้องใช้ความรู้จากศาสตร์หลายแขนง ได้แก่

- ฐานข้อมูล (database systems, data warehouses, On-Line Analytical Processing – OLAP) ซึ่งเป็นหลักในการจัดเก็บ รวบรวม และเตรียมข้อมูลที่ใช้ในการทำเหมืองข้อมูล
- การเรียนรู้ของเครื่อง (machine learning) ใช้เป็นอัลกอริทึมหลักที่ใช้ในการค้นหา รูปแบบและความสัมพันธ์ที่ซ่อนอยู่ในข้อมูล
- หลักสถิติ (statistical and data analysis methods) สำหรับวิเคราะห์ข้อมูลเบื้องต้นซึ่งอาจจะชี้ให้เห็นถึงรูปแบบและความสัมพันธ์ของข้อมูลที่ซ่อนอยู่ได้
- การทำจินตทัศน์ (visualization) เพื่อแสดงผลลัพธ์ รูปแบบ และความสัมพันธ์ของข้อมูลออกมาให้ผู้ใช้งานเข้าใจง่ายที่สุด ซึ่งเป็นประโยชน์ต่อการตีความและการนำผลลัพธ์นั้นไปใช้
- การโปรแกรมทางคณิตศาสตร์ (mathematical programming)
- การคำนวณประสิทธิภาพสูง (high performance computing) เนื่องจากข้อมูลที่มีมากจะทำให้การทำเหมืองข้อมูลใช้เวลานาน จึงจำเป็นต้องมีการคำนวณที่รวดเร็วรองรับ

นอกจากความรู้เหล่านี้แล้ว การทำเหมืองข้อมูลยังต้องอาศัยความรู้ในตัวข้อมูลที่ใช้ในการทำเหมืองด้วย เช่น ในการทำเหมืองข้อมูลทางชีววิทยา จำเป็นต้องมีความรู้ด้านชีววิทยา หรือ ในการทำเหมืองข้อมูลเพื่อใช้ในด้านธุรกิจ จำเป็นต้องมีความรู้ในธุรกิจนั้นๆ

2.4.2 สถาปัตยกรรมของการทำเหมืองข้อมูล

กระบวนการทำเหมืองข้อมูลจะประกอบไปด้วยองค์ประกอบย่อยๆดังนี้

- 2.4.2.1 แหล่งข้อมูล เกิดจากการเก็บรวบรวมข้อมูลในรูปของคลังข้อมูล ฐานข้อมูล หรือเพิ่มข้อมูล ซึ่งข้อมูลเหล่านี้ต้องผ่านกระบวนการเตรียมข้อมูลให้อยู่ในสภาพที่พร้อมในการประมวลผล
- 2.4.2.2 เซิร์ฟเวอร์ มีหน้าที่ในการเรียกดูข้อมูลที่ใช้ต้องการเพื่อนำมาทำเหมืองข้อมูล
- 2.4.2.3 ฐานความรู้ เป็นที่เก็บรวบรวมผลลัพธ์ที่ได้จากการทำเหมืองข้อมูล รวมถึงความรู้ที่ผู้เชี่ยวชาญสามารถใส่เข้าสู่ระบบเพื่อช่วยให้การทำเหมืองข้อมูลประสบความสำเร็จได้
- 2.4.2.4 ตัวประมวลผลเหมืองข้อมูล ประกอบไปด้วยขั้นตอนวิธีการทำเหมืองข้อมูลหลายรูปแบบ เช่น ตัวจำแนกประเภท (Classification) ขั้นตอนวิธีการจัดกลุ่มข้อมูล (Clustering) เป็นต้น
- 2.4.2.5 ตัวประเมินคุณภาพของความรู้ที่ได้จากการทำเหมืองข้อมูล ได้แก่ ฟังก์ชันวัดค่าความถูกต้องของตัวจำแนก (Accuracy)
- 2.4.2.6 ส่วนต่อประสานไปยังผู้ใช้ เป็นอินเทอร์เฟซที่ช่วยให้ผู้ใช้สามารถติดต่อกับระบบได้ในรูปแบบของการสั่งงานและเรียกดูผลลัพธ์ของการทำเหมืองข้อมูล

2.4.3 ขั้นตอนในการทำเหมืองข้อมูล

การทำเหมืองข้อมูลประกอบด้วยขั้นตอนหลักดังต่อไปนี้

2.4.3.1 การทำความเข้าใจปัญหา ประกอบด้วยกระบวนการย่อยดังนี้

- ตั้งเป้าหมายว่าการทำเหมืองข้อมูลครั้งนี้ต้องการที่จะแก้ปัญหาใด เช่น ทำเหมืองข้อมูลเพื่อต้องการเพิ่มยอดขายสินค้า เป็นต้น
- ตั้งเกณฑ์วัดความสำเร็จในการทำเหมืองข้อมูล ซึ่งอาจเป็นได้ทั้งความสำเร็จในด้านรูปธรรม เช่น สามารถเพิ่มยอดขายสินค้าได้ 5% และความสำเร็จในนามธรรม เช่น สามารถค้นพบความรู้ใหม่จากข้อมูล

- ประเมินสถานการณ์ในด้านต่างๆ เช่น ความรู้พื้นฐานในเรื่องที่จะทำเหมือง ข้อมูลมีเพียงพอหรือไม่ และผลประโยชน์จากการทำเหมืองข้อมูลจะคุ้มค่ากับต้นทุนที่เสียไปหรือไม่ เป็นต้น
- ตั้งเป้าหมายในเชิงการทำเหมืองข้อมูล ซึ่งต่างไปจากเป้าหมายหลักในการแก้ปัญหา เช่น เป้าหมายหลักคือต้องการเพิ่มยอดขายสินค้า เป้าหมายในเชิงการทำเหมืองข้อมูล คือ การหาลักษณะของลูกค้าที่มีแนวโน้มจะซื้อสินค้า
- วางแผนการทำเหมืองข้อมูลว่าจะเก็บข้อมูลอย่างไร และใช้อัลกอริทึมใดในการทำเหมืองข้อมูล

2.4.3.2 ทำความเข้าใจข้อมูล ประกอบด้วยกระบวนการย่อยดังนี้

- เก็บรวบรวมข้อมูล
- กำหนดคุณสมบัติของข้อมูลที่เก็บมาได้
- ตรวจสอบข้อมูลอย่างคร่าวๆ ถึงค่าสถิติต่างๆของข้อมูล
- ตรวจสอบข้อมูลขั้นต้น โดยตรวจสอบทั้งความสมบูรณ์และความถูกต้องของข้อมูล

2.4.3.3 การเตรียมข้อมูล ประกอบด้วยกระบวนการย่อยดังนี้

- คัดเลือกข้อมูลที่จะนำมาใช้
- ปรับเปลี่ยนรูปแบบข้อมูล เช่น นำสองตารางในฐานข้อมูลมาเชื่อมต่อกัน
- ทำความสะอาดข้อมูล เป็นกระบวนการเตรียมข้อมูลให้เหมาะสมที่สุดเพื่อนำไปใช้ในขั้นตอนต่อไป ซึ่งมีวิธีการต่างๆ หลายวิธี ได้แก่ การแก้ไขข้อมูลให้ถูกต้องสมบูรณ์ เช่น การแก้ไขค่าว่างของข้อมูล โดยใส่ค่า 0 การปรับเปลี่ยนข้อมูลให้มีค่าเหมาะสมในการตัดสินใจ

2.4.3.4 การสร้างแบบจำลอง ประกอบด้วยกระบวนการย่อยดังนี้

- เลือกอัลกอริทึมที่เหมาะสมในการทำเหมืองข้อมูล
- กำหนดรูปแบบการทดสอบผลลัพธ์
- สร้างแบบจำลองตามอัลกอริทึมที่เลือก
- ทดสอบแบบจำลองที่ได้มานั้นว่ามีความถูกต้องและน่าเชื่อถือเพียงใด

2.4.3.5 การประเมิน

อาจจะประเมินแบบจำลองที่สร้างขึ้นด้วยการลองนำไปใช้กับสถานการณ์จริง หรือนำไปใช้

2.4.4 การทำเหมืองข้อมูลเว็บ (Web Mining)

การทำเหมืองข้อมูลเว็บ (Web Mining) คือ การใช้เทคนิคการทำเหมืองข้อมูลเพื่อค้นหาและสกัดข้อมูลและสารสนเทศจากเอกสารเว็บและบริการบนเว็บ โดยอัตโนมัติ เพื่อนำความรู้ที่ได้มาแก้ปัญหาที่ต้องการทั้งทางตรงและทางอ้อม นอกจากนี้ยังได้แบ่งประเภทของการทำเหมืองข้อมูลเว็บ โดยพิจารณาจากข้อมูลที่นำมาวิเคราะห์ออกเป็น 3 ประเภท คือ Web Content Mining, Web Structure Mining และ Web Usage Mining

2.4.4.1 Web Content Mining

เป็นการค้นหาข้อมูลที่มีประโยชน์จากข้อมูลที่อยู่ภายในเว็บ เช่น ข้อความ รูปภาพ เป็นต้น โดย Web Content Mining สามารถแบ่งออกเป็น 2 ประเภทตามมุมมองคือ มุมมองทางด้านการสืบค้นสารสนเทศ (Information Retrieval) และมุมมองทางด้านฐานข้อมูล (Database) สำหรับเป้าหมายของ Web Content Mining จากมุมมองของการสืบค้นสารสนเทศคือการทำเหมืองข้อมูลเว็บเพื่อปรับปรุงการหาข้อมูลหรือกรองข้อมูลให้ผู้ใช้โดยพิจารณาจากข้อมูลที่ใช้อ้างอิงหรือร้องขอ ในขณะที่เป้าหมายของ Web Content Mining ในมุมมองของฐานข้อมูลส่วนใหญ่พยายามจำลองข้อมูลบนเว็บและรวมข้อมูลนั้น เพื่อให้การสอบถามทำงานดีขึ้นมากกว่าการใช้คำหลักเป็นตัวค้นหาเพียงอย่างเดียว

2.4.4.2 Web Structure Mining

เป็นวิธีการที่พยายามค้นหารูปแบบโครงสร้างการเชื่อมโยงที่สำคัญและซ่อนอยู่ในเว็บ ซึ่งรูปแบบนี้จะขึ้นอยู่กับรูปแบบการเชื่อมโยงเอกสารภายในเว็บ โดยนำรูปแบบที่ได้มาใช้ในการจัดกลุ่มเว็บเพจและใช้สร้างข้อมูลสารสนเทศที่เป็นประโยชน์ เช่น นำมาใช้ในการปรับโครงสร้างของเว็บให้สามารถให้บริการผู้ใช้ได้อย่างรวดเร็ว

2.4.4.3 Web Usage Mining

เป็นวิธีการที่พยายามค้นหาความหมายของข้อมูลที่สร้างจากช่วงการทำงานหนึ่งของผู้ใช้หรือสร้างจากพฤติกรรมของผู้ใช้เรียกอีกชื่อหนึ่งว่า Web Log Mining โดยในขณะที่ Web Content Mining และ Web Structure Mining ใช้ประโยชน์จากข้อมูลจริง หรือข้อมูลพื้นฐานบนเว็บแต่ Web

Usage Mining ทำการค้นหาความรู้จากข้อมูลการติดต่อสื่อสารระหว่างกันของผู้ใช้ที่ติดต่อกับเว็บ โดย Web Usage Mining ทำการรวบรวมข้อมูลจากบันทึกในการดำเนินการต่างๆ เช่น บันทึกการใช้งานของ Proxy (Proxy Server Log) ข้อมูลการลงทะเบียน (Registration Data) หรือข้อมูลอื่นอันเป็นผลจากการทำงานร่วมกันมาใช้วิเคราะห์ ดังนั้น Web Usage Mining จึงเป็นวิธีการทำงานที่เน้นใช้เทคนิคที่สามารถทำนายพฤติกรรมของผู้ใช้ในขณะที่ผู้ใช้ทำงานกับเว็บ กระบวนการทำงานของ Web Usage Mining สามารถแบ่งออกเป็น 2 วิธีคือ

1. ทำการจับคู่ข้อมูลการใช้งานของเครื่องให้บริการเว็บให้อยู่ในรูปของตารางความสัมพันธ์ ก่อนที่นำข้อมูลนี้มาปรับใช้กับเทคนิคการทำเหมืองข้อมูลการใช้เว็บ
2. ใช้ประโยชน์จากข้อมูลในบันทึกการใช้งานโดยตรงซึ่งจะใช้เทคนิคการเตรียมข้อมูล (Preprocessing) เพื่อเตรียมข้อมูลก่อนหาความสัมพันธ์ (Pattern Discovery) และวิเคราะห์รูปแบบ (Pattern Analysis)

2.4.5 เทคนิคการทำเหมืองข้อมูลเว็บ

2.4.5.1 Statistical Analysis การวิเคราะห์ทางด้านสถิติ เป็นวิธีการที่จะค้นหาความรู้เกี่ยวกับผู้เยี่ยมชมเว็บไซต์ โดยการวิเคราะห์ส่วนของไฟล์ เป็นวิธีหนึ่งที่สามารถบอกชนิดความแตกต่างเกี่ยวกับการวิเคราะห์ทางสถิติบนตัวแปร เช่น หน้าที่ดู เวลาการดูและความยาวของเส้นทางเว็บไซต์ เครื่องมือการวิเคราะห์การจราจรเว็บส่วนมากจะรายงานข้อมูลเกี่ยวกับสถิติ เช่น ความถี่ของหน้าที่เข้าถึงบ่อยๆ เวลาเฉลี่ยของหน้าหรือความยาวเฉลี่ยของเส้นทางเว็บไซต์ รายงานนี้อาจจะรวมถึงการวิเคราะห์ข้อผิดพลาดที่จำกัด เช่น การสืบทายการที่ไม่ให้สิทธิเข้าถึงหรือการค้นหา URL ที่ไม่สมบูรณ์ รูปแบบของความรู้สามารถเป็นประโยชน์อย่างมากสำหรับการปรับปรุงระบบ การเพิ่มความปลอดภัยของระบบ ทำให้สะดวกในการปรับปรุงเว็บไซต์และการสนับสนุนการตัดสินใจทางการตลาด

2.4.5.2 Association Rule เป็นการค้นหาความสัมพันธ์ของข้อมูล โดยค้นหาความสัมพันธ์ของข้อมูลทั้งสองชุดหรือมากกว่าสองชุดขึ้นไปไว้ด้วยกัน ความสำคัญของกฎ ทำการวัดโดยใช้ข้อมูลสองตัวด้วยกัน คือค่าสนับสนุน (Support) ซึ่งเป็นเปอร์เซ็นต์ของการดำเนินการที่กฎสามารถนำไปใช้ หรือเป็นเปอร์เซ็นต์ของการดำเนินการที่กฎที่ใช้มีความถูกต้อง และข้อมูลตัวที่สองที่นำมาใช้วัดคือค่าความมั่นใจ (Confidence) ซึ่งเป็นจำนวนของกรณีที่กฎถูกต้องโดยสัมพันธ์

กับจำนวนของกรณีที่สามารถนำไปใช้ได้ ในการหาความสัมพันธ์นั้นจะมีขั้นตอนวิธีการหาหลายวิธีด้วยกัน แต่ขั้นตอนวิธีที่เป็นที่รู้จักและใช้อย่างแพร่หลายคือ อัลกอริทึม Apriori

2.4.5.3 Clustering คือการจัดกลุ่มข้อมูลซึ่งมีลักษณะคล้ายกับการแบ่งประเภทแต่จะไม่เหมือนกัน โดยการแบ่งประเภทจะวิเคราะห์ข้อมูลตามต้นแบบ แต่สำหรับการแบ่งกลุ่มเป็นการวิเคราะห์โดยไม่พิจารณาจัดกลุ่มตามประเภทที่มีหรือที่รู้จัก แต่จะใช้ขั้นตอนวิธีการจัดกลุ่มเพื่อค้นหากลุ่มที่สามารถยอมรับได้เพื่อจัดเข้ากลุ่ม กล่าวคือ กลุ่มของวัตถุมีการสร้างขึ้น โดยเปรียบเทียบวัตถุที่มีความเหมือนกันจัดเข้ากลุ่มเดียวกัน

2.4.5.4 Classification เป็นการจัดแบ่งประเภทของข้อมูล โดยหาชุดต้นแบบหรือชุดของการทำงานที่อธิบายและแบ่งประเภทข้อมูล วัตถุประสงค์เพื่อให้สามารถใช้เป็นต้นแบบทำนายประเภทของวัตถุหรือข้อมูลที่ไม่มีการระบุประเภทหรือชนิดของข้อมูล ซึ่งต้นแบบสร้างจากการวิเคราะห์ชุดของข้อมูลฝึกสอน (Training Data) โดยอาจจะเป็นกลุ่มข้อมูลที่มีการระบุประเภทหรือกลุ่มเรียบร้อยแล้ว รูปแบบของต้นแบบแสดงได้หลายแบบเช่น Classification Rules Decision Trees หรือ Neural Networks เป็นต้น

2.4.5.5 Sequential Pattern เป็นเทคนิคการสืบค้นรูปแบบลำดับเหตุการณ์ เป็นการหาความสัมพันธ์ของข้อมูลระหว่างทรานแซกชันซึ่งทำให้มีเวลาเข้ามาเกี่ยวข้อง ด้วยรูปแบบลำดับที่น่าสนใจ จะแสดงว่าถ้าเกิดเหตุการณ์นี้แล้วหรือพบกลุ่มของข้อมูลชุดนี้แล้วมีแนวโน้มที่จะเกิดเหตุการณ์อะไรหรือพบกลุ่มของข้อมูลใดตามมาจากภายหลัง

2.4.5.6 Dependency Modeling เป็นแบบจำลองการพึ่งพา เป็นการค้นพบรูปแบบอื่นๆ ที่มีประโยชน์ในการทำเหมืองข้อมูลเว็บ เป้าหมายคือการพัฒนาต้นแบบที่สามารถทำได้ของตัวแทนที่สำคัญของการพึ่งพาท่ามกลางตัวแปรต่างๆ ในโดเมนเว็บ เช่น อาจจะใช้วิธีการสร้างแบบจำลองสถานที่แตกต่างกัน ให้ผู้เยี่ยมชมได้เห็นในขณะที่เลือกซื้อสินค้าทางออนไลน์ มีหลายเทคนิคการเรียนรู้ที่เป็นไปได้ สามารถถูกใช้เพื่อเป็นแบบของการค้นหาพฤติกรรมของผู้ใช้ เช่น ตัวแบบฮิดเดนมาร์คอฟ (Hidden Markov Model) และแนวคิดข่ายงานความเชื่อเบย์ (Bayesian Belief Network)

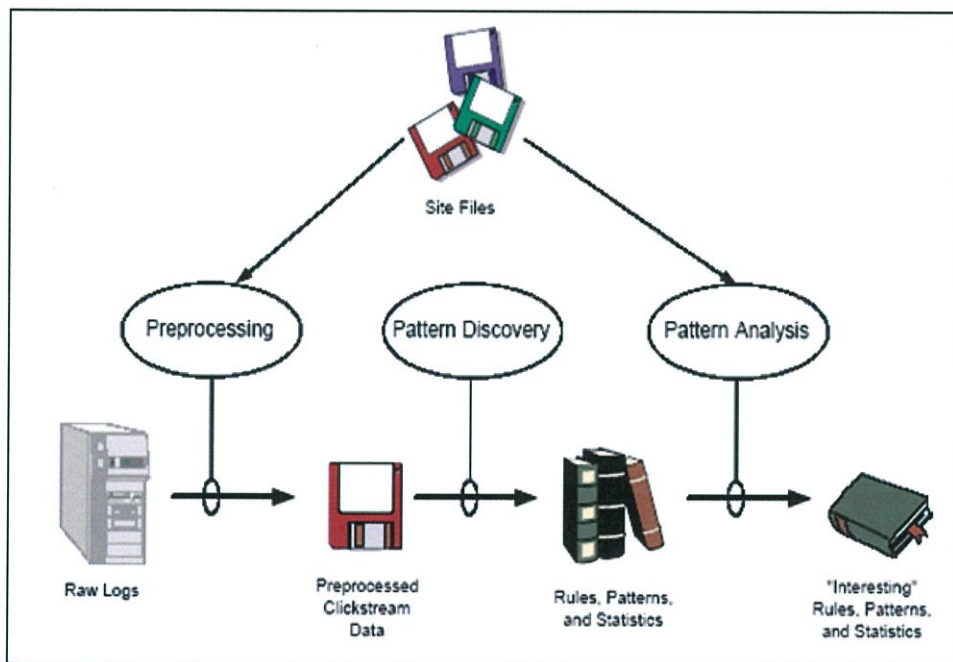
2.4.6 ขั้นตอนการทำเหมืองข้อมูลเว็บ

กระบวนการทำเหมืองข้อมูลเว็บ ประกอบด้วยกระบวนการ 3 ขั้นตอนด้วยกันคือ

2.4.6.1 การเตรียมข้อมูล (Preprocessing) เป็นขั้นตอนการแปลงข้อมูลเว็บให้อยู่ในรูปแบบที่เหมาะสมสำหรับการค้นหาความสัมพันธ์ เช่น Data Cleaning, Data Filtering

2.4.6.2 การค้นหารูปแบบ (Pattern Discovery) เป็นขั้นตอนที่นำเทคนิคการทำเหมืองข้อมูล เช่น Association Rule Discovery, Sequential Pattern Discovery เพื่อค้นหาความสัมพันธ์หรือรูปแบบจากข้อมูลเว็บที่ผ่านขั้นตอนการเตรียมข้อมูลแล้ว

