การวิเคราะห์ต้นทุนของเทคนิคการเรียนรู้ด้วยเครื่องในการตรวจจับการบุกรุก

น.ส.ไปรยา ตั้งจาตุรโสภณ

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
สาขาวิชาวิทยาศาสตร์คอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2561
ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

COST ANALYSIS OF MACHINE LEARNING TECHNIQUES IN INTRUSION DETECTION

Miss Praiya Tungjaturasopon

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Science in Computer Science
Department of Computer Engineering
Faculty of Engineering
Chulalongkorn University
Academic Year 2018

Copyright of Chulalongkorn University

หัวข้อวิทยานิพนธ์	การวิเคราะห์ต้นทุนของเทคนิคการเรียนรู้ด้วยเครื่องในการตรวจจับการบุกรุก
โดย	น.ส.ไปรยา ตั้งจาตุรโสภณ
สาขาวิชา	วิทยาศาสตร์คอมพิวเตอร์
อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก	ผู้ช่วยศาสตราจารย์ ดร.เกริก ภิรมย์โสภา

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต

..... คณบดีคณะวิศวกรรมศาสตร์
(รองศาสตราจารย์ ดร.สุพจน์ เตชวรสินสกุล)

คณะกรรมการสอบวิทยานิพนธ์

..... ประธานกรรมการ
(ผู้ช่วยศาสตราจารย์ ดร.ณัฐวุฒิ หนูไพโรจน์)

..... อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก
(ผู้ช่วยศาสตราจารย์ ดร.เกริก ภิรมย์โสภา)

..... กรรมการ
(ดร.กุลวดี ศรีพานิชกุลชัย)

..... กรรมการภายนอกมหาวิทยาลัย
(ดร.พงศ์ธวัช ชีพพิมลชัย)

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ไปรยา ตั้งจาตุรโสภณ : การวิเคราะห์ต้นทุนของเทคนิคการเรียนรู้ด้วยเครื่องในการตรวจจับการบุกรุก. (
COST ANALYSIS OF MACHINE LEARNING TECHNIQUES IN INTRUSION DETECTION) อ.ที่ปรึกษาหลัก : ผศ. ดร.เกริก ภิรมย์โสภา

งานวิจัยนี้ได้นำเสนอการวิเคราะห์ต้นทุนของเทคนิคการเรียนรู้ด้วยเครื่องในการตรวจจับการบุกรุก โดยวิเคราะห์เวลาที่ใช้ในการสร้างแบบจำลองที่ใช้กับข้อมูลในการตรวจจับการบุกรุก ซึ่งการเรียนรู้ด้วยเครื่อง (machine learning) เป็นวิทยาการคอมพิวเตอร์ที่ทำให้คอมพิวเตอร์สามารถเรียนรู้ด้วยตนเอง สามารถทำนายหรือตัดสินใจจากข้อมูลที่เข้ามาได้ ว่าเป็นภัยคุกคามทางเครือข่ายหรือไม่ ซึ่งคุณสมบัตินี้จะช่วยให้สามารถตรวจสอบภัยคุกคามทางเครือข่ายรูปแบบใหม่ๆ ที่ไม่เคยตรวจพบมาก่อนได้ ดังนั้นการเพิ่มประสิทธิภาพความถูกต้องของเครื่องมือการตรวจจับการบุกรุก กลายเป็นเรื่องที่เปิดกว้างและได้รับความสนใจจากกลุ่มการวิจัย อย่างไรก็ตามต้นทุนทางด้านเวลามักถูกมองข้ามในกลุ่มการวิจัย งานวิจัยนี้จะช่วยให้ผู้ดูแลระบบสามารถตัดสินใจได้ดีขึ้นเกี่ยวกับวิธีเลือกฮาร์ดแวร์ที่เหมาะสมสำหรับการตรวจจับการบุกรุกในสภาพแวดล้อมต่างๆ อีกทั้งได้เสนอแบบจำลองสำหรับการประมาณเวลาในการสร้างแบบจำลองระบบตรวจจับการบุกรุกของแต่ละรูปแบบ และได้นำเสนอสมรรถนะของเวลาในการประมวลผลที่เพิ่มขึ้น (Speedup ratio) และสัดส่วนของงานที่สามารถประมวลผลแบบขนานได้ในแต่ละรูปแบบวิธี เพื่อเป็นแนวทางในการตัดสินใจ เลือกรูปแบบเทคนิคการเรียนรู้ด้วยเครื่องในการสร้างระบบตรวจจับการบุกรุกที่เหมาะสม สำหรับการลงทุน

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

สาขาวิชา วิทยาศาสตร์คอมพิวเตอร์ ลายมือชื่อนิสิต .....
ปีการศึกษา 2561 ลายมือชื่อ อ.ที่ปรึกษาหลัก .....

5970941221 : MAJOR COMPUTER SCIENCE
KEYWORD: INTRUSION DETECTION, COST ANALYSIS, MACHINE LEARNING

Praiya Tungjaturasopon :

COST ANALYSIS OF MACHINE LEARNING TECHNIQUES IN INTRUSION DETECT ION. Advisor: Asst. Prof. Krerk Piromsopa, Ph.D.

In our study presents the performance analysis of machine learning techniques in intrusion detection. We analyze time to build (and to retrain) the models used by Intrusion Detection System. Machine Learning is a branch of computer science that allows the computer to learn by themselves without programming sequence. These techniques can be applied to detect the new threat that has never seen before. Due to the large volumes of security audit data as well as complex and dynamic properties of intrusion behaviors, optimizing the accuracy of IDS becomes an important open problem that is receiving attention from the research community. However, the performance (time and space required) is usually ignored. Our study allows administrators to work make better decisions about how to select the proper hardware for intrusion detection in various environments. We proposed the models for estimating the time to build each model, presented the speedup ratio and the fraction of work that can be processed in parallel in each method. To be a guideline for choosing a machine learning technique to build the models used by Intrusion Detection System.

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

Field of Study: Computer Science Student's Signature .....

Academic Year: 2018 Advisor's Signature .....

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้สำเร็จลุล่วงไปได้ด้วยความอนุเคราะห์อย่างยิ่งของอาจารย์ ดร. เกริก ภิรมย์โสภา อาจารย์ที่ปรึกษา ซึ่งท่านได้ให้ความรู้ แนะนำแนวทางการวิจัย ตรวจสอบให้คำแนะนำ และสนับสนุนเป็นอย่างดี พร้อมทั้งให้กำลังใจเสมอมา จนทำให้การวิจัยในครั้งนี้สำเร็จออกมาด้วยดี

ขอขอบพระคุณ อาจารย์ ดร. ณัฐวุฒิ หนูไพโรจน์ อาจารย์ ดร. พงศ์ธวัช ชีพพิมลชัย และอาจารย์ ดร.กุลวดี ศรีพานิชกุลชัย กรรมการสอบวิทยานิพนธ์ ที่กรุณาเสียสละเวลา ให้คำแนะนำตรวจสอบ และแก้ไขวิทยานิพนธ์ฉบับนี้

ท้ายที่สุด ผู้เสนอวิทยานิพนธ์ขอขอบคุณครอบครัว หัวหน้าและเพื่อนร่วมงาน รวมทั้งเพื่อน ๆ ทุก ๆ คน ที่คอยติดตาม ให้กำลังใจและสนับสนุน รวมถึงท่านอื่น ๆ ที่มิได้กล่าวชื่อไว้ ณ ที่นี้ที่มีส่วนช่วยให้วิทยานิพนธ์สำเร็จได้ด้วยดี

ไปรยา ตั้งจาตุรโสภณ

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

สารบัญ

หน้า

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

สารบัญภาพ

หน้า

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

สารบัญตาราง

หน้า

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

บทที่ 1
ที่มา และความสำคัญ

1.1 ความเป็นมาและความสำคัญ

เนื่องจากภัยคุกคามและอาชญากรรมในโลกคอมพิวเตอร์เติบโตขึ้นเรื่อยๆ อีกทั้งมีรูปแบบและลักษณะที่ซับซ้อนมากขึ้นอีกด้วย ดังนั้นเครื่องมือในการช่วยรักษาความมั่นคงปลอดภัยของระบบจึงจำเป็นต้องพัฒนาตามไปด้วย ระบบตรวจจับการบุกรุก (IDS) จึงกลายเป็นส่วนประกอบสำคัญสำหรับการรักษาความปลอดภัยเครือข่าย โดยสามารถแยกแยะ IDS ได้จากวิธีการตรวจจับ ได้แก่ การตรวจจับตามรูปแบบอัตลักษณ์(signature-based) และการตรวจหาจากสิ่งผิดปกติ (anomaly-based) ระบบตรวจจับการบุกรุกจากสิ่งผิดปกติสามารถแบ่งออกเป็น 3 กลุ่มตามเทคนิคการตรวจจับ ได้แก่ ตรวจจับตามข้อมูลทางสถิติ (statistical based), ตรวจจับตามฐานข้อมูลความรู้ (knowledge based) และตรวจจับโดยใช้การเรียนรู้ด้วยเครื่อง (machine learning based) ซึ่งการเรียนรู้ด้วยเครื่อง (machine learning) เป็นวิทยาการคอมพิวเตอร์ที่ทำให้คอมพิวเตอร์สามารถเรียนรู้ด้วยตนเอง สามารถทำนายหรือตัดสินใจจากข้อมูลที่เข้ามาได้ ว่าเป็นภัยคุกคามทางเครือข่ายหรือไม่ โดยปราศจากการทำงานตามลำดับคำสั่งโปรแกรม ซึ่งจะช่วยให้สามารถตรวจสอบภัยคุกคามรูปแบบใหม่ๆ ที่ไม่เคยตรวจพบมาก่อนได้อย่างประสิทธิภาพ คุณภาพ และเพิ่มความมั่นคงของระบบให้มากขึ้น

การเพิ่มประสิทธิภาพความถูกต้องของระบบตรวจจับการบุกรุก (IDS) กลายเป็นปัญหาเปิดกว้างที่ได้รับความสนใจในการวิจัย แต่อย่างไรก็ตาม เรื่องของต้นทุนการใช้งานทรัพยากรด้านเวลาในการตรวจจับมักถูกละเลยไป งานวิจัยนี้จะช่วยให้ผู้ดูแลระบบ สามารถตัดสินใจได้ดีขึ้น เกี่ยวกับการเลือกฮาร์ดแวร์ที่เหมาะสมสำหรับใช้สร้างระบบการตรวจจับการบุกรุกในสภาพแวดล้อมต่าง ๆ จึงได้เสนอแบบจำลอง สำหรับการประมาณเวลาในการสร้างการตรวจจับแต่ละรูปแบบ และสมการเวกเตอร์ของจุดตัด เพื่อกำหนดจำนวนหน่วยประมวลผลขั้นต่ำที่จำเป็นสำหรับการสร้างการตรวจจับโดยใช้รูปแบบจำลองต้นไม้ตัดสินใจ (Decision tree model) และรูปแบบซัพพอร์ตเวกเตอร์แมชชีน (support vector machine model)

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

1.2 วัตถุประสงค์ของการวิจัย

การวิจัยมีวัตถุประสงค์ ดังนี้

1. เพื่อศึกษารูปแบบและการทำงานของการสร้างโมเดลการจำแนกประเภทต่าง ๆ และนำมาใช้กับข้อมูล NSL-KDD เพื่อใช้ในระบบการตรวจจับการบุกรุก
2. เพื่อทดลอง วิเคราะห์ และเปรียบเทียบต้นทุนด้านเวลา และนำเสนอสมการสำหรับประมาณเวลาในการสร้างแบบจำลองในแต่ละรูปแบบวิธี
3. วิเคราะห์ และนำเสนอสมรรถนะของเวลาในการประมวลผลที่เพิ่มขึ้น (Speedup ratio) และสัดส่วนของงานที่สามารถประมวลผลแบบขนานได้ในแต่ละรูปแบบวิธี

1.3 ขอบเขตของการวิจัย

ขอบเขตของการวิจัยถูกกำหนดไว้ ดังนี้

1. ทำการทดลองกับเครื่องระบบปฏิบัติการวินโดวส์เซิฟเวอร์2016 (Windows server 2016) บนแพลตฟอร์ม google cloud ที่มีคุณสมบัติคือ ใช้หน่วยประมวลผลกลาง Intel (Intel Core Processor) และ หน่วยความจำ 16 กิกะไบต์
2. มุ่งเน้นการศึกษาข้อมูลเกี่ยวกับการบุกรุกทางด้านเครือข่าย โดยใช้กลุ่มข้อมูล NSL-KDD ในการวิจัย
3. กำหนดรูปแบบการจำแนกประเภทข้อมูลในการทดลอง โดยใช้การเรียนรู้ของเครื่องแบบมีผู้สอน (Supervised Machine Leaning) ดังนี้ รูปแบบจำลองต้นไม้ตัดสินใจ (Decision tree model), รูปแบบซัพพอร์ตเวกเตอร์แมชชีน (support vector machine model) และรูปแบบโครงข่ายประสามเทียมแบบ Multi-Layer Perceptron (Feedforward Neural Network)
4. การสร้างโมเดลที่นำมาทดลอง ต้องมีค่าความถูกต้องในการตรวจจับข้อมูลผิดปกติที่มากกว่า 90% โดยใช้วิธี k-fold Cross Validation ในการวัดประสิทธิภาพ

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

1.4 ขั้นตอนและวิธีดำเนินการวิจัย

วิธีดำเนินการวิจัย ถูกแบ่งเป็น 6 ขั้นตอน ดังนี้

1. ศึกษางานวิจัยเกี่ยวข้องกับข้อมูล NSL-KDD และการสร้างโมเดลให้มีค่าความถูกต้องในการตรวจจับข้อมูลผิดปกติที่สูง โดยใช้ รูปแบบจำลองต้นไม้ตัดสินใจ (Decision tree model), รูปแบบซัพพอร์ตเวกเตอร์แมชชีน (support vector machine model) และรูปแบบโครงข่ายประสามเทียมแบบ Multi-Layer Perceptron (Feedforward Neural Network)
2. ค้นคว้ารวบรวมความรู้พื้นฐาน และทฤษฎีที่เกี่ยวข้องกับงานวิจัย
3. ออกแบบการทดลอง วิเคราะห์เปรียบเทียบเครื่องมือที่จะใช้ดำเนินการ และวางแผนขอบเขตการดำเนินการ
4. ทดลองเพื่อปรียบเทียบประสิทธิภาพด้านเวลาในการสร้างโมเดลในแต่ละรูปแบบวิธี
5. วัดผลการทดลอง เพื่อเปรียบเทียบและวิเคราะห์เวลาในการสร้างโมเดลของการจำแนกประเภทต่าง ๆ และสร้างแบบจำลองในการตัดสินใจเลือกรูปแบบวิธีที่มีต้นทุนทางด้านเวลาที่ดีและเหมาะสมกับระบบ
6. สรุปผลการวิจัยและข้อเสนอแนะ และจัดทำวิทยานิพนธ์

1.5 ประโยชน์ที่คาดว่าจะได้รับ

ประโยชน์ที่คาดว่าจะได้รับจากการวิจัย ได้แก่

1. เข้าใจวิธีในการสร้างโมเดลการจำแนกประเภทต่าง ๆ ได้แก่ รูปแบบจำลองต้นไม้ตัดสินใจ (Decision tree model), รูปแบบซัพพอร์ตเวกเตอร์แมชชีน (support vector machine model) และรูปแบบโครงข่ายประสามเทียมแบบ Multi-Layer Perceptron (Feedforward Neural Network)
2. ได้รับความรู้เกี่ยวกับรูปแบบวิธีในการสร้างโมเดลการจำแนกประเภทต่าง ๆ รวมทั้งผลกระทบในด้านต้นทุนของเวลา และการใช้หน่วยประมวลผลกลาง
3. ได้แบบจำลองในการตัดสินใจเพื่อสรุปและเลือกรูปแบบวิธีในการสร้างโมเดลการจำแนกข้อมูลประเภทต่าง ๆ ที่มีต้นทุนทางด้านเวลาที่ดีและเหมาะสมกับระบบ โดยสามารถนำไปปรับใช้กับระบบการตรวจจับการบุกรุก เพื่อให้มีความยืดหยุ่นและเหมาะสมกับสภาพแวดล้อมที่ต่างกันได้อย่างมีประสิทธิภาพ
4. สามารถนำความรู้จากผลการวิจัยนี้ไปประยุกต์ใช้จริงในระบบการตรวจจับการบุกรุก ต่อไปในอนาคต

1.6 ลำดับการจัดเรียงเนื้อหาในวิทยานิพนธ์

วิทยานิพนธ์นี้แบ่งเนื้อหาออกเป็น 5 บท ดังต่อไปนี้ บทที่ 1 เป็นบทนำซึ่งกล่าวถึง ความเป็นมาและความสำคัญของปัญหา รวมถึงวัตถุประสงค์ของการวิจัย บทที่ 2 กล่าวถึงทฤษฎีและงานวิจัยที่เกี่ยวข้องกับการวิจัยนี้ บทที่ 3 กล่าวถึงแนวทางการออกแบบการทดลอง บทที่ 4 กล่าวถึงการทดลองในแต่ละรูปแบบวิธีการจำแนกประเภทข้อมูล รวมถึงบทวิเคราะห์และสรุปผลการทดลอง และบทที่ 5 กล่าวถึงบทสรุปของการวิจัย

1.7 ผลงานที่ตีพิมพ์จากวิทยานิพนธ์

ส่วนหนึ่งของวิทยานิพนธ์นี้ได้รับการตอบรับให้ตีพิมพ์เป็นบทความทางวิชาการในหัวข้อเรื่อง “Performance Analysis of Machine Learning Techniques in Intrusion Detection” โดย นางสาวไปรยา ตั้งจาตุรโสภณ และอาจารย์ ดร. เกริก ภิรมย์โสภา ในงานประชุมวิชาการ “International Conference on Network Security (ICNS 2018)” ณ เมืองไทเป ประเทศไต้หวัน วันที่ 14-16 ธันวาคม พ.ศ. 2561

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

บทที่ 2
ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

2.1 ทฤษฎีที่เกี่ยวข้อง

2.1.1 NSL-KDD

KDD99 เป็นชุดข้อมูลเกี่ยวกับการบุกรุกโจมตีทางเครือข่ายที่ใช้งานกันอย่างกว้างขวาง เพื่อใช้ในการวิเคราะห์หาความสัมพันธ์ลักษณะการโจมตีต่าง ๆ โดยมีทั้งชุดข้อมูลสำหรับการสอนแบบจำลอง (Training dataset) และชุดข้อมูลในการทดสอบแบบจำลอง (Testing dataset) ซึ่งสามารถนำมาใช้พัฒนาแบบจำลอง ทางการวิเคราะห์ทางสถิติได้ แต่ต่อมาพบว่ามี ประเด็นสำคัญในชุดข้อมูลที่ส่งผลต่อประสิทธิภาพของระบบและส่งผลให้มีการประมาณความผิดปกติอย่างมาก จึงมีวิธีการตรวจสอบ เพื่อแก้ปัญหาเหล่านี้ กลายเป็นชุดข้อมูลใหม่เป็นคือ NSL-KDD ที่มีความสมบูรณ์มากขึ้น[1] ข้อดีของชุดข้อมูล NSL-KDD คือ

1. ไม่มีข้อมูลที่ซ้ำซ้อนในชุดข้อมูลสำหรับการสอนแบบจำลอง (NSL-KDD Training dataset) ดังนั้นจะช่วยลดผลลัพธ์ของการวิเคราะห์ที่มีความลำเอียง (biased result)
2. ไม่มีข้อมูลที่ซ้ำกันในชุดข้อมูลสำหรับการทดสอบแบบจำลอง (NSL-KDD Testing dataset) ซึ่งจะส่งผลให้ได้ผลการวิเคราะห์ที่ดีขึ้น
3. ข้อมูลที่เลือกจากแต่ละกลุ่มมีสัดส่วนผกผัน กับเปอร์เซ็นต์ของชุดข้อมูล KDD ต้นฉบับ เป็นผลให้อัตราการจัดหมวดหมู่ของวิธีการเรียนรู้ (Classification) ที่แตกต่างกันมีประสิทธิภาพมากขึ้น มีการประเมินผลที่ถูกต้องของเทคนิคการเรียนรู้ที่แตกต่างกันมากขึ้น

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

No	Attribute Name	Description	Sample Data
1	duration	Length of time duration of the connection	0
2	protocol_type	Protocol used in the connection	Tcp
3	service	Destination network service used	ftp_data
4	flag	Status of the connection – Normal or Error	SF
5	src_bytes	Number of data bytes transferred from source to destination in	491

		single connection	
6	dst_bytes	Number of data bytes transferred from destination to source in single connection	0
7	land	if source and destination IP addresses and port numbers are equal then, this variable takes value 1 else 0	0
8	wrong_fragment	Total number of wrong fragments in this connection	0
9	urgent	Number of urgent packets in this connection. Urgent packets are packets with the urgent bit activated	0
10	hot	Number of „hot" indicators in the content such as: entering a system directory, creating programs and	
11	executing programs	0	
12	num_failed_logins	Count of failed login attempts	0
13	logged_in	Login Status: 1 if successfully logged in; 0 otherwise	0
14	num_compromised	Number of ``compromised' ' conditions	0
15	root_shell	1 if root shell is obtained; 0 otherwise	0
16	su_attempted	1 if ``su root" command attempted or used; 0 otherwise	0
17	num_root	Number of ``root" accesses or number of operations performed as a root in the connection	0

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

18	num_file_creations	Number of file creation operations in the connection	0
19	num_shells	Number of shell prompt	0
20	num_access_files	Number of operations on access control files	0
21	num_outbound_cmds	Number of outbound commands in an ftp session	0
22	is_host_login	1 if the login belongs to the ``hot" list i.e., root or admin; else 0	0
23	is_guest_login	1 if the login is a ``guest" login; 0 otherwise	0
24	count	Number of connections to the same destination host as the current connection in the past two seconds	2
25	srv_count	Number of connections to the same service (port number) as the current connection in the past two seconds	2
26	serror_rate	The percentage of connections that have activated the flag (4) s0, s1, s2 or s3, among the connections aggregated in count (23)	0
27	srv_serror_rate	The percentage of connections that have activated the flag (4) s0, s1, s2 or s3, among the connections aggregated in srv_count (24)	0
28	rerror_rate	The percentage of connections	0

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

		that have activated the flag (4) REJ, among the connections aggregated in count (23)	
29	srv_rerror_rate	The percentage of connections that have activated the flag (4) REJ, among the connections aggregated in srv_count (24)	0
30	same_srv_rate	The percentage of connections that were to the same service, among the connections aggregated in count (23)	1
31	diff_srv_rate	The percentage of connections that were to different services, among the connections aggregated in count (23)	0
32	srv_diff_host_rate	The percentage of connections that were to different destination machines among the connections aggregated in srv_count (24)	0
33	dst_host_count	Number of connections having the same destination host IP address	150
34	dst_host_srv_count	Number of connections having the same port number	25
35	dst_host_same_srv_rate	The percentage of connections that were to the same service, among the connections aggregated in dst_host_count (32)	0.17
36	dst_host_diff_srv_rate	The percentage of connections that were to different services, among the connections	0.03

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

		aggregated in dst_host_count (32)	
37	dst_host_same_src_port_rate	The percentage of connections that were to the same source port, among the connections aggregated in dst_host_srv_count (33)	0.17
38	dst_host_srv_diff_host_rate	The percentage of connections that were to different destination machines, among the connections aggregated in dst_host_srv_c	0
39	dst_host_serror_rate	The percentage of connections that have activated the flag (4) s0, s1, s2 or s3, among the connections aggregated in dst_host_count (32)	0
40	dst_host_srv_serror_rate	The percent of connections that have activated the flag (4) s0, s1, s2 or s3, among the connections aggregated in dst_host_srv_count (33	0
41	dst_host_rerror_rate	The percentage of connections that have activated the flag (4) REJ, among the connections aggregated in dst_host_count (32)	0.05

ตารางที่ 1 รายละเอียดองค์ประกอบของข้อมูล NSL-KDD

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

Type	Features
Nominal	Protocol_type(2), Service(3), Flag(4)
Binary	Land(7), logged_in(12), root_shell(14), su_attempted(15), is_host_login(21), is_guest_login(22)
Numeric	Duration(1), src_bytes(5), dst_bytes(6), wrong_fragment(8), urgent(9), hot(10), num_failed_logins(11), num_compromised(13), num_root(16), num_file_creations(17), num_shells(18), num_access_files(19), num_outbound_cmds(20), count(23), srv_count(24), serror_rate(25), srv_serror_rate(26), rerror_rate(27), srv_rerro_rate(28), same_srv_rate(29), diff_srv_rate(30), srv_diff_host_rate(31), dst_host_count(32), dst_host_srv_count(33), dst_host_same_srv_rate(34), dst_host_diff_srv_rate(35), dst_host_same_src_port_rate(36), dst_host_srv_diff_host_rate(37), dst_host_serror_rate(38), dst_host_srv_serror_rate(39), dst_host_rerror_rate(40), dst_host_srv_rerror_rate(41)

ตารางที่ 2 รายละเอียดของ NSL-KDD โดยแยกตามลักษณะขององค์ประกอบ

โดยข้อมูลของ NSL-KDD แต่ละบรรทัด มีองค์ประกอบทั้งหมด 41 attributes[2]ซึ่งมีรายละเอียดตามตารางที่ (1) และหากทำการแยกแต่ละองค์ประกอบตามลักษณะของข้อมูล[3] เพื่อเอาไปใช้ประโยชน์ต่อไปในการวิเคราะห์ จะสามารถแยกได้ ดังตารางที่ (2)
การแยกผลลัพธ์ของข้อมูล NSL-KDD สามารถแบ่งข้อมูลเป็นลักษณะ 2 แบบ (Binary data) โดยจะแบ่งเป็นข้อมูลปกติ ไม่เข้าข่ายเป็นการโจมตี และเป็นข้อมูลที่เข้าข่ายการโจมตี แต่หากแยกตามประเภทการบุกรุก จะพิจารณาว่ามีลักษณะข้อมูลทั้งหมด 5 แบบ[4] ดังนี้

1. Normal Traffic เป็นการติดต่อภายในเครือข่ายที่ปกติ ไม่เข้าข่ายเป็นการโจมตี
2. Denial of Service (DoS) attack เป็นการโจมตีที่ตั้งใจใช้งานทรัพยากรของระบบให้หมด เพื่อไม่ให้ระบบสามารถตอบสนองต่อการเรียกใช้งานอื่น ๆ
3. User to Root (U2R) Attack เป็นการเข้าถึงสิทธิ์เครื่องคอมพิวเตอร์ของผู้อื่นโดยไม่ได้รับอนุญาต โดยมีลักษณะโจมตี เริ่มจากเข้าใช้งานระบบด้วยสิทธิ์ผู้ใช้งาน ทั่วไป จากนั้นจะใช้ช่องโหว่ของระบบเพื่อให้ตัวเองมีสิทธิ์เท่าผู้ดูแลระบบ (root)

4. Probe Attack เป็นรูปแบบที่ผู้บุกรุกพยายามทำการสำรวจข้อมูลของระบบ ว่ามีระบบใดบ้างที่ไม่มีระบบรักษาความปลอดภัยที่ดี และมีการให้บริการงานในรูปแบบใด เพื่อเก็บรวบรวมข้อมูล และใช้เป็นประโยชน์ต่อการโจมตีตามช่องโหว่ที่ได้จากการสำรวจนั้นๆ
5. Root to Local (R2L) Attack เป็นการโจมตีโดยการอาศัยช่องโหว่เพื่อควบคุม และเข้าใช้เครื่องของผู้อื่น

Training Dataset		**Testing dataset**	
Attack Class	**Quantity**	**Attack Class**	**Quantity**
Normal	67343	Normal	9711
DoS	45927	DoS	7458
Probe	11656	Probe	2421
R2l	995	R2L	2754
U2R	52	U2R	200
Total	125973	Total	22544

ตารางที่ 3 จำนวนข้อมูลของ NSL-KDD โดยแยกตามประเภทการโจมตี

Category	**Actual Attacks in Training**	**Additional Attacks in Test set**
DoS	Neptune, smurf, teardrop, pod, land, back	apache2, mailbomb, processtable, udpstorm
Probing	satan, ipsweep, nmap, portsweep	mscan, saint
R2L	imap, warezmaster, phf, multihop, guess passwd, spy, warezclient, ftp write	httptunnel, named, sendmail, snmp, getattack, xlock, xsnoop
U2R	loadmodule, buffer over flow, rootkit, perl	ps, snmpguess, sqlattack, worm, xterm

ตารางที่ 4 วิธีการบุกรุกทางเครือข่ายในกลุ่มข้อมูล NSL-KDD โดยแยกตามประเภทการโจมตี

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ซึ่งรายละเอียดจำนวนข้อมูลของแต่ละประเภทการบุกรุกทางเครือข่าย[5] เป็นไปตามข้อมูลในตารางที่ (3) และหากพิจารณากลุ่มข้อมูล NSL-KDD แล้ว จะพบว่าประเภทการบุกรุกหลักทั้ง 4 กลุ่ม ยังมีวิธีการบุกรุกต่าง ๆตามแต่ละรูปแบบ ซึ่งสรุปได้ดังตารางที่ (4)

2.1.2 การเรียนรู้ด้วยเครื่อง (Machine Leaning)

การเรียนรู้ของเครื่อง เป็นสาขาหนึ่งของปัญญาประดิษฐ์ ที่พัฒนามาจากการศึกษา การรู้จำรูปแบบ และการสร้างอัลกอริทึมที่สามารถเรียนรู้ข้อมูลและทำนายข้อมูลได้ อัลกอริทึมนั้นจะทำงานโดยอาศัยโมเดลที่สร้างมาจากชุดข้อมูลตัวอย่างขาเข้า เพื่อการทำนายหรือตัดสินใจในภายหลังแทนที่จะทำงานตามลำดับของคำสั่งโปรแกรมคอมพิวเตอร์ การเรียนรู้ของเครื่องมีเกี่ยวข้องอย่างมากกับสถิติศาสตร์ เนื่องจากทั้งสองสาขาศึกษาการวิเคราะห์ข้อมูลเพื่อการทำนายเช่นกัน นอกจากนี้ยังมีความสัมพันธ์กับสาขาการหาค่าเหมาะที่สุดในทางคณิตศาสตร์ที่แง่ของวิธีการ ทฤษฎี และการประยุกต์ใช้ การเรียนรู้ของเครื่องสามารถนำไปประยุกต์ใช้งานได้หลากหมาย ไม่ว่าจะเป็นการกรองจดหมายอิเล็กทรอนิกส์ขยะ การรู้จำตัวอักษร เครื่องมือค้นหา และคอมพิวเตอร์วิทัศน์

ส่วนการเรียนรู้ของเครื่องกับการทำเหมืองข้อมูลมักจะใช้วิธีการคล้ายกัน และมีส่วนสัมพันธ์กันอย่างเห็นได้ชัด สิ่งที่แตกต่างระหว่างสองศาสตร์นี้คือ การเรียนรู้ของเครื่องเน้นเรื่องการพยากรณ์ข้อมูลจากคุณสมบัติที่ได้เรียนรู้มาจากข้อมูลชุดสอน ส่วนการทำเหมืองข้อมูล เน้นเรื่องการค้นหาคุณสมบัติที่ไม่ทราบจากข้อมูลที่ได้มา กล่าวได้ว่าเป็นขั้นตอนการวิเคราะห์เพื่อค้นหา ความรู้ใหม่ในฐานข้อมูล

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

สองศาสตร์นี้มีส่วนสัมพันธ์กันคือ การทำเหมืองข้อมูลใช้วิธีการทางการเรียนรู้ของเครื่อง แต่มักจะมีเป้าหมายที่แตกต่างออกไปเล็กน้อย ส่วนการเรียนรู้ของเครื่อง มักใช้วิธีการของการทำเหมืองข้อมูลบางอย่าง เช่น การเรียนรู้แบบไม่มีผู้สอน หรือขั้นตอนการเตรียมข้อมูลเพื่อปรับปรุงความถูกต้องของการเรียนรู้ การผสมสองศาสตร์นี้เข้าด้วยกัน ทำให้ประสิทธิภาพของการเรียนรู้ของเครื่องมักจะดีขึ้นหากมีความสามารถในการรู้ความรู้บางอย่าง ในขณะที่การค้นหาความรู้และการทำเหมืองข้อมูลนั้น สิ่งสำคัญคือการค้นหาความรู้ที่ไม่รู้มาก่อน หากมีการวัดประสิทธิภาพจากสิ่งที่ไม่รู้มาก่อน วิธีการเรียนรู้แบบมีผู้สอนของการเรียนรู้ของเครื่อง ก็มักจะให้ผลได้ดีกว่าการใช้วิธีการเรียนรู้แบบไม่มีผู้สอนอย่างเดียว และสิ่งสำคัญที่สุดของการเรียนรู้ของเครื่องคือ การทำให้โมเดลมีความทั่วไป (general) มากขึ้นจากข้อมูลที่ได้มา การทำให้มีความทั่วไปมากขึ้นนี้จะทำให้เครื่องสามารถพยากรณ์หรือทำงานกับตัวอย่างข้อมูลที่ไม่เคยเห็นมาก่อนได้อย่างแม่นยำมากขึ้น

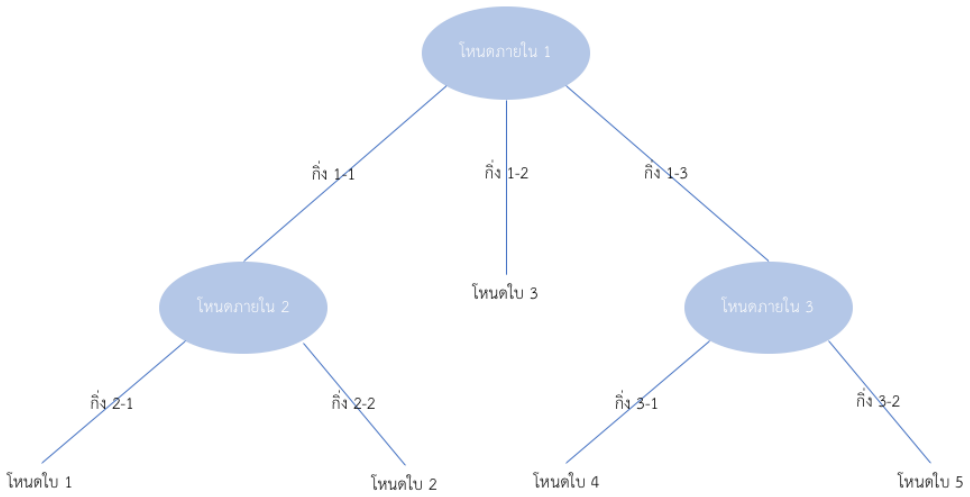
2.1.2.1 ต้นไม้ตัดสินใจ (Decision tree)

ต้นไม้ตัดสินใจนับว่าเป็นวิธีการเรียนรู้ที่ใช้มากที่สุดแบบหนึ่งในการเรียนรู้ของเครื่อง[6] การเรียนรู้แบบนี้เป็นการเรียนรู้โดยการแยกแยะ (classification) ข้อมูลออกเป็นกลุ่ม (class) ต่าง ๆ โดยใช้ คุณสมบัติ (attribute) ของข้อมูลในการแยกแยะ ต้นไม้ตัดสินใจที่ได้จากการเรียนรู้ ทำให้ทราบว่าคุณสมบัติใดของข้อมูลที่เป็นตัวกำหนดการแยกแยะ และคุณสมบัติแต่ละตัวของข้อมูลมีความสำคัญมากน้อยต่างกันอย่างไร ซึ่งเป็นประโยชน์ช่วยให้ผู้ใช้สามารถวิเคราะห์ข้อมูลและตัดสินใจได้ถูกต้องยิ่งขึ้น

การแทนต้นไม้ตัดสินใจ (Decision Tree Representation) ผลลัพธ์ของการเรียนรู้ต้นไม้ตัดสินใจจะแสดงในรูปต้นไม้ ซึ่งประกอบไปด้วย

1. โหนดภายใน (internal node) คือ คุณสมบัติต่าง ๆ ของข้อมูล ซึ่งเมื่อข้อมูลใด ๆ ตกลงมา ที่โหนด จะใช้คุณสมบัตินี้เป็นตัวตัดสินใจว่าข้อมูลจะไปในทิศทางใด โดยโหนดภายในที่เป็น จุดเริ่มต้นของต้นไม้ เรียกว่าโหนดราก
2. กิ่ง (branch, link) เป็นค่าคุณสมบัติของคุณสมบัติในโหนดภายในที่แตกกิ่งนี้ออกมา ซึ่ง โหนดภายในจะแตกกิ่งเป็นจำนวนเท่ากับจำนวนค่าคุณสมบัติของโหนดภายในนั้น
3. โหนดใบ (leaf node) คือกลุ่มต่าง ๆ ซึ่งเป็นผลลัพธ์ในการแยกแยะข้อมูล ตัวอย่างของต้นไม้ตัดสินใจแสดงในรูปที่ (1)

จุฬาลงกรณ์มหาวิทยาลัย



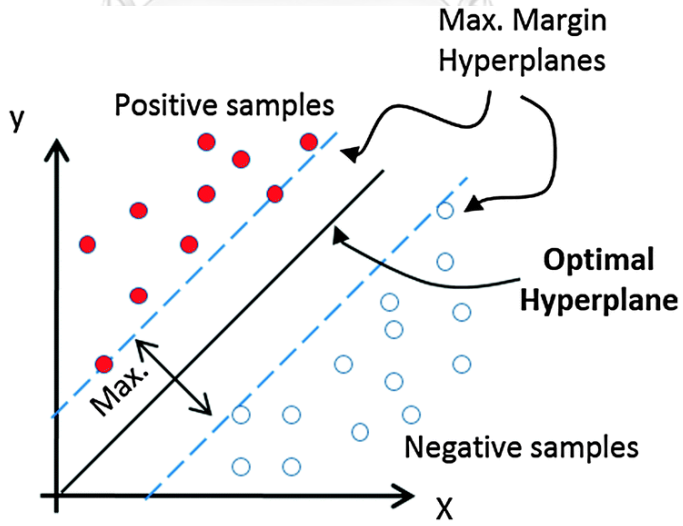
รูปที่ 1 การแทนต้นไม้ตัดสินใจ

ลักษณะการเรียนรู้ของต้นไม้ตัดสินใจ

- ผลการเรียนรู้แสดงอยู่ในรูปที่เข้าใจง่าย จึงง่ายต่อการแยกแยะกลุ่มต่าง ๆ
- แต่ละรูปแบบจากโหนดรากถึงโหนดใบสามารถแสดงให้อยู่ในรูปแบบเงื่อนไขได้
- มีความทนทานต่อข้อมูลที่มีสัญญาณรบกวน เช่น ค่าคุณสมบัติที่ผิดพลาดหรือขาดหาย และคุณสมบัติที่ไม่เกี่ยวข้อง
- การเรียนรู้ใช้เวลาน้อย เมื่อเทียบกับอัลกอริทึมสำหรับแยกแยะชนิดอื่น

2.1.2.2 ซัพพอร์ตเวกเตอร์แมชชีน (support vector machine)

เป็นวิธีการที่นำมาแยกกลุ่มของข้อมูล[7] โดยอาศัยระนาบสำหรับแบ่งเขตแดนของกลุ่มข้อมูลออกจากกันเป็นสองฝั่งโดยที่มีเวกเตอร์ (Vector) แทนจำนวนเซตของคุณลักษณะ ที่ซึ่ง Hyperplane เป็นตัวแยกกลุ่มของเวกเตอร์ ส่วน Margin เป็นระยะห่างจากเส้นตรง Hyperplane ถึง เส้นตรงที่ผ่านข้อมูลที่ใกล้ที่สุดและขนานกับ Hyperplane ของทั้งสองกลุ่ม โดยที่ SVM จะเลือก Hyperplane ที่มีค่า Margin สูงสุดดัง แสดงในรูปที่ (2)



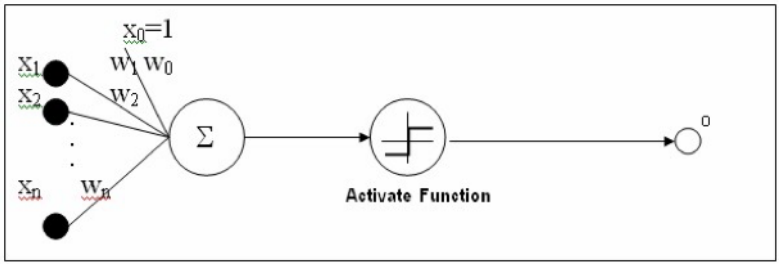
รูปที่ 2 ซัพพอร์ตเวกเตอร์แมชชีนแบบสองมิติ

วิธีการจัดหมวดหมู่ แบบซัพพอร์ตเวกเตอร์แมชชีน (Support Vector Classification) คือการพัฒนาที่ค่อนข้างทันสมัย ในการจดจำรูปแบบทางสถิติ วิธีการนี้เป็นการจัดหมวดหมู่ที่ดีที่สุดของ

ข้อมูลสองกลุ่มที่แยกกัน ทำให้บรรลุผลโดยอาศัยความกว้างมากที่สุดของพื้นที่ว่างเปล่า (Maximum Margin) ของระหว่างข้อมูลสองกลุ่ม ซึ่งความกว้าง จากขอบ (Margin) เป็นระยะทางระหว่าง การกำหนดเส้นเขตแดน (Hypersurface) ในพื้นที่แอตทริบิวต์n มิติ กับรูปแบบการเรียนรู้ที่ใกล้กำหนดเส้นเขตแดนที่สุด หรือที่เรียกว่าซัพพอร์ตเวกเตอร์ (Support Vectors) ที่ซึ่งรูปแบบการเรียนรู้ที่ใกล้ที่สุด ทำให้สามารถระบุฟังก์ชันการจำแนก เห็นได้ชัดว่าความสามารถในการระบุเส้นเขตแดนของการแบ่งแยกที่ดีที่สุด ระหว่างข้อมูลทั้งสอง กลุ่มในกรณีที่มีหลายเส้นเขตแดนที่แบ่งแยกข้อมูล เป็นจุดเด่นที่สำคัญของวิธีการนี้ และช่วยในการ ลดปัญหาระหว่างการการเรียนรู้ของการที่โมเดลจดจำรูปแบบของข้อมูลได้ดีเกินไปจนทำให้ นำไปใช้สำหรับทำนายข้อมูลอื่นได้ไม่ดี (Overfitting) ในทางทฤษฎีเส้นเขตแดนที่เหมาะสมที่สุด (Optimal Hypersurface) มีความจุ (Capacity) ต่ำที่สุด จากความต้องทางสถิติ ทฤษฎีการเรียนรู้ของ Vapnikและ Chervonenkisวิธีดั้งเดิมของวิธีการจัดหมวดหมู่แบบซัพพอร์ตเวกเตอร์ ได้รับการพัฒนาสำหรับการแยกเชิงเส้น (Linear Separation) ของข้อมูลสองกลุ่มซึ่งยังเป็นข้อจำกัด โดยภายหลังได้พัฒนามาเป็นการเรียนรู้การแยกที่ไม่เป็นเส้นตรง (Non-Linearly Separable) ของข้อมูลการเรียนรู้ ส่วนข้อเสียของซัพพอร์ตเวกเตอร์ (SVM) เป็นเรื่องของประสิทธิภาพในการสอนแบบจำลอง ที่ค่อนข้างใช้เวลานาน และใช้ทรัพยากรกับข้อมูลจำนวนมาก

2.1.2.3 โครงข่ายประสามเทียม (neural networks)

แนวคิดของนิวรอลเน็ตเวิรก์ได้มาจากการจำลองการทำงานของเซลล์สมองของมนุษย์ โดยหน่วยที่ย่อยที่สุดของนิวรอลเน็ตเวิรก์เรียกว่าเพอร์เซ็ปตรอน (Perceptron) ซึ่งเทียบได้กับเซลล์สมองของมนุษย์หนึ่งนิวรอน (neuron) เพอร์เซ็ปตรอนนี้จะทำหน้าที่รับอินพุตซึ่งเป็นเวคเตอร์ของจำนวนจริงเข้ามา พร้อมคำนวณค่าเหล่านี้ โดยให้น้ำหนักของอินพุตแต่ละตัวแตกต่างกันดังแสดงในรูปที่ (3) เอาต์พุตที่ได้จะถูกนำไปคำนวณค่า ผิดพลาด (error) เพื่อนำมาปรับน้ำหนักของอินพุตต่อไป



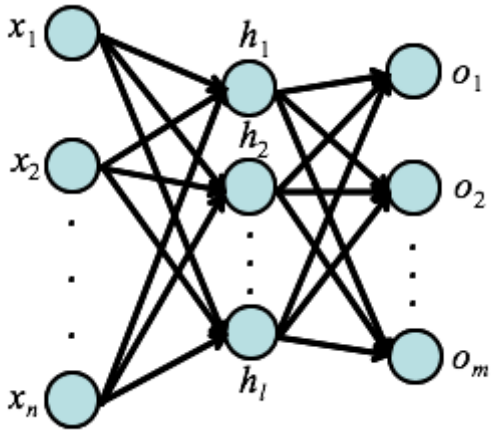
รูปที่ 3 เพอร์เซ็ปตรอน (Perceptron)