2.4.6.3 การวิเคราะห์รูปแบบ (Pattern Analysis) เป็นรูปแบบการนำผลลัพธ์จากการค้นหา มาทำการวิเคราะห์เพื่อช่วยในการตัดสินใจหรือการวางแผนทางธุรกิจ



รูปที่ 2.12 ขั้นตอนการทำ Web Usage Mining (Srivastava, 2000)

2.5 เว็บแอปพลิเคชัน (Web Application)

ปัจจุบันการพัฒนาด้านโปรแกรมเพื่อใช้งานบนเว็บไซต์ ได้รับความนิยมมากขึ้นอย่างต่อเนื่อง เนื่องจากเว็บแอปพลิเคชันสามารถตอบสนองความต้องการได้จากทุกที่ทุกเวลา ทำงานร่วมกันจากหลาย ๆ แห่งได้พร้อมกัน โดยผู้พัฒนาไม่ต้องติดตั้งโปรแกรมลงบนเครื่องของผู้ใช้งานทุกคน แต่ใช้วิธีการนำโปรแกรมไปติดตั้งไว้บนเครื่องคอมพิวเตอร์ Server จากนั้นผู้ใช้งานโปรแกรมจะสามารถเรียกใช้โปรแกรมได้ผ่านทางเว็บเบราว์เซอร์ (Web Browser) ซึ่งมีให้เลือกใช้หลายผลิตภัณฑ์ เช่น Internet Explorer, Google Chrome, Mozilla Firefox เป็นต้น

2.5.1 ประเภทของเว็บแอปพลิเคชัน

แบ่งได้เป็น 3 ประเภทหลัก ดังนี้

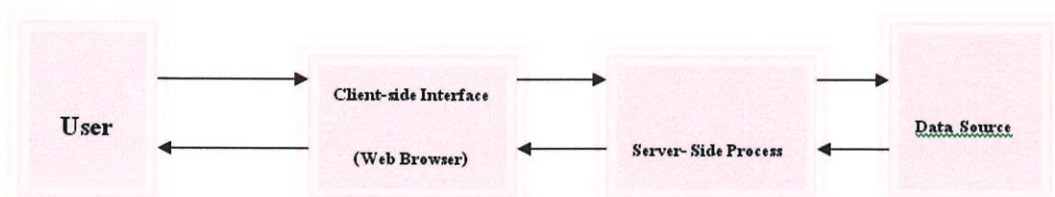
- 1) Static Web Application คือ เว็บแอปพลิเคชันที่ไม่มีการโต้ตอบระหว่างผู้ใช้งานกับเว็บเบราว์เซอร์ เช่น HTML
- 2) Dynamic Web Application คือ เว็บแอปพลิเคชันที่มีการโต้ตอบระหว่างผู้ใช้งานกับ Web Server
- 3) Web Services คือ แอปพลิเคชันที่อยู่บนเว็บ ข้อมูลที่ได้จากแอปพลิเคชันจะอยู่ในรูปแบบของเอกสาร XML โดยสามารถรองรับได้หลาย Platforms

2.5.2 ความหมายของเว็บแอปพลิเคชัน

เว็บแอปพลิเคชัน เป็นแอปพลิเคชันที่เข้าถึงได้ด้วยเว็บเบราว์เซอร์ผ่านคอมพิวเตอร์อย่าง อินเทอร์เน็ตหรืออินทราเน็ต ซึ่งเว็บแอปพลิเคชันเป็นที่นิยมมากในการพัฒนาโปรแกรม เนื่องจากความสามารถในการอัปเดต และดูแล โดยการติดตั้งซอฟต์แวร์บนเครื่องผู้ใช้หรือการสร้างโปรแกรมให้ทำงานอัตโนมัติอยู่บนเว็บไซต์ เช่น การนำเสนอข้อมูลอัตโนมัติ การรับบันทึกข้อมูลรายละเอียดต่างๆ เข้าสู่ฐานข้อมูล เพื่อนำไปใช้ในการวิเคราะห์ข้อมูลต่างๆ หรือติดต่อสื่อสารกับผู้ใช้เข้าชมเว็บไซต์แบบออนไลน์ โดยตัวอย่างเว็บแอปพลิเคชัน ที่เป็นที่รู้จักกันทั่วไป เช่น เว็บกระดานสนทนา บล็อก เว็บเมล เป็นต้น

2.5.3 ลักษณะการทำงานของเว็บแอปพลิเคชัน

เว็บแอปพลิเคชัน มีการพัฒนาเพื่อใช้ในการเขียนโปรแกรมอย่างต่อเนื่อง จึงไม่มีวิธีการที่ตายตัวแน่นอนในการพัฒนา เนื่องจากผู้พัฒนาสามารถเลือกใช้เทคโนโลยีที่มีอยู่อย่างหลากหลายมาใช้ร่วมกันได้ เช่น การเลือกใช้งาน CGI, ASP, .NET, Java Servlet เป็นต้น และการใช้งานข้อมูลในรูปแบบของฐานข้อมูล XML, HTML, CSS เป็นต้น แต่การพัฒนาเทคโนโลยีทั้งหมดนี้ มีหลักการทำงานพื้นฐานของเว็บแอปพลิเคชันเหมือนกันหมด โดยมีขั้นตอนการทำงาน ดังนี้



รูปที่ 2.13 ขั้นตอนการทำงานของเว็บแอปพลิเคชัน

- 1) **Provide a Query Interface** เว็บแอปพลิเคชันจะแสดงแบบฟอร์มผ่านทางเว็บเบราว์เซอร์ เพื่อให้ผู้ใช้สามารถป้อนข้อมูลเข้ามาในโปรแกรมได้ โดยข้อมูลที่ถูกป้อนเข้ามาในระบบนี้จะเรียกว่า “Query” หรือ “Request” Data ซึ่งหมายถึงเป็นข้อมูลที่ใช้ในการร้องขอบริการจาก Web Server เช่น การค้นหาข้อมูลจากฐานข้อมูลการส่งหนังสือ การป้อนข้อมูลการลงทะเบียนผ่านอินเทอร์เน็ต เป็นต้น
- 2) **Transmit User-Defined Query** เมื่อผู้ใช้ป้อนข้อมูลเข้ามาในระบบแล้ว ข้อมูลเหล่านั้นจะถูกส่งไปยัง Web Server ผ่านเครือข่ายอินเทอร์เน็ต
- 3) **Perform Server Side Processing** Web Server จะทำการประมวลผลข้อมูลที่ได้รับมาจากผู้ใช้ โดยใช้ Middleware
- 4) **Message Data** เมื่อทำการประมวลผลเสร็จแล้ว Web Server จะส่งผลลัพธ์ออกมาจากโปรแกรมเพื่อส่งไปให้ผู้ใช้ในขั้นตอนต่อไป
- 5) **Transmit Query Results** เมื่อข้อมูลถูกส่งไปถึงผู้ใช้งานแล้ว ข้อมูลจะแสดงผลผ่านเว็บเบราว์เซอร์
- 6) **Perform Client Side Processing** เว็บเบราว์เซอร์จะทำหน้าที่แสดงข้อมูลที่ส่งกลับมาจาก Web Server

2.5.4 ประโยชน์ของเว็บแอปพลิเคชัน

- 1) ใช้งานง่าย เพียงแค่มีเว็บเบราว์เซอร์ ก็สามารถใช้งานได้
- 2) ใช้งานได้ทุกที่ทุกเวลา โดยใช้งานได้โดยไม่ต้องติดตั้งโปรแกรม
- 3) สะดวกในการบริหาร และควบคุมจากส่วนกลาง (Centralized Control)
- 4) ใช้ได้หลากหลายแพลตฟอร์มทั้ง Windows, Linux และ Mac
- 5) ข้อมูลจัดเก็บที่เดียว ง่ายต่อการจัดการ และไม่เกิดความซ้ำซ้อน
- 6) ประหยัดค่าใช้จ่าย เพราะไม่จำเป็นต้องมีคอมพิวเตอร์ที่มีประสิทธิภาพสูง
- 7) เหมาะสำหรับองค์กรที่มีการติดตั้งระบบเครือข่าย
- 8) มีความยืดหยุ่น และสามารถขยายขีดความสามารถได้
- 9) เชื่อมต่อกับเว็บแอปพลิเคชันหรือบริการออนไลน์อื่นๆ ได้ง่าย

2.6 ASP.NET

2.6.1 ความหมายของ ASP.NET

ASP.NET คือ เฟรมเวิร์ก (Framework) ที่ใช้ในการสร้างเว็บไซต์ หรือเว็บแอปพลิเคชัน และเว็บเซอร์วิส โดยคำว่า เฟรมเวิร์ก (Framework) หรือ .NET Framework นั้น จะหมายถึงแพลตฟอร์ม (Platform) ที่ประกอบไปด้วยเครื่องมือ รวมไปถึงไลบรารีต่างๆ ที่ใช้ในการพัฒนาโปรแกรมซึ่ง .NET Framework นั้นถูกสร้างขึ้นโดย Microsoft ที่สามารถติดตั้งได้ในระบบปฏิบัติการ Windows เช่น WindowsXP, Windows Vista, Windows 7, Windows Server เป็นต้นซึ่ง ASP.NET มีต้นกำเนิดมาจากภาษา ASP (Active Server Pages) ซึ่งเป็นเว็บเทคโนโลยีที่ใช้ในการสร้างเว็บไซต์

ใน .NET Framework รองรับการทำงานของภาษาหลายๆภาษาด้วยกัน ซึ่งเป็นข้อดีเพราะมีความยืดหยุ่นสูง ตัวอย่างของภาษาที่ .NET Framework เช่น C#, VB.NET, C++/CU, F# เป็นต้น จากตัวอย่างภาษาข้างต้นสามารถเลือกใช้ต่างๆได้ หากต้องการสร้างแอปพลิเคชันหรือซอฟต์แวร์ตัวหนึ่งที่รองรับการทำงานของ .NET Framework แต่ถ้าหากต้องการสร้างเว็บไซต์ด้วย ASP.NET สามารถเลือกใช้ได้ 2 ภาษา คือ C# และ VB.NET

2.6.2 ASP.NET MVC

MVCย่อมาจาก Model View Controller เป็นรูปแบบอย่างหนึ่งที่ใช้ในการแบ่งแยกแอปพลิเคชันออกเป็นมุมมอง 3 ส่วน คือ

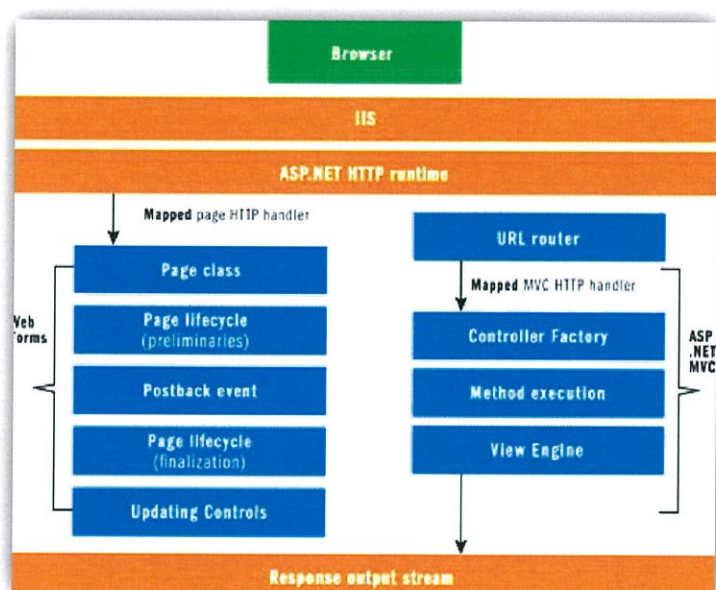
- Model คือ ส่วนที่ใช้ในการติดต่อกับฐานข้อมูลซึ่งทำหน้าที่ในการดึงข้อมูลขึ้นมาจากฐานข้อมูลขึ้นมาและทำการจัดการข้อมูลต่างๆไว้ในรูปแบบที่เหมาะสม
- View คือ ส่วนที่จะนำข้อมูลจาก Model ไปใช้แสดงผลให้ผู้ใช้เห็นผลลัพธ์ออกมาใน User Interface
- Controller คือ ส่วนที่จะคอยรับ Input จาก Client เข้ามาแล้วนำคำสั่งไปประมวลผลเพื่อสั่งงาน View และ Model ต่ออีกทีว่าจะให้ประมวลผลออกมาเป็นอย่างไร

MVC จะแบ่งส่วนการทำงานชัดเจนช่วยให้สามารถจัดการและดูแลตัวระบบที่มีขนาดใหญ่ได้ง่ายมากขึ้นอีกทั้งยังช่วยให้การพัฒนาเพิ่มเติม หรือแก้ไข ปรับปรุงระบบสะดวกมากยิ่งขึ้นอีกด้วย

หลักการทำงานของ MVC

- 1) Client ทำการ Request ข้อมูลผ่าน View
- 2) Request จะถูกส่งผ่านไป Controller เพื่อ Filter Action ต่างๆ
- 3) จากนั้น Controller ก็จะร้องขอไปยัง Model จาก Action นั้นๆ
- 4) เมื่อ Model รับการร้องขอก็จะทำการ Query ข้อมูลตามที่ Controller ส่งมา
- 5) Model จะส่ง Response ข้อมูลที่ได้กลับมามาหา Controller
- 6) Controller ก็จะทำการกำหนดค่าลงในตัวแปรเพื่อส่งให้ View ต่อไป
- 7) View ก็จะนำตัวแปรเหล่านั้นไปทำการแสดงผลตามต้องการได้

2.6.3 ความแตกต่างระหว่าง ASP.NET Web Form และ ASP.NET MVC



รูปที่ 2.14 ความแตกต่างระหว่าง ASP.NET Web Form และ ASP.NET MVC

จากรูปที่ 2.14 ทางส่วนด้านซ้ายจะเป็นโปรเซสการทำงานของ ASP.NET Web Form จะพบว่ามีการใช้งาน Page Class และอาศัยการทำงานของ Postback event ซึ่งทำให้การทำงานนั้นมีความซับซ้อน ส่วนทางด้านขวาจะเป็นโปรเซสการทำงานของ ASP.NET MVC จะเห็นได้ว่าไม่มีการ Postback event ทำให้ดูเรียบง่ายและ Maintain ได้ง่ายกว่า

2.6.4 ข้อดีของ MVC Pattern

1) SoC (Separate of Concern) ช่วยลดความยุ่งยากในการพัฒนา และมีการจัดเป็นสัดส่วน ได้แก่ Model View และ Controller ทำการแยกการทำงานออกจากกัน ทั้ง User Interface, Business Logic, Data Access Logic หรืออาจจะเขียนแยกฟังก์ชันและเมธอดสำหรับการนำมาใช้ใหม่ (Reuse)

2) รองรับการทำ test-driven development (TDD) เนื่องจากโครงสร้างของ ASP.NET MVC นั้นได้ถูกออกแบบด้วย INTERFACE-BASED Programming ทำให้สามารถจำลองข้อมูล (mock object) เพื่อใช้ในการอินพุต/เอาต์พุตในเมธอดหรือคลาสที่ต้องการทดสอบได้ จึงไม่จำเป็นต้องกด Run Debug (F5) เพื่อทำการทดสอบระบบเหมือนกับ ASP.NET Web Form ที่คอยต้องนำข้อมูลเข้าไปเพื่อทำการทดสอบโค้ดและข้อดีอีกประการหนึ่งคือสามารถเปลี่ยน unit-testing framework ที่พัฒนาบน .NET Framework ได้ตามต้องการอีกด้วย

3) Extensible/Pluggable Abilities ความสามารถในการต่อเติมขยายระบบ เนื่องจาก Component ของ ASP.NET MVC นั้นได้ถูกพัฒนามาให้เราสามารถแก้ไขเพิ่มเติม หรือหาอะไรมาทดแทนการทำงานได้ เช่น สามารถนำ View Engine มาใช้งาน หรือจะกำหนดการทำงานของ URL Routing เอง ซึ่ง ASP.NET MVC นั้นรองรับการใช้งานในลักษณะ Dependency Injection (DI) คือการใส่ Object เข้าไปยังคลาสแทนการเรียกใช้ตรง ๆ โดยอ้างอิงจาก Interface Contract และยังรองรับการทำงานแบบ Inversion of Control (IoC) ซึ่งเป็นการ reverse ของการเรียกใช้ Object นั้นเอง โดยไปอ่านจาก Configที่กำหนดไว้ โดยทั้ง DI/LoC สามารถนำมาใช้งานเข้ากับ ASP.NET MVC ได้ทุกส่วน ซึ่งการใช้งานวิธีการเหล่านี้จะช่วยทำให้แยกทดสอบระบบได้อย่างอิสระมากขึ้น

4) Modern Web Programming Support เนื่องจากความสามารถของ Routing ใน ASP.NET MVC มาใช้ในการทำ URL Mapping ทำให้ URL ของนั้นดูง่ายและไม่ซับซ้อน ตลอดจนยังสามารถซ่อน Extension ของไฟล์ (.aspx) และพวก Query String ทำให้ URL ดูง่ายและจดจำง่าย ซึ่งความสามารถนี้ยังช่วยในการทำ Search Engine Optimize (SEO) และ REpresentational State Transfer (REST) อีกด้วย

5) ASP.NET Programming Model นั้น Support โค้ดที่เคยได้เขียนไว้ใน ASP.NET Web Form คือยังคงนำมาใช้งานได้ด้วยเช่นกัน ไม่ว่าจะเป็น asp.net page (.aspx), user control (.ascx) และ master page (.master) nested master page ซึ่งไฟล์เหล่านี้นำมาใช้เป็น View Template นั้นเอง อีกทั้งการโค้ดที่ใช้ใน View เราจะใช้ in-line expression (<%= %>) ในการควบคุม logic ของ view

นั้น ๆ นอกจากนั้นก็ยังมีการใช้ Server control + template, การทำ data binding, localization และอีกมากมาย

6) เหมาะสำหรับงานของเว็บแอปพลิเคชันซึ่งรูปแบบการพัฒนาสามารถรองรับระบบงานใหญ่และการแบ่งงานกันทำงานภายในทีมพัฒนาทุกขนาด

2.7 ไวยากรณ์พื้นฐาน

2.7.1 การประกาศตัวแปร

ตัวแปร คือ สิ่งที่ใช้ในการเก็บค่าหรือข้อมูลต่างๆ เพื่อให้ส่วนอื่นๆทำการเรียกใช้ต่อไปซึ่งการประกาศตัวแปรในภาษา C# และ Visual Basic มีดังนี้ในภาษา C# จะขึ้นต้นด้วยชนิดตัวแปรแล้วตามด้วยชื่อตัวแปร เช่น

```
Int a = 10;
stringstr = "Hello";
```

สำหรับข้อความที่เป็น string (string = ตัวอักษรรวมถึงอักขระต่างๆมากกว่าหนึ่งตัวมารวมกัน) จะใช้เครื่องหมาย double quote (“ ”) ครอบตรงส่วนที่อยู่หลังเครื่องหมายเท่ากับเราเรียกว่าค่าของตัวแปร เช่น 10 Hello เป็นต้น

2.7.1.1 การตั้งชื่อตัวแปร

ในการตั้งชื่อตัวแปรสามารถตั้งชื่อได้ตามต้องการ แต่ชื่อตัวแปรจะต้องไม่ซ้ำกับ Keyword และต้องขึ้นต้นด้วยเครื่องหมาย Underscore (_) หรือตัวอักษรเท่านั้น แต่หลังจากตัวอักษรตัวแรกไปจะมีตัวเลขก็ได้ ซึ่งในการตั้งชื่อตัวแปรต้องตั้งให้เข้าใจง่าย โดยปกติจะเป็นชื่อที่เกี่ยวข้องกับข้อมูลที่เก็บค่านั้นๆ เช่น ถ้าต้องการเก็บข้อมูลเกี่ยวกับรหัสนักรของศึกษา ชื่อที่สอดคล้องอาจจะเป็น StudentID, Student_ID เป็นต้น ซึ่งสำหรับตัวอักษรพิมพ์เล็กและพิมพ์ใหญ่ในภาษา C# จะให้ค่าไม่เหมือนกัน

2.7.1.2 ชนิดของตัวแปร

ชนิดของตัวแปรใน C# มีดังตารางต่อไปนี้

ตารางที่ 2.1 ชนิดของตัวแปรในภาษา C#

Data Types	Size	Values
sbyte	8 bit	-128 to 127
byte	8 bit	0 to 255
short	16 bit	-32,768 to 32,767
ushort	16 bit	0 to 65,535
int	32 bit	-2,147,483,648 to 2,147,483,647
uint	32 bit	0 to 4,294,967,295
long	64 bit	-9,223,372,036,854,775,808 to 9,223,372,036,854,775,807
ulong	64 bit	0 to 18,446,744,073,709,551,615
char	16 bit	0 to 65535
float	32 bit	-1.5 x 10 ⁴⁵ to 3.4 x 10 ³⁸
double	64 bit	-5 x 10 ³²⁴ to 1.7 x 10 ³⁰⁸
decimal	128 bit	-1028 to 7.9 x 10 ²⁸
bool	---	True or false
string	Variable length	0-2 billion Unicode characters

หากต้องการประกาศตัวแปรแบบไม่กำหนดชนิดของตัวแปร สามารถประกาศโดยใช้คีย์เวิร์ดคำว่า var สำหรับภาษา C# ได้ ซึ่งหลังประกาศ var แล้วไม่จำเป็นต้องกำหนดชนิดตัวแปรอีก เพราะคอมไพเลอร์จะทำการกำหนดชนิดของข้อมูลให้เองโดยอัตโนมัติ สามารถเขียนคำสั่งได้ดังนี้

```
protected void Page_Load (object sender, EventArgs e)
{
    var i = 50;
    var j = "String"
    Reponse.Write(i.ToString());
    Reponse.Write(j);
}
```

2.7.2 การกำหนดเงื่อนไข

ในกรณีที่ต้องการเขียนคำสั่งให้โปรแกรมทำการตัดสินใจทำงานบางอย่าง สามารถกำหนดเงื่อนไขต่างๆให้กับโปรแกรมได้ เช่น ถ้าเงื่อนไขแรกไม่เป็นจริงก็ให้โปรแกรมทำงานที่เงื่อนไขถัดไป ในภาษา C# จะมีคำสั่งที่ใช้กำหนดเงื่อนไขคือ if, if-else

2.7.2.1 การใช้งาน if

```
int i = 10;
int j = 13;
if (i>j)
{
    Response.Write ("i>j");
}
```

ในตัวอย่างนี้เป็นการตรวจสอบเงื่อนไข if ถ้าค่าตัวแปร i มากกว่า j จะให้แสดงข้อความว่า i > j ซึ่งในตัวอย่างแรกนี่จะเป็นการตรวจสอบแค่เงื่อนไข if เท่านั้น ถ้าเงื่อนไข if ไม่เป็นจริง จะออกจากเงื่อนไขทันทีแล้วทำในคำสั่งถัดไป

2.7.2.2 การใช้งาน if-else

```
int i = 10;
int j = 13;
if (i>j)
{
    Response.Write("i>j");
}
else
{
    Response.Write("i<j");
}
```

ในตัวอย่างจะมีการเพิ่ม else{} เข้ามา เพื่อทำงานในกรณีเงื่อนไข if ไม่เป็นจริง ก็จะเข้าทำงานในเงื่อนไข else

2.7.2.3 การใช้งาน if-elseif

```
int i = 10;
int j = 13;
if (i > j)
{
    Response.Write("i > j");
}
else if (i < j)
{
    Response.Write("i < j");
}
else
{
    Response.Write("i = j");
}
```

ในตัวอย่างนี้มีการเพิ่ม else if เข้ามา ซึ่งจะมีการตรวจสอบเงื่อนไขใน if ก่อน หากไม่เป็นจริงจะมาตรวจเงื่อนไขใน else if ถ้าเป็นจริงจะทำงานตามคำสั่งที่อยู่ใน else if แต่ถ้าหากเงื่อนไขใน else if ไม่เป็นจริง จะทำงานในตามคำสั่งที่อยู่ใน else

2.7.2.4 การใช้งาน switch/Select Case

ในกรณีที่มีทางเลือกหลายๆทาง เราสามารถที่จะดูค่านั้นว่าตรงกับทางเลือกไหนได้โดยใช้ switch สำหรับภาษา C# ได้ดังนี้

```
int inputNumber = 20;
switch (inputNumber)
{
    case 5;
    Response.Write("InputNumber = 5");
    break;
    case 10;
    Response.Write("InputNumber = 10");
}
```

```

        break;
        case 15;
Response.Write("InputNumber = 15");
        break;
        default;
Response.Write("Invalid InputNumber");
        break;
}

```

ในตัวอย่างนี้มีการประกาศตัวแปร InputNumber ซึ่งกำหนดค่าไว้ 20 แล้วนำค่าตัวแปรนี้มาตรวจสอบ โดยใช้ switch ว่าค่าของตัวแปรตรงกับ case ไหนบ้างใน switch ซึ่งสามารถกำหนด case ได้ตามต้องการและในแต่ละ case ต้องปิดท้ายด้วย break; เพื่อเป็นการบอกว่าหากตรงกับเงื่อนไขใน case ใดให้ทำงานตามคำสั่งใน case นั้นแล้วออกจากคำสั่ง switch ทันที โดยไม่มีการตรวจสอบ case อื่นๆอีก แต่ในกรณีที่ไม่มีตรงกับ case ใดเลยๆ จะเข้าทำงานที่ default;

2.7.3 การวนลูป (Loop)

การวนลูปเป็นกระบวนการทำงานที่มีลักษณะการทำงานซ้ำๆกัน โดยเมื่อทำงานตามคำสั่งในลูปเสร็จแล้ว จะเริ่มย้อนกลับมาทำงานที่ลูปถัดไปและจะหยุดทำงานเมื่อครบจำนวนครั้งของการวนลูป แล้วออกจากการวนลูป

2.7.3.1 การใช้งาน for

เป็นการวนลูปที่มีการกำหนดค่าเริ่มต้นที่จะวนลูปก่อนซึ่งมีรูปแบบดังนี้

```

for (int i = 0; i < 5; i++)
{
Response.write (i+ "<br/>");
}

```

ผลลัพธ์ของการใช้ for

```

0
1
2
3
4

```

ในการใช้ for นั้น เริ่มแรกให้ทำการกำหนดค่าเริ่มต้นที่จะวนลูปก่อน ในตัวอย่างนี้จะกำหนด $i = 0$ หมายความว่าให้วนลูปของตัวแปร i โดยเริ่มจาก 0 ไปเรื่อยๆ ถึง i มีค่าน้อยกว่า 5 และให้เพิ่มค่า i ทีละ 1 ($i++$)

2.7.3.2 การใช้งาน while

การใช้ while จะมีลักษณะคล้ายๆ กับ for แต่แตกต่างกันที่ while มีการตรวจสอบเงื่อนไขการวนลูปก่อนว่าเป็นจริงหรือไม่ ถ้าเป็นจริงจึงทำงานตามคำสั่งใน while

```
int i = 10;
    while (i<15)
    {
Response.Write(i + " ");
        i++;
    }
```

ผลลัพธ์ที่ได้คือ 10 11 12 13 14 15 จากตัวอย่างทำการกำหนดค่าเริ่มต้นให้กับตัวแปร i เป็น 10 จากนั้นทำการตรวจสอบเงื่อนไขใน while ถ้า i น้อยกว่า 15 ให้ทำงานตามคำสั่งในลูป while นั่นคือ แสดงค่าของตัวแปร i จากนั้นทำการเพิ่มค่าของ i ($i++$) ทีละ 1 ซึ่งส่วนนี้ถือเป็นส่วนสำคัญที่ต้องกำหนดให้มีการเพิ่มค่าของ i ไม่เช่นนั้นจะเกิดการวนลูปไม่รู้จบ (Infinite Loop)

2.7.3.3 การใช้งาน do while

```
int i = 10;
    do
    {
Response.Write(i + " ");
        i++;
    }while (i<15)
```

การใช้ do while จะมีลักษณะตรงกันข้ามกับ while นั่นคือ มีการลูปก่อนแล้วจึงตรวจสอบเงื่อนไข ในการใช้ do while จะทำงานในส่วนของ do ก่อน จากนั้นทำการตรวจสอบเงื่อนไข

while ว่าเป็นจริงหรือไม่ ถ้าเป็นจริงจะทำการวนลูปทำงานซ้ำในส่วนของ do ทำซ้ำจนกว่าเงื่อนไขใน while จะเป็นเท็จจึงออกจากลูป

2.7.3.4 การใช้งาน foreach

```
foreach (char c in "Hello")
{
    Respons.Write(c);
}
```

สำหรับการใช้งาน foreach จะมีการกำหนดชนิดตัวแปรที่รับค่า จากค่าที่ต้องการวนลูป ซึ่งในตัวอย่างให้แสดงตัวอักษร (char) ของข้อความ Hello ออกมาทีละตัวอักษร จากนั้นมีการแสดงค่าของตัวแปร c ออกมาผ่านคำสั่ง Respons.Write

2.7.4 การสร้าง Array

Array หรืออาร์เรย์เป็นชนิดของข้อมูลที่ทำหน้าที่เก็บข้อมูลต่างๆ ไว้ในลักษณะเป็นกลุ่มซึ่งการประกาศตัวแปรที่เป็น Array จะมีลักษณะดังนี้

```
string[] strData = new string[3];
```

จากการประกาศ Array ข้างต้นมีการกำหนดตัวแปร strData เป็นตัวแปร Array ที่เก็บข้อมูลชนิด string โดยมีขนาดเป็น 3 (สามารถมีสมาชิกใน Array ได้ 3 ตัว) ใช้สัญลักษณ์ [] เพื่อบอกว่าเป็นตัวแปร Array ซึ่งหลังจากประกาศตัวแปรแล้ว ต่อไปเป็นการกำหนดสมาชิกในตัวแปร strData ซึ่งมีวิธีดังนี้

```
strData[0] = "Car";
    strData[1] = "Rabbit";
strData[2] = "Cow";
```

สำหรับสมาชิกใน Array จะมีตำแหน่งเริ่มต้นที่ 0 เสมอ หลังจากที่เรากำหนดสมาชิกใน Array ครบ 3 ตัวแล้ว ขั้นตอนต่อไปเป็นการใช้ลูปในการแสดงสมาชิกใน Array ดังนี้

```

strData[0] = "Car";
    strData[1] = "Rabbit";
    strData[2] = "Cow";
    foreach (string str in strData)
    {
        Response.Write(str + "<br/>");
    }

```

ผลลัพธ์ที่ได้

Car

Rabbit

Cow

2.8 การกำหนด Style ด้วย CSS

2.8.1 ความหมายของ Cascading style Sheet (CSS)

CSS ย่อมาจาก Cascading Style Sheets เป็นภาษาที่มีรูปแบบการเขียน Syntax ที่เฉพาะ และถูกกำหนดมาตรฐานโดย W3C (World Wide Web Consortium) เช่นเดียวกับ HTML และ XHTML ใช้สำหรับตกแต่งเอกสาร HTML/ XHTML ให้มีหน้าตา สี สัน ตัวอักษร เส้นขอบ พื้นหลัง ระยะห่าง ฯลฯ อย่างที่ผู้ใช้ต้องการ ด้วยการกำหนดคุณสมบัติให้กับ Element ต่างๆ ของ HTML เช่น <body>, <p>, <h1> เป็นต้น

2.8.2 พื้นฐานการใช้งาน CSS

2.3.2.1 Comment

Comment เป็นการใส่คำอธิบายลงไปใน CSS หรือถ้าโค้ด CSS บางอย่างที่ไม่ได้ใช้แต่ไม่ต้องการลบออก ก็สามารถที่จะ Comment โค้ดส่วนนั้นไว้ได้ ทาง Browser จะไม่นำโค้ดส่วนนั้นไปประมวลผล ซึ่งใช้สัญลักษณ์ /* สำหรับเริ่มต้นการ Comment และปิดท้ายด้วย */ โดยสัญลักษณ์ /*

จะอ่านว่า Forward Slash and Asterisk และ */ อ่านว่า Asterisk and Forward Slash ลักษณะการ Comment ใน CSS จะมีดังนี้

```
<style type = "text/css">
    .myclass
    {
        /*background-color: Lime;*/
    }
```

ในตัวอย่างนี้ได้ทำการ Comment ส่วนของ background-color: Lime ไว้โดย browser จะไม่ทำการประมวลผลในส่วนนี้ แต่ถ้าหากต้องการใส่คำอธิบายหลังโค้ด สามารถเขียนได้ดังนี้

```
<style type = "text/css">
    .myclass
    {
        background-color: Lime; /*คำสั่งนี้ใช้สำหรับการกำหนดสีพื้นหลัง*/
    }
```

2.8.2.2 Selectors

Selectors เป็นการสร้างจุดอ้างอิงที่ประกาศ CSS เพื่อให้แท็ก HTML หรือ Control ต่างๆ ใน ASP.NET สามารถเรียกใช้ CSS ผ่าน Selectors นี้ได้ โดย Selectors จะมีอยู่หลายแบบด้วยกัน แต่ในส่วนนี้ขอกล่าวถึงเพียง 2 แบบเท่านั้น เนื่องจากเป็น Selectors ที่เป็นพื้นฐาน ซึ่งมีดังนี้

1. Class Selectors เป็นการสร้าง Style Class โดยเริ่มต้นด้วยสัญลักษณ์จุด (.) หรือ dot แล้วตามด้วยชื่อคลาส

```
<style type = "text/css">
    .class
    {
        position: fixed;
    }
```

```

.class2
{
    background-color: Blue;
}
.format_border
{
    border-top: 2px solid;
    border-bottom: 2px solid;
}
</style>

```

สามารถสร้าง Class Selectors ได้ไม่จำกัด โดยชื่อคลาสนั้นควรตั้งให้เหมาะสมกับการใช้งาน เพื่อที่จะได้เรียกใช้งานคลาสได้ถูกต้อง หลังจากที่ทำการสร้าง Class Selectors แล้ว สามารถเรียกใช้ได้โดยใช้คุณสมบัติของชื่อคลาส ดังนี้

```

<div class= "class2">
</div>

```

ในกรณีที่มี Class Selectors มากกว่า 1 คลาสแล้ว หากต้องการเรียกใช้งานคลาสหลายๆ คลาส สามารถทำได้ โดยการกำหนดชื่อคลาสแล้วเว้นห่าง 1 ช่อง แล้วตามด้วยชื่อคลาสถัดไป ดังนี้

```

<div class= "class2 format_border">
</asp:Panel>

```

ถ้าต้องการ Class Selectors ใน Control ของ ASP.NET สามารถใช้คุณสมบัติที่ชื่อ CssClass = "ชื่อคลาส" ได้ดังนี้

```

<asp:Panel ID= "Panel1" runat= "server" CssClass= "class1">
</asp:Panel>

```

2. ID Selectors มีลักษณะคล้ายๆกับ Class Selector แต่จะเริ่มต้นด้วยเครื่องหมาย # (Hash Mark) แทน ได้ดังนี้

```
<style type= "text/css">

#my_bg

{

    background: rgb(11, 22, 33);

}

</style>
```

ในการเรียกใช้งาน ID Selector จะทำการกำหนดชื่อ id = "ชื่อ" ดังนี้

```
<body id = "my_bg">

<form id= "form1" runat= "server">

<div class= "class2">

</div>

</form>

</body>
```

2.8.2.3 การกำหนดสี

สามารถกำหนดโดยใช้ชื่อ กำหนดเป็น RGB และรหัสสีในเลขฐาน 16 ดังนี้

1. การกำหนดเป็นชื่อสี วิธีนี้สามารถใส่ชื่อตรงๆได้เลย เช่น blue, green ดังตัวอย่าง

```

<style type= "text/css">

div

{

    background-color: Blue;

}

</style>

```

2. การกำหนดเป็นค่าสี RGB วิธีนี้ต้องเพิ่มคำสั่ง rgb ซึ่ง RGB จะเป็นระบบแม่สี มี 3 สี คือ Red, Green, Blue โดยเราสามารถกำหนดค่าตัวเลขให้กับ rgb นี้ได้ตั้งแต่ 0 ถึง 255 ดังตัวอย่าง

```

<style type= "text/css">

div

{

    background-color: rgb(22,156,233);

}

</style>

```

3. การกำหนดสีโดยใช้เลขฐาน 16 ในการกำหนดต้องขึ้นต้นเครื่องหมาย # แล้วตามด้วยอักษร A ถึง F หรือ 0 ถึง 9 ซึ่งชื่อสีจะประกอบไปด้วยตัวอักษรหรือตัวเลขรวมได้ 6 ตัว ดังตัวอย่าง

```
<style type= "text/css">
div
{
background-color: #AABB35
}
</style>
```

2.8.2.4 การจัดการกับตัวอักษร

ใน CSS จะมีคุณสมบัติต่างๆที่ในการจัดการกับตัวอักษร (font) หรือข้อความ เช่น การกำหนดรูปแบบ การเว้นระยะห่าง การกำหนดชนิดตัวอักษร เป็นต้น

1. การกำหนดระยะห่างระหว่างตัวอักษร ใช้คุณสมบัติที่ชื่อ letter-spacing : ค่าระยะห่างดังนี้

```
<style type= "text/css">
p
{
letter-spacing: 5px;
}
</style>
```

ในการเรียกใช้งานให้เราสร้างแท็ก p ขึ้นมา ดังนี้

```
<p> This is Letter-Spacing </p>
```

ผลลัพธ์ที่ได้คือ

This is Letter Spacing

2. การกำหนดชนิดตัวอักษรให้กับข้อความใช้คำสั่ง font-family: ชื่อแบบ; ในการกำหนดชนิดของตัวอักษร ซึ่งชนิดตัวอักษรนั้นจะต้องกำหนดชื่อตามที่ได้มีการติดตั้งไว้ในเครื่องด้วย เช่น Arial, Scan Serif เป็นต้น

```
<style type= "text/css">
p
{
font-family: Arial;
}
</style>
```

3. การกำหนดขนาดตัวอักษร ใช้คำสั่ง font-size: ขนาด; ซึ่งขนาดนั้นมีหลายหน่วย เช่น percent (%), pixel (px) หรือจะใช้หน่วยมาตรฐานก็ได้ ดังนี้

```
<style type= "text/css">
P
{
font-size:10px;
}
</style>
```

2.8.2.5 การกำหนด Style ด้วย Background

สำหรับ Background นั้น จะกำหนดสีพื้นหลังด้วยคำสั่ง background-color ใน CSS สามารถกำหนดได้ดังนี้

```

<style type= "text/css">

P

{

background-color: url('myimage1.jpg');

}

</style>

```

2.9 การติดต่อฐานข้อมูล

2.9.1 ความหมายของฐานข้อมูล

ฐานข้อมูล หรือ Database คือ แหล่งข้อมูล (Source) ที่ทำหน้าที่เก็บข้อมูลต่างๆไว้ สามารถดึงข้อมูล que เก็บมาใช้งานได้ และนอกจากนี้ยังสามารถเพิ่มข้อมูล เปลี่ยนแปลงข้อมูล รวมถึงลบข้อมูลออกจากฐานข้อมูลได้ โดยประโยชน์ของฐานข้อมูลก็คือ จัดการกับข้อมูลและเก็บข้อมูลต่างๆ ที่สำคัญนั่นเอง ตัวอย่างการนำฐานข้อมูลมาใช้งาน เช่น

- เก็บข้อมูลเกี่ยวกับพนักงานขององค์กรหรือบริษัท เช่น ตำแหน่งของพนักงานแต่ละคน แผนกงาน ที่อยู่ เบอร์โทรศัพท์ เป็นต้น
- เก็บข้อมูลนักเรียนหรือนักศึกษา เช่น เก็บข้อมูลชื่อ คณะหรือสาขาที่เรียน ผลการเรียน เป็นต้น
- เก็บข้อมูลห้องสมุด เช่น เก็บรายชื่อหนังสือ ผู้ยืมหนังสือ วัน-เวลาในการยืมคืน หมวดหมู่ของหนังสือ เป็นต้น

จากตัวอย่างการเก็บข้อมูลในฐานข้อมูล เห็นว่าสามารถเก็บข้อมูลได้หลากหลายรูปแบบ ขึ้นอยู่กับหน่วยงาน หรือนักคนที่ต้องการนำฐานข้อมูลไปใช้ นอกจากฐานข้อมูลแล้ว ข้อมูลที่ต้องการเก็บจะอยู่ในรูปของตาราง (Table) ฐานข้อมูล 1 ฐานข้อมูลสามารถมีได้หลายตาราง เช่น ฐานข้อมูลของบริษัทจะมี ตารางพนักงาน ตารางแผนก เป็นต้น ซึ่งเรื่องของตารางขึ้นอยู่กับผู้ออกแบบฐานข้อมูลเช่น

ตาราง Employee มีฟิลด์ ดังนี้

- EmployeeID (เก็บชื่อรหัสพนักงาน)
- EmployeeFirstName (เก็บชื่อพนักงาน)
- EmployeeLastName (เก็บนามสกุลพนักงาน)
- EmployeeBirthDate (เก็บวันเกิดพนักงาน)
- EmployeeAddress (เก็บที่อยู่พนักงาน)
- DepartmentID (เก็บรหัสแผนกที่พนักงานคนนี้อยู่)

ตาราง Department มีฟิลด์ ดังนี้

- DepartmentID (เก็บรหัสแผนก)
- DepartmentName (เก็บชื่อแผนก)

หลังจากออกแบบแล้วเสร็จเรียบร้อยแล้ว ต่อไปเป็นการเชื่อมโยงความสัมพันธ์ระหว่างตาราง เช่น จากตัวอย่างที่ผ่านมาในตาราง Employee จะเห็นฟิลด์ DepartmentID ซึ่งเป็นฟิลด์สำหรับเก็บรหัสแผนก ในตารางนี้จะเชื่อมโยงกันกับตารางที่ชื่อ Department ว่าพนักงานคนนั้นทำอยู่ในแผนกใด โดยใช้ DepartmentID เป็นตัวเชื่อมโยง เช่น สมมติว่าในตาราง Employee มีพนักงานชื่อ A พนักงานคนนี้มี DepartmentID เป็น 103 นั่นหมายความว่า พนักงานคนนีทำงานอยู่ในแผนกที่มีรหัสเป็น 103 ดังนั้น สมมติว่ามีข้อมูลดังตารางที่ 2.2

ตารางที่ 2.2 ตาราง Department

DepartmentID	DepartmentName
100	Customer Sevice
101	Logistic
102	Marketing
103	IT
104	Sale