ในการเรียนรู้ของเพอร์เซ็ปตรอนมีกระบวนการดังนี้

- เริ่มจากการสุ่มค่าน้ำหนัก Wi

- เทียบเพอร์เซ็ปตรอนกับทุกตัวอย่างที่สอนทีละตัว และแก้ไขน้ำหนักเมื่อเพอร์เซ็ปตรอนแยกตัวอย่างผิดพลาด
- วนทำซ้ำกับตัวอย่างที่สอน จนกระทั่งเพอร์เซ็ปตรอนแยกตัวอย่างได้ถูกต้องทั้งหมด

เพอร์เซ็ปตรอนเดี่ยวสามารถแสดงระนาบตัดสินใจแบบเชิงเส้น (linear decision surface) เท่านั้น ต้องใช้เน็ตเวิร์กแบบหลายชั้น (multilayer network) ถึงจะสามารถแสดงระนาบตัดสินใจแบบไม่เชิงเส้น (non-linear decision surface) ซึ่งมีความที่ซับซ้อนมากกว่าได้[6] ซึ่งในงานวิจัยนี้เราจะใช้ ขั้นตอนวิธีแบ็กพรอพาเกชันขั้นตอนวิธีในการเรียนรู้ แบ็กพรอพาเกชันนิวรอลเน็ตเวิรก์ (Backpropagation neural network) เป็นเน็ตเวิรก์ที่มีได้หลายนิวรอนและมีได้หลายชั้น (multilayer) และทำงานกับฟังกชันซิกมอยด์ (Sigmoid function) ซึ่งเป็นฟังก์ชันที่สามารถแยกตัวอย่างได้แบบไม่เชิงเส้น ทำให้ทำงานได้ดีกว่าเพอร์เซ็ปตรอนเดี่ยวๆ โครงสร้างของแบ็กพรอพาเกชันนิวรอลเน็ตเวิรก์แสดงในรูปที่ (4)



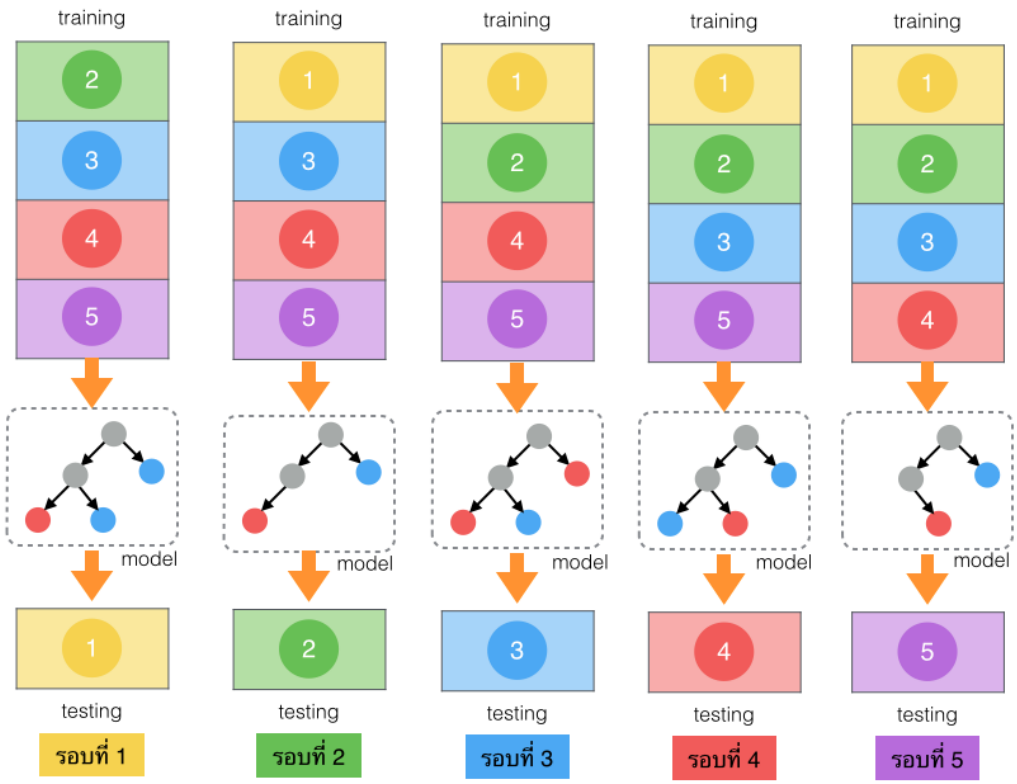
รูปที่ 4แบ็กพรอพาเกชันนิวรอลเน็ตเวิรก์

ตัวอย่างในรูปที่ (4) แสดงเน็ตเวิรก์ป้อนไปหน้าแบบหลายชั้น ซึ่งประกอบด้วยชั้นอินพุต ชั้นฮิดเดนหรือชั้นซ้อน และชั้นเอาท์พุต ในรูปแสดงชั้นฮิดเดนเพียงชั้นเดียว แต่อาจมีมากกว่าหนึ่งชั้นได้ และเส้นเชื่อมจะเชื่อมต่อเป็นชั้น ๆ ไม่ข้ามชั้น จากชั้นอินพุตไปชั้นฮิดเดน ถ้ามีชั้นฮิดเดนมากกว่า หนึ่งชั้น ก็เชื่อมต่อกันไป และสุดท้ายจากชั้นฮิดเดนไปชั้นเอาท์พุตเน็ตเวิรก์ป้อนไปหน้าแบบหลายชั้น นี้จะ

ไม่มีเส้นเชื่อมย้อนกลับจะมีแต่เส้นเชื่อมไปข้างหน้าอย่างเดียว กล่าวคือ ไม่มีเส้นเชื่อมจากโหนด (node) ในชั้นเอาท์พุตส่งกลับมายังโหนดในชั้นฮิดเดนหรือชั้นอินพุต

2.1.3 การทดสอบประสิทธิภาพของแบบจำลอง ด้วยวิธี Cross-validation

วิธีนี้เป็นวิธีที่นิยมในการทำงานวิจัย เพื่อใช้ในการทดสอบประสิทธิภาพของแบบจำลอง เนื่องจากผลที่ได้มีความน่าเชื่อถือ การวัดประสิทธิภาพด้วยวิธี Cross-validation นี้จะทำการแบ่งข้อมูลออกเป็นหลายส่วน (มักจะแสดงด้วยค่า k) เช่น 5-fold cross-validation การแบ่งข้อมูลแบบ K - fold cross-validation คือการแบ่งข้อมูลออกเป็น K ชุดเท่า ๆกัน และทำการคำนวณค่าความผิดพลาด K รอบ โดยแต่ละรอบการคำนวณข้อมูลชุดหนึ่งจากข้อมูล K ชุดจะถูกเลือกออกมาเพื่อเป็นข้อมูลทดสอบ และข้อมูลอีก K - 1 ชุดจะถูกใช้เป็นข้อมูลสำหรับการ เรียนรู้ดังตัวอย่างต่อไปนี้ K - fold Cross Validation (K = 5)



รูปที่ 5การวัดประสิทธิภาพด้วยวิธี K - fold Cross Validation (K = 5)

จากรูปที่ 5 แบ่งข้อมูลสอนออกเป็น 5 ส่วนที่มีจำนวนเท่ากัน หลังจากนั้นทำการทดสอบประสิทธิภาพของโมเดล 5 ครั้ง ดังนี้[8]

- รอบที่ 1 ใช้ข้อมูลส่วนที่ 2,3,4 และ 5 สร้างแบบจำลอง และใช้แบบจำลองทำนายข้อมูลส่วนที่ 1 เพื่อทำการทดสอบ
- รอบที่ 2 ใช้ข้อมูลส่วนที่ 1,3,4 และ 5 สร้างแบบจำลอง และใช้แบบจำลองทำนายข้อมูลส่วนที่ 2 เพื่อทำการทดสอบ
- รอบที่ 3 ใช้ข้อมูลส่วนที่ 1,2,4 และ 5 สร้างแบบจำลอง และใช้แบบจำลองทำนายข้อมูลส่วนที่ 3 เพื่อทำการทดสอบ
- รอบที่ 4 ใช้ข้อมูลส่วนที่ 1,2,3 และ 5 สร้างแบบจำลอง และใช้แบบจำลองทำนายข้อมูลส่วนที่ 4 เพื่อทำการทดสอบ
- รอบที่ 5 ใช้ข้อมูลส่วนที่ 1,2,3 และ 4 สร้างแบบจำลอง และใช้แบบจำลองทำนายข้อมูลส่วนที่ 5 เพื่อทำการทดสอบ

วิธีการนี้คือข้อมูลในแต่ละชุดที่ทำการแบ่งจะถูกทดสอบอย่างน้อย 1 ครั้ง และถูกเรียนรู้ทั้งหมด K -1 ครั้ง โดยในขั้นตอนเหล่านี้ สามารถกำหนดได้ว่าต้องการขนาดข้อมูลขนาดใด และต้องการทำการคำนวณเป็นจำนวนรอบเท่าใด ซึ่งค่า K ที่นิยมและถือว่าเป็นมาตรฐานก็คือ K=10 เพราะว่าจะทำให้เหลือข้อมูลไว้สำหรับฝึกฝนถึง 90% ในแต่ละรอบ (training data) และมีข้อมูลอีก 10% ไว้สำหรับทดสอบ (testing data) แสดงว่าหากนำแบบจำลองที่ได้นั้นไปใช้ ความเที่ยงตรงของการวัดนี้จะอยู่ในระดับสูงประมาณ 90%

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

2.1.4 กฎของ Amdahl

ในการปรับปรุงสมรรถนะของระบบคอมพิวเตอร์ ไม่ได้หมายความว่าการประมวลผลจะได้ความเร็วตามจำนวนเท่าของการปรับปรุงระบบ เนื่องจากในการทำงานนั้น ประกอบไปด้วย งานที่สามารถปรับปรุงสมรรถนะได้ และงานที่ไม่สามารถปรับปรุงได้ โดยกฎของ Amdahl ได้กำหนดการวัด Speedup หรือสมรรถนะที่เพิ่มขึ้นว่า หากมีการพัฒนาให้ส่วนหนึ่งส่วนใดของระบบมีสมรรถนะมากขึ้น [9]

$$\text{Overall Speedup} = \frac{ExecutionTime_{old}}{ExecutionTime_{new}} \quad \text{(x)}$$

ทั้งนี้สมมติให้ส่วนที่ได้รับการปรับปรุงมีสัดส่วนเป็น P และอัตราส่วนที่ได้รับการปรับปรุงมีค่าเป็น n จึงแสดงการคำนวณค่า Speedup ตามแนวทางของ Amdahl ได้ดังนี้

$$\text{Overall Speedup} = \frac{1}{(1-P)+\frac{P}{n}} \quad \text{(x)}$$

หลักการนี้ ถูกนำมาใช้ในการปรับปรุงสมรรถนะของระบบคอมพิวเตอร์ กล่าวคือ หากต้องเลือกการปรับปรุงให้ส่วนใดส่วนหนึ่งของระบบเร็วขึ้น ควรเลือกส่วนที่มีการใช้งานเป็นปริมาณมาก เพื่อให้ได้สมรรถนะโดยรวมที่ดีที่สุดแทนการเลือกส่วนที่ใช้งานน้อย

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

2.2 งานวิจัยที่เกี่ยวข้อง

งานวิจัยที่เกี่ยวข้องกับการวิเคราะห์ประสิทธิภาพของเทคนิคการเรียนรู้ด้วยเครื่องในการตรวจจับการบุกรุกมีหลายงานด้วยกัน ได้แก่

2.2.1 EFFICIENT FEATURE SELECTION TECHNIQUE FOR NETWORK INTRUSION DETECTION SYSTEM USING DISCRETE DIFFERENTIAL EVOLUTION AND DECISION TREE[2]

ศึกษาชุดข้อมูล NSL-KDD โดยใช้เครื่องมือ WEKA ในการวิจัยวัดความค่าถูกต้องของการสร้างโมเดล โดยไม่ได้มีการปรับคุณสมบัติของกลุ่มข้อมูลสอน สรุปผลงานนี้สามารถอธิบายได้ดังนี้ ต้นไม้ตัดสินใจ (J48) ได้ค่าความถูกต้อง 81.05%, ซัพพอร์ตเวกเตอร์แมชชีน (SVM) ได้ค่าความถูกต้อง 69.52% และ โครงข่ายประสามเทียม (MLP) ได้ค่าความถูกต้อง 77.41% โดยพบว่ามีการนำวิธีการต่าง ๆ มาใช้ในการเพิ่มความถูกต้องในการจำแนกประเภท หรือเพื่อลดเวลาในการฝึกสอนและทดสอบ วิธีการหนึ่งคือการหาค่าพารามิเตอร์ที่เหมาะสมที่สุดของตัวแยกประเภท และอีกวิธีหนึ่งคือการลดคุณลักษณะที่ใช้ เพื่อให้ได้เวลาการฝึกอบรมและเวลาในการทดสอบที่เร็วขึ้น

2.2.2 PERFORMANCE COMPARISON FOR INTRUSION DETECTION SYSTEM USING NEURAL NETWORK WITH KDD DATASET[10]

วิเคราะห์และเปรียบเทียบอัลกอริทึมต่าง ๆเกี่ยวกับชุดข้อมูล KDD 99 สำหรับ 41 คุณสมบัติ โดยไม่ได้มีการปรับคุณสมบัติของกลุ่มข้อมูลสอน ความถูกต้องของ ซัพพอร์ตเวกเตอร์แมชชีน (SVM) คือ 90.7 และความถูกต้องของ โครงข่ายประสามเทียม (MLP) เท่ากับ 92.47%

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

2.2.3 DOS ATTACKS PREVENTION USING IDS AND DATA MINING[11]

ศึกษาและวิเคราะห์ประสิทธิภาพของอัลกอริทึมต่าง ๆสำหรับชุดข้อมูล KDD 99 สำหรับ 41 คุณสมบัติ โดยวัดผลได้ค่าความถูกต้องของต้นไม้ตัดสินใจ (Decision Tree) เท่ากับ 98.42% และความถูกต้องของซัพพอร์ตเวกเตอร์แมชชีน (SVM) เท่ากับ 97.336% ซึ่งจะสังเกตเห็นว่าประสิทธิภาพของ SVM ค่อนข้างดี แต่ยังน้อยกว่าประสิทธิภาพของ ต้นไม้ตัดสินใจ (Decision Tree) และจากการศึกษายังพบว่า ซัพพอร์ตเวกเตอร์แมชชีน (SVM) ใช้เวลาในการสร้างโมเดลช้ากว่า ต้นไม้ตัดสินใจ (Decision Tree) เล็กน้อย

2.2.4 MULTI-LEVEL HYBRID SUPPORT VECTOR MACHINE AND EXTREME LEARNING MACHINE BASED ON MODIFIED K-MEANS FOR INTRUSION DETECTION SYSTEM[12]

ศึกษาและวิเคราะห์ประสิทธิภาพของอัลกอริทึมซัพพอร์ตเวกเตอร์แมชชีน (SVM) และ K-means พร้อมทั้งแสดงข้อเสียของการลดเวลาการฝึกของอัลกอริทึมซัพพอร์ตเวกเตอร์แมชชีน (SVM) ว่าเวลาฝึกอบรมโมเดลนั้น จะถูกทำให้นานขึ้นเมื่อชุดข้อมูลการฝึกอบรมมีขนาดใหญ่มากขึ้น

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

บทที่ 3
แนวทางการออกแบบการทดลองและแบบจำลอง

ในบทนี้แบ่งออกเป็น 3 ส่วนคือ การออกแบบการทดลอง ขั้นตอนการทดลองและแบบจำลองเบื้องต้น ในส่วนแรกกล่าวถึงรูปแบบและรายละเอียดของการทดลองเพื่อวัดและศึกษาเปรียบเทียบประสิทธิภาพของระบบสำหรับการสร้างโมเดลของการจำแนกประเภทต่าง ๆ ส่วนที่สองอธิบายรายละเอียดขั้นตอนของการดำเนินการวิจัย และในส่วนที่สุดท้ายได้เสนอแบบจำลองเบื้องต้นเพื่อแสดงแนวทางและภาพรวมของการทดลองในบทถัดไป

3.1 การออกแบบการทดลอง

งานวิจัยนี้ออกแบบเพื่อรวบรวมข้อมูล เปรียบเทียบ และวิเคราะห์หาความสัมพันธ์ด้านเวลาในการสร้างโมเดลระบบการตรวจจับการบุกรุกทางด้านเครือข่าย ระหว่างจำนวนของข้อมูล จำนวนของตัวประมวลผลกลาง และอัลกอริธึมการเรียนรู้ด้วยเครื่องในหลายๆกรณี โดยมีรายละเอียดดังนี้

1. ทำการทดลองกับเครื่องระบบปฏิบัติการวินโดวส์เซิฟเวอร์2016 (Windows server 2016) บนแพลตฟอร์ม google cloud ที่มีคุณสมบัติคือ ใช้ตัวประมวลผลกลาง Intel (Intel Core Processor) และ หน่วยความจำ 16 กิกะไบต์
2. มุ่งเน้นการศึกษาข้อมูลเกี่ยวกับการบุกรุกทางด้านเครือข่าย โดยใช้กลุ่มข้อมูล NSL-KDD ในการวิจัย โดยเริ่มต้นจากจำนวนข้อมูล 10,000 บรรทัด และเพิ่มจำนวนมากขึ้นทุก ๆ 10,000 บรรทัดในแต่ละกรณี
3. กำหนดรูปแบบการจำแนกประเภทข้อมูลในการทดลอง โดยใช้การเรียนรู้ของเครื่องแบบมีผู้สอน (Supervised Machine Leaning) ดังนี้ รูปแบบจำลองต้นไม้ตัดสินใจ (Decision tree model), รูปแบบซัพพอร์ตเวกเตอร์แมชชีน (support vector machine model) และรูปแบบโครงข่ายประสามเทียมแบบ Multi-Layer Perceptron (Feedforward Neural Network)
4. การสร้างโมเดลที่นำมาทดลอง ต้องมีค่าความถูกต้องในการตรวจจับข้อมูลผิดปกติที่มากกว่า 90% โดยใช้วิธี 10-fold Cross Validation ในการวัดประสิทธิภาพ และใช้คุณสมบัติทั้งหมด โดยไม่ได้มีการปรับลดคุณสมบัติของกลุ่มข้อมูลสอน

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

5. ในแต่ละกรณีทดลองจะดำเนินการทดลอง 5 ครั้ง เพื่อทำการตัดข้อมูลที่อาจจะเป็นข้อมูลรบกวน โดยจะทำการตัดข้อมูลที่เป็นขอบบน และขอบล่างของกลุ่มผลลัพธ์ที่ได้จากการทดลองออก และหาค่าเฉลี่ยของกลุ่มผลลัพธ์ที่เหลือ เพื่อใช้ในการเปรียบเทียบและวิเคราะห์ต่อไป
6. เครื่องมือที่ใช้ในการทดลองคือ WEKA 3.8 ซึ่งเป็นโปรแกรมที่ใช้ในการวิเคราะห์ข้อมูลด้วยเทคนิคการเรียนรู้ด้วยเครื่องประเภทต่าง ๆ[13]
7. แยกการทดลองเป็น 2 ประเภทตามลักษณะการใช้งานของหน่วยประมวลผล คือ การทดลองที่ไม่ได้มีการปรับการใช้งานของหน่วยประมวลผล และ การทดลองที่มีการปรับให้สามารถใช้งานหน่วยประมวลผลได้ทุกหน่วยในเวลาเดียวกัน

จากรายละเอียดที่กล่าวไปข้างต้น สามารถแบ่งการทดลองทั้งหมดเป็น 6รูปแบบตามเทคนิคการจำแนกประเภทข้อมูลและลักษณะการใช้งานของหน่วยประมวลผล

Scenario no.	Algorithm	Workloads	Number of CPU	ลักษณะการใช้งานของ CPU
1-14	Decision Tree	10,000 -140,000 (increase every 10,000)	1 Core	ไม่ได้มีการปรับการใช้งานของหน่วยประมวลผล
15-28			2 Cores	
29-42			4 Cores	
43-56			8 Cores	
57-70			16 Cores	
71-82	Support Vector Machine	10,000 -120,000 (increase every 10,000)	1 Core	
83-94			2 Cores	
95-106			4 Cores	
107-118			8 Cores	
119-130			16 Cores	
131-142	Multi-Layer Perceptron	10,000 -120,000 (increase every 10,000)	1 Core	
143-154			2 Cores	
155-166			4 Cores	
167-178			8 Cores	
179-190			16 Cores	

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

191-202	Decision Tree	10,000 -120,000 (increase every 10,000)	1 Core	มีการปรับให้สามารถใช้งานหน่วยประมวลผลได้ทุกหน่วยในเวลาเดียวกัน
203-214			2 Cores	
215-226			4 Cores	
227-238			8 Cores	
239-250			16 Cores	
251-262	Support Vector Machine	10,000 -120,000 (increase every 10,000)	1 Core	
263-274			2 Cores	
275-286			4 Cores	
287-298			8 Cores	
299-310			16 Cores	
311-322	Multi-Layer Perceptron	10,000 -120,000 (increase every 10,000)	1 Core	
323-334			2 Cores	
335-346			4 Cores	
347-358			8 Cores	
359-360			16 Cores	

ตารางที่ 5 กรณีทดลองทั้งหมดของงานวิจัย

รูปแบบที่ 1 : ต้นไม้ตัดสินใจ (Decision tree model)

ในรูปแบบนี้จะใช้เทคนิคต้นไม้ตัดสินใจ ในการทดลอง ซึ่งมีทั้งหมด 70 กรณี คือกรณีที่ 1-70 อ้างอิงตามตารางที่ (5) โดยมีการทดลองถึงช่วงข้อมูล 140,000 บรรทัด มากกว่ากรณีทดลองของรูปแบบอื่นๆ เนื่องจากเพื่อให้เห็นแนวโน้มของข้อมูลที่ชัดเจนมากขึ้น

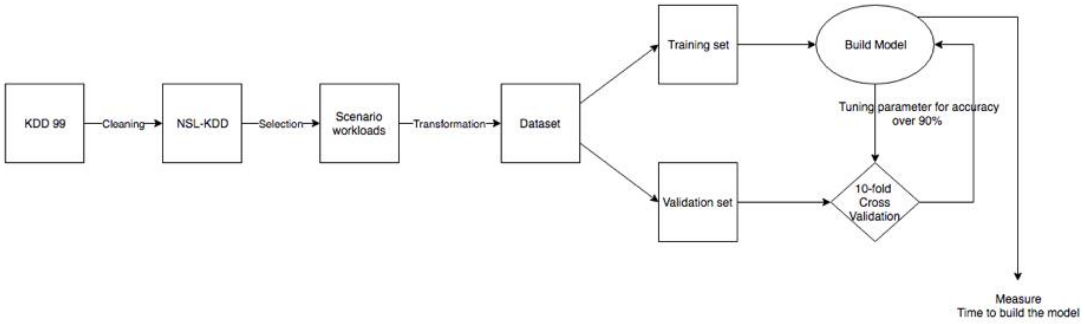
รูปแบบที่ 2 : ซัพพอร์ตเวกเตอร์แมชชีน (support vector machine model)