ตารางที่ 2.3 ตาราง Employee

EmployeeID	EmployeeName	DepartmentID
101000	A	100
101001	B	104
101002	C	103

Primary Key (PK) คือ คีย์หลักของแต่ละตารางที่ต้องมี โดยคีย์หลักนี้ต้องไม่ซ้ำกัน เช่น ตาราง Department จะมี DepartmentID เป็นคีย์หลัก นั่นหมายความว่าต้องมีชื่อ DepartmentID ซ้ำกัน

Foreign Key (FK) คือ คีย์รองที่ใช้ในการเชื่อมระหว่างตารางเข้าด้วยกัน โดย Foreign Key จะทำการนำ Primary Key ของอีกตารางหนึ่งมาเป็น Foreign Key ของอีกตารางหนึ่ง เพื่อที่จะเชื่อมโยงกันได้ จากตัวอย่างตาราง Employee มีการนำ DepartmentID มาเป็น Foreign Key

2.9.2 คำสั่งในภาษา SQL

ภาษา SQL (Standard Query Language) เป็นส่วนหนึ่งของระบบฐานข้อมูลแบบรีเลชันแนล (Relational Database) ที่ได้รับความนิยมมากเพราะง่ายต่อความเข้าใจและอยู่ในรูปของภาษาอังกฤษ

ประเภทคำสั่งของ SQL

1) คำสั่งกำหนดประเภทข้อมูล (Data Definition Language Command: DDL) เป็นกลุ่มคำสั่งที่ใช้สำหรับการปรับเปลี่ยนโครงสร้างของฐานข้อมูลประกอบด้วยคำสั่ง Create, Replace, Alter, Truncate, Rename, Drop

2) คำสั่งในการควบคุมโครงสร้างข้อมูล (Data Control Language Command: DCL) ประกอบด้วยคำสั่งที่ใช้ในการควบคุมการเกิดภาวะพร้อมกันหรือการป้องกันการเกิดเหตุการณ์ที่ผู้ใช้หลายคนเรียกใช้ข้อมูลพร้อมกันและคำสั่งที่เกี่ยวข้องกับการควบคุมความปลอดภัยของข้อมูลด้วยการกำหนดสิทธิของผู้ใช้ที่แตกต่างกันเช่นคำสั่ง Grant และ Revoke

3) คำสั่งในการปรับปรุงข้อมูล (Data Manipulation Language Command: DML) ประกอบด้วยคำสั่งที่ใช้ในการเรียกใช้ข้อมูลการเปลี่ยนแปลงข้อมูลการเพิ่มหรือลบข้อมูลซึ่งได้แก่ คำสั่ง Insert, Delete และ Update

4) คำสั่งที่ใช้ในการค้นหาข้อมูล (Data Retrieval Command) มีหน้าที่ในการค้นหาข้อมูล เพื่อแสดงรายการข้อมูลหรือคำนวณ โดยมีคำสั่งเพียงคำสั่งเดียวนั้นคือคำสั่ง Select

5) คำสั่งในการควบคุมการทำรายการข้อมูล (Transaction Control Command) เป็นคำสั่งที่ใช้ในการยืนยันรายการต่างๆที่ผู้ใช้งานได้กระทำกับฐานข้อมูลโดยคำสั่งในกลุ่มนี้จะประกอบด้วย “Commit” และ “Rollback”

2.9.2.1 คำสั่ง Insert

สามารถใช้คำสั่ง INSERT ในการเพิ่มข้อมูลต่อท้ายตารางได้โดยการใช้คำสั่ง INSERT ร่วมกับวลี VALUE ซึ่งมีรูปแบบดังนี้

```
INSERT INTO tablename (column1, column2, column..n)
VALUES (Value1, Value2, Value..n )
```

เมื่อ	tablename	คือ	ชื่อของตารางที่จะทำการเพิ่มข้อมูล
	column	คือ	ชื่อของคอลัมน์ที่จะทำการเพิ่มข้อมูล
	value	คือ	ค่าต่างๆที่จะเพิ่มให้กับคอลัมน์นั้นๆ

ตัวอย่างเช่น

```
Insert Into Student (ID, Name, Age, Tel)
```

```
Values ('1020', 'Egkarin', 28, '089001234')
```

2.9.2.2 คำสั่ง Update

คำสั่ง Update ใช้สำหรับการปรับปรุงข้อมูลเดิมโดยสามารถกำหนดขอบเขตในการแก้ไขข้อมูลได้โดยใช้คำสั่ง Where โดยไม่มีการกำหนดเงื่อนไขที่ใช้ในคำสั่ง Update จะหมายถึงการแก้ไขข้อมูลในทุกรายการ (Record) ซึ่งมีรูปแบบในการใช้งานดังนี้

```
Update tablename Set column1 = value1 [, column2 = value2,...]
```

```
[Where condition]
```

เมื่อ	tablename	คือ	ชื่อของตารางที่จะทำการเพิ่มข้อมูล
	column	คือ	ชื่อของคอลัมน์ที่จะทำการเพิ่มข้อมูล
	value	คือ	ค่าต่างๆที่จะเพิ่มให้กับคอลัมน์นั้นๆ
	condition	คือ	เงื่อนไขที่ระบุขอบเขตในการแก้ไขรายการในตาราง

ตัวอย่างเช่น

```
Update student Set Name = 'เอกรินทร์' , Tel = '045123456'
```

```
Where ID = '1020'
```

2.9.2.3 คำสั่ง Delete

เป็นคำสั่งที่ใช้สำหรับลบรายการข้อมูลภายในตารางสามารถกำหนดขอบเขตที่ใช้สำหรับลบรายการได้เช่นเดียวกับคำสั่ง Update โดยมีรูปแบบการใช้งานดังนี้

```
Delete From tablename [Where condition]
```

เมื่อ	tablename	คือ	ชื่อของตารางที่จะทำการเพิ่มข้อมูล
	condition	คือ	เงื่อนไขที่ระบุขอบเขตในการแก้ไขรายการในตาราง

ตัวอย่างเช่น

```
Update student Set Name = 'เอกรินทร์' , Tel = '045123456' Where ID = '1020'
```

2.9.2.4 คำสั่ง Select

เป็นคำสั่งที่มีความสำคัญอย่างมากที่สุดในกลุ่มของคำสั่ง SQL เนื่องจากจะเป็นเสมือนเครื่องมือของผู้ใช้งานระบบฐานข้อมูลสามารถแสดงผลลัพธ์ในรูปแบบรายการทำให้ทราบได้ว่าข้อมูลที่มีอยู่ในฐานข้อมูลมีอะไรบ้างข้อมูลที่ได้นั้นก็แก้ไขหรือลบนั้นเป็นอย่างไรนอกจากนี้ยังสามารถใช้เพื่อการคำนวณหรือประมวลผลในรูปแบบต่างๆได้อีกด้วย รูปแบบของคำสั่ง Select มีดังนี้

```

SELECT select_list
FROM table_source
[ WHERE search_condition]
[ GROUP BY group_by_expression]
[ HAVING search_condition]
[ ORDER BY order_expression[ ASC | DESC ] ]

```

เมื่อ	<i>select_list</i>	คือ	รายการคอลัมน์ที่ต้องการแสดงผล
	<i>table_source</i>	คือ	ชื่อตาราง
	<i>search_condition</i>	คือ	เงื่อนไขที่ใช้กำหนดขอบเขตการค้นหารายการ
	<i>group_by_expression</i>	คือ	คอลัมน์ที่ใช้ในการจัดทำ group by
	<i>order_expression</i>	คือ	คอลัมน์ที่ใช้ในการจัดเรียงข้อมูล

ตัวอย่างเช่น

```
Select * From student ;
```

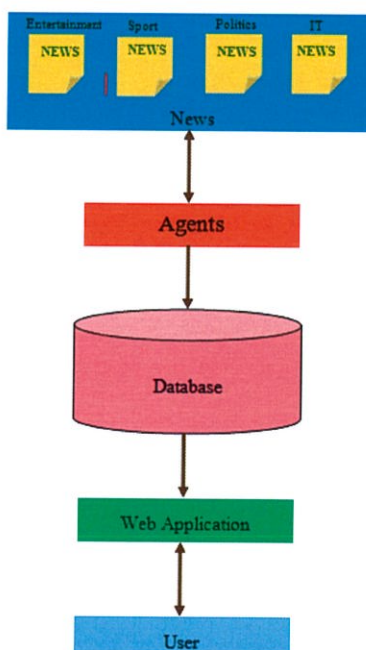
```
Select name, age From student Where age > 15
```

บทที่ 3

ขั้นตอนการดำเนินงาน

ในการพัฒนาเว็บแอปพลิเคชันสำหรับจำแนกกลุ่มเอกสารบนเว็บ จะทำการจำแนกข่าวคือ ข่าวการเมือง ข่าวกีฬา ข่าวบันเทิง และข่าวไอทีโดยใช้ตัวแทนปัญญา (Intelligent Agent) หรือซอฟต์แวร์ที่ทำหน้าที่เปรียบเสมือนตัวแทนโดยทำการสอนให้ตัวแทนปัญญานี้สามารถตัดสินใจได้ว่าข่าวนี้เป็นข่าวประเภทใด โดยจะใช้เทคนิคการเรียนรู้ของเครื่อง (Machine Learning)

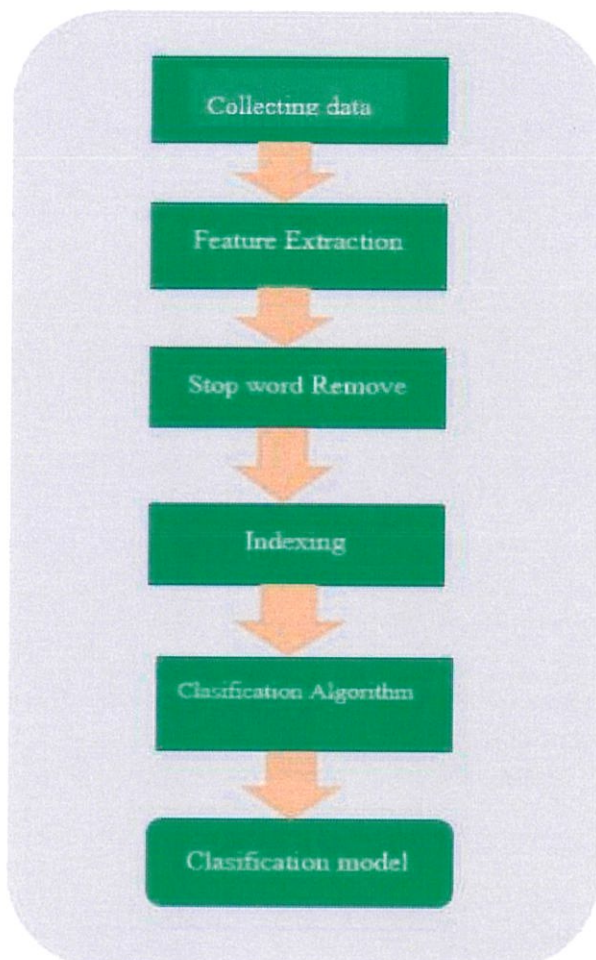
ขั้นตอนในการพัฒนาระบบอยู่ 2 ส่วนหลักคือ ส่วนการเลือกสร้างแบบจำลองสำหรับการรู้จำข่าวแต่ละประเภทและส่วนของการพัฒนาเว็บแอปพลิเคชัน



รูปที่ 3.1 แผนภาพแสดงภาพรวมของการพัฒนาเว็บแอปพลิเคชัน

จากรูปที่ 3.1 แสดงขั้นตอนโดยรวมของการพัฒนาเว็บแอปพลิเคชันสำหรับการจำแนกกลุ่มเอกสารบนเว็บไซค์ โดยเริ่มจากตัวแทนปัญญาจะทำการดึงข่าวจากเว็บไซค์ข่าวต่างๆมาเก็บในฐานข้อมูลตามประเภทของข่าว โดยจะมีข่าวอยู่ 4 ประเภท คือ ข่าวการเมือง ข่าวกีฬา ข่าวบันเทิง และข่าวไอที เมื่อผู้ใช้ทำการเลือกข่าว ระบบจะทำการดึงรายการข่าวจากฐานข้อมูล แล้วแสดงรายการข่าวผ่านหน้าเว็บแอปพลิเคชัน

3.1 การสร้างแบบจำลองสำหรับการจำแนกข่าวแต่ละประเภท



รูปที่ 3.2 โครงสร้างของกระบวนการสร้างแบบจำลอง

กระบวนการสร้างแบบจำลองประกอบด้วย 4 ขั้นตอน ดังนี้

1. ขั้นตอนการรวบรวมข้อมูล (Collecting data) เป็นขั้นตอนในการรวบรวมข้อมูลเพื่อใช้ในการเรียนรู้
2. ขั้นตอนในการเลือกคุณลักษณะ (Feature Extraction) เป็นการเลือกคำสำคัญ (Keywords) ที่ใช้ในจำแนกประเภทข่าวโดยใช้โปรแกรม Copernic Summarizer และใช้กระบวนการตัดคำที่ไม่สำคัญออก (Stop Word Removed) เข้ามาใช้ในการเลือกคุณลักษณะ

3. ขั้นตอนการการสร้างดัชนี (Indexing) เป็นขั้นตอนการแปลงเอกสารให้อยู่ในรูปแบบที่คอมพิวเตอร์สามารถเรียนรู้ได้
4. ขั้นตอนการใช้อัลกอริทึมในการจำแนกประเภท (Classification Algorithm) เพื่อใช้ในการสร้างแบบจำลองและวัดประสิทธิภาพของแบบจำลอง

3.1.1 ขั้นตอนการรวบรวมข้อมูล (Collecting data)

ทำการเก็บและรวบรวมข่าว 4 ประเภท ได้แก่ ข่าวการเมือง ข่าวไอที ข่าวบันเทิง และข่าวกีฬา ให้อยู่ในรูปแบบไฟล์นามสกุล .txt โดยมีจำนวนกลุ่มข่าวตัวอย่างทั้งหมด 480 เอกสาร โดยเป็นกลุ่มข้อมูลฝึกสอน (Training) จำนวน 360 เอกสาร และกลุ่มข้อมูลทดสอบ (Testing) จำนวน 120 เอกสาร

3.1.2 ขั้นตอนในการเลือกคุณลักษณะ (Feature Extraction)

นำข้อมูลฝึกสอน (Training) มาทำการคัดเลือกคุณลักษณะ (Feature Extraction) โดยการตัดคำที่ไม่สำคัญ (Stop Word) และคำซ้ำออกไป ผลลัพธ์ที่ได้จากการตัดคำจะได้เป็นคำเดี่ยว (Single Word) จำนวนมากเพื่อมาใช้เป็นตัวแทนเอกสารในการเรียนรู้ซึ่งคำที่จัดว่าเป็น Stop Word ได้แก่

- คำสรรพนาม สันธาน บุพบท เช่น I, are, in, or เป็นต้น
- อักขระโรมัน A ถึง Z
- ตัวเลข
- สัญลักษณ์ต่างๆ เช่น { , * , “ ,) เป็นต้น

ตารางที่ 3.1 ตัวอย่างบางส่วนของคำที่ไม่เกี่ยวข้องในการค้นหาหรือสืบค้น (Stop Word)

A	EITHER	IF	SOME
ABOUT	EVERY	MOSTLY	SUCH
BE	FIRST	NEXT	THAN
BEFORE	HAVE	NOR	WOULD
CAN	HE	NOT	YOU
CO	I	OTHERS	YET

3.1.3 ขั้นตอนการสร้างดัชนี (Indexing)

ทำการสร้างดัชนี (Index) เป็นการแปลงเอกสารให้อยู่ในรูปแบบที่คอมพิวเตอร์สามารถทำการเรียนรู้ได้ โดยเริ่มจากการสร้างเวกเตอร์ตัวแทนเอกสารจากนั้นจะสร้างเมตริกซ์ของกลุ่มเอกสารขึ้นจากเวกเตอร์ตัวแทนเอกสาร ซึ่งจะได้รูปแบบที่แสดงความสัมพันธ์ระหว่างคำ (Words) และเอกสารทั้งหมด (Documents) ด้วยเวกเตอร์ 2 มิติ ดังตารางที่ 3.2 ซึ่งตัวเลขในเวกเตอร์ตัวแทนเอกสาร คือ ความถี่ของคำซ้ำ โดยจะนำความถี่ของคำซ้ำนี้ไปเป็นอินพุตในขั้นตอนต่อไป

ตารางที่ 3.2 เวกเตอร์ตัวแทนเอกสาร

	word	Doc1	Doc2	Doc3	...	DocM
1	Actor	1	2	0		1
2	Music	0	0	1		1
3	Hollywood	2	1	1		0
4	Superstar	0	0	0		1
5	Singer	0	0	4		0
...						
M	magazine	1	0	1		0

3.1.4 ขั้นตอนการใช้อัลกอริทึมในการแบ่งประเภท (Classification Algorithm)

เข้าสู่ขั้นตอนการเรียนรู้ของเครื่องแบบมีผู้สอน โดยใช้เทคนิคโครงข่ายประสาทเทียมแบบหลายชั้น (Multilayer Perceptrons: MLP) เพื่อใช้ในการสร้างแบบจำลองสำหรับการจำแนกประเภทข่าว เนื่องจากมีความซับซ้อนในการหาค่าผลลัพธ์

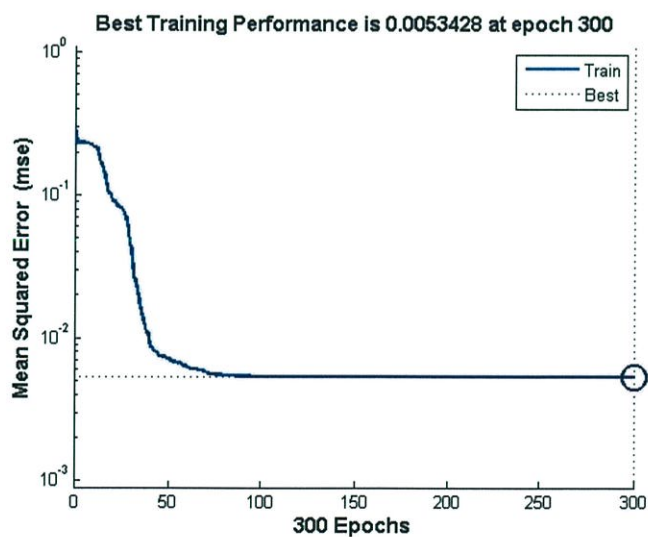
3.1.4.1 การฝึกสอนโครงข่าย (Training Process)

ขั้นตอนที่ 1 นำความถี่ของคำซ้ำที่ได้จากขั้นตอนการสร้างดัชนีมาเป็นอินพุต (Input)

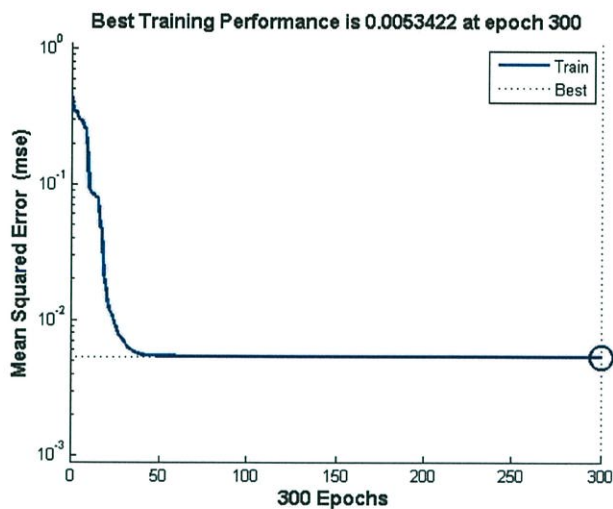
	1	2	3	4	5	6	7	8	9	10	11	12	13
1	0	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	0	0	0	0	0	0	1
3	0	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	1	0	0	0	0	0	0	0	0	0
10	0	0	0	0	0	0	0	0	0	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	1	2	0
12	0	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0	0
16	0	0	0	0	0	0	0	0	0	0	0	0	0
17	0	0	0	0	0	0	0	0	0	0	0	0	0
18	0	0	0	0	0	0	0	0	0	0	0	0	0
19	0	0	0	0	0	0	0	0	0	0	0	0	0
20	0	0	0	0	0	0	0	0	2	0	0	0	0
21	0	0	0	0	0	0	0	0	0	0	0	0	0
22	0	0	0	0	0	0	0	0	0	0	0	0	0

รูปที่ 3.3 ข้อมูลอินพุตสำหรับการฝึกสอน

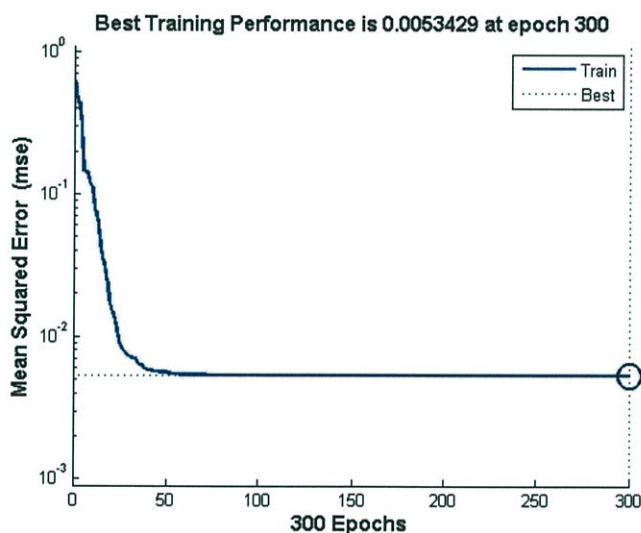
ขั้นตอนที่ 2 กำหนดจำนวนโหนดในชั้นซ่อน (Hidden) โดยทำการทดลองหาจำนวนโหนดในชั้นซ่อนที่ให้ค่าความผิดพลาดน้อยที่สุด โดยกำหนดจำนวนโหนดในชั้นซ่อนเป็น 5 10 และ 15 โหนด



รูปที่ 3.4 ค่าความผิดพลาดเฉลี่ยเมื่อกำหนดให้จำนวนโหนดในชั้นซ่อนมี 5 โหนด



รูปที่ 3.5 ค่าความผิดพลาดเฉลี่ยเมื่อกำหนดให้จำนวนโหนดในชั้นซ่อนมี 10 โหนด



รูปที่ 3.6 ค่าความผิดพลาดเฉลี่ยเมื่อกำหนดให้จำนวนโหนดในชั้นซ่อนมี 15 โหนด

ตารางที่ 3.3 ผลการทดลองในการหาจำนวนโหนดในชั้นซ่อนที่ทำให้ค่าความผิดพลาดน้อยที่สุด

จำนวนโหนดในชั้นซ่อน	ค่าความผิดพลาดเฉลี่ย
5	0.0053428
10	0.0053422
15	0.0053429

สรุปผลการทดลองจำนวนโหนดในชั้นซ่อนที่ทำให้เกิดความผิดพลาดเฉลี่ยน้อยที่สุดคือ 10 โหนด ซึ่งมีค่าความผิดพลาดเฉลี่ยเท่ากับ 0.0053422

ขั้นตอนที่ 3 กำหนดจำนวนเอาต์พุต (Output) เป็นคลาสของข่าว 4 ประเภท โดยคลาสของข่าวบันเทิงคือ 1 0 0 0 คลาสของข่าวไอทีคือ 0 1 0 0 คลาสของข่าวการเมืองคือ 0 0 0 1 และคลาสของข่าวกีฬาคือ 0 0 1 0

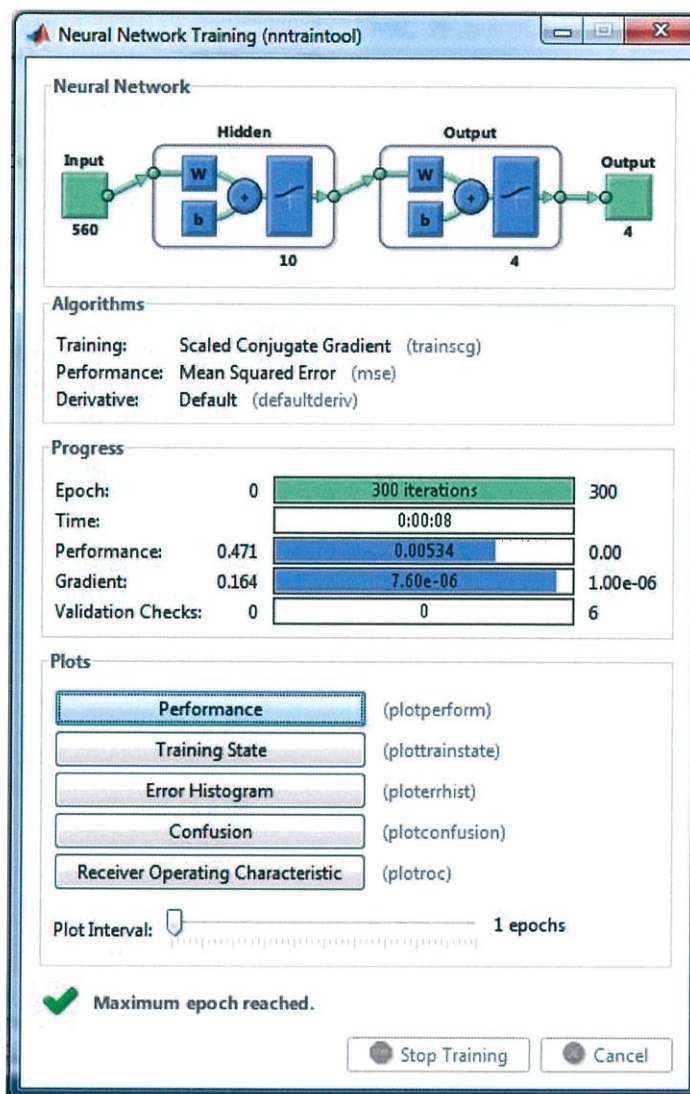
ขั้นตอนที่ 4 กำหนดค่าเป้าหมาย (Targets) ให้กับอินพุต

	1	2	3	4	5	6	7	8	9	10	11	12
1	1		1	1	1	1	1	1	1	1	1	1
2	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0
5												
6												
7												
8												
9												
10												
11												
12												
13												
14												
15												
16												

รูปที่ 3.7 ค่าเป้าหมายสำหรับการฝึกสอน

ขั้นตอนที่ 5 กำหนดอัตราการเรียนรู้ (Learning Rate) เท่ากับ 0.001 และจำนวนรอบสูงสุดที่ใช้ในการฝึกสอนเท่ากับ 300 รอบ

ขั้นตอนที่ 6 Neural Network จะทำการคำนวณเอาต์พุตออกมา และนำค่าเอาต์พุตที่ได้มาเปรียบเทียบกับค่าเป้าหมาย (Target) ถ้าค่าเอาต์พุตกับค่าเป้าหมายมีค่าไม่เท่ากัน จะทำการปรับค่าน้ำหนัก (Weight) โดยเริ่มจากการปรับค่าน้ำหนักของชั้นเอาต์พุตก่อน แล้วจึงปรับค่าน้ำหนักของชั้นซ่อน ซึ่งเรียกวิธีการนี้ว่า การเรียนรู้แบบแพร่ย้อนกลับ (Back propagation Learning) ดังนั้น ในขั้นตอนสุดท้ายจะได้ค่าน้ำหนักที่ทำให้เกิดความผิดพลาดน้อยที่สุด



รูปที่ 3.8 การฝึกสอน โครงข่ายโดยใช้ Neural Network Toolbox ใน โปรแกรม Matlab

3.1.4.2 การทดสอบโครงข่าย (Testing Process) มีขั้นตอนดังนี้ คือ นำเข้าชุดข้อมูลทดสอบ โดยนำค่าน้ำหนัก และค่าไบแอส ที่ได้จากการสอน มาทำการคำนวณค่าเอาต์พุต บันทึกค่าความผิดพลาด (Mean Square Error) ของแบบจำลองที่ได้ทำการพัฒนาเพื่อหาแบบจำลองที่มีค่าความผิดพลาดน้อยที่สุด เพื่อนำมาเป็นต้นแบบที่ใช้ในการพัฒนาเว็บแอปพลิเคชันในขั้นตอนถัดไป

Variable Editor - TestInputs

File Edit View Graphics Debug Desktop Window Help

Stack: Base No valid plots for TestInpu...

TestInputs <560x120 double>

	1	2	3	4	5	6	7	8	9	10	11	12
1	0	0	0	0	0	0	0	0	0	0	0	0
2	0	0	0	0	0	0	1	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0
5	0	0	0	0	0	0	0	0	0	0	0	0
6	0	0	0	0	0	0	0	0	0	0	0	0
7	0	0	0	0	0	0	0	0	0	0	0	0
8	0	0	0	0	0	0	0	0	0	0	0	0
9	0	0	0	1	0	0	0	0	0	0	2	0
10	0	1	0	0	0	0	0	1	0	0	0	0
11	0	0	0	0	0	0	0	0	0	0	0	0
12	0	0	0	0	0	0	0	0	0	0	0	0
13	0	0	0	0	0	0	0	0	0	0	0	0
14	0	0	0	0	0	0	0	0	0	0	0	0
15	0	0	0	0	0	0	0	0	0	0	0	0

รูปที่ 3.9 ข้อมูลอินพุตสำหรับการทดสอบ

Variable Editor - TestTargets

File Edit View Graphics Debug Desktop Window Help

Stack: Base No valid plots for TestTarg...

TestTargets <4x120 double>

	1	2	3	4	5	6	7	8	9	10	11	12
1	1	1	1	1	1	1	1	1	1	1	1	1
2	0	0	0	0	0	0	0	0	0	0	0	0
3	0	0	0	0	0	0	0	0	0	0	0	0
4	0	0	0	0	0	0	0	0	0	0	0	0
5												
6												
7												
8												
9												
10												
11												
12												
13												
14												
15												

รูปที่ 3.10 ค่าเป้าหมายสำหรับการทดสอบ

3.2 ขั้นตอนการออกแบบและพัฒนาเว็บแอปพลิเคชัน

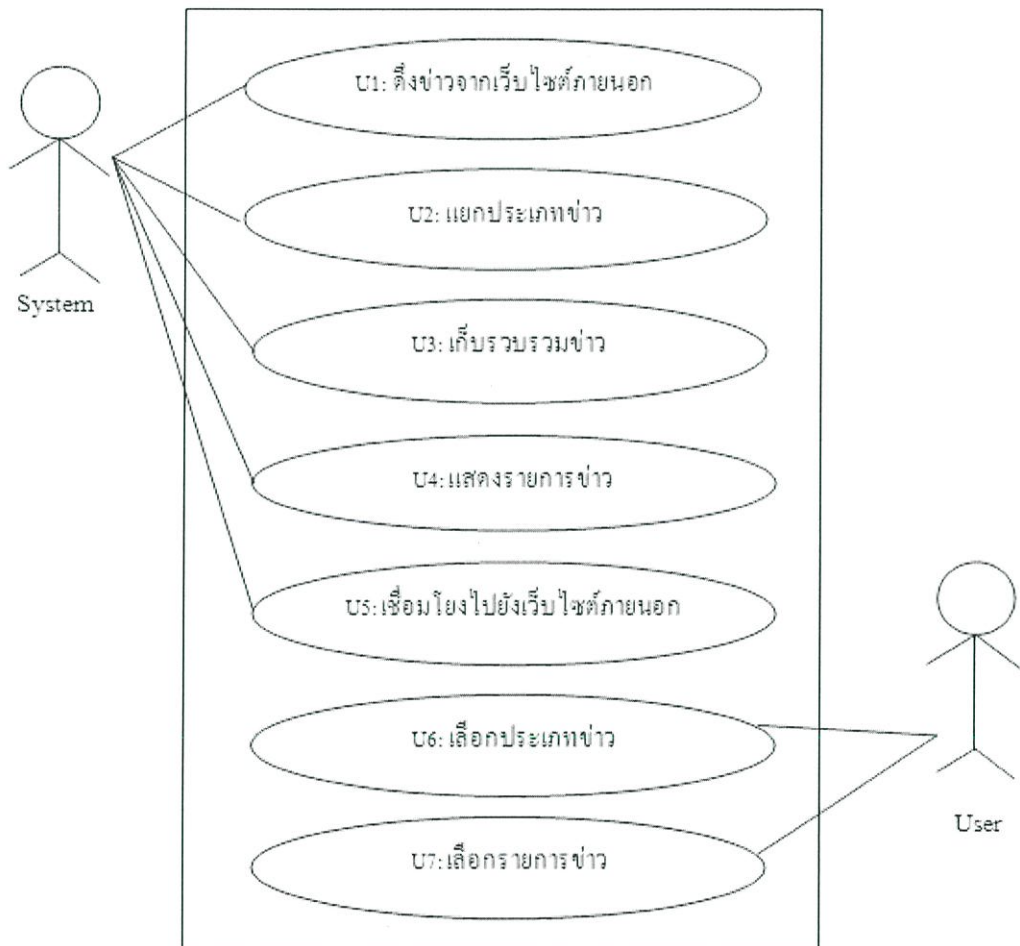
3.2.1 การวิเคราะห์ระบบ

เว็บแอปพลิเคชันนี้ได้พัฒนาขึ้นมาเพื่อรวบรวมข่าว 4 ประเภทคือ ข่าวกีฬา ข่าวการเมือง ข่าวบันเทิง และข่าวไอที เพื่อให้ผู้ใช้สะดวกในการค้นหาข่าวแบบเฉพาะเจาะจง

3.2.1.1 ความต้องการของระบบ

- ระบบสามารถแยกประเภทข่าวได้
- ระบบสามารถแสดงรายการข่าวผ่านหน้าเว็บแอปพลิเคชัน

3.2.1.2 Use Case Diagram



รูปที่ 3.11 Use Case Diagram ของระบบ

3.2.1.3 คำอธิบาย Use Case

ตารางที่ 3.4 คำอธิบายของ U1: ดึงข่าวจากเว็บไซต์ภายนอก

Use Case Name :	ดึงข่าวจากเว็บไซต์ภายนอก	
Scenario :	ระบบทำการดึงข่าวจากเว็บไซต์ภายนอก	
Trigger Event :	มีข่าวใหม่อัพเดท	
Brief Description :	เมื่อมีข่าวใหม่อัพเดท ระบบทำการดึงข่าวจากเว็บไซต์ภายนอก	
Actor :	ระบบ	
Related Use Case :	U2	
Stakeholders :	ระบบ	
Precondition :	-	
Post condition :	แยกประเภทข่าว	
Flow of Activity :	Actor	System
		1.ดึงข่าวจากเว็บไซต์ภายนอก
Exception Condition	-	

ตารางที่ 3.5 คำอธิบายของ U2: แยกประเภทข่าว

Use Case Name :	แยกประเภทข่าว	
Scenario :	ระบบทำการแยกประเภทข่าว	
Trigger Event :	เมื่อมีการดึงข่าวจากเว็บไซต์ภายนอก	
Brief Description :	ระบบทำการแยกประเภทข่าวออกเป็น 4 ประเภท คือ ข่าวกีฬา ข่าวไอที ข่าวบันเทิง และข่าวการเมือง	
Actor :	ระบบ	
Related Use Case :	U1, U3	
Stakeholders :	ระบบ	
Precondition :	ทำการดึงข่าวจากเว็บไซต์ภายนอก	
Post condition :	เก็บข่าวแต่ละประเภทลงในฐานข้อมูล	
Flow of Activity :	Actor	System
		1.แยกประเภทข่าว
Exception Condition	-	

ตารางที่ 3.6 คำอธิบายของ U3: เก็บรวบรวมข่าว

Use Case Name :	เก็บรวบรวมข่าว	
Scenario :	ระบบรวบรวมข่าวแล้วเก็บลงในฐานข้อมูล	
Trigger Event :	เมื่อมีการแยกประเภทข่าว	
Brief Description :	ระบบทำการรวบรวมข่าวที่ได้จากการแยกประเภท จากนั้นเก็บข่าวแต่ละประเภทลงในฐานข้อมูล	
Actor :	ระบบ	
Related Use Case :	U2	
Stakeholders :	ระบบ	
Precondition :	แยกประเภทข่าว	
Post condition :	-	
Flow of Activity :	Actor	System
		1.เก็บข่าวแต่ละประเภทลงในฐานข้อมูล
Exception Condition	-	

ตารางที่ 3.7 คำอธิบายของ U4: แสดงรายการข่าว

Use Case Name :	แสดงรายการข่าว	
Scenario :	แสดงรายการข่าวตามที่ผู้ใช้งานร้องขอ	
Trigger Event :	ผู้ใช้เลือกประเภทข่าวที่ต้องการ	
Brief Description :	ระบบทำการแสดงรายการข่าวตามที่ผู้ใช้งานร้องขอ	
Actor :	ระบบ	
Related Use Case :	U3, U6	
Stakeholders :	ระบบ	
Precondition :	ผู้ใช้ทำการเลือกประเภทข่าว	
Post condition :	เชื่อมโยงไปยังเว็บไซต์ภายนอก	
Flow of Activity :	Actor	System
	1. เลือกประเภทข่าว	2. แสดงรายการข่าว
	3. เลือกรายการข่าว	
Exception Condition	-	

ตารางที่ 3.8 คำอธิบายของ U5: เชื่อมโยงไปยังเว็บไซต์ภายนอก

Use Case Name :	เชื่อมโยงไปยังเว็บไซต์ภายนอก	
Scenario :	ระบบทำการเชื่อมโยงไปยังเว็บไซต์ภายนอก	
Trigger Event :	เมื่อมีการเลือกประเภทข่าว	
Brief Description :	ระบบทำการเชื่อมโยงไปยังเว็บไซต์ภายนอกเพื่อแสดงเนื้อหาข่าว	
Actor :	ระบบ	
Related Use Case :	U6	
Stakeholders :	ระบบ	
Precondition :	ผู้ใช้ทำการเลือกรายการข่าว	
Post condition :	-	
Flow of Activity :	Actor	System
	1. เลือกรายการข่าว	2. เชื่อมโยงไปยังเว็บไซต์ภายนอก
	3. อ่านข่าว	
Exception Condition	-	

ตารางที่ 3.9 คำอธิบายของ U6: เลือกประเภทข่าว

Use Case Name :	เลือกประเภทข่าว	
Scenario :	เลือกประเภทข่าวจากหน้าเว็บแอปพลิเคชัน	
Trigger Event :	เมื่อต้องการเลือกประเภทข่าวที่สนใจ	
Brief Description :	ผู้ใช้เลือกข่าวที่ต้องการจาก 4 ประเภท คือ ข่าวไอที ข่าวบันเทิง ข่าวกีฬา และข่าวการเมือง จากหน้าเว็บไซต์	
Actor :	ผู้ใช้งาน	
Related Use Case :	-	
Stakeholders :	ผู้ใช้งาน	
Precondition :	-	
Post condition :	แสดงรายการข่าว	
Flow of Activity :	Actor	System
	1. เข้ามาที่หน้าเว็บแอปพลิเคชัน	3. แสดงรายการข่าว
	2. เลือกประเภทข่าวที่ต้องการ	
Exception Condition	-	

ตารางที่ 3.10 คำอธิบายของ U7: เลือกรายการข่าว

Use Case Name :	เลือกรายการข่าว	
Scenario :	ผู้ใช้งานทำการเลือกรายการข่าว	
Trigger Event :	เมื่อต้องการอ่านรายการข่าวที่สนใจ	
Brief Description :	ผู้ใช้งานทำการเลือกรายการข่าวจากที่แสดงในหน้าเว็บแอปพลิเคชัน	
Actor :	ผู้ใช้งาน	
Related Use Case :	U4	
Stakeholders :	ผู้ใช้งาน	
Precondition :	ผู้ใช้งานต้องทำการเลือกประเภทข่าวก่อน	
Post condition :	เชื่อมต่อไปยังเว็บไซต์ภายนอก	
Flow of Activity :	Actor	System
	1. เลือกรายการข่าว	2. เชื่อมต่อไปยังเว็บไซต์ภายนอก
	3. อ่านข่าว	
Exception Condition	-	

3.2.1.4 ตารางของระบบฐานข้อมูล

ชื่อตาราง : News

คีย์หลัก : NewsId

ตารางที่ 3.11 ตาราง News

ลำดับ	ชื่อคอลัมน์	ชนิดข้อมูล	ความหมาย	ประเภทคีย์
1.	NewsId	int	รหัสข่าว	Primary Key
2.	NewsTitle	nvarchar(max)	หัวข้อข่าว	
3.	NewsLink	nvarchar(max)	ลิงค์ข่าว	
4.	NewsDate	date	วันที่	
5.	NewsCategory	int	ประเภทข่าว	

3.2.2 การออกแบบเว็บแอปพลิเคชัน

สำหรับการออกแบบเว็บแอปพลิเคชันสำหรับจำแนกกลุ่มเอกสารบนเว็บนี้มีการออกแบบ 2 ส่วนหลัก คือ ส่วนของหน้าเว็บแอปพลิเคชันและส่วนการติดต่อผู้ใช้



รูปที่ 3.12 โครงสร้างโดยรวมของเว็บแอปพลิเคชัน

3.2.2.1 ส่วนของเว็บแอปพลิเคชัน

การทำงานของเว็บแอปพลิเคชัน ที่มีหน้าแสดงผลข่าวทั้งหมด 4 ประเภท คือ ข่าวกีฬา ข่าวบันเทิง ข่าวการเมือง และข่าวไอที มีขั้นตอนการทำงานดังนี้

- 1) เมื่อผู้ใช้เลือกประเภทข่าวที่ต้องการ ระบบจะทำการดึงรายการข่าวจากฐานข้อมูลมาแสดงตามประเภทที่ผู้ใช้ได้ทำการเลือกไป
- 2) เมื่อผู้ใช้เลือกรายการข่าวที่สนใจ ระบบจะทำการเชื่อมโยงไปยังเว็บไซต์ภายนอกเพื่อแสดงรายละเอียดและเนื้อหาข่าว

3.2.2.2 ส่วนติดต่อกับผู้ใช้

ส่วนติดต่อกับผู้ใช้ คือ ส่วนของเว็บแอปพลิเคชันที่ให้ผู้ใช้งานหรือเลือกส่วนที่ต้องการได้ ซึ่งผู้ใช้งานสามารถใช้งานเว็บแอปพลิเคชันได้ ดังนี้

- 1) เริ่มจากที่ผู้ใช้ เลือกประเภทข่าวที่ต้องการจากทั้ง 4 ประเภท คือ ข่าวกีฬา ข่าวบันเทิง ข่าวการเมือง และข่าวไอที
- 2) จากนั้นผู้ใช้งานสามารถเลือกรายการข่าวที่สนใจได้จากรายการที่แสดง