ในรูปแบบนี้จะใช้เทคนิคซัพพอร์ตเวกเตอร์แมชชีน ในการทดลอง ซึ่งมีทั้งหมด 60 กรณี คือ กรณีที่ 71-130 อ้างอิงตามตารางที่ (5)

รูปแบบที่ 3 : โครงข่ายประสามเทียมแบบ Multi-Layer Perceptron (Feedforward Neural Network)

ในรูปแบบนี้จะใช้เทคนิคโครงข่ายประสามเทียมแบบ Multi-Layer Perceptron ในการทดลอง ซึ่งมีทั้งหมด 60 กรณี คือกรณีที่ 131-190 อ้างอิงตามตารางที่ (5)

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY



รูปที่ 6 ขั้นตอนการทดลอง

3.2 ขั้นตอนการทดลอง

1. กลุ่มข้อมูลที่นำมาใช้คือ กลุ่มข้อมูล NSL-KDD ซึ่งเป็นกลุ่มข้อมูลที่ผ่านขั้นตอนการทำความสะอาดข้อมูล (cleaning data) มาจากกลุ่มข้อมูล KDD 99 เรียบร้อยแล้ว
2. นำกลุ่มข้อมูล NSL-KDD มาทำการคัดเลือก (Selection data) โดยวิธีการสุ่มเพื่อให้ได้จำนวนข้อมูลตามรูปแบบการทดลองของแต่ละกรณี
3. เมื่อได้จำนวนข้อมูลตามแต่ละกรณีทดลองแล้ว จึงนำข้อมูลมาเข้าขั้นตอนแปลงข้อมูล (Transformation data) โดยเปลี่ยนชนิดของคุณสมบัติให้เป็นไปตามตารางที่ (2) เพื่อให้คุณสมบัติเป็นไปตามลักษณะข้อมูล ก่อนที่จะทำการทดลอง
4. ทำการสร้างแบบจำลอง (Model) โดยปรับค่าพารามิเตอร์ เพื่อให้ได้ค่าความถูกต้องในการตรวจจับข้อมูลผิดปกติที่มากกว่า 90% โดยใช้วิธี 10-fold Cross Validation ในการวัดประสิทธิภาพ และใช้คุณสมบัติทั้งหมด โดยไม่ได้มีการปรับลดคุณสมบัติของกลุ่มข้อมูลสอน
5. ในแต่ละกรณีทดลองจะดำเนินการทดลองทั้งหมด 5 ครั้ง โดยจะวัดค่าเวลาที่ใช้ในการสร้างแบบจำลองในทุกๆครั้ง และนำข้อมูลในแต่ละกรณีมาพิจารณากำจัดค่าผิดปกติออก (Outliner) พร้อมทั้งหาค่าเฉลี่ย
6. รวบรวมข้อมูลค่าเฉลี่ยของทุกกรณีทดลอง เพื่อจะนำไปวิเคราะห์ และสรุปผลในบทต่อไป

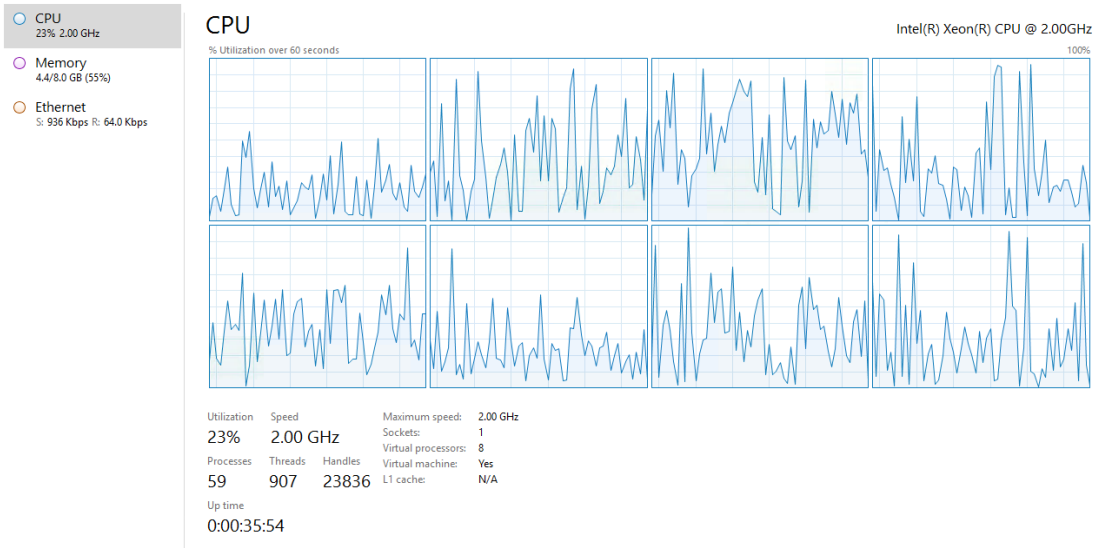
CHULALONGKORN UNIVERSITY

3.3 ลักษณะการใช้งานของหน่วยประมวลผล

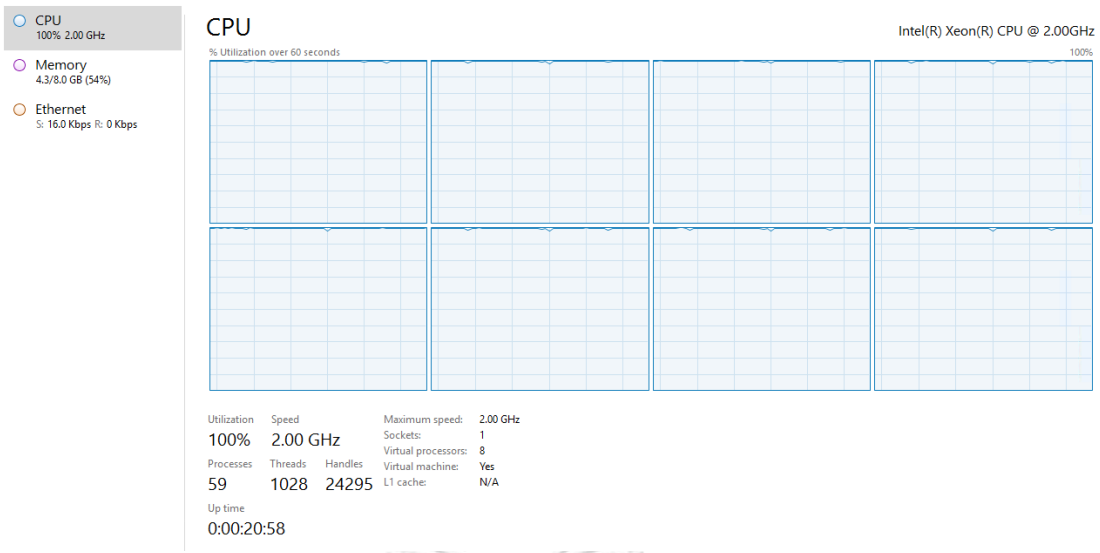
สำหรับเครื่องมือที่ใช้ในการทดลองคือ Weka นั้น โดยปกติจะมีลักษณะการทำงานที่ไม่ได้ใช้หน่วยประมวลผลทุกหน่วย ในการทำงานหนึ่งๆ หากต้องการให้การทำงานของโปรแกรมใช้หน่วยประมวลผลทุกหน่วยพร้อมกันในการทำงานหนึ่งๆนั้น เพื่อที่จะสามารถใช้งานหน่วยประมวลผลได้อย่างเต็มประสิทธิภาพ จึงจำเป็นต้องมีการเรียกใช้ API Wekaserver มาช่วยให้สามารถปรับค่าพารามิเตอร์ที่สามารถระบุจำนวนหน่วยประมวลผลที่ให้ใช้พร้อมกัน ในการทำงานหนึ่งๆได้ โดยจะใช้คำสั่งดังนี้

“java weka.RunWekaServer -host localhost -port 8085 -slot <ระบุจำนวนหน่วยประมวลผล>”

ดังเช่นตัวอย่างในรูปที่ 8 ได้ทำการระบุจำนวนหน่วยประมวลเท่ากับ 8 จะเห็นได้ว่ามีการใช้งานหน่วยประมวลผลทุกหน่วยพร้อมกันในการทำงาน ทุกหน่วยประมวลผลถูกใช้งานอย่างเต็มประสิทธิภาพ ซึ่งแตกต่างจากรูปที่ 7 ที่ไม่ได้มีการปรับค่าพารามิเตอร์ดังกล่าว อีกทั้งรูปที่ 9 แสดงให้เห็นถึง Weka ที่มีการปรับค่าพารามิเตอร์ จะมีการแตกงานออกย่อย ๆ เพื่อส่งให้แต่ละหน่วยประมวลผลทุกหน่วย ช่วยกันทำงานต่อไป



รูปที่ 7 การใช้งานหน่วยประมวลผล หากไม่มีการปรับค่าพารามิเตอร์



รูปที่ 8 การใช้งานหน่วยประมวลผล เมื่อการปรับค่าพารามิเตอร์ให้โปรแกรมใช้งานหน่วยประมวลผล 8 หน่วยพร้อมกัน

Weka Server (localhost:8085)

Number of execution slots: 8
Number of tasks executing: 8
Number of tasks queued: 0
Load adjust factor: 1.000
Server load ((#executing + #queued) * loadFactor / #execution_slots): 1.0

Memory (free/total/max) in bytes: 1,557,510,344 / 2,110,783,488 / 3,817,865,216

Tasks

Task name	ID	Server	Last execution	Next execution	Status	Purge
DecisionTable-CV_fold_1	2eb2d2ab-36b5-473b-a497-aeab9c81294a	local	2019-02-24T10:41:00	-	Finished executing	Remove
DecisionTable-CV_fold_1	4751f86c-9960-4254-8c9a-032c99a95adb	local	2019-02-24T10:45:00	-	Processing...	--
DecisionTable-CV_fold_10	830c38c0-4809-4cdc-97e9-6f86ebddc94a	local	2019-02-24T10:49:00	-	Processing...	--
DecisionTable-CV_fold_2	40bd0f43-f9bf-426e-915a-3dad883a2b6a	local	2019-02-24T10:41:00	-	Finished executing	Remove
DecisionTable-CV_fold_2	ba5ddf45-c5fa-4ebf-a530-9b7f46994843	local	2019-02-24T10:45:00	-	Finished executing	Remove
DecisionTable-CV_fold_3	e4be9fbc-c5f8-4abc-b80d-19b128a87251	local	2019-02-24T10:45:00	-	Processing...	--
DecisionTable-CV_fold_4	e69f53a2-f3ff-45b5-9ee5-23cf789809a0	local	2019-02-24T10:45:00	-	Processing...	--
DecisionTable-CV_fold_5	46a66650-645b-4490-8b97-b4c4bac6506b	local	2019-02-24T10:45:00	-	Processing...	--
DecisionTable-CV_fold_6	15803f3d-99e0-478a-84f3-b11ec9c156b7	local	2019-02-24T10:45:00	-	Processing...	--
DecisionTable-CV_fold_7	3e492e9f-acdc-47a5-b0dc-e118db64aed2	local	2019-02-24T10:45:00	-	Finished executing	Remove
DecisionTable-CV_fold_8	f951175c-14ec-4235-9ae7-c51b381fc775	local	2019-02-24T10:45:00	-	Processing...	--
DecisionTable-CV_fold_9	43557c9b-6ffa-4691-8fc8-c376d0bc6b45	local	2019-02-24T10:48:00	-	Processing...	--

รูปที่ 9 การแตกงานออกย่อยๆ ของโปรแกรม Weka

3.4 แบบจำลองเบื้องต้น

สำหรับเทคนิคการเรียนรู้ด้วยเครื่องแต่ละประเภท มีปัจจัยหลายอย่างที่มีผลต่อเวลาในการสร้างแบบจำลองระบบการตรวจจับการบุกรุกทางด้านเครือข่าย โดยมีจำนวนของข้อมูลเป็นปัจจัยสำคัญ เมื่อวิเคราะห์แล้วพบว่า เวลาในการสร้างโมเดลเกี่ยวข้องโดยตรงกับจำนวนข้อมูล ซึ่งแทนค่าด้วยตัวแปร w และจำนวนตัวประมวลผลกลาง แทนค่าด้วยตัวแปร c โดยสามารถร่างแบบจำลองเบื้องต้นของเวลาที่ใช้ในการสร้างแบบจำลองระบบ(T) ในแต่ละรูปแบบวิธี ได้ดังสมการที่ (1)

$$T_{Algo}=T(c,w) \quad (1)$$

สำหรับการศึกษาทดลองเพื่อวัดเวลาที่ใช้ในการสร้างแบบจำลองระบบที่ใช้เทคนิคการเรียนรู้ด้วยเครื่องแต่ละประเภท ตามสมการของแบบจำลองข้างต้นจะกล่าวถึงในบทถัดไป

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

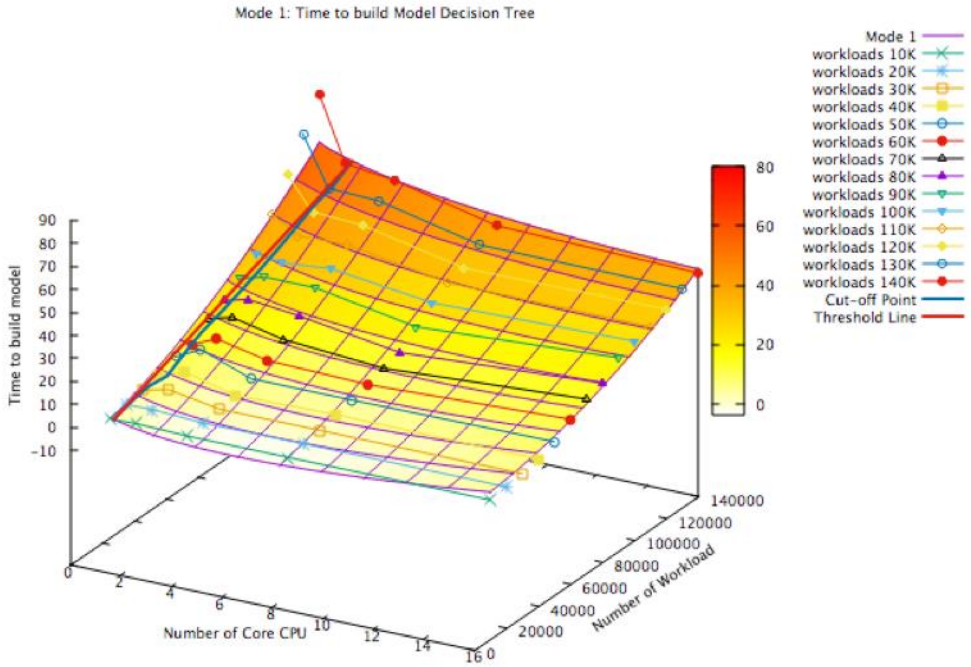
บทที่ 4
ผลการทดลอง การวิเคราะห์และสรุปผลการทดลอง

ในบทนี้จะกล่าวถึงการทดลองเพื่อวัดเวลาที่ใช้ในการสร้างแบบจำลองระบบที่ใช้เทคนิคการเรียนรู้ด้วยเครื่องแต่ละประเภท โดยปรับค่าพารามิเตอร์ของแต่ละกรณีทดสอบ ซึ่งแบ่งการทดลองทั้งหมดเป็น 3 รูปแบบตามเทคนิคการจำแนกประเภทข้อมูล ดังที่ได้กล่าวไว้ในหัวข้อที่ 3.1

4.1. รูปแบบที่ 1 : ต้นไม้ตัดสินใจ (Decision tree model)

4.1.1 ลักษณะการใช้งานของหน่วยประมวลผล: ไม่ได้มีการปรับการใช้งานของหน่วยประมวลผล

สำหรับรูปแบบนี้ จะทำการสร้างแบบจำลองระบบที่ใช้เทคนิคต้นไม้ตัดสินใจ โดยทำการทดลอง และวัดผลประสิทธิภาพด้านเวลาในการสร้างแบบจำลอง จากแบบจำลองเบื้องต้นในหัวข้อ 3.2 แสดงให้เห็นถึงพารามิเตอร์ที่ส่งผลกับเวลาที่ใช้ในรูปแบบวิธีนี้อันได้แก่ จำนวนข้อมูล (w) และจำนวนตัวประมวลผลกลาง (c) การทดลองจึงทำการวัดเวลาที่ใช้ในการสร้างแบบจำลองกับขนาดจำนวนข้อมูลต่างกันตั้งแต่ 10,000 บรรทัด จนถึง 140,000 บรรทัด บนเครื่องที่มีจำนวนตัวประมวลผลกลางตั้งแต่ 1 ถึง 16 Cores ผลการทดลองที่ได้แสดงดังรูปที่ (10)



รูปที่ 10 เวลาที่ใช้ในการสร้างแบบจำลองระบบที่ใช้เทคนิคต้นไม้ตัดสินใจ โดยไม่ได้มีการปรับการใช้งานของหน่วยประมวลผล

ในอุดมคติแล้วเมื่อจำนวนตัวประมวลผลกลางเพิ่มมากขึ้น เวลาที่ใช้ย่อมลดลง อย่างไรก็ตาม รูปที่ (10) แสดงให้เห็นว่าเวลาที่ใช้ในการสร้างแบบจำลองระบบไม่ได้ลดลงเสมอไปเมื่อใช้จำนวนตัวประมวลผลกลางในระบบมากขึ้น เวลาที่ใช้สำหรับแต่ละขนาดจำนวนข้อมูลลดลงอย่างรวดเร็วในช่วงแรก (เมื่อใช้จำนวนตัวประมวลผลกลางน้อย) แต่เมื่อเพิ่มจำนวนตัวประมวลผลกลางไปจนถึงจุดหนึ่งจะพบว่าเวลาค่อนข้างคงที่ เห็นได้ชัดว่ากราฟของเวลาถูกแบ่งเป็นสองบริเวณคือ บริเวณที่เมื่อเพิ่มจำนวนตัวประมวลผลกลางแล้วจะส่งผลให้เวลาที่ใช้ลดลงและบริเวณที่เมื่อเพิ่มจำนวนตัวประมวลผลกลางแล้วเวลาที่ใช้ค่อนข้างคงที่ สังเกตได้ว่ากราฟของเวลาสำหรับแต่ละขนาดของจำนวนข้อมูลจะมีจุดแบ่ง (Cut-Off Point) ซึ่งเมื่อจำนวนตัวประมวลผลกลางของเครื่องมากกว่าจุดนี้จะทำให้การเพิ่มจำนวนตัวประมวลผลกลางไม่เป็นประโยชน์ในการที่จะทำให้เวลาการสร้างแบบจำลองระบบลดลง

เมื่อพิจารณาบริเวณที่เมื่อเพิ่มจำนวนตัวประมวลผลกลางแล้วจะส่งผลให้เวลาที่ใช้ลดลงอย่างรวดเร็ว โดยมีแนวโน้มการลดลงแบบ Exponential สมการสำหรับประมาณเวลาที่ใช้ในการสร้างแบบจำลอง(T_{DT}) ในรูปของฟังก์ชันที่แปรผันตามจำนวนตัวประมวลผลกลางของเครื่อง (c) และจำนวนข้อมูล (w) สามารถแสดงได้ดังนี้

$$T_{DT}(c,w) = k_1c^m + k_2w^2 + k_3(cw)^n + k_4c + k_5w + k_6cw + k_7 \qquad (2)$$

จากสมการ (2) สามารถประมาณค่าสัมประสิทธิ์ k_1-k_7,n และ m จากระเบียบวิธีกำลังสองน้อยที่สุด (Least Square Method) ได้ดังนี้

k_1 = -11.6433 k_2 = 1.55986e-09

k_3 = 5.03903 k_4 = 5.19553

k_5 = 0.00022554 k_6 = -1.1694e-05s

k_7 = 2.72791 m = 0.716225

n = -0.0590303

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ผลการทดลองวัดเวลาที่ใช้ในการสร้างแบบจำลองระบบเทียบกับจำนวนตัวประมวลผลกลางของเครื่องและจำนวนข้อมูล สามารถแสดงดังรูปที่ (10) เส้นตรงทึบสีม่วงแสดงถึงแบบจำลองของสมการที่ใช้ประมาณเวลาในการสำรองข้อมูล $T_{DT}(w,c)$ ในบริเวณแรกซึ่งลดลงอย่างรวดเร็วเมื่อเพิ่มจำนวนตัวประมวลผลกลางของเครื่องเส้นตรงทึบสีน้ำเงินเป็นเส้นเชื่อมจุดแบ่ง (Cut-off point) ของกราฟเวลาที่ใช้ในแต่ละขนาดจำนวนข้อมูล สมการของเส้นขีดแบ่ง (Threshold Line) ในรูปของสมการเวกเตอร์ (Vector Equation) ซึ่งเป็นสมการเส้นตรงซึ่งประมาณค่าจุดแบ่งของกราฟเวลาที่ใช้ในแต่ละขนาดจำนวนข้อมูล แสดงด้วยเส้นตรงทึบสีแดงของรูปที่ (10) มีสมการดังนี้

$$\begin{pmatrix} c \\ w \\ T_{DT} \end{pmatrix} = \begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix} + t \begin{pmatrix} b_1 \\ b_2 \\ b_3 \end{pmatrix} \quad (3)$$