บทที่ 4

ผลการดำเนินงาน

เว็บแอปพลิเคชันสำหรับจำแนกกลุ่มเอกสารบนเว็บนี้ จะใช้ข้อมูลข่าวที่ได้รวบรวมจากเว็บไซต์ข่าวต่างประเทศ จากนั้นนำข้อมูลที่ได้มาสร้างแบบจำลอง เพื่อคัดแยกประเภทข่าวออกเป็น 4 ประเภทคือ ข่าวไอที ข่าวบันเทิง ข่าวการเมือง และข่าวกีฬา โดยใช้เทคนิคโครงข่ายประสาทเทียม ซึ่งจะนำผลที่ได้จากแบบจำลองนี้มาพัฒนาเว็บแอปพลิเคชันต่อไป

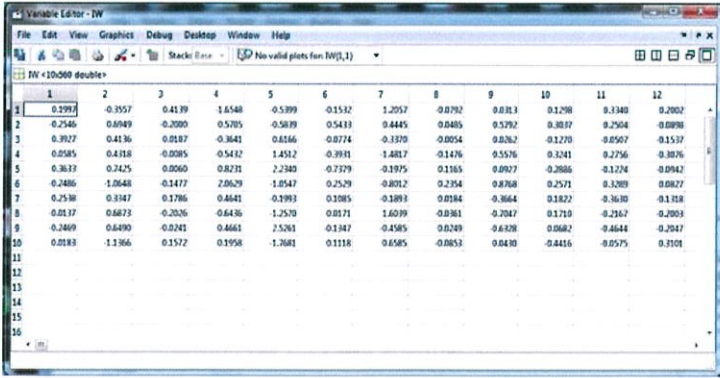
โดยผลการดำเนินงานแบ่งออกเป็นสองส่วนคือ

- 4.1 ผลการดำเนินงานในส่วนของการสร้างแบบจำลอง
- 4.2 ผลการดำเนินงานในส่วนของการสร้างเว็บแอปพลิเคชัน

4.1 ผลการดำเนินงานในส่วนของการสร้างแบบจำลอง

4.1.1 ผลการดำเนินงานจากการฝึกสอนโครงข่าย (Training Process)

ผลลัพธ์ของการฝึกสอนโครงข่ายคือ ค่าน้ำหนักระหว่างชั้นซ่อนกับชั้นเอาต์พุต ค่าน้ำหนักระหว่างชั้นซ่อนกับชั้นอินพุตและค่าไบแอสในชั้นซ่อนและชั้นเอาต์พุต



	1	2	3	4	5	6	7	8	9	10	11	12
1	0.2997	-0.2957	0.4139	1.6548	-0.5399	-0.1512	1.2957	-0.0792	0.0113	0.1298	0.2340	0.2902
2	-0.2546	0.0949	-0.2090	0.5705	-0.5879	0.5411	0.4445	0.0405	0.5792	0.3837	0.2504	-0.0908
3	0.3927	0.4136	0.0187	-0.3641	0.6166	-0.0774	-0.3370	-0.0054	0.0362	-0.1270	-0.0507	-0.1537
4	0.0585	0.4118	-0.0085	-0.5432	1.4512	-0.3911	1.4817	-0.1476	0.5576	0.3241	0.2756	0.3076
5	0.3633	0.2425	0.0060	0.8271	2.2340	-0.7379	-0.1975	0.1165	0.0927	-0.2086	-0.1274	-0.0942
6	-0.2486	1.0648	-0.1477	2.0629	1.0547	0.2529	-0.0012	0.2354	0.8768	0.2571	0.1209	0.0827
7	0.2538	0.3347	0.1786	0.4641	-0.1993	0.3085	-0.1893	0.0184	-0.8664	0.1822	-0.3630	-0.1118
8	0.0137	0.6873	-0.2026	0.6436	1.2570	0.0171	1.6039	-0.0361	-0.2047	0.1710	-0.2367	-0.2003
9	-0.2409	0.4490	-0.0241	0.4661	2.5261	0.1347	0.4585	0.0249	0.6108	0.0682	-0.4644	0.2047
10	0.0183	1.1366	0.1572	0.1958	-1.7081	0.1118	0.4585	-0.0853	0.6430	-0.4416	-0.0575	0.2101
11												
12												
13												
14												
15												
16												

รูปที่ 4.1 ค่าน้ำหนักระหว่างชั้นซ่อนกับชั้นอินพุต

Variable Editor - LW

File Edit View Graphics Debug Desktop Window Help

Stacks: Base - No valid plots for LW(1,1)

LW <4x12 double>

	1	2	3	4	5	6	7	8	9	10	11	12
1	3.0453	-1.4047	0.5855	6.0756	-5.0679	6.6324	-6.3382	-2.6644	-6.1260	3.1299		
2	-9.1364	1.0234	1.1548	-5.2356	-0.9411	5.2061	4.6992	-0.9139	0.2989	2.9814		
3	-1.3842	-5.1301	1.5400	6.4273	2.9790	-6.2996	2.1654	-4.5496	6.8326	-2.8645		
4	4.7048	-1.5085	0.5296	-5.6064	-4.5271	-7.4738	1.8217	8.0454	-2.0998	1.0019		

รูปที่ 4.2 คำนวณหาค่าระหว่างชั้นซ้อนกับชั้นเอทท์พูด

Variable Editor - b1

File Edit View Graphics Debug Desktop Window Help

Stacks: Base - No valid plots for b1(1,1)

b1 <10x12 double>

	1	2	3	4	5	6	7	8	9	10	11	12
1	-0.1729											
2	-0.6265											
3	-1.0121											
4	-0.3085											
5	-1.5704											
6	0.0510											
7	0.0208											
8	-0.0982											
9	-2.0559											
10	2.0015											

รูปที่ 4.3 ค่าไบแอสในชั้นซ้อน

Variable Editor - b2

File Edit View Graphics Debug Desktop Window Help

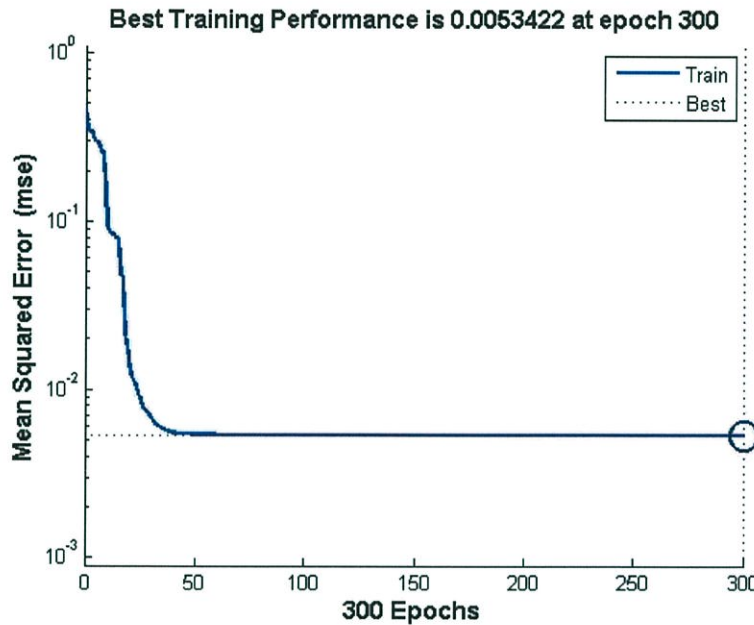
Stacks: Base - No valid plots for b2(1,1)

b2 <4x12 double>

	1	2	3	4	5	6	7	8	9	10	11	12
1	-4.2199											
2	-1.0870											
3	-2.8158											
4	-1.7477											

รูปที่ 4.4 ค่าไบแอสในชั้นเอทท์พูด

4.1.2 ผลการดำเนินงานจากการทดสอบโครงข่าย (Testing Process)



รูปที่ 4.5 ความสัมพันธ์ระหว่างค่า MSE กับค่า Epochs

ค่าความผิดพลาดเฉลี่ย (Mean Squared Error: MSE) ถ้าค่าความผิดพลาดเฉลี่ยมีค่าน้อยกว่าค่าที่ยอมรับได้ ให้จบการเรียนรู้ แต่ถ้าค่าความผิดพลาดเฉลี่ยมากกว่าค่าที่ยอมรับได้ให้ตรวจสอบว่าได้ทำการเรียนรู้ครบตามจำนวนรอบที่กำหนดไว้หรือไม่ ถ้าครบแล้วให้จบการเรียนรู้ ถ้ายังไม่ครบให้กลับไปเริ่มขั้นตอนการเรียนรู้ใหม่

จากรูปที่ 4.5 แสดงให้เห็นว่า ค่าความผิดพลาดเฉลี่ย (MSE) แปรผกผันกับจำนวนรอบในการเรียนรู้ (Epochs) คือ ค่าความผิดพลาดเฉลี่ยจะลดลงเมื่อจำนวนรอบในการเรียนรู้เพิ่มขึ้น ถ้าค่าความผิดพลาดเฉลี่ยเข้าใกล้ศูนย์มากแสดงว่าเกิดความผิดพลาดน้อย ในที่นี้มีค่าความผิดพลาดเฉลี่ยเป็น 0.0027779

ตารางที่ 4.1 Confusion Matrix ของแบบจำลอง

Output	Target			
	Ent.	IT	Sport	Politic
Ent.	23	2	2	1
IT	3	28	0	0
Sport	0	0	26	1
Politic	4	0	2	28
รวม	30	30	30	30

จากตารางที่ 4.1 แสดงผลการทำนายชุดข้อมูลทดสอบ ซึ่งสามารถทำนายเป็นข่าวบันเทิงได้ถูกต้องเป็นจำนวน 23 ข่าวจากทั้งหมด 30 ข่าว ทำนายเป็นข่าวไอทีได้ถูกต้องเป็นจำนวน 28 ข่าวจากทั้งหมด 30 ข่าว ทำนายเป็นข่าวกีฬาได้ถูกต้องเป็นจำนวน 26 ข่าว จากทั้งหมด 30 ข่าว และทำนายเป็นข่าวการเมืองได้ถูกต้องเป็นจำนวน 28 ข่าว จากทั้งหมด 30 ข่าว

ตารางที่ 4.2 การวัดประสิทธิภาพการทำงานของแบบจำลอง

Output	Precisions	Recall	F-Measure
Ent.	$23/(23+2+2=1) \times 100$ = 82.1%	$23/30 \times 100$ = 76.7%	$2(82.1 \times 76.7)/(82.1+76.7)$ = 79.31%
IT	$28/(28+3) \times 100$ = 90.3%	$28/30 \times 100$ = 93.3%	$2(90.3 \times 93.3)/(90.3+93.3)$ = 91.78%
Sport	$26/(26+1) \times 100$ = 96.3%	$26/30 \times 100$ = 86.7%	$2(96.3 \times 86.7)/(96.3+86.7)$ = 91.25%
Politic	$28/(28+4+2) \times 100$ = 82.4%	$28/30 \times 100$ = 93.3%	$2(82.4 \times 93.3)/(82.4+93.3)$ = 87.51%

จากตารางที่ 4.2 เป็นการวัดประสิทธิภาพของแบบจำลองโดยค่าที่ใช้ในการวัดประสิทธิภาพมีทั้งหมด 4 ค่าคือค่าความแม่นยำ (Accuracy) ค่าความเที่ยง (Precision) ค่าความระลึก (Recall) และค่าถ่วงดุล (Tradeoff)

จาก Confusion Matrix สามารถคำนวณหาค่า Accuracy ได้ดังนี้

$$\begin{aligned} \text{Accuracy} &= \frac{23+28+26+28}{30+30+30+30} \\ &= 87.5\% \end{aligned}$$

สรุปได้ว่า

ในคลาสที่ 1	ข่าวบันเทิง	มีค่า Precisions 82.1%	ค่า Recall 76.7%	ค่า F-Measure 79.31%
ในคลาสที่ 2	ข่าวไอที	มีค่า Precisions 90.3%	ค่า Recall 93.3%	ค่า F-Measure 91.78%
ในคลาสที่ 3	ข่าวกีฬา	มีค่า Precisions 96.3%	ค่า Recall 86.7%	ค่า F-Measure 91.25%
ในคลาสที่ 4	ข่าวการเมือง	มีค่า Precisions 82.4%	ค่า Recall 93.3%	ค่า F-Measure 87.51%

แบบจำลองนี้มีค่าความแม่นยำ 87.5% ซึ่งเป็นความแม่นยำที่ยอมรับได้และสามารถนำไปพัฒนาในขั้นต่อไป

4.2 ผลการดำเนินงานในส่วนของการสร้างเว็บแอปพลิเคชัน

ผู้ใช้สามารถเลือกดูข่าวที่ต้องการได้จากหน้าเว็บแอปพลิเคชัน โดยจะแยกเป็นประเภทข่าวในแต่ละหน้า รวมทั้งหมด 4 หน้า คือ หน้าของข่าวไอที ข่าวบันเทิง ข่าวการเมืองและข่าวกีฬาซึ่งในแต่ละหน้าจะแสดงลิงค์ข่าวที่ได้จากการรวบรวมข่าวจากเว็บไซต์ 4 เว็บไซต์ ดังนี้ <http://news-briefs.ew.com/>, <http://sportsnewsinternational.com>, <http://blogs.telegraph.co.uk/news> และ <http://extremetech.com>

4.2.1 ความสามารถในการเข้าใช้ระบบของผู้ใช้

- สามารถเลือกประเภทข่าวที่ต้องการได้ทั้งหมด 4 ประเภท คือ ข่าวกีฬา ข่าวบันเทิง ข่าวการเมือง และข่าวไอที
- สามารถเลือกหัวข้อข่าวที่สนใจได้จากรายการลิงค์ที่แสดงในหน้าเว็บไซต์

1. หน้าแรกของเว็บแอปพลิเคชัน

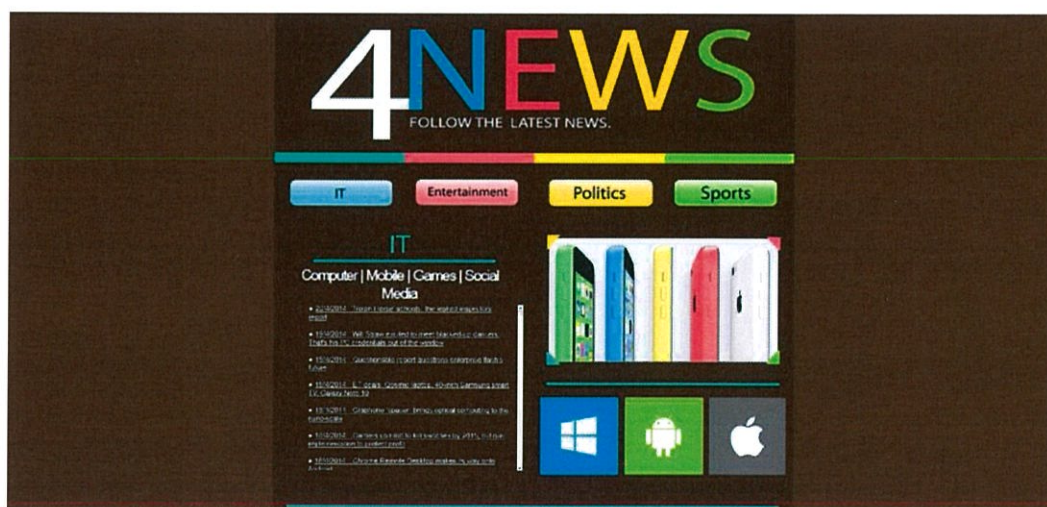
ในหน้านี้จะมีให้เลือกประเภทข่าวทั้งหมด 4 ประเภท คือ ข่าวไอที ข่าวบันเทิง ข่าวการเมือง และข่าวกีฬา



รูปที่ 4.6 หน้าแรกของเว็บแอปพลิเคชัน

2. หน้าแสดงลิงค์ข่าวไอที

ในหน้านี้จะแสดงลิงค์ข่าวไอที



รูปที่ 4.7 หน้าหลักของข่าวไอที

3. หน้าแสดงลิงค์ข่าวบันเทิง

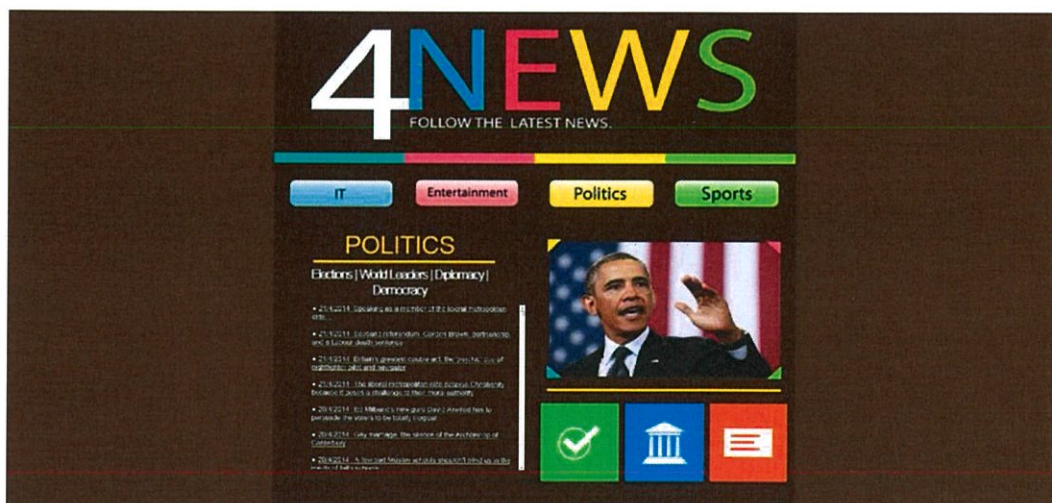
ในหน้านี้จะแสดงลิงค์ข่าวบันเทิง



รูปที่ 4.8 หน้าหลักของข่าวบันเทิง

4. หน้าแสดงลิงค์ข่าวการเมือง

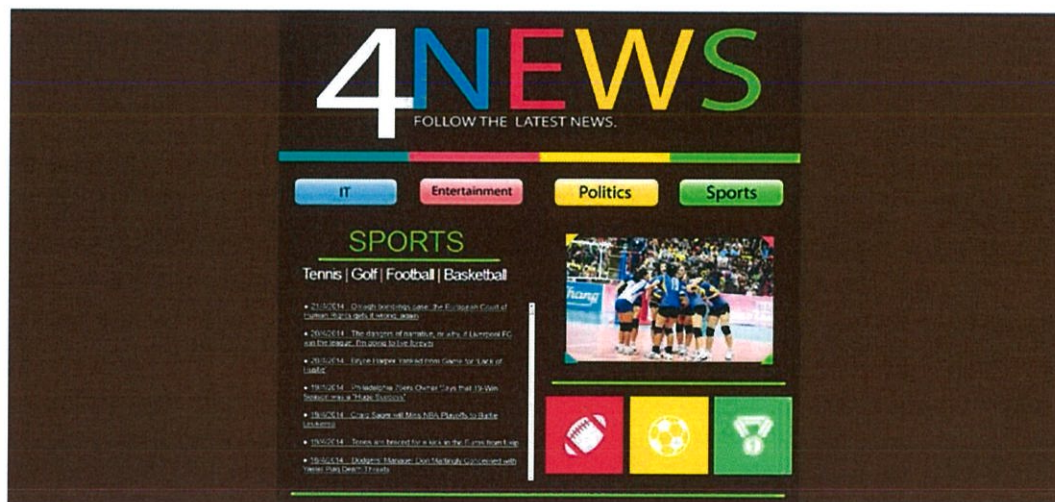
ในหน้านี้จะแสดงลิงค์ข่าวการเมือง



รูปที่ 4.9 หน้าหลักของข่าวการเมือง

5. หน้าแสดงลิงค์ข่าวกีฬา

ในหน้านี้จะแสดงลิงค์ข่าวการกีฬา



รูปที่ 4.10 หน้าหลักของข่าวกีฬา

บทที่ 5

สรุปผลการดำเนินงานและข้อเสนอแนะ

5.1 สรุปผลการดำเนินงาน

จากการศึกษาและพัฒนาเว็บแอปพลิเคชันอัจฉริยะสำหรับการจำแนกกลุ่มเอกสารบนเว็บ โดยใช้ตัวแทนอัจฉริยะ ทางคณะผู้จัดทำได้พิจารณาว่า เนื่องจากในปัจจุบันมีเว็บไซต์ที่ให้บริการรวบรวมข่าวสารเกิดขึ้นเป็นจำนวนมาก ซึ่งมีข้อมูลเอกสารทั้งที่เป็นประโยชน์และไม่เป็นประโยชน์ กระจายอยู่ตามเว็บไซต์ต่างๆ เริ่มต้นจากการศึกษาและรวบรวมข้อมูลต่างๆ โดยการรวบรวมข่าวทั้งหมด 4 ประเภท คือ ข่าวไอที ข่าวบันเทิง ข่าวการเมือง และข่าวกีฬา จากเว็บไซต์ข่าวต่างประเทศโดยใช้จำนวนกลุ่มข่าวตัวอย่างทั้งหมด 360 เอกสาร แบ่งเป็นกลุ่มข่าวตัวอย่างเรียนรู้ (Training) จำนวน 240 เอกสารและกลุ่มทดสอบ (Testing) จำนวน 120 เอกสารจากนั้นนำข้อมูลข่าวนั้นมาสร้างเป็นแบบจำลองโดยใช้โครงข่ายประสาทเทียมแบบ Multilayer Perceptrons (MLP) สำหรับการจำแนกประเภทข่าวจากนั้น ทำการคำนวณค่าความผิดพลาดของแบบจำลองที่ได้ทำการพัฒนาเพื่อหาแบบจำลองที่ดีที่สุดหรือมีค่าความผิดพลาดน้อยที่สุดเพื่อนำมาใช้ในการพัฒนาเว็บแอปพลิเคชันต่อไป