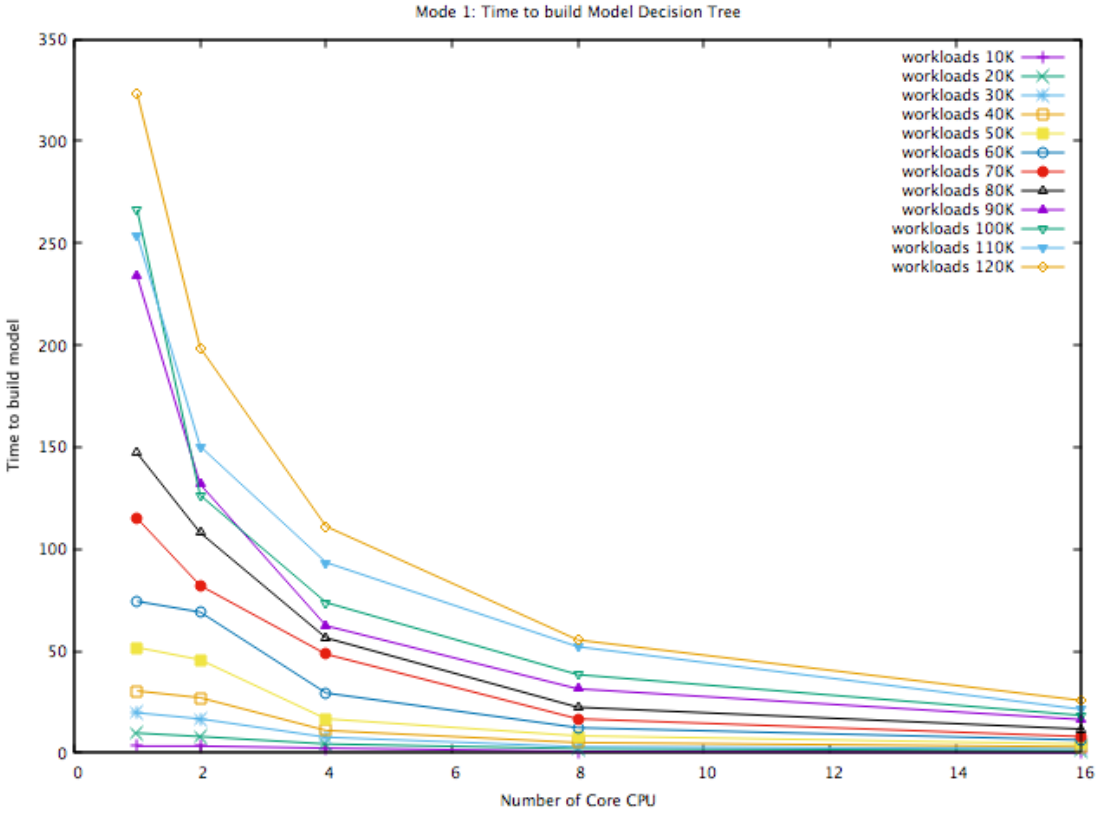
โดยมีค่าสัมประสิทธิ์ดังนี้

a1 = 1.1	a2 = 10,000	a3 = 1.1691
b1 = 2.3	b2 = 140,000	b3 = 54.3021

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

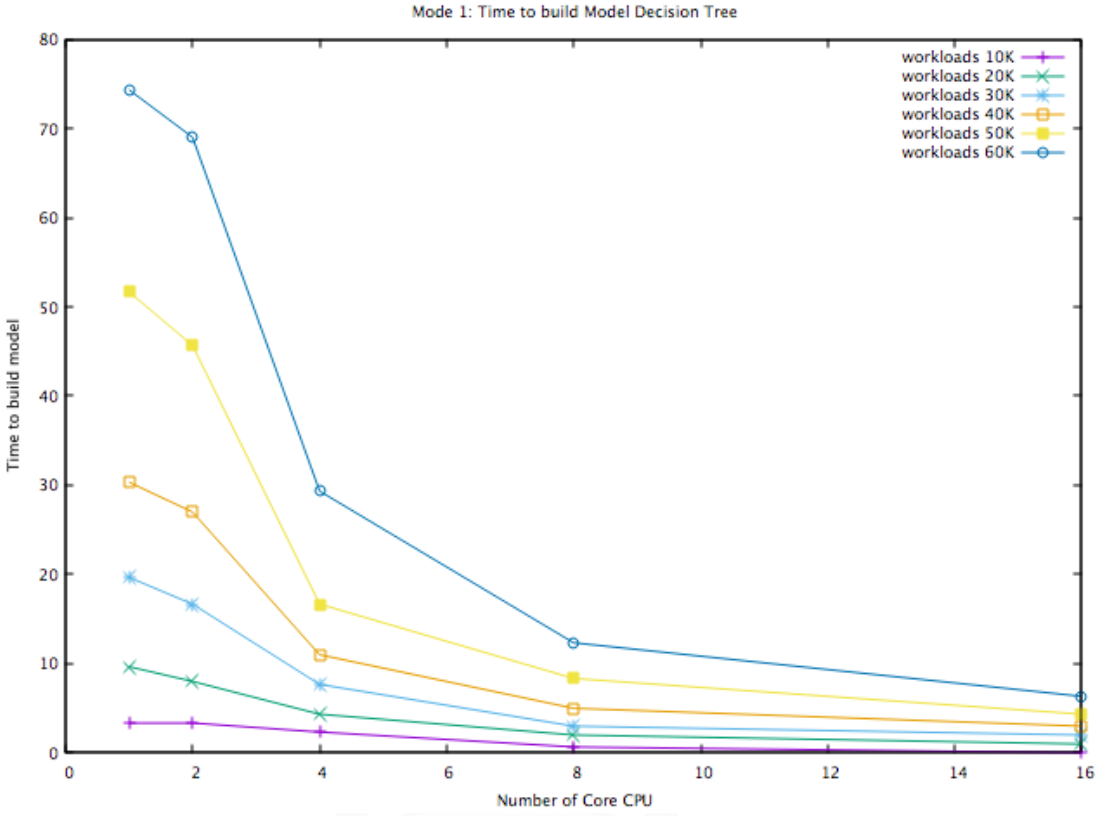
4.1.2 ลักษณะการใช้งานของหน่วยประมวลผล: มีการปรับให้สามารถใช้งานหน่วยประมวลผลได้ทุกหน่วยในเวลาเดียวกัน

สำหรับรูปแบบนี้ จะทำการสร้างแบบจำลองระบบที่ใช้เทคนิคต้นไม้ตัดสินใจ โดยทำการทดลอง และวัดผลประสิทธิภาพด้านเวลาในการสร้างแบบจำลอง จากแบบจำลองเบื้องต้นในหัวข้อ 3.2 แสดงให้เห็นถึงพารามิเตอร์ที่ส่งผลกับเวลาที่ใช้ในรูปแบบวิธีนี้อันได้แก่ จำนวนข้อมูล (w) และจำนวนตัวประมวลผลกลาง (c) การทดลองจึงทำการวัดเวลาที่ใช้ในการสร้างแบบจำลองกับขนาดจำนวนข้อมูลต่างกันตั้งแต่ 10,000 บรรทัด จนถึง 120,000 บรรทัด บนเครื่องที่มีจำนวนตัวประมวลผลกลางตั้งแต่ 1 ถึง 16 Cores และได้มีการปรับค่าพารามิเตอร์ให้สามารถใช้งานหน่วยประมวลผลทุกหน่วยได้ในเวลาเดียวกัน โดยผลการทดลองที่ได้แสดงดังรูปที่ (11)



รูปที่ 11 เวลาที่ใช้ในการสร้างแบบจำลองระบบ(เทคนิคต้นไม้ตัดสินใจ) โดยปรับค่าการใช้งานหน่วยประมวลผล

จากผลการทดลองจะเห็นได้ว่า แนวโน้มของเวลาที่ใช้ในการสร้างแบบจำลองนั้นมีลักษณะ 2 แบบ คือแนวโน้มของเวลาที่ใช้ในการสร้างแบบจำลองของช่วงจำนวนข้อมูลน้อยกว่าเท่ากับ 60,000 บรรทัด และแนวโน้มของช่วงจำนวนข้อมูลมากกว่า 60,000 บรรทัด ซึ่งสามารถแสดงได้ดังรูป 12 และ 13 ตามลำดับ



รูปที่ 12 เวลาที่ใช้ในการสร้างแบบจำลองระบบ(เทคนิคต้นไม้ตัดสินใจ) โดยปรับค่าการใช้งานหน่วยประมวลผลในช่วงข้อมูลน้อยกว่าเท่ากับ 60,000 บรรทัด

เมื่อพิจารณาผลการทดลองในช่วงข้อมูลน้อยกว่าเท่ากับ 60,000 บรรทัดจะเห็นได้ว่าเมื่อเพิ่มจำนวนตัวประมวลผลกลางแล้วจะส่งผลให้เวลาที่ใช้มีแนวโน้มการลดลงแบบ Exponential แต่แนวโน้มของช่วงข้อมูลนี้ จะสังเกตได้ว่า เมื่อใช้หน่วยประมวลผลกลาง 1 หน่วย และ 2 หน่วย เวลาที่ใช้ในการสร้างแบบจำลองนั้นมีแนวโน้มลดลงไม่มาก แต่หากเพิ่มหน่วยประมวลผลกลางในการทำงานเป็น 4 หน่วย กลับส่งผลให้แนวโน้มของเวลาในการสร้างแบบจำลองนั้นมีการลดลงอย่างรวดเร็ว แต่เมื่อเพิ่มหน่วยประมวลผลกลางมากขึ้นเป็น 16 หน่วย จะเห็นได้ว่าการเพิ่มนั้น ไม่ได้ช่วยเพิ่มให้แนวโน้มของการใช้เวลาในการทำงานลดลงไปด้วย อีกทั้งยังทำให้จำนวนข้อมูลมีผลต่อเวลาในการสร้างแบบจำลองน้อยลงมาก เมื่อเทียบกับเวลาที่ใช้สร้างแบบจำลองในสภาวะที่มีหน่วยประมวลผลกลางจำนวนน้อย ๆ

ซึ่งสมการสำหรับประมาณเวลาที่ใช้ในการสร้างแบบจำลอง($T_{DT\_Tuning \le 6}$) ในรูปของฟังก์ชันที่แปรผันตามจำนวนตัวประมวลผลกลางของเครื่อง (c) และจำนวนข้อมูล (w) สามารถแสดงได้ดังนี้

$$T_{DT\_Tuning \le 6}(c,w) = k_1c^m+k_2w^2+k_3(cw)^n+k_4c+k_5w+k_6cw+k_7 \quad (4)$$

จากสมการ (4) สามารถประมาณค่าสัมประสิทธิ์ k_1-k_7,n และ m จากระเบียบวิธีกำลังสองน้อยที่สุด (Least Square Method) ได้ดังนี้

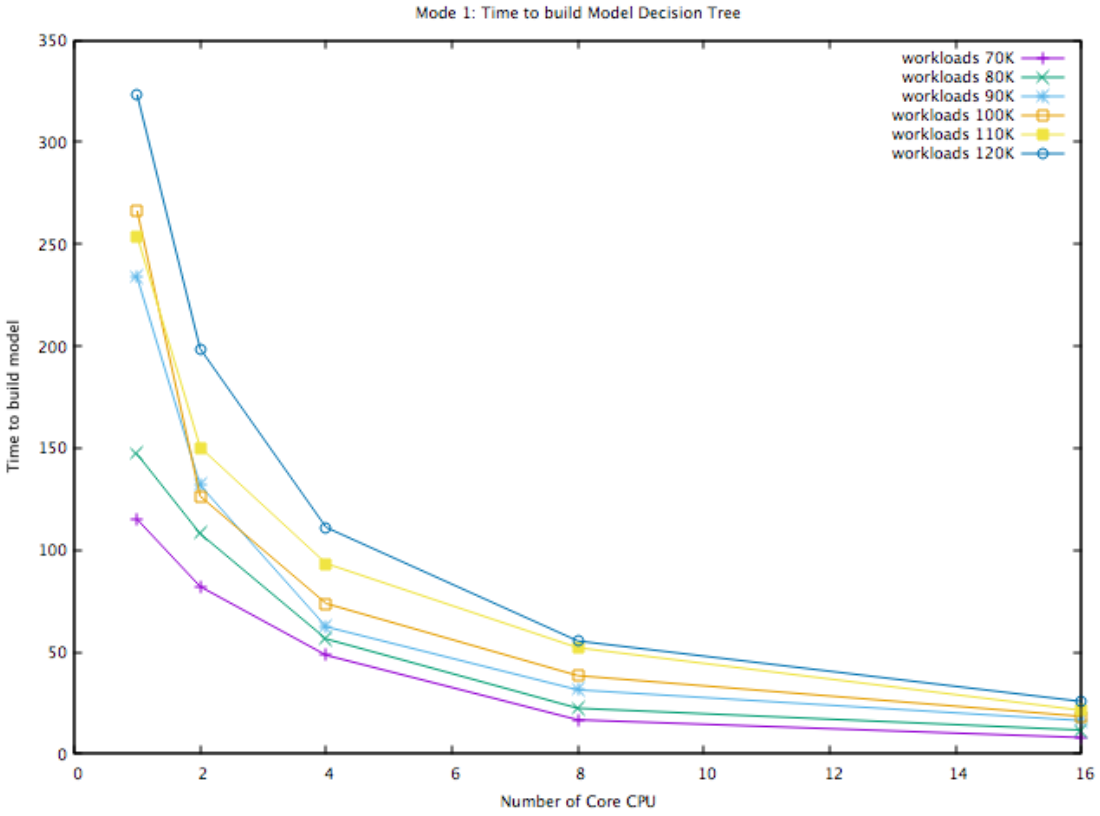
k_1 = 2.72645 k_2 = 1.25359e-08

k_3 = 5.61034 k_4 = -8.36903

k_5 = 0.00034239 k_6 = -8.21754e-05

k_7 = 4.53902 m = 1.42662

n = -0.00815126



รูปที่ 13 เวลาที่ใช้ในการสร้างแบบจำลองระบบ(เทคนิคต้นไม้ตัดสินใจ) โดยปรับค่าการใช้งานหน่วยประมวลผลในช่วงข้อมูลมากกว่า 60,000 บรรทัด

เมื่อพิจารณาผลการทดลองในช่วงข้อมูลมากกว่า 60,000 บรรทัดจะเห็นได้ว่าเมื่อเพิ่มจำนวนตัวประมวลผลกลางแล้วจะส่งผลให้เวลาที่ใช้มีแนวโน้มการลดลงแบบ Exponential เช่นเดียวกับช่วงข้อมูลน้อยกว่าเท่ากับ 60,000 บรรทัด แต่แนวโน้มของช่วงข้อมูลนี้ จะสังเกตได้ว่า เมื่อเริ่มเพิ่มหน่วยประมวลผลกลาง ส่งผลให้เวลาที่ใช้ในการสร้างแบบจำลองนั้นมีแนวโน้มลดลงอย่างรวดเร็วในทันที แตกต่างจากแนวโน้มของช่วงข้อมูลน้อยกว่าเท่ากับ 60,000 บรรทัด แต่เมื่อเพิ่มหน่วยประมวลผลกลางเป็น 16 หน่วย จะเห็นได้ว่าการเพิ่มนั้น ไม่ได้ช่วยเพิ่มให้แนวโน้มของการใช้เวลาในการทำงานลดลงมากนัก อีกทั้งยังทำให้จำนวนข้อมูลมีผลต่อเวลาในการสร้างแบบจำลองน้อยลงมาก เมื่อเทียบกับเวลาที่ใช้สร้างแบบจำลองในสภาวะที่มีหน่วยประมวลผลกลางจำนวนน้อยๆ

ซึ่งสมการสำหรับประมาณเวลาที่ใช้ในการสร้างแบบจำลอง($T_{DT\_Tuning\ >6}$) ในรูปของฟังก์ชันที่แปรผันตามจำนวนตัวประมวลผลกลางของเครื่อง (c) และจำนวนข้อมูล (w) สามารถแสดงได้ดังนี้

$$T_{DT\_Tuning\ >6}(c,w)= k_1c^m+k_2w^2+k_3(cw)^n+k_4c+k_5w+k_6cw+k_7 \qquad (5)$$

จากสมการ (5) สามารถประมาณค่าสัมประสิทธิ์ k_1-k_7,n และ m จากกระเบียบวิธีกำลังสองน้อยที่สุด (Least Square Method) ได้ดังนี้

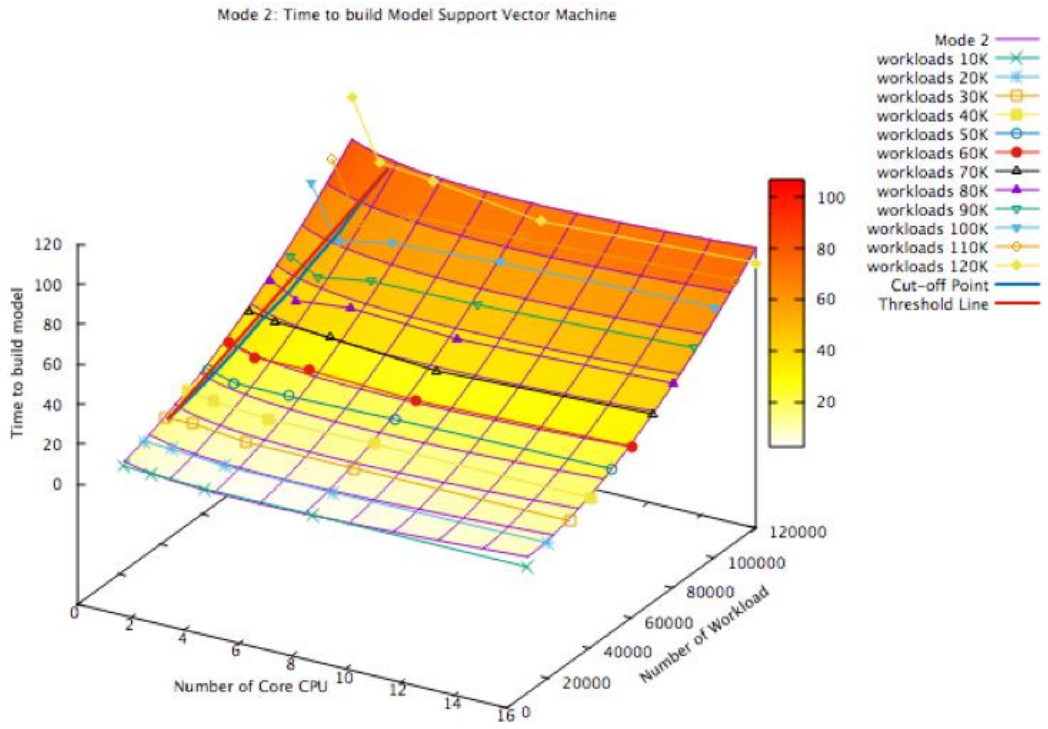
k_1	= 426.252	k_2	= -7.29334e-10
k_3	= 12.6312	k_4	= -488.067
k_5	= 0.00298259	k_6	= -0.000184742
k_7	= 0.0931737	m	= 1.04978
n	=-0.0666003		

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

4.2. รูปแบบที่ 2 : ซัพพอร์ตเวกเตอร์แมชชีน (support vector machine model)

4.2.1 ลักษณะการใช้งานของหน่วยประมวลผล: ไม่ได้มีการปรับการใช้งานของหน่วยประมวลผล

สำหรับรูปแบบนี้ จะทำการสร้างแบบจำลองระบบที่ใช้เทคนิคซัพพอร์ตเวกเตอร์แมชชีน โดยทำการทดลอง และวัดผลประสิทธิภาพด้านเวลาในการสร้างแบบจำลอง จากแบบจำลองเบื้องต้นในหัวข้อ 3.2 แสดงให้เห็นถึงพารามิเตอร์ที่ส่งผลกับเวลาที่ใช้ในรูปแบบวิธีนี้อันได้แก่ จำนวนข้อมูล (w) และจำนวนตัวประมวลผลกลาง (c) การทดลองจึงทำการวัดเวลาที่ใช้ในการสร้างแบบจำลองกับขนาดจำนวนข้อมูลต่างกันตั้งแต่ 10,000 บรรทัด จนถึง 120,000 บรรทัด บนเครื่องที่มีจำนวนตัวประมวลผลกลางตั้งแต่ 1 ถึง 16 Cores ผลการทดลองที่ได้แสดงดังรูป (14)



รูปที่ 14 เวลาที่ใช้ในการสร้างแบบจำลองระบบที่ใช้เทคนิคซัพพอร์ตเวกเตอร์แมชชีน

จากการทดลอง พบว่าเวลาที่ใช้ในการสร้างแบบจำลองระบบไม่ได้ลดลงเสมอไปเมื่อใช้จำนวนตัวประมวลผลกลางในระบบมากขึ้น เวลาที่ใช้สำหรับแต่ละขนาดจำนวนข้อมูลลดลงอย่างรวดเร็วในช่วงแรก เช่นเดียวกับรูปแบบที่ 1 ที่ได้ทำการทดลองโดยใช้เทคนิคต้นไม้ตัดสินใจ หากเมื่อพิจารณาแล้ว จะได้สมการสำหรับประมาณเวลาที่ใช้ในการสร้างแบบจำลอง(T_{SVM}) ในรูปของฟังก์ชันที่แปรผันตามจำนวนตัวประมวลผลกลางของเครื่อง (c) และจำนวนข้อมูล (w) ดังนี้

$$T_{SVM}(c,w) = k_1 c^m + k_2 w^2 + k_3 (cw)^n + k_4 c + k_5 w + k_6 cw + k_7 \quad (6)$$

จากสมการ (6) สามารถประมาณค่าสัมประสิทธิ์ k_1-k_7,n และ m จากระเบียบวิธีกำลังสองน้อยที่สุด (Least Square Method) ได้ดังนี้

k_1	= -50.5927	k_2	= 3.34379e-09
k_3	= -0.0127763	k_4	= 1.47306
k_5	= 0.000325611	k_6	= 9.87862e-06
k_7	= 54.7895	m	= 0.554196
n	= 0.0939622		

ผลการทดลองวัดเวลาที่ใช้ในการสร้างแบบจำลองระบบเทียบกับจำนวนตัวประมวลผลกลางของเครื่องและจำนวนข้อมูล สามารถแสดงดังรูปที่ (14) เส้นตรงทึบสีม่วงแสดงถึงแบบจำลองของสมการที่ใช้ประมาณเวลาในการสำรองข้อมูล T_{svm}(w,c) ในบริเวณแรกซึ่งลดลงอย่างรวดเร็วเมื่อเพิ่มจำนวนตัวประมวลผลกลางของเครื่องเส้นตรงทึบสีน้ำเงินเป็นเส้นเชื่อมจุดแบ่ง (Cut-off point) ของกราฟเวลาที่ใช้ในแต่ละขนาดจำนวนข้อมูล สมการของเส้นขีดแบ่ง (Threshold Line) ในรูปของสมการเวกเตอร์ (Vector Equation) ซึ่งเป็นสมการเส้นตรงซึ่งประมาณค่าจุดแบ่งของกราฟเวลาที่ใช้ในแต่ละขนาดจำนวนข้อมูล แสดงด้วยเส้นตรงทึบสีแดงของรูปที่ (14) มีสมการดังนี้

$$\begin{pmatrix} c \\ w \\ T_{svm} \end{pmatrix} = \begin{pmatrix} a_1 \\ a_2 \\ a_3 \end{pmatrix} + t \begin{pmatrix} b_1 \\ b_2 \\ b_3 \end{pmatrix} \quad (7)$$

โดยมีค่าสัมประสิทธิ์ดังนี้

a1	= 1.1	a2	= 30,000	a3	= 13.7351
b1	= 2.5	b2	= 120,000	b3	= 74.7293

จากสมการ (3) และ (7) สามารถจัดรูปสมการใหม่ได้ โดยเขียนความสัมพันธ์ของจำนวนตัวประมวลผลกลางของเครื่องในรูปฟังก์ชันของขนาดของจำนวนข้อมูล (Cartesian Equation) ดังแสดงในสมการ (8) ซึ่งสามารถนำมาใช้ประมาณหรือทำนายจำนวนตัวประมวลผลกลางที่เพียงพอและเหมาะสมสำหรับการสร้างแบบจำลองระบบตรวจสอบการบุกรุกที่ใช้เทคนิคต้นไม้ตัดสินใจและเทคนิคซัพพอร์ตเวกเตอร์แมชชีน โดยกำหนดขนาดของจำนวนข้อมูล ซึ่งเมื่อเพิ่มจำนวนตัวประมวลผลกลางมากกว่าค่านี้แล้วเวลาที่ใช้แทบจะไม่ลดลงเลย

$$c = \left(\frac{b_1}{b_2}\right)(w - a_2) + a_1 \qquad (8)$$

สำหรับ เทคนิคต้นไม้ตัดสินใจ

a1 = 1.1 a2 = 10,000

b1 = 2.3 b2 = 140,000

สำหรับ เทคนิคซัพพอร์ตเวกเตอร์แมชชีน

a1 = 1.1 a2 = 30,000

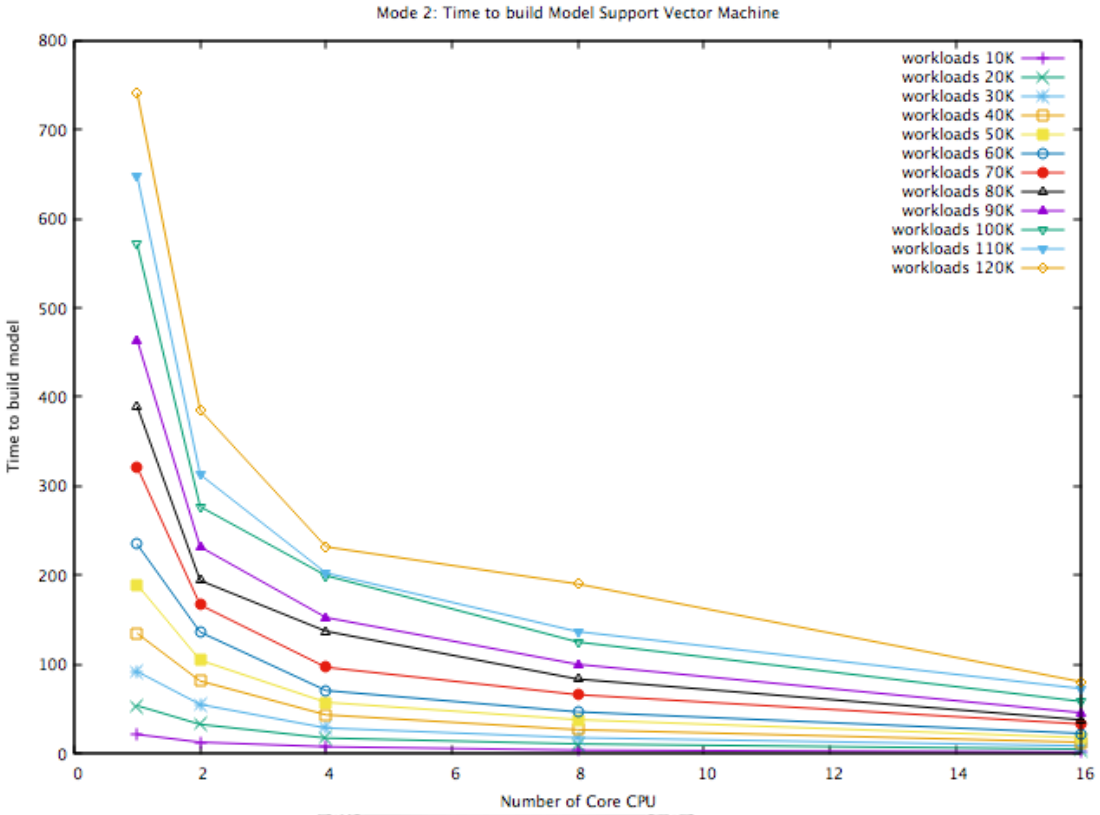
b1 = 2.5 b2 = 120,000

ยกตัวอย่างเช่น เมื่อมีจำนวนข้อมูล 50,000 บรรทัด (w = 50000) และใช้เทคนิคซัพพอร์ตเวกเตอร์แมชชีน เมื่อประมาณด้วยสมการ (8) จะให้ค่า c = 1.51667 นั่นคือในการสร้างระบบการตรวจจับการบุกรุกที่จำนวนข้อมูลดังกล่าว จำนวนตัวประมวลผลกลางของเครื่องที่มากเพียงพอเพื่อที่จะใช้เวลาน้อยที่สุดคือ 2 Cores เนื่องจากเมื่อเพิ่มจำนวนมากกว่านี้จะไม่เกิดประโยชน์ในการที่จะช่วยลดเวลาอีกต่อไป

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

หากพิจารณาแล้ว จะพบว่าเมื่อจำนวนข้อมูลเพิ่มขึ้น ค่าประมาณจำนวนตัวประมวลผลกลางจะมีค่ามากขึ้นตามด้วย อีกทั้ง (a1,a2) และ (b1,b2) เป็นจุดตัดแบ่ง(Cut-off point) ของกราฟรูปที่ (10),(14) ซึ่ง a1,b1 เป็นค่าจำนวนตัวประมวลผลกลางที่ใช้ ที่แปรผันตรง เมื่อจำนวนข้อมูลเท่ากับ b1,b2 ตามลำดับ จึงเห็นได้ว่า แบบจำลองที่ใช้ในการประมาณจำนวนตัวประมวลผลกลางที่เพียงพอและเหมาะสมสำหรับการสร้างระบบตรวจสอบการบุกรุกมีความสัมพันธ์โดยตรงกับจำนวนข้อมูล

4.2.2 ลักษณะการใช้งานของหน่วยประมวลผล: มีการปรับให้สามารถใช้งานหน่วยประมวลผลได้ทุกหน่วยในเวลาเดียวกัน



รูปที่ 15 เวลาที่ใช้ในการสร้างแบบจำลองระบบ(เทคนิคซัพพอร์ตเวกเตอร์แมชชีน) โดยปรับค่าการใช้งานหน่วยประมวลผล

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

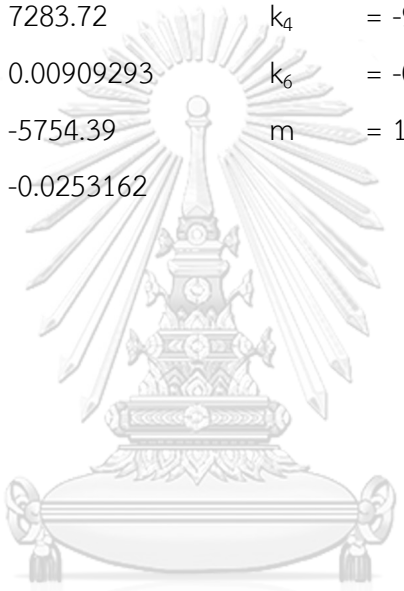
เมื่อพิจารณาผลการทดลองจะเห็นได้ว่าเมื่อเพิ่มจำนวนตัวประมวลผลกลางแล้วจะส่งผลให้เวลาที่ใช้มีแนวโน้มการลดลงแบบ Exponential โดยเมื่อเริ่มเพิ่มหน่วยประมวลผลกลาง จะส่งผลให้เวลาที่ใช้ในการสร้างแบบจำลองนั้นมีแนวโน้มลดลงอย่างรวดเร็วในทันที แต่เมื่อเพิ่มหน่วยประมวลผลกลางมากขึ้นเป็น 16 หน่วย จะเห็นได้ว่าการเพิ่มนั้น ไม่ได้ช่วยเพิ่มให้แนวโน้มของการใช้เวลาในการทำงานลดลงมากนัก อีกทั้งยังทำให้จำนวนข้อมูลมีผลต่อเวลาในการสร้างแบบจำลองน้อยลงมาก เมื่อเทียบกับเวลาที่ใช้สร้างแบบจำลองในสภาวะที่มีหน่วยประมวลผลกลางจำนวนน้อย ๆ

ซึ่งสมการสำหรับประมาณเวลาที่ใช้ในการสร้างแบบจำลอง(T_{SVM_Tuning}) ในรูปของฟังก์ชันที่แปรผันตามจำนวนตัวประมวลผลกลางของเครื่อง (c) และจำนวนข้อมูล (w) สามารถแสดงได้ดังนี้

$$T_{SVM\_Tuning}(c,w) = k_1c^m + k_2w^2 + k_3(cw)^n + k_4c + k_5w + k_6cw + k_7 \quad (9)$$

จากสมการ (9) สามารถประมาณค่าสัมประสิทธิ์ k_1-k_7,n และ m จากระเบียบวิธีกำลังสองน้อยที่สุด (Least Square Method) ได้ดังนี้

k_1	= 1010.65	k_2	= -1.29748e-08
k_3	= 7283.72	k_4	= -995.094
k_5	= 0.00909293	k_6	= -0.000299281
k_7	= -5754.39	m	= 1.00395
n	= -0.0253162		

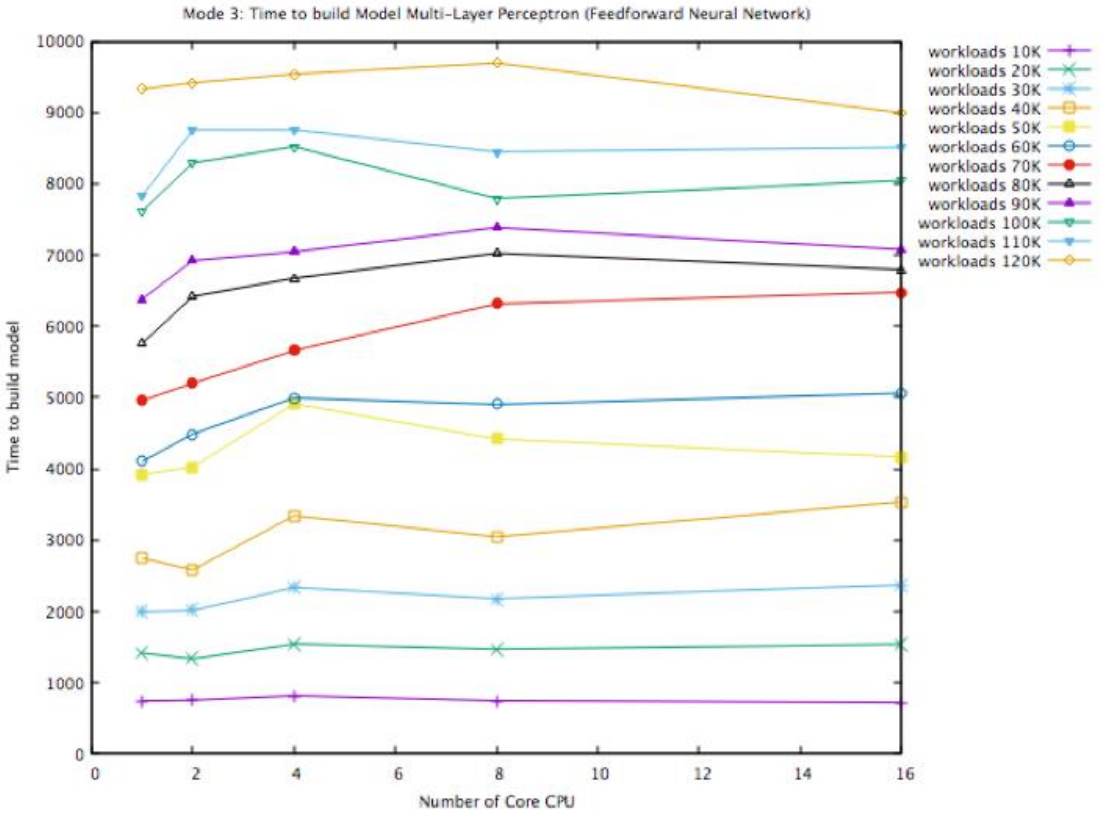


จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

4.3. รูปแบบที่ 3 : โครงข่ายประสามเทียมแบบ Multi-Layer Perceptron (Feedforward Neural Network)

4.3.1 ลักษณะการใช้งานของหน่วยประมวลผล: ไม่ได้มีการปรับการใช้งานของหน่วยประมวลผล

สำหรับรูปแบบนี้ ได้ทำการทดลองเช่นเดียวกับรูปแบบที่ 1 และ2 โดยใช้เทคนิคโครงข่ายประสาทเทียมแบบ Multi-Layer Perceptron ผลการทดลองที่ได้แสดงดังรูป (16)



รูปที่ 16 เวลาที่ใช้ในการสร้างแบบจำลองระบบที่ใช้เทคนิคโครงข่ายประสามเทียมแบบ Multi-Layer Perceptron (Feedforward Neural Network)

เมื่อวิเคราะห์ผลจากการทดลองในรูปแบบนี้แล้ว พบว่าไม่ได้มีแนวโน้มการลดลงของเวลาในการสร้างแบบจำลอง เช่นเดียวกับรูปแบบการทดลองที่ 1 และ 2 ที่มีแนวโน้มลดลงแบบ Exponential แต่จะเห็นได้ว่า ประสิทธิภาพด้านเวลาในการสร้างแบบจำลองรูปแบบนี้มีสัดส่วนโดยตรงกับจำนวนของข้อมูล เพราะเมื่อเพิ่มจำนวนตัวประมวลผลกลางมากขึ้น แทบจะไม่ได้ช่วยลดเวลาลงไป โดยพิจารณาผลการทดลองแล้ว จะได้สมการสำหรับประมาณเวลาที่ใช้ในการสร้างแบบจำลอง(T_{FNN}) ในรูปของฟังก์ชันที่แปรผันตามจำนวนตัวประมวลผลกลางของเครื่อง (c) และจำนวนข้อมูล (w) ดังนี้

$$T_{FNN}(c,w) = k_1 c + k_2 w + k_3 \quad (10)$$

จากสมการ (10) สามารถประมาณค่าสัมประสิทธิ์ k_1-k_3 จากระเบียบวิธีกำลังสองน้อยที่สุด (Least Square Method) ได้ดังนี้

k_1 = 155.223

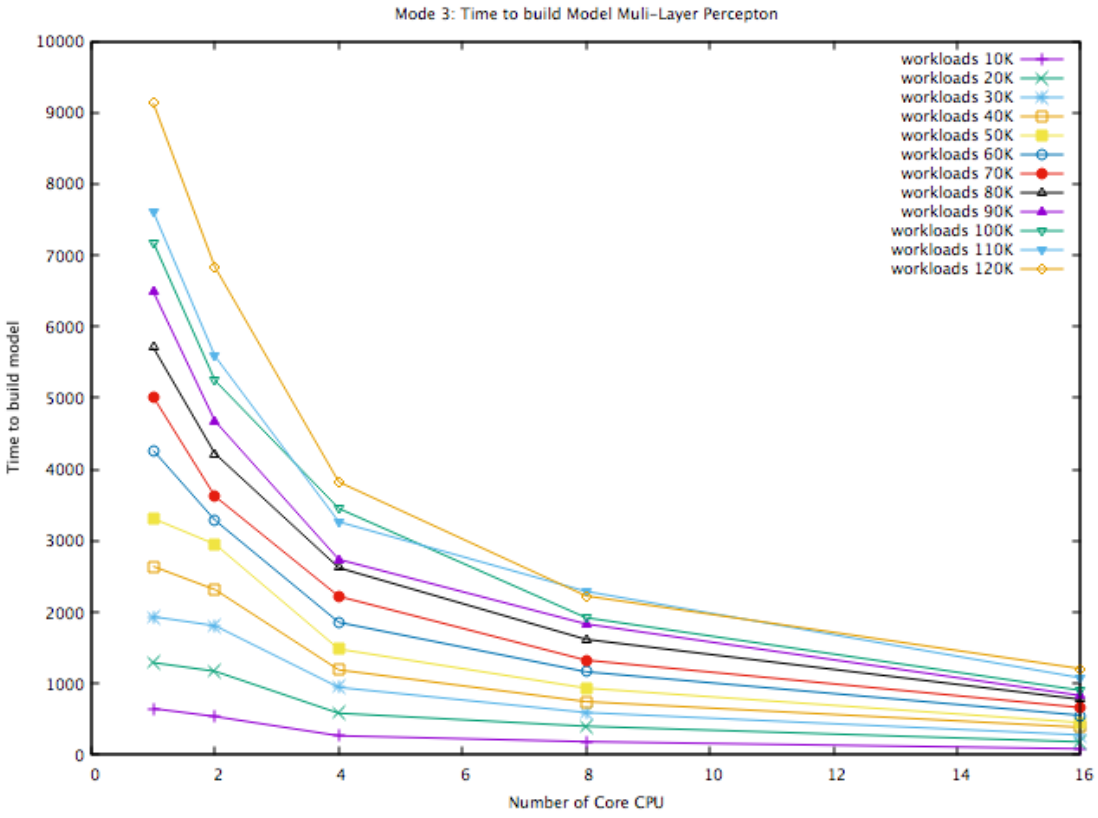
k_2 = 0.0787788

k_3 = -458.342

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

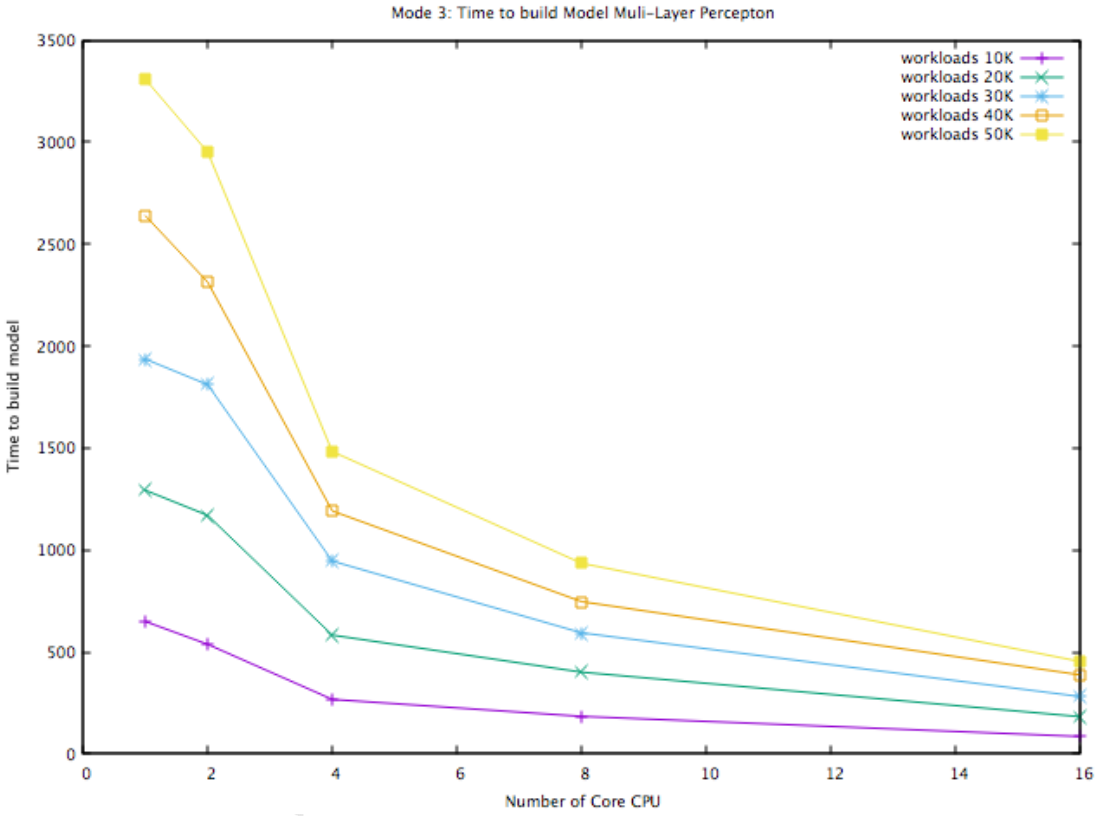
4.3.2 ลักษณะการใช้งานของหน่วยประมวลผล: มีการปรับให้สามารถใช้งานหน่วยประมวลผลได้ทุกหน่วยในเวลาเดียวกัน

สำหรับรูปแบบนี้ จะทำการสร้างแบบจำลองระบบที่ใช้เทคนิคโครงข่ายประสาทเทียม โดยทำการทดลอง และวัดผลประสิทธิภาพด้านเวลาในการสร้างแบบจำลอง จากแบบจำลองเบื้องต้นในหัวข้อ 3.2 แสดงให้เห็นถึงพารามิเตอร์ที่ส่งผลกับเวลาที่ใช้ในรูปแบบวิธีนี้อันได้แก่ จำนวนข้อมูล (w) และจำนวนตัวประมวลผลกลาง (c) การทดลองจึงทำการวัดเวลาที่ใช้ในการสร้างแบบจำลองกับขนาดจำนวนข้อมูลต่างกันตั้งแต่ 10,000 บรรทัด จนถึง 120,000 บรรทัด บนเครื่องที่มีจำนวนตัวประมวลผลกลางตั้งแต่ 1 ถึง 16 Cores และได้มีการปรับค่าพารามิเตอร์ให้สามารถใช้งานหน่วยประมวลผลทุกหน่วยได้ในเวลาเดียวกัน โดยผลการทดลองที่ได้แสดงดังรูปที่ (17)



รูปที่ 17 เวลาที่ใช้ในการสร้างแบบจำลองระบบ(เทคนิคโครงข่ายประสาทเทียม) โดยปรับค่าการใช้งานหน่วยประมวลผล

จากผลการทดลองจะเห็นได้ว่า แนวโน้มของเวลาที่ใช้ในการสร้างแบบจำลองนั้นมีลักษณะ 2 แบบ คือแนวโน้มของเวลาที่ใช้ในการสร้างแบบจำลองของช่วงจำนวนข้อมูลน้อยกว่าเท่ากับ 50,000 บรรทัด และแนวโน้มของช่วงจำนวนข้อมูลมากกว่า 50,000 บรรทัด ซึ่งสามารถแสดงได้ดังรูป 18 และ 19 ตามลำดับ



รูปที่ 18 เวลาที่ใช้ในการสร้างแบบจำลองระบบ (เทคนิคโครงข่ายประสาทเทียม) โดยปรับค่าการใช้งานหน่วยประมวลผลในช่วงข้อมูลน้อยกว่าเท่ากับ 50,000 บรรทัด

เมื่อพิจารณาผลการทดลองในช่วงข้อมูลน้อยกว่าเท่ากับ 50,000 บรรทัดจะเห็นได้ว่าเมื่อเพิ่มจำนวนตัวประมวลผลกลางแล้วจะส่งผลให้เวลาที่ใช้มีแนวโน้มการลดลงแบบ Exponential แต่แนวโน้มของช่วงข้อมูลนี้ จะสังเกตได้ว่า เมื่อใช้หน่วยประมวลผลกลาง 1 หน่วย และ 2 หน่วย เวลาที่ใช้ในการสร้างแบบจำลองนั้นมีแนวโน้มลดลงไม่มาก แต่หากเพิ่มหน่วยประมวลผลกลางในการทำงานเป็น 4 หน่วย กลับส่งผลให้แนวโน้มของเวลาในการสร้างแบบจำลองนั้นมีการลดลงอย่างรวดเร็ว และเมื่อมีการเพิ่มหน่วยประมวลผลกลางมากขึ้นเป็น 16 หน่วย จะเห็นได้ว่าการเพิ่มนั้น ไม่ได้ช่วยเพิ่มให้แนวโน้มของการใช้เวลาในการทำงานลดลงไปมากนัก