5.1.1 สรุปผลการทำนายโดยใช้แบบจำลองโครงข่ายประสาทเทียม

จากการรวบรวมข่าวจากหลากหลายเว็บไซต์ เพื่อนำมาเป็นข้อมูลทดสอบของแบบจำลองโครงข่ายประสาทเทียม ซึ่งสามารถวัดประสิทธิภาพการทำงานได้ดังนี้

- คลาสที่ 1 คือ ข่าวบันเทิง มีค่า Precisions 82.1% ค่า Recall 76.7% ค่า F-Measure 79.31%
- คลาสที่ 2 คือ ข่าวไอที มีค่า Precisions 90.3% ค่า Recall 93.3% ค่า F-Measure 91.78%
- คลาสที่ 3 คือ ข่าวกีฬา มีค่า Precisions 96.3% ค่า Recall 86.7% ค่า F-Measure 91.25%
- คลาสที่ 4 คือ ข่าวการเมือง มีค่า Precisions 82.4% ค่า Recall 93.3% ค่า F-Measure 87.51%

5.1.2 สรุปผลการพัฒนาเว็บแอปพลิเคชัน

จากการพัฒนาระบบสามารถแก้ไขปัญหาได้ตามการวิเคราะห์และออกแบบ คือ ผู้ใช้สามารถเลือกประเภทข่าวที่ต้องการจากทั้ง 4 ประเภท คือ ข่าวกีฬา ข่าวบันเทิง ข่าวการเมือง และข่าวไอที โดยได้จัดทำเป็นเว็บไซต์จำนวน 4 หน้า แยกเป็นหน้าหลักของข่าวแต่ละประเภท ซึ่งในแต่ละหน้านั้นได้ทำการจัดเรียงลิงค์ข่าวแต่ละประเภทไว้ ซึ่งผู้ใช้สามารถเลือกหัวข้อข่าวที่สนใจได้จากลิงค์รายการที่แสดง ทำให้ผู้ใช้เลือกดูข่าวได้สะดวกและรวดเร็ว

ตารางที่ 5.1 Confusion Matrix ของเว็บแอปพลิเคชัน

Output	Target			
	Ent.	IT	Sport	Politic
Ent.	145	3	66	11
IT	7	191	3	6
Sport	2	0	204	10
Politic	56	16	27	183
รวม	210	210	300	210

จากตารางที่ 5.1 แสดงผลการแยกประเภทข่าวของเว็บแอปพลิเคชัน ที่เอเจนต์ได้ทำการดึงและแยกประเภทข่าวมาจากเว็บไซต์ 4 เว็บไซต์ ดังนี้ <http://sportsnewsinternational.com>, <http://blogs.telegraph.co.uk/news>, <http://extremetech.com> และ <http://news-briefs.ew.com/> ซึ่งสามารถทำนายเป็นข่าวนบันเทิงได้ถูกต้องเป็นจำนวน 145 ข่าวจากทั้งหมด 210 ข่าว ทำนายเป็นข่าวไอทีได้ถูกต้องเป็นจำนวน 191 ข่าว จากทั้งหมด 210 ข่าว ทำนายเป็นข่าวกีฬาได้ถูกต้องเป็นจำนวน 204 ข่าว จากทั้งหมด 300 ข่าว และ ทำนายเป็นข่าวการเมืองได้ถูกต้องเป็นจำนวน 183 ข่าว จากทั้งหมด 210 ข่าว

ตารางที่ 5.2 การวัดประสิทธิภาพการทำงานของเว็บแอปพลิเคชัน

Output	Precision	Recall	F-Measure
Ent.	64.4%	69.1%	66.7%
IT	92.3%	91%	91.6%
Sport	94.5%	68%	79.1%
Politic	64.9%	87.1%	74.4%
Average	79.0%	78.8%	78%

ตารางที่ 5.2 เป็นการวัดประสิทธิภาพการทำงานของเว็บแอปพลิเคชัน ได้ผลดังนี้

- คลาสที่ 1 คือ ข่าวบันเทิง มีค่า Precisions 64.4% ค่า Recall 69.1% ค่า F-Measure 66.7%
- คลาสที่ 2 คือ ข่าวไอที มีค่า Precisions 92.3% ค่า Recall 91% ค่า F-Measure 91.6%
- คลาสที่ 3 คือ ข่าวกีฬา มีค่า Precisions 94.5% ค่า Recall 68% ค่า F-Measure 79.1%
- คลาสที่ 4 คือ ข่าวการเมือง มีค่า Precisions 64.9% ค่า Recall 87.1% ค่า F-Measure 74.4%

5.2 ปัญหาที่พบในการดำเนินงาน

ปัญหาที่เกิดจากการหาศัพท์เวิร์ดในงานวิจัยนี้ ใช้วิธีการนับจำนวนคำที่เกิดขึ้นบ่อยในแต่ละข่าว ซึ่งวิธีการนี้ยังไม่สามารถหาศัพท์เวิร์ดที่เป็นตัวแทนของเนื้อหาข่าวนั้นอย่างแท้จริงได้ นอกจากนี้ยังพบปัญหาการทับซ้อนกันของข่าว เนื่องจากมีข่าวบางข่าวที่สามารถจัดอยู่ในหัวข้อข่าวได้มากกว่าหนึ่งหัวข้อ เมื่อเป็นเช่นนี้แล้ว จึงทำให้ผลของการจัดกลุ่มด้วยแบบจำลองเกิดความผิดพลาด

5.3 ข้อเสนอแนะ

จากการพัฒนาเว็บแอปพลิเคชันอัจฉริยะสำหรับการจำแนกกลุ่มเอกสารบนเว็บโดยใช้ตัวแทนอัจฉริยะนั้น สามารถดึงถึงคำแต่ละประเภทได้ตามที่ต้องการ ซึ่งทางคณะผู้จัดทำเห็นว่าควรมีการพัฒนาเว็บแอปพลิเคชันนี้เพิ่มเติมในอนาคต เช่น การเพิ่มประเภทของข่าวให้มากขึ้น และการพัฒนาหน้าเว็บแอปพลิเคชันให้สามารถรองรับการแสดงผลได้บนหลากหลายอุปกรณ์ เช่น สมาร์ทโฟน หรือ แท็บเล็ต เป็นต้น

เอกสารอ้างอิง

- นวลวรรณ สุนทรภิชัย. 2552. **ปัญญาประดิษฐ์**. พิมพ์ครั้งที่ 2. กรุงเทพฯ: ภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์ มหาวิทยาลัยเกษตรศาสตร์
- กิตติ ภักดีวัฒนกุล และจำลอง ครูอุตสาหะ. 2550. **ระบบฐานข้อมูล (Database Systems)**. พิมพ์ครั้งที่ 9. กรุงเทพฯ : เคทีพี คอมพ์ แอนด์ คอนซัลท์.
- ณัฐพงษ์ วารีประเสริฐ และณรงค์ คำดี. 2552. **ปัญญาประดิษฐ์ (Artificial Intelligence)**. กรุงเทพฯ : เคทีพี คอมพ์ แอนด์ คอนซัลท์.
- ผศ.ดร.ชนวัฒน์ ศรีสอ้าน. 2550. **ฐานข้อมูล คลังข้อมูลและเหมืองข้อมูล**. ปทุมธานี : มหาวิทยาลัยรังสิต.
- ดวงพร ขอเจริญพร. 2546. **เขียนโปรแกรม Java บน Web ด้วย Servlets และ JSP**. กรุงเทพฯ : เคทีพี คอมพ์ แอนด์ คอนซัลท์.
- กริฐ อัสวคุปตานนท์. 2555. **ASP.NET & SQL Server by C# เรียนรู้สุดยอดเทคโนโลยีล่าสุดในการพัฒนา Web ขั้นสูง**. กรุงเทพฯ : เน็ตดีไซน์ พับลิชซิ่ง.
- สมคิด แสนเสนา. **การแบ่งกลุ่มเอกสารโดยใช้เทคนิคการประมวลผลข้อความด้วยโครงข่ายประสาทเทียมที่เรียนรู้แบบหาผู้ชนะ**. วิทยานิพนธ์วิทยาศาสตรมหาบัณฑิต. สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง, 2547.
- พงศ์สันต์ นุ่นสุข. **การพิสูจน์ลายเซ็นแบบออนไลน์โดยใช้โครงข่ายประสาทเทียม**. วิทยานิพนธ์วิศวกรรมศาสตรมหาบัณฑิต. สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง, 2550.
- ธนทัสน์ ไทยชนวัฒน์ และ พยุง มีสัจ. **ระบบกรองอีเมลขยะด้วยโครงข่ายประสาทเทียม Spam Filtering Using Neural Network Algorithm**. [ออนไลน์]. ภาควิชาเทคโนโลยีสารสนเทศ คณะเทคโนโลยีสารสนเทศ และ ภาควิชาครุศาสตร์ไฟฟ้า คณะครุศาสตร์อุตสาหกรรม มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ. เข้าถึงได้จาก: 202.44.34.144/nccitedoc/admin/nccit_files/NCCIT-2011260257.pdf. (วันที่ค้นข้อมูล: 10 พฤศจิกายน 2556)

เอกสารอ้างอิง (ต่อ)

บุษรา ลิ้มพิพัฒนางกูร. การพยากรณ์โอกาสการสำเร็จการศึกษาของนักศึกษา โดยใช้จີเน็ตคัลอัลกอริทึมแบบหลายวัตถุประสงค์ เปรียบเทียบกับวิธีทางระบบโครงข่ายประสาทเทียม : กรณีศึกษานักศึกษาภาควิชาวิทยาการคอมพิวเตอร์และสารสนเทศ.[ออนไลน์]. วิทยานิพนธ์วิทยาศาสตรมหาบัณฑิต. กรุงเทพฯ : มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าพระนครเหนือ, 2549 เข้าถึงได้จาก:
www.gits.kmutnb.ac.th/ethesis/data/4640580090.pdf. (วันที่ค้นข้อมูล: 10 พฤศจิกายน 2556)

เอกวิทย์ หายักวงษ์ และ วิจิตร กิณเรศ. “การตรวจจับฮาร์มอนิกโดยการใช้โครงข่ายประสาทเทียมเพอร์เซ็ปตรอนหลายชั้น Harmonic Detection by Using Multilayer Perceptron Neural Network” วิศวกรรมลาดกระบัง คณะวิศวกรรมศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง. 22 (1): 65-70 [ออนไลน์]. เข้าถึงได้จาก:
www.kmitl.ac.th/lej/PDFjournal48/Volume22_No1_MAR2548.pdf. (วันที่ค้นข้อมูล: 10 พฤศจิกายน 2556)

เอกรินทร์ วัฒนัญเลิศสกุล. 2556. สร้าง Query สืบค้นข้อมูลด้วย SQL Server 2008. [ออนไลน์]. เข้าถึงได้จาก: <http://www.thai-library.org/Resource/0000006226.pdf> (วันที่ค้นข้อมูล: 20 พฤศจิกายน 2556)

เพชร เครือพานิช. ASP.NET MVC 3 กับ HTML5 ตอนที่ 1. [ออนไลน์]. เข้าถึงได้จาก:
https://www.microsoft.com/thailand/msdn/ASP.NET_MVC_Music_Store_pt3_02.aspx. (วันที่ค้นข้อมูล: 8 พฤศจิกายน 2556)

คุณากร ศิริคุปต์. ASP.NET MVC 3 Introduction. [ออนไลน์]. เข้าถึงได้จาก:
<http://joycerookie.blogspot.com/2013/05/aspnet-mvc-3-introduction.html>. (วันที่ค้นข้อมูล: 8 พฤศจิกายน 2556)

เอกสารอ้างอิง (ต่อ)

จารวี ฉันทสิทธิพร. การจำแนกยาเม็ดจากภาพถ่ายโดยใช้เทคนิคเครือข่ายประสาท. [ออนไลน์].

วิทยานิพนธ์วิทยาศาสตรมหาบัณฑิต. กรุงเทพฯ : มหาวิทยาลัยศิลปากร, 2548 เข้าถึงได้จาก:
www.thapra.lib.su.ac.th/objects/thesis/fulltext/snamcn/.../Fulltext.pdf. (วันที่ค้นข้อมูล: 10
พฤศจิกายน 2556)

ความหมายของ **Data Mining**. [ออนไลน์]. เข้าถึงได้จาก:

<http://dllibrary.spu.ac.th:8080/dspace/bitstream/123456789/96/2/%E0%B8%9A%E0%B8%97%E0%B8%97%E0%B8%B5%E0%B9%88%202.pdf>. (วันที่ค้นข้อมูล: 20 พฤศจิกายน 2556)

การทำเหมืองข้อมูล (**Data mining**). [ออนไลน์]. เข้าถึงได้จาก:

<http://www.science.cmru.ac.th/scienceblog/admin/blog/file/90112174716.pdf> .
(วันที่ค้นข้อมูล: 20 พฤศจิกายน 2556)

DATA Mining. [ออนไลน์]. เข้าถึงได้จาก:<http://alaska.reru.ac.th/text/datamining.pdf>. (วันที่ค้นข้อมูล: 20
พฤศจิกายน 2556)

การทำเหมืองข้อมูลและการสร้างเว็บแอปพลิเคชัน (**web application**). [ออนไลน์]. เข้าถึงได้จาก:

<http://km.cpc.ac.th/jintana/files/2012/10/%E0%B8%9A%E0%B8%97%E0%B8%97%E0%B8%B5%E0%B9%882.pdf>. (วันที่ค้นข้อมูล: 20 พฤศจิกายน 2556)

คู่มือปฏิบัติงานสำหรับผู้ดูแลระบบ (**Administrator**) หลักสูตร การใช้งานและบริหารฐานข้อมูล (**SQL Server 2008 R2**). [ออนไลน์]. เข้าถึงได้จาก:

http://plan.hss.moph.go.th/uploadFiles/document/D00000000395_23731.pdf (วันที่ค้นข้อมูล: 20
พฤศจิกายน 2556)

การทำเหมืองข้อมูล (**Data Mining**). [ออนไลน์]. เข้าถึงได้จาก:

http://compcenter.bu.ac.th/index.php?option=com_content&task=view&id=75&Itemid=172.
(วันที่ค้นข้อมูล: 20 พฤศจิกายน 2556)

เอกสารอ้างอิง (ต่อ)

ทฤษฎีการสกด. [ออนไลน์] เข้าถึงได้จาก:

research.dusit.ac.th/menu/abstra/abstract/full/sci/Amonrat.../ch2.pdf. (วันที่ค้นข้อมูล: 14 พฤศจิกายน 2556)

Artificial Neural Network โครงข่ายประสาทเทียม. [ออนไลน์]. เข้าถึงได้จาก:

<http://alaska.reru.ac.th/text/ann.pdf> (วันที่ค้นข้อมูล: 14 พฤศจิกายน 2556)

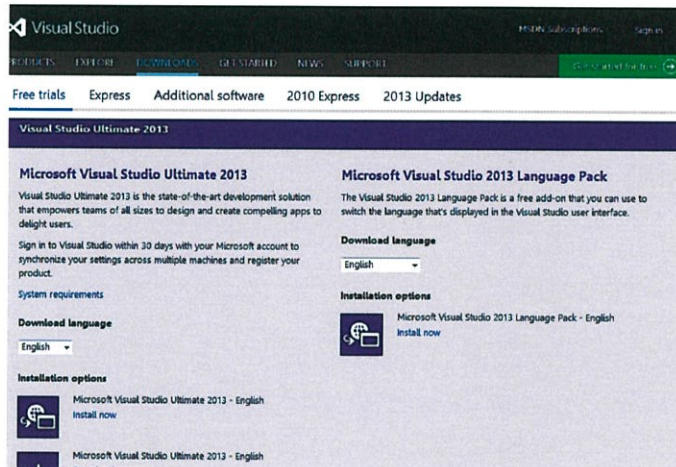
ภาคผนวก ก.

วิธีการติดตั้งโปรแกรม Visual Studio Ultimate 2013

ภาคผนวก ก.

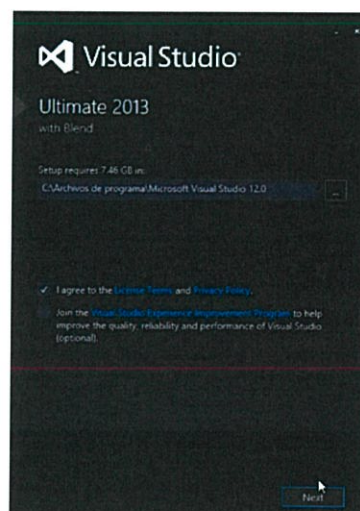
วิธีการติดตั้งโปรแกรม Visual Studio Ultimate 2013

1. ทำการดาวน์โหลด Visual Studio Ultimate 2013 (รองรับ windows 7 ขึ้นไป) ได้จาก <http://www.visualstudio.com/downloads/download-visual-studio-vs>



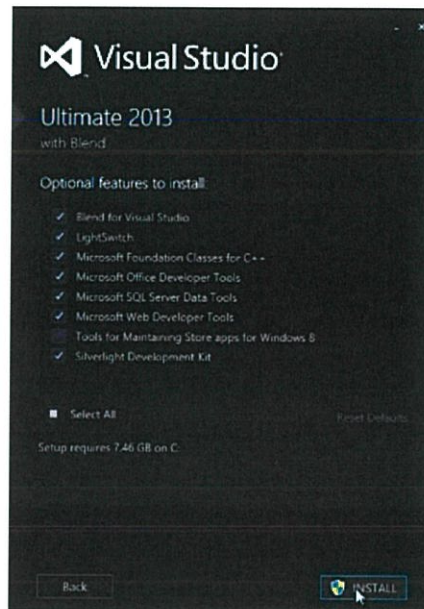
รูปที่ ก.1 หน้าเว็บไซต์ที่ทำการดาวน์โหลด

2. ทำการเปิดไฟล์ vs_ultimate.exe ที่ได้ทำการดาวน์โหลดมาจะปรากฏหน้าต่างดังรูปที่ โดยการสามารถเลือกไดเรกทอรีพาทสำหรับการติดตั้งได้ จากนั้นทำการเลือก I agree to the License Terms and Privacy Policy แล้วคลิก “Next”



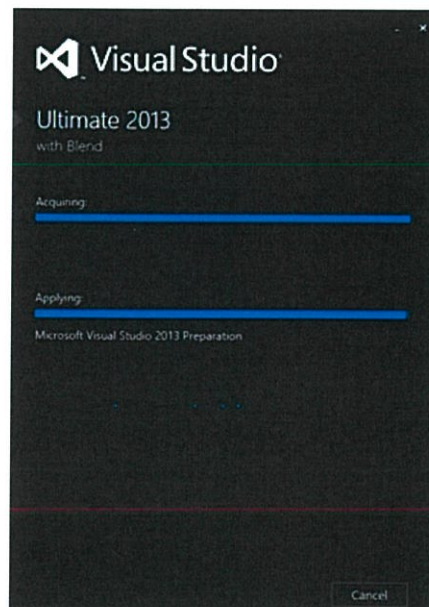
รูปที่ ก.2 หน้ารายละเอียด

3. หลังจากคลิก “Next” จะปรากฏหน้าต่างดังรูปที่ ให้ทำการเลือกคุณสมบัติที่ต้องการจากนั้นให้คลิก “INSTALL” เพื่อเริ่มต้นการติดตั้ง



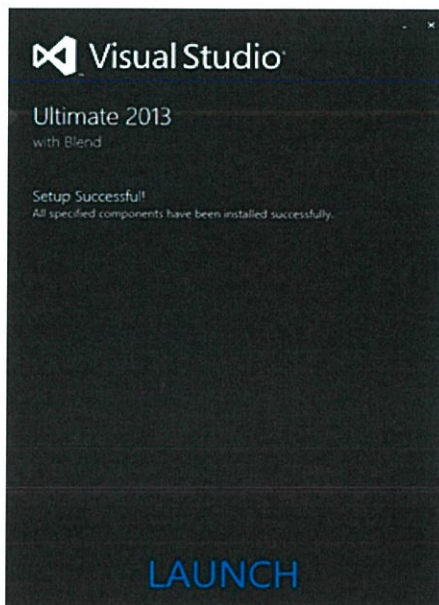
รูปที่ ก.3 คุณสมบัติของ Visual Studio ที่จะทำการติดตั้ง

4. หลังจากทำการคลิก “INSTALL” จะปรากฏหน้าต่างแสดงการติดตั้งโปรแกรม



รูปที่ ก.4 หน้าต่างแสดงการติดตั้ง

5. หลังจากทำการติดตั้งเสร็จเรียบร้อยแล้วจะปรากฏหน้าต่างดังรูปที่ จากนั้นทำการคลิก “LAUNCH” เพื่อเริ่มต้นการใช้งาน Visual Studio 2013



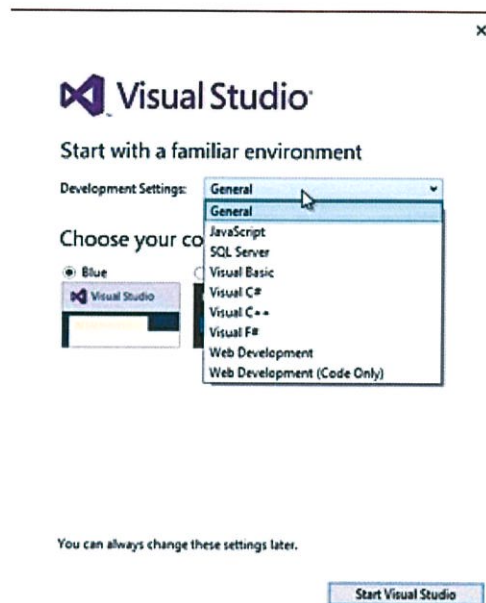
รูปที่ ก.5 หน้าต่างหลังจากการติดตั้งเสร็จเรียบร้อยแล้ว

6. หลังจากทำการคลิก “LAUNCH” แล้วจะปรากฏหน้าต่างสำหรับการสำหรับการเข้าสู่บัญชีของไมโครซอฟท์โดยคลิกที่ “Sign In” หากไม่ต้องการเข้าสู่ระบบให้คลิก “Not now, maybe later”



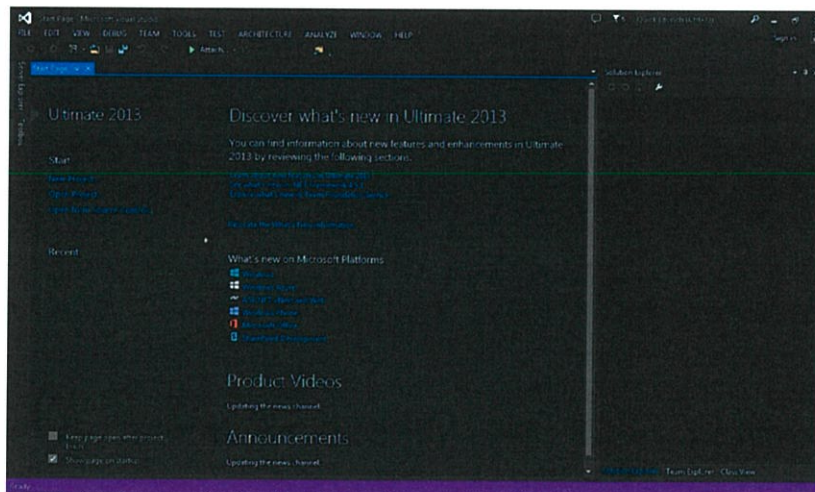
รูปที่ ก.6 หน้าต่างสำหรับการเข้าสู่บัญชีไมโครซอฟท์

7. เลือกธีมตามที่ต้องการ จากนั้นคลิก “Start Visual Studio” เพื่อทำการเปิดโปรแกรม



รูปที่ ก.7 รูปแบบของ โปรแกรม

8. หลังจากคลิก “Start Visual Studio” จะปรากฏหน้าต่างเริ่มต้นของโปรแกรม Visual Studio 2013



รูปที่ ก.8 หน้าเริ่มต้นของ Visual Studio 2013

ภาคผนวก ข.