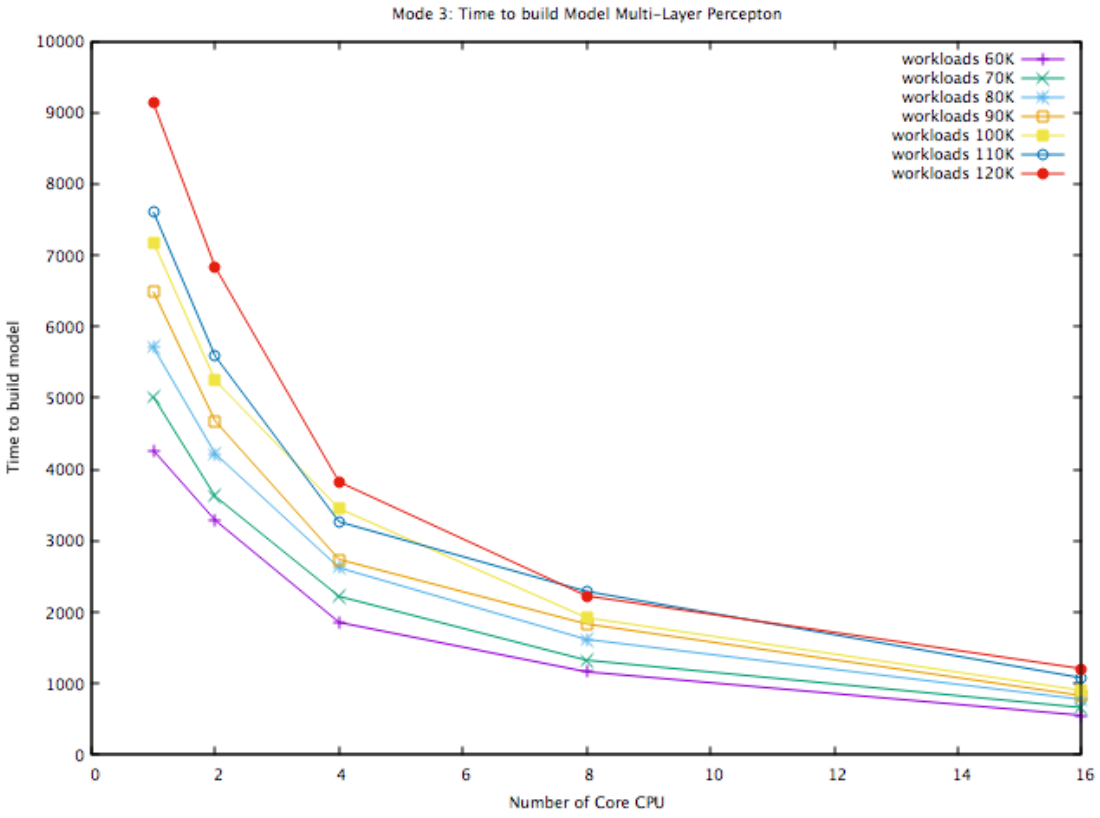
อีกทั้งยังทำให้จำนวนข้อมูลมีผลต่อเวลาในการสร้างแบบจำลองน้อยลงมาก เมื่อเทียบกับเวลาที่ใช้สร้างแบบจำลองในสภาวะที่มีหน่วยประมวลผลกลางจำนวนน้อยๆ

ซึ่งสมการสำหรับประมาณเวลาที่ใช้ในการสร้างแบบจำลอง($T_{FNN\_Tuning \le 5}$) ในรูปของฟังก์ชันที่แปรผันตามจำนวนตัวประมวลผลกลางของเครื่อง (c) และจำนวนข้อมูล (w) สามารถแสดงได้ดังนี้

$$T_{FNN\_Tunging \le 5}(c,w) = k_1c^m+k_2w^2+k_3(cw)^n+k_4c+k_5w+k_6cw+k_7 \qquad (11)$$

จากสมการ (11) สามารถประมาณค่าสัมประสิทธิ์ k_1-k_7,n และ m จากระเบียบวิธีกำลังสองน้อยที่สุด (Least Square Method) ได้ดังนี้

k_1	= 1.36912e-07	k_2	= 1.85304e-07
k_3	= 0.923946	k_4	= 0.182808
k_5	= 0.0555877	k_6	= -0.00502668
k_7	= 0.943224	m	= 8.11747
n	= -0.131454		



รูปที่ 19 เวลาที่ใช้ในการสร้างแบบจำลองระบบ(เทคนิคโครงข่ายประสาทเทียม) โดยปรับค่าการใช้งานหน่วยประมวลผลในช่วงข้อมูลมากกว่า 50,000 บรรทัด

เมื่อพิจารณาผลการทดลองในช่วงข้อมูลมากกว่า 50,000 บรรทัดจะเห็นได้ว่าเมื่อเพิ่มจำนวนตัวประมวลผลกลางแล้วจะส่งผลให้เวลาที่ใช้มีแนวโน้มการลดลงแบบ Exponential เช่นเดียวกับช่วงข้อมูลน้อยกว่าเท่ากับ 50,000 บรรทัด แต่แนวโน้มของช่วงข้อมูลนี้ จะสังเกตได้ว่า เมื่อเริ่มเพิ่มหน่วยประมวลผลกลาง ส่งผลให้เวลาที่ใช้ในการสร้างแบบจำลองนั้นมีแนวโน้มลดลงอย่างรวดเร็วในทันที แตกต่างจากแนวโน้มของช่วงข้อมูลน้อยกว่าเท่ากับ 50,000 บรรทัด แต่เมื่อเพิ่มหน่วยประมวลผลกลางเป็น 16 หน่วย จะเห็นได้ว่าการเพิ่มนั้น ไม่ได้ช่วยเพิ่มให้แนวโน้มของการใช้เวลาในการทำงานลดลงมากนัก อีกทั้งยังทำให้จำนวนข้อมูลมีผลต่อเวลาในการสร้างแบบจำลองน้อยลงมาก เมื่อเทียบกับเวลาที่ใช้สร้างแบบจำลองในสภาวะที่มีหน่วยประมวลผลกลางจำนวนน้อยๆ

ซึ่งสมการสำหรับประมาณเวลาที่ใช้ในการสร้างแบบจำลอง($T_{FNN\_Tunging\ >5}$) ในรูปของฟังก์ชันที่แปรผันตามจำนวนตัวประมวลผลกลางของเครื่อง (c) และจำนวนข้อมูล (w) สามารถแสดงได้ดังนี้

$$T_{FNN\_Tuning\ >5}\ (c,w)= k_1c^m+k_2w^2+k_3(cw)^n+k_4c+k_5w+k_6cw+k_7 \qquad (12)$$

จากสมการ (12) สามารถประมาณค่าสัมประสิทธิ์ k_1-k_7,n และ m จากระเบียบวิธีกำลังสองน้อยที่สุด (Least Square Method) ได้ดังนี้

k_1	= 0.191287	k_2	= 1.41711e-07
k_3	= 0.911408	k_4	= 0.780849
k_5	= 0.0550981	k_6	= -0.00679726
k_7	= 1.00014	m	= 3.62852
n	= -0.198714		

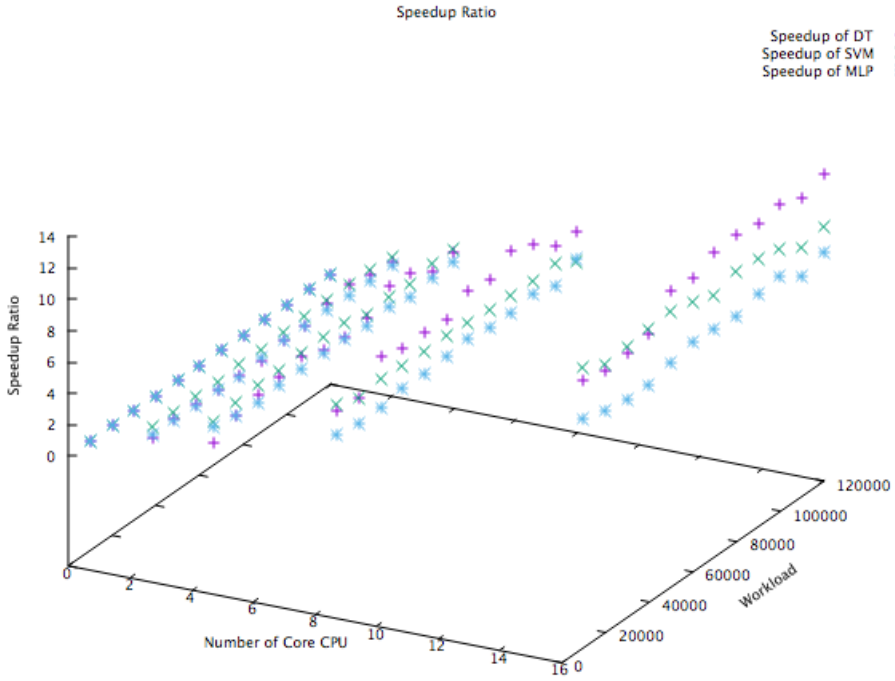
จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

4.4. การวิเคราะห์ผลการทดลองในเชิงการทำงานของหน่วยประมวลผลแบบขนาน

จากการทดลองที่ได้มีการปรับค่าการสร้างแบบจำลอง ให้ใช้หน่วยประมวลผลได้ทุกหน่วยในเวลาเดียวกัน(หน่วยประมวลผลแบบขนาน) สามารถวิเคราะห์เป็น 3 เรื่อง

4.4.1 สมรรถนะของเวลาในการประมวลผลที่เพิ่มขึ้น (Speedup Ratio)

เมื่อพิจารณาเปรียบเทียบ สมรรถนะของเวลาในการประมวลผลที่เพิ่มขึ้น (Speedup Ratio) ของการสร้างแบบจำลองระบบในทุกกรณีการทดลอง อ้างอิงตามรูปที่ (20)

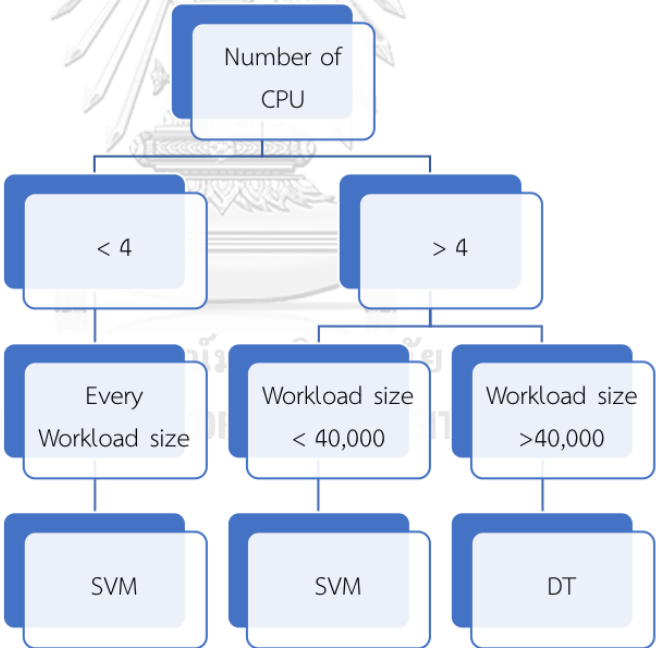


รูปที่ 20 สมรรถนะที่เพิ่มขึ้นของการสร้างแบบจำลองระบบ

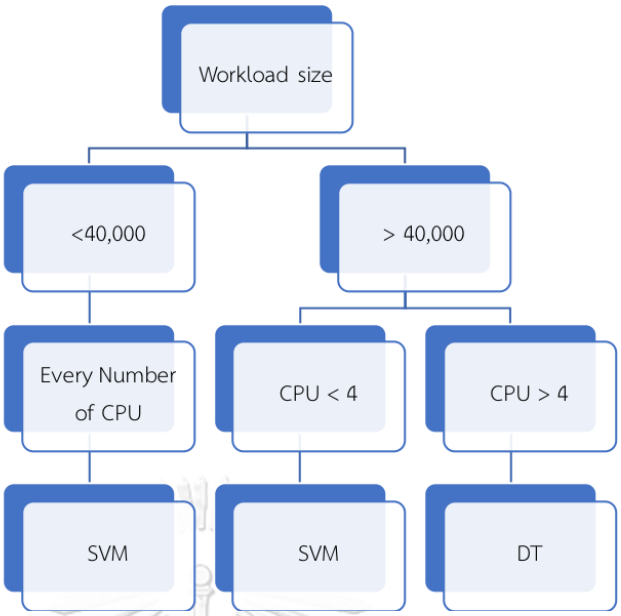
ซึ่งสามารถวิเคราะห์ได้ว่า

1. เทคนิคโครงข่ายประสาทเทียม มีค่าสมรรถนะของเวลาในการประมวลผล (Speedup Ratio) ที่น้อยกว่าเทคนิคอื่น ๆ เป็นส่วนมาก และใช้เวลาการสร้างแบบจำลองที่มากกว่าเทคนิคอื่น ๆ อยู่เสมออีกด้วย จึงกล่าวได้ว่า เป็นเทคนิคที่ใช้ต้นทุนมากที่สุด ไม่คุ้มค่าต่อการลงทุนที่จะเลือกใช้ หากเทียบกับรูปแบบเทคนิคอื่น ๆ

2. สำหรับเทคนิคต้นไม้การสินใจและเทคนิคซัพพอร์ตเวกเตอร์แมชชีน ในช่วงหน่วยประมวลผลไม่เกิน 4 หน่วยนั้น ทั้ง 2 เทคนิค ไม่ว่าจำนวนข้อมูลจะมากหรือน้อย จะให้ค่าสมรรถนะของเวลาในการประมวลผลที่เพิ่มขึ้น (Speedup Ratio) ที่ใกล้เคียงกัน โดยที่เทคนิคซัพพอร์ตเวกเตอร์แมชชีน จะให้ค่าสมรรถนะของเวลาในการประมวลผล (Speedup Ratio) ที่มากกว่าอยู่เล็กน้อย
3. สำหรับเทคนิคต้นไม้การสินใจและเทคนิคซัพพอร์ตเวกเตอร์แมชชีน ในช่วงหน่วยประมวลผลตั้งแต่ 4 หน่วยขึ้นไป จะเห็นได้ว่า ในช่วงข้อมูลที่น้อยๆ (น้อยกว่า 40,000) เทคนิคซัพพอร์ต-เวกเตอร์แมชชีน จะให้ค่าสมรรถนะของเวลาในการประมวลผลที่เพิ่มขึ้น (Speedup Ratio) ที่มากกว่า แต่เมื่อจำนวนข้อมูลเพิ่มขึ้น สมรรถนะของเวลาในการประมวลผล (Speedup Ratio) ของเทคนิคต้นไม้การสินใจ จะให้ค่าที่ดีกว่า ทั้งนี้จำนวนเลขช่วงข้อมูลอ้างอิงจากข้อมูลการทดลอง อาจเปลี่ยนแปลงได้ตามสภาพแวดล้อมของการใช้งานนั้น ๆ จึงสรุปเป็นเงื่อนไขในการแนะนำการเลือกใช้เทคนิคต่าง ๆ แสดงได้ดังรูปที่ (21-22)

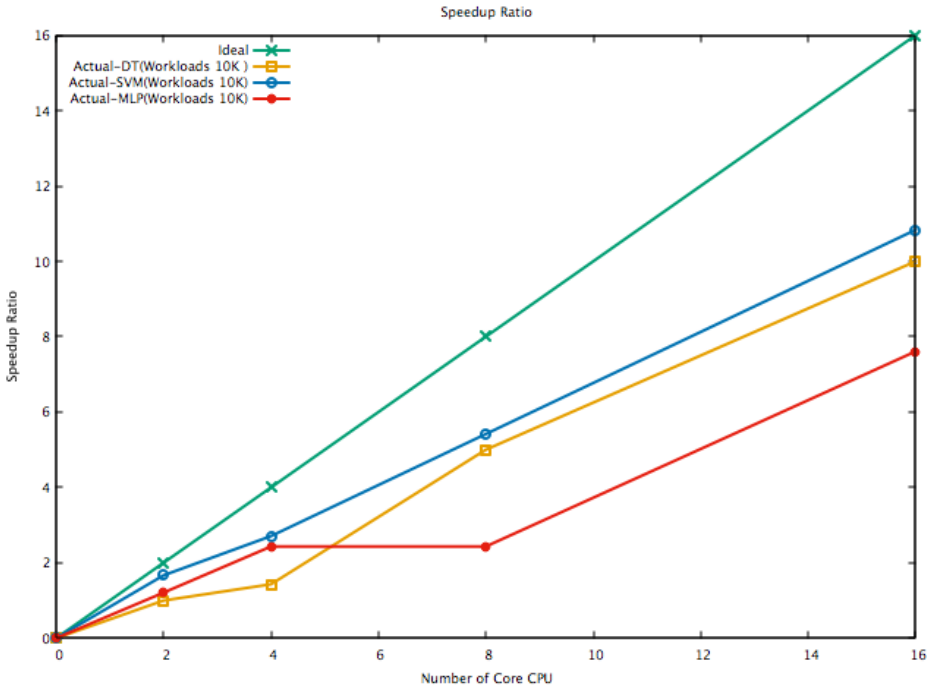


รูปที่ 21 เงื่อนไขในการแนะนำการเลือกใช้เทคนิคการเรียนรู้จากจำนวนหน่วยประมวลผลกลาง

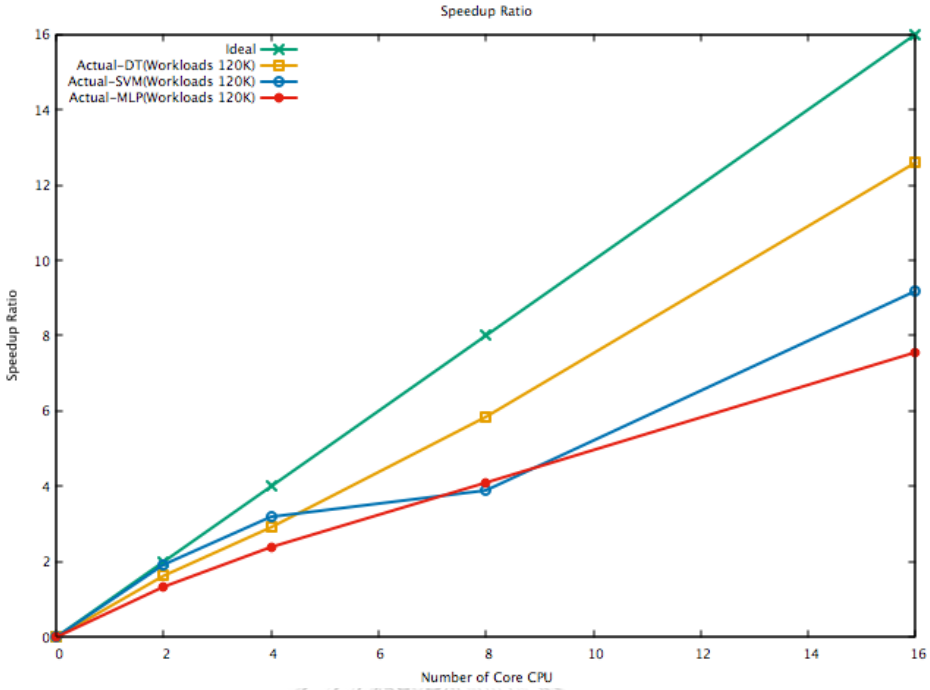


รูปที่ 22 เงื่อนไขในการแนะนำการเลือกใช้เทคนิคการเรียนรู้จากจำนวนข้อมูล

โดยเมื่อวิเคราะห์สมรรถนะของเวลาในการประมวลผลที่เพิ่มขึ้น (Speedup Ratio) ของจำนวนข้อมูลน้อยสุด (10,000 บรรทัด) จากรูปที่ (23) และจำนวนข้อมูลมากที่สุด (120,000 บรรทัด) จากรูปที่ (24) พบว่า แนวโน้มยังเป็นไปตามข้อสรุปด้านบน ตามรูปที่ (21-22) จึงนำไปสู่การให้รายละเอียด และวิเคราะห์ข้อมูลโดยใช้แนวโน้มของค่าเฉลี่ยในหัวข้อถัด ๆ ไป



รูปที่ 23 สมรรถนะที่เพิ่มขึ้นของของจำนวนข้อมูลน้อยสุด

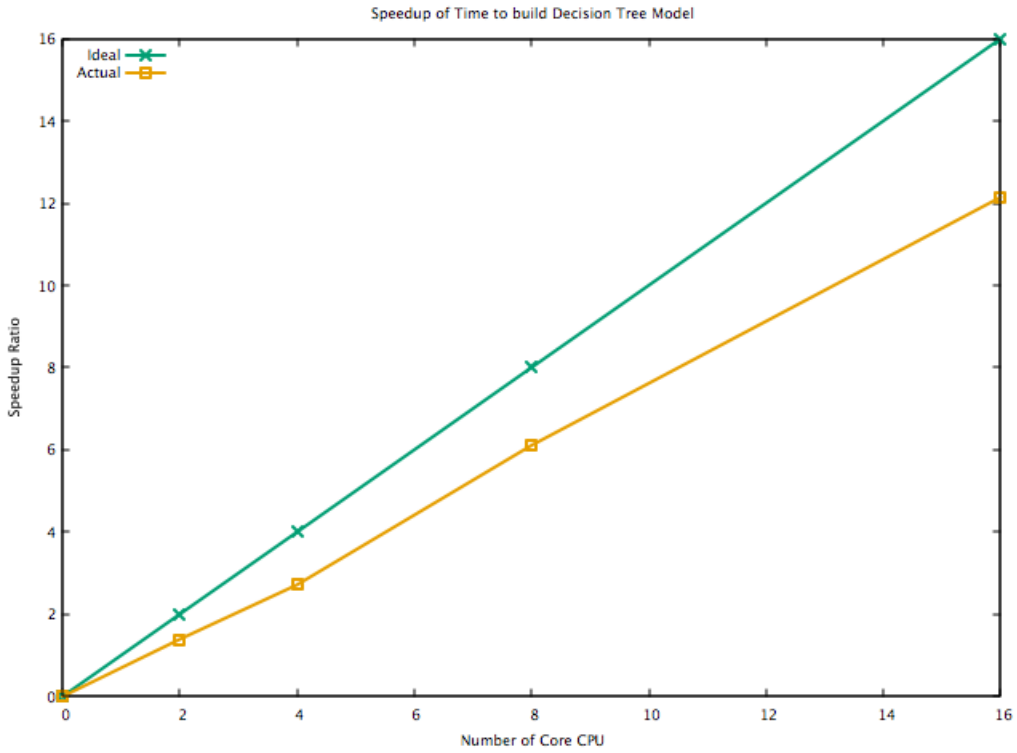


รูปที่ 24 สมรรถนะที่เพิ่มขึ้นของจำนวนข้อมูลมากที่สุด

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

4.4.1.1. รูปแบบที่ 1 : ต้นไม้ตัดสินใจ (Decision tree model)