วิธีการติดตั้งโปรแกรม MATLAB

ภาคผนวก ข.

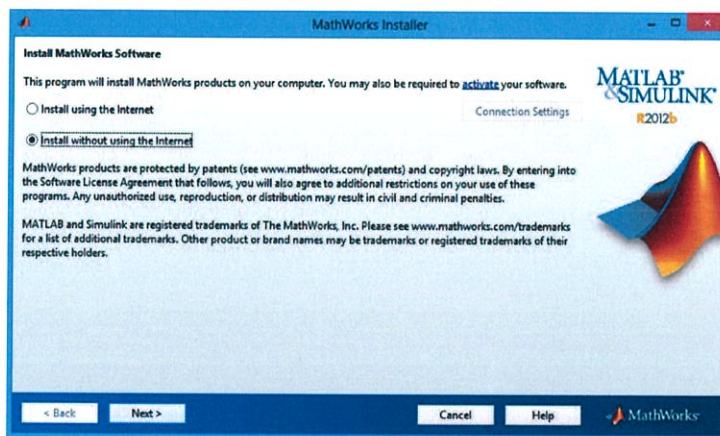
วิธีการติดตั้งโปรแกรม MATLAB

1. ให้ Double Click file setup.exe

(For Win32 CD Drive:\setup, For Win64 CD Drive:\bin\Win64\setup)

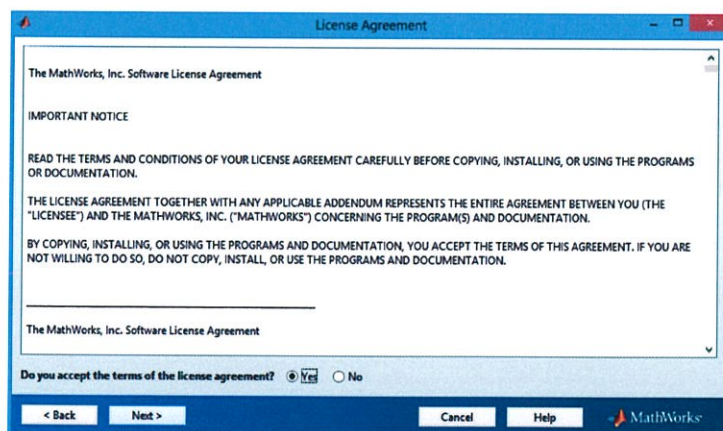


2. หลังจากนั้นให้เลือก Install without using the Internet หลังจากนั้นให้กด Next



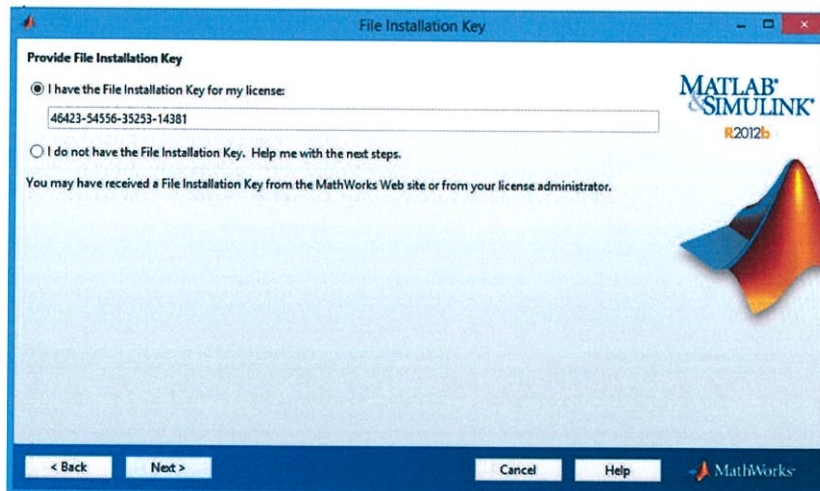
รูปที่ ข.1 หน้าเริ่มต้นการติดตั้ง

3. ให้ ตอบ Yes ใน หน้าต่าง License Agreement หลังจากนั้นให้กด Next



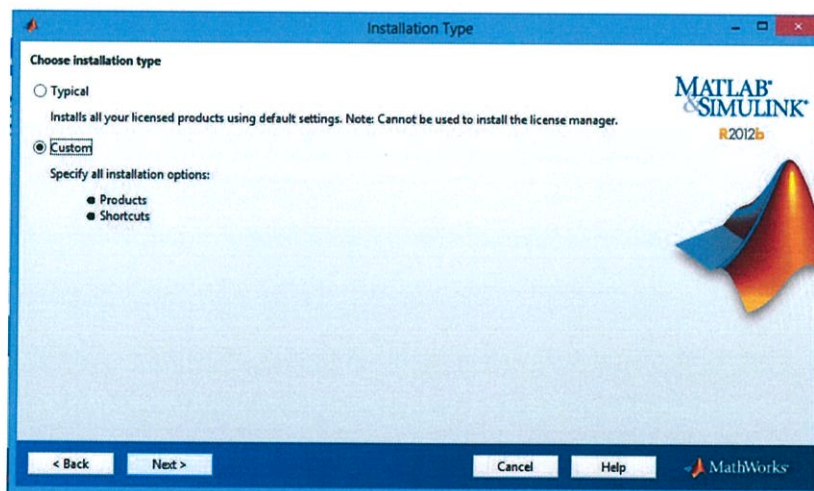
รูปที่ ข.2 หน้ารายละเอียด

3. เลือก I have the File Installation Key for my license แล้วป้อนตัวเลข Installation Key ที่ใน file FIK.txt ที่ได้รับทาง email ใส่ในช่องว่าง ดังรูป หลังจากนั้นให้กด Next



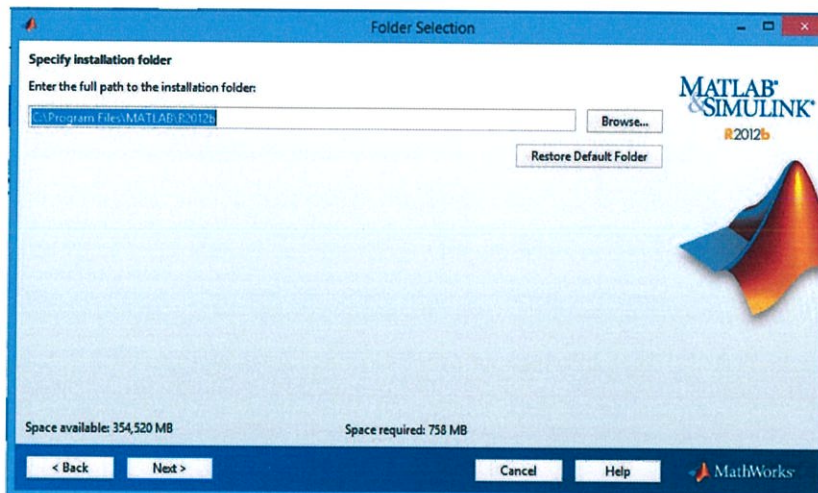
รูปที่ ข.3 หน้าต่างสำหรับการป้อน Installation Key

4. กดเลือกการติดตั้งแบบ Custom แล้วทำการกด Next



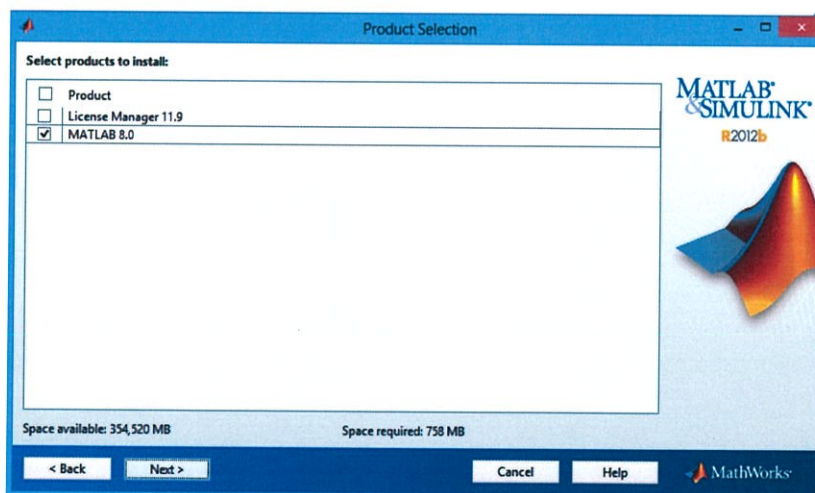
รูปที่ ข.4 รูปแบบการติดตั้ง

5. ให้ทำการ เลือก folder เพื่อติดตั้ง โปรแกรม ดังรูป แล้วทำการกด Next



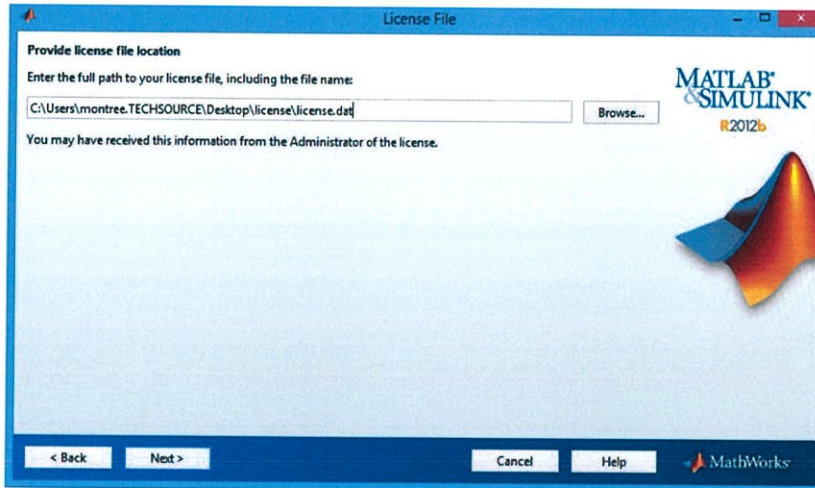
รูปที่ ข.5 หน้าต่างสำหรับการเลือกไอเดคทอรีพาทสำหรับการติดตั้ง

6. ให้ทำการเลือกติดตั้ง MATLAB และ toolbox ที่ต้องการติดตั้ง (อย่าเลือก License Manager) หลังจากนั้น กดปุ่ม Next



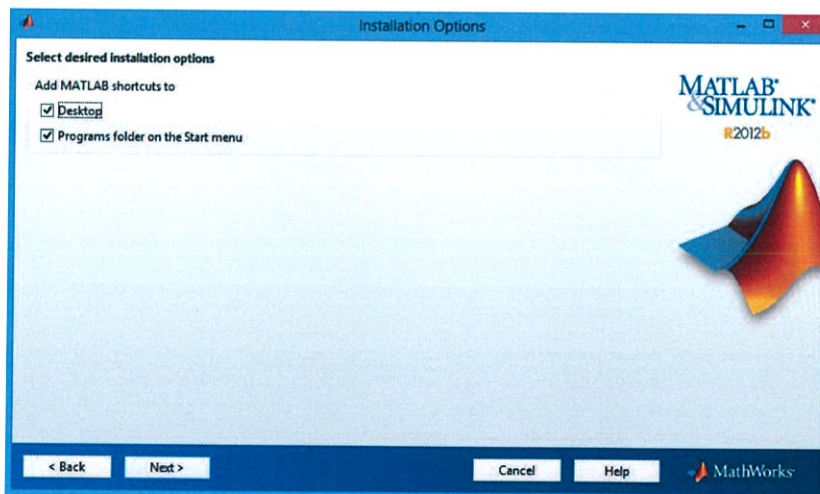
รูปที่ ข.6 หน้าต่างคุณสมบัติที่ต้องการติดตั้ง

7. หลังจากนั้นให้ทำการ Browse License.dat file ที่ได้รับจากเครื่อง server (ซึ่งจะอยู่ที่ CD Drive:\License file\License.dat) หลังจากนั้นให้ทำการกด Next



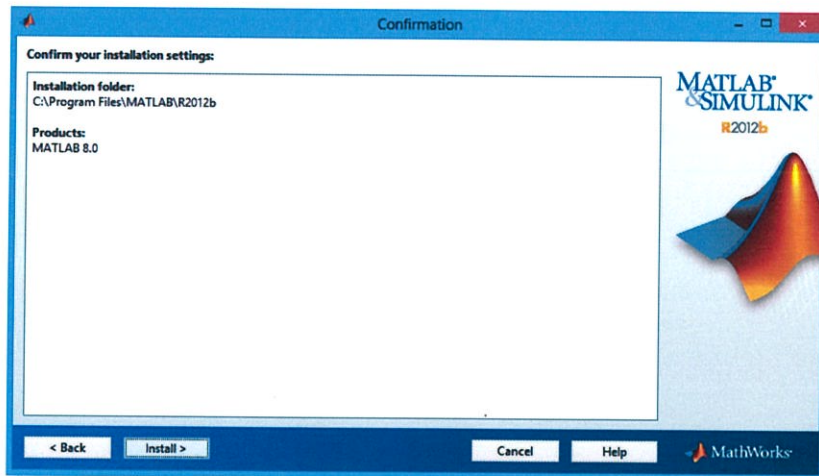
รูปที่ ข.7 หน้าต่างการสำหรับการเลือก License.dat file

8. ให้ทำการเลือก location ของไอคอน MATLAB



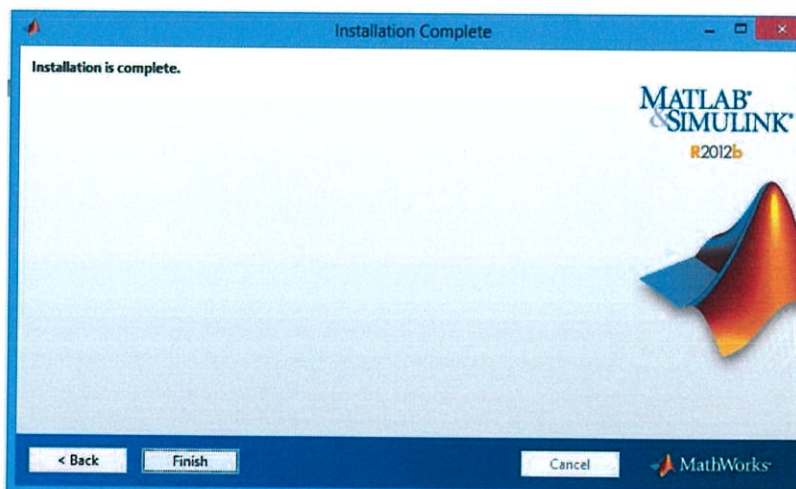
รูปที่ ข.8 หน้าต่าง Location

9. ให้กด Install เพื่อ ทำการติดตั้ง โปรแกรม



รูปที่ ข.9 หน้าต่างข้อมูลรายละเอียด

10. เมื่อทำการติดตั้งเสร็จสิ้น ให้กดปุ่ม Finish



รูปที่ ข.10 ทำการติดตั้งเรียบร้อย

ภาคผนวก ค.

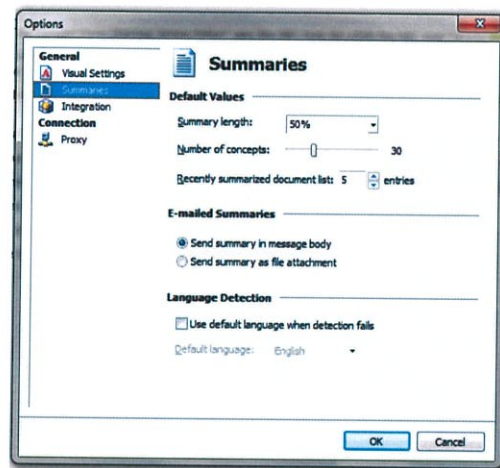
วิธีการติดตั้งโปรแกรม Copernic Summarizer

ภาคผนวก ก.

วิธีการติดตั้งโปรแกรม Copernic Summarizer

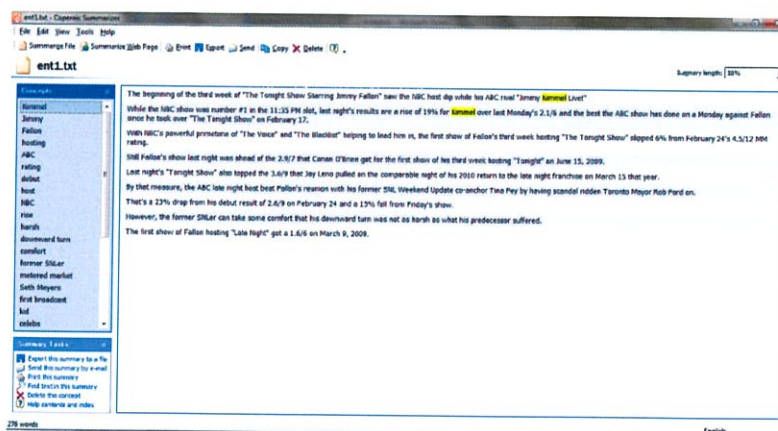
โปรแกรม Copernic Summarizer เป็นโปรแกรมที่มีความสามารถในการหาคำสำคัญของเอกสารที่เป็นไฟล์ได้หลายรูปแบบเช่น การหาคำสำคัญจากหน้าเว็บเพจ จากเท็กซ์ไฟล์ เป็นต้น ซึ่งผู้พัฒนาโปรแกรมนี้คือ บริษัท Copernic Technologies สามารถดาวน์โหลดโปรแกรมมาใช้งานได้ที่ www.copernic.com

หลังจากที่ได้ทำการติดตั้งโปรแกรมแล้ว ให้ทำการปรับค่าพารามิเตอร์ชื่อ Number of concepts ซึ่งเป็นพารามิเตอร์สำคัญที่เรากำหนดค่าเพื่อให้โปรแกรมประมวลผลโดยหาคำสำคัญที่พบมากที่สุด ในไฟล์เรียงตามลำดับตามจำนวนที่เรากำหนด ดังแสดงรูปที่



รูปที่ ก.1 การตั้งค่าพารามิเตอร์

ต่อจากนั้น จะสั่งรันโปรแกรมตัวอย่าง ผลลัพธ์ที่ได้จากการรันไฟล์โดยคำสำคัญจะแสดงในกรอบด้านซ้ายมือ



รูปที่ ก.2 ผลลัพธ์ที่ได้จากการรันโปรแกรม Copernic Summarizer