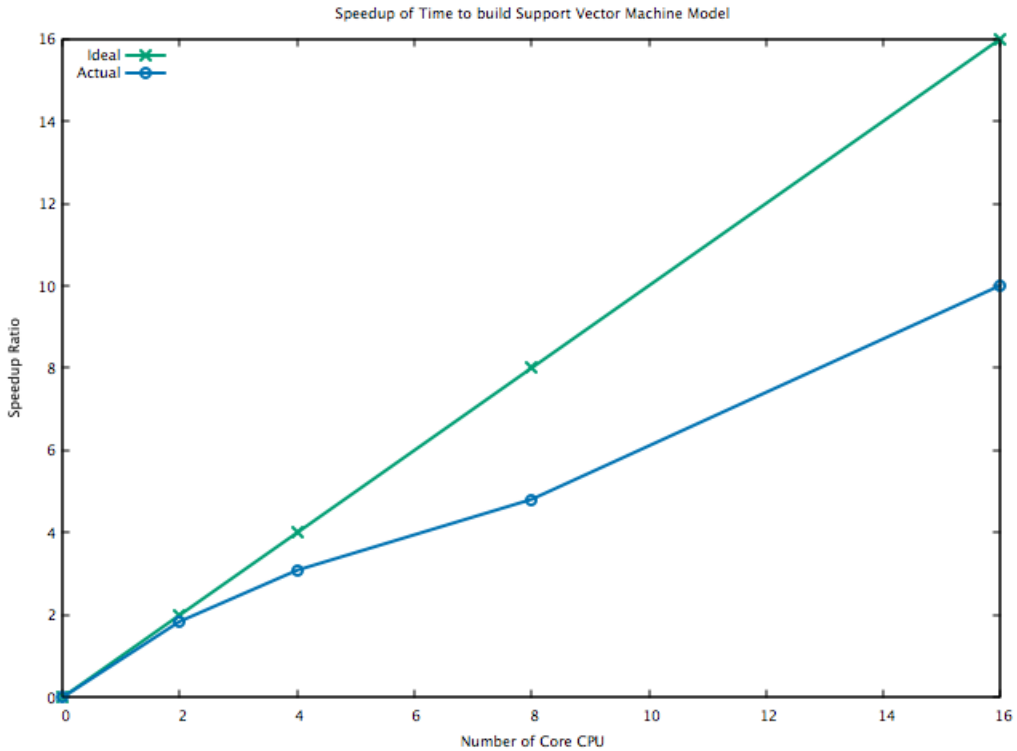
เมื่อพิจารณาค่าสมรรถนะของเวลาในการประมวลผลที่เพิ่มขึ้น (Speedup Ratio) โดยเฉลี่ยของการสร้างแบบจำลองระบบที่ใช้รูปแบบต้นไม้การสินใจ จะพบว่า เมื่อจำนวนหน่วยประมวลผลเพิ่มขึ้น จะยังคงส่งผลช่วยให้ประมวลผลได้เร็วขึ้น แต่ประสิทธิภาพของการประมวลผลแบบขนานนั้นค่อยๆน้อยลง ดังจะเห็นจากรูปที่ (25) ได้ว่า ระยะห่างระหว่างเส้นสมรรถนะของเวลาในการประมวลผลที่เพิ่มขึ้น (Speedup Ratio) ในอุดมคติ กับเส้นที่ได้จากข้อมูลในความเป็นจริงนั้น มีระยะห่างมากขึ้นเรื่อย ๆ โดยเส้นที่ได้จากข้อมูลในความเป็นจริง จะมีค่าน้อยกว่าเส้นในอุดมคติอยู่เสมอ



รูปที่ 25 สมรรถนะที่เพิ่มขึ้นของเทคนิคต้นไม้การสินใจ

4.4.1.2. รูปแบบที่ 2 : ซัพพอร์ตเวกเตอร์แมชชีน (support vector machine model)

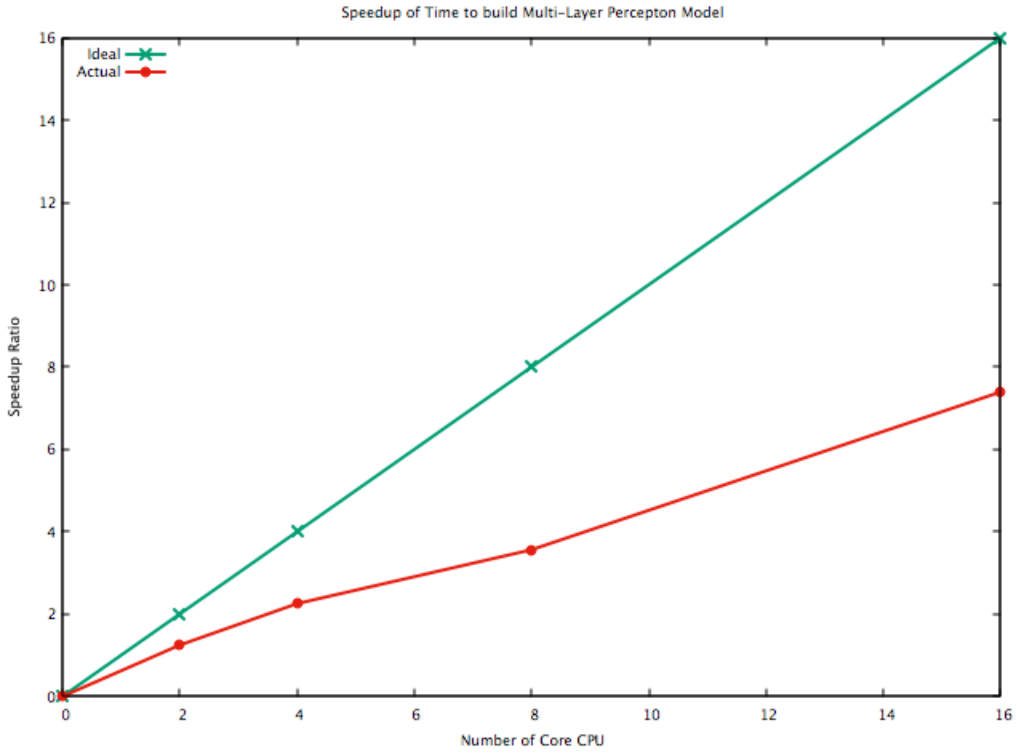
เมื่อพิจารณาค่าสมรรถนะของเวลาในการประมวลผลที่เพิ่มขึ้น (Speedup Ratio) โดยเฉลี่ยของการสร้างแบบจำลองระบบที่ใช้รูปแบบซัพพอร์ตเวกเตอร์แมชชีน จะพบว่า เมื่อจำนวนหน่วยประมวลผลเพิ่มขึ้น จะยังคงส่งผลช่วยให้ประมวลผลได้เร็วขึ้น แต่ประสิทธิภาพของการประมวลผลแบบขนานนั้นค่อยๆน้อยลง เช่นเดียวกับรูปแบบต้นไม้การสินใจ ดังจะเห็นจากรูปที่ (26) ได้ว่า ระยะห่างระหว่างเส้นสมรรถนะของเวลาในการประมวลผลที่เพิ่มขึ้น (Speedup Ratio) ในอุดมคติ กับเส้นที่ได้จากข้อมูลในความเป็นจริงนั้น มีระยะห่างมากขึ้นเรื่อย ๆ โดยเส้นที่ได้จากข้อมูลในความเป็นจริง จะมีค่าน้อยกว่าเส้นในอุดมคติอยู่เสมอ อีกทั้งหากพิจารณาเพิ่มเติม จะพบว่า สำหรับการประมวลผลแบบขนาน โดยการใช้หน่วยประมวลผล 2 หน่วย จะให้ส่งผลใกล้เคียงกับค่าในอุดมคติ ถือว่าเป็นการลงทุนที่ได้รับผลลัพธ์ที่คุ้มค่ามากที่สุด ของเทคนิคซัพพอร์ตเวกเตอร์แมชชีน



รูปที่ 26 สมรรถนะที่เพิ่มขึ้นของเทคนิคซัพพอร์ตเวกเตอร์แมชชีน

4.4.1.3. รูปแบบที่ 3 : โครงข่ายประสามเทียมแบบ Multi-Layer Perceptron (Feedforward Neural Network)

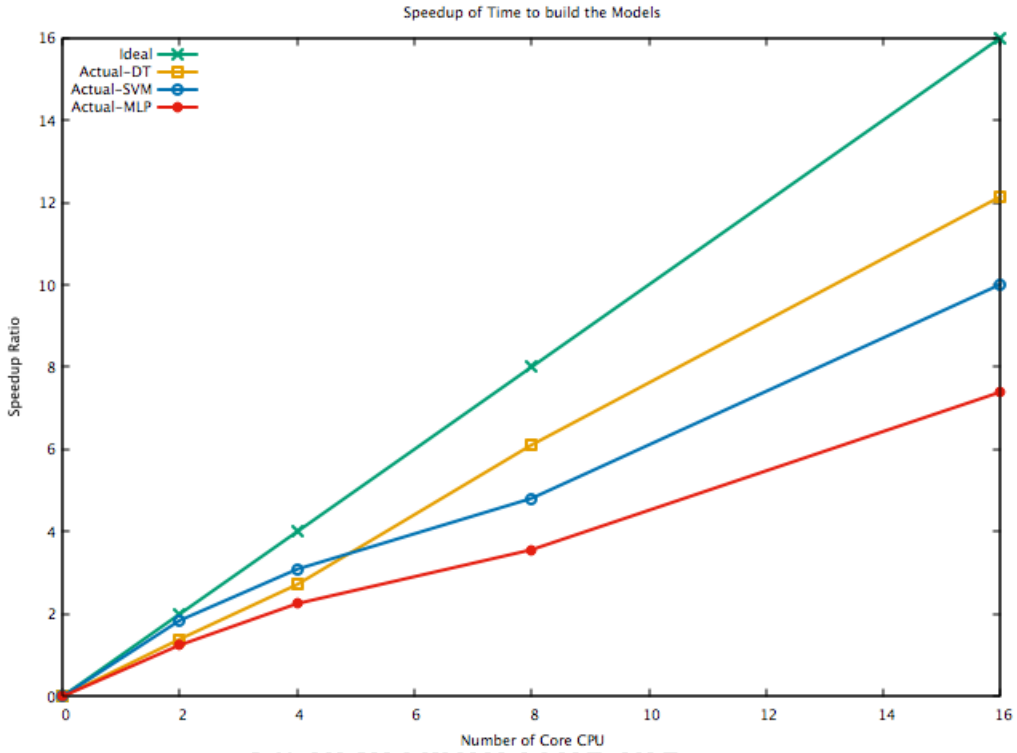
เมื่อพิจารณาค่าสมรรถนะของเวลาในการประมวลผลที่เพิ่มขึ้น (Speedup Ratio) โดยเฉลี่ยของการสร้างแบบจำลองระบบที่ใช้รูปแบบโครงข่ายประสามเทียมแบบ Multi-Layer Perceptron จะพบว่า เมื่อจำนวนหน่วยประมวลผลเพิ่มขึ้น จะยังคงส่งผลช่วยให้ประมวลผลได้เร็วขึ้น แต่ประสิทธิภาพของการประมวลผลแบบขนานนั้นค่อยๆน้อยลง เช่นเดียวกับรูปแบบต้นไม้การสินใจและซัพพอร์ตเวกเตอร์แมชชีน ดังจะเห็นจากรูปที่ (27) ได้ว่า ระยะห่างระหว่างเส้นสมรรถนะของเวลาในการประมวลผลที่เพิ่มขึ้น (Speedup Ratio) ในอุดมคติ กับเส้นที่ได้จากข้อมูลในความเป็นจริงนั้น มีระยะห่างมากขึ้นเรื่อย ๆ โดยเส้นที่ได้จากข้อมูลในความเป็นจริง จะมีค่าน้อยกว่าเส้นในอุดมคติอยู่เสมอ



รูปที่ 27 สมรรถนะที่เพิ่มขึ้นของเทคนิคโครงข่ายประสามเทียม

ซึ่งหากพิจารณาเปรียบเทียบ สมรรถนะของเวลาในการประมวลผลที่เพิ่มขึ้น (Speedup Ratio) โดยเฉลี่ยของทุกรูปแบบนั้น แสดงดังรูปที่ (28) จะพบว่า สำหรับการประมวลผลแบบขนานที่จำนวนหน่วยประมวลไม่เกิน 4 หน่วย ประสิทธิภาพการทำงานของการประมวลผลแบบขนานของ

รูปแบบซัพพอร์ตเวกเตอร์แมชชีน ดีกว่ารูปแบบต้นไม้การสินใจ และรูปแบบโครงข่ายประสามเทียมแบบ Multi-Layer Perceptron ตามลำดับ แต่เมื่อใช้หน่วยประมวลผลมากกว่า 4 หน่วยขึ้นไป ประสิทธิภาพการทำงานของการประมวลผลแบบขนานของรูปแบบต้นไม้การสินใจ ดีกว่ารูปแบบซัพพอร์ตเวกเตอร์แมชชีน และรูปแบบโครงข่ายประสามเทียมแบบ Multi-Layer Perceptron ตามลำดับ ซึ่งสอดคล้องกับเงื่อนไขในการแนะนำการเลือกใช้เทคนิคต่าง ๆ แสดงได้ดังรูปที่ (21-22)



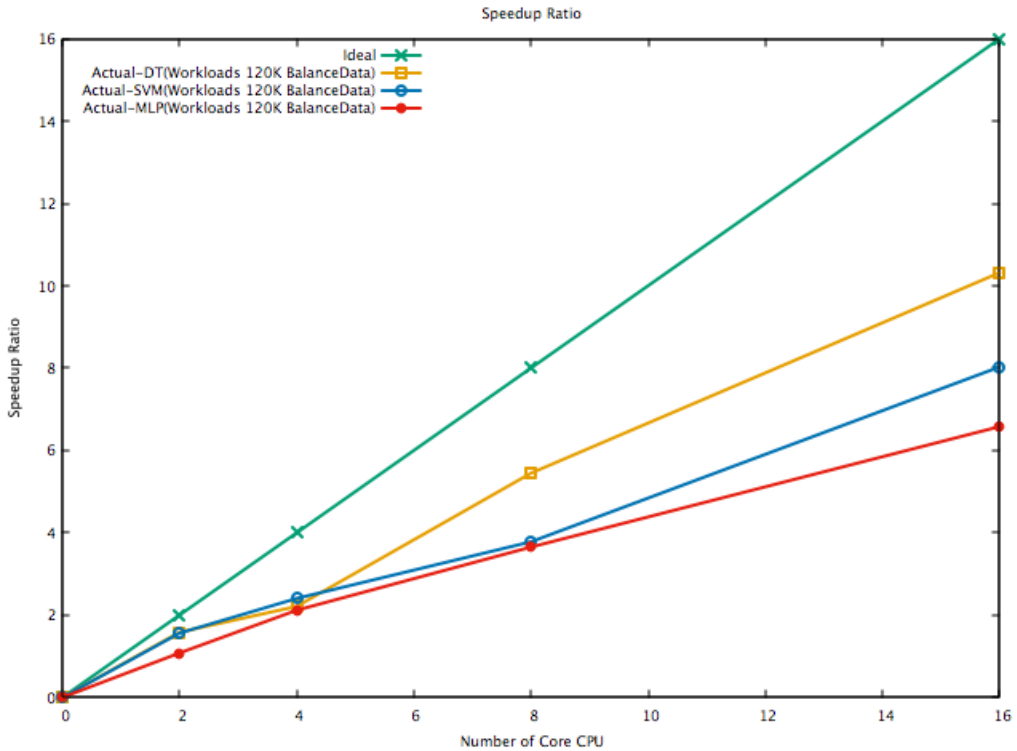
รูปที่ 28 สมรรถนะที่เพิ่มขึ้นของการสร้างแบบจำลองระบบโดยเฉลี่ย

CHULALONGKORN UNIVERSITY

4.4.2 สมรรถนะของเวลาในการประมวลผลที่เพิ่มขึ้น (Speedup Ratio) เมื่อทำการปรับค่าโน้มเอียงของข้อมูล

เนื่องจากชุดข้อมูล NSL-KDD เป็นข้อมูลการโจมตีทางด้านเครือข่ายที่มีความโน้มเอียงเป็นข้อมูลเชิงบวกมากกว่า จึงได้ทำการทดลอง โดยทำการปรับค่าโน้มเอียง ระหว่างข้อมูลที่เป็นข้อมูลใช้งานปกติ และข้อมูลที่เป็นการโจมตีชนิดต่าง ๆ ให้มีจำนวนเท่ากัน เพื่อวิเคราะห์หาความแตกต่างในการใช้ต้นทุนด้านเวลา และสมรรถนะของเวลาในการประมวลผลที่เพิ่มขึ้น (Speedup Ratio) โดยทำการตัวอย่างเฉพาะ ข้อมูลจำนวน 120,000 บรรทัด

จากการทดลอง พบว่าทุกเทคนิค เมื่อมีการปรับค่าโน้มเอียงของข้อมูล มีผลทำให้ค่าความถูกต้อง (Accuracy) มีค่าลดลง และสมรรถนะของเวลาในการประมวลผลที่เพิ่มขึ้น (Speedup Ratio) มีค่าน้อยลง แสดงได้ดังรูปที่ (29) แต่พิจารณาแล้วยังคงมีแนวโน้มไปในทิศทางเดียวกับ Speedup ของข้อมูลที่ไม่ได้ทำการปรับค่าโน้มเอียงของข้อมูล ดังรูปที่ (24) และมีแนวโน้มไปในทิศทางเดียวกับ Speedup โดยเฉลี่ยของทุก Workloads ดังรูปที่ (28)



รูปที่ 29 สมรรถนะที่เพิ่มขึ้นของของจำนวนข้อมูลมากที่สุด โดยมีการปรับค่าโน้มเอียงข้อมูล

4.4.3 ประสิทธิภาพของเวลาในการประมวลผลแบบขนาน

จากการวิเคราะห์สมรรถนะของเวลาในการประมวลผลที่เพิ่มขึ้น (Speedup Ratio) ของแต่ละรูปแบบ พบว่าเมื่อใช้การประมวลผลแบบขนานในการทำงาน เวลาที่ใช้ในการสร้างแบบจำลองไม่ได้ลดเป็น อัตราเดียวกับการเพิ่มจำนวนหน่วยประมวลผล เนื่องจากในการทำงานทั้งหมด ประกอบไปด้วย งานที่สามารถประมวลผลขนานกันได้ และงานที่ไม่สามารถทำงานขนานได้

จากกฎของ Amdahl ได้กำหนดการวัดสมรรถนะของเวลาในการประมวลผลที่เพิ่มขึ้น (Speedup Ratio) และค่าประสิทธิภาพในการเพิ่มหน่วยประมวลผล (Efficiency) ในรูปของฟังก์ชันที่แปรผันตามจำนวนตัวประมวลผลกลางของเครื่อง (c) และ สัดส่วนของงานที่สามารถประมวลผลขนานกันได้(P) สามารถแสดงได้ดังนี้

$$\text{Overall Speedup (c)} = \frac{1}{(1-P)+\frac{P}{c}} \tag{13}$$

$$\text{Efficiency} = \frac{Speedup}{c} \tag{14}$$

จากสมการ (13) สามารถประมาณค่าสัมประสิทธิ์ P จากระเบียบวิธีกำลังสองน้อยที่สุด (Least Square Method) ได้ดังนี้

รูปแบบ	สัดส่วนของงานที่ประมวลผลแบบขนาน(P)
ต้นไม้ตัดสินใจ (Decision tree model)	0.976405
ซัพพอร์ตเวกเตอร์แมชชีน (support vector machine model)	0.95426
โครงข่ายประสามเทียมแบบ Multi-Layer Perceptron (Feedforward Neural Network)	0.905175

ตารางที่ 6 สัดส่วนของงานที่ประมวลผลแบบขนาน(P) ของการสร้างแบบจำลองระบบ

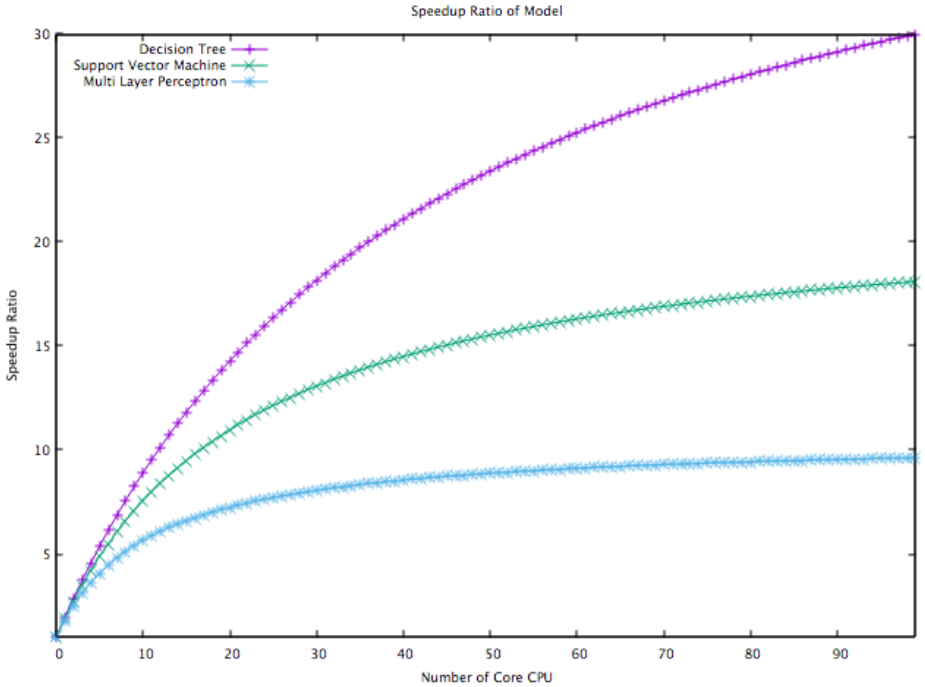
พิจารณาแล้วพบว่า การสร้างแบบจำลองโดยใช้เทคนิคต้นไม้ตัดสินใจ มีสัดส่วนของงานที่สามารถประมวลผลขนานได้(P) มากที่สุด โดยเทคนิคซัพพอร์ตเวกเตอร์แมชชีน และเทคนิคโครงข่ายประสามเทียมแบบ Multi-Layer Perceptron มีค่าน้อยกว่าตามลำดับ

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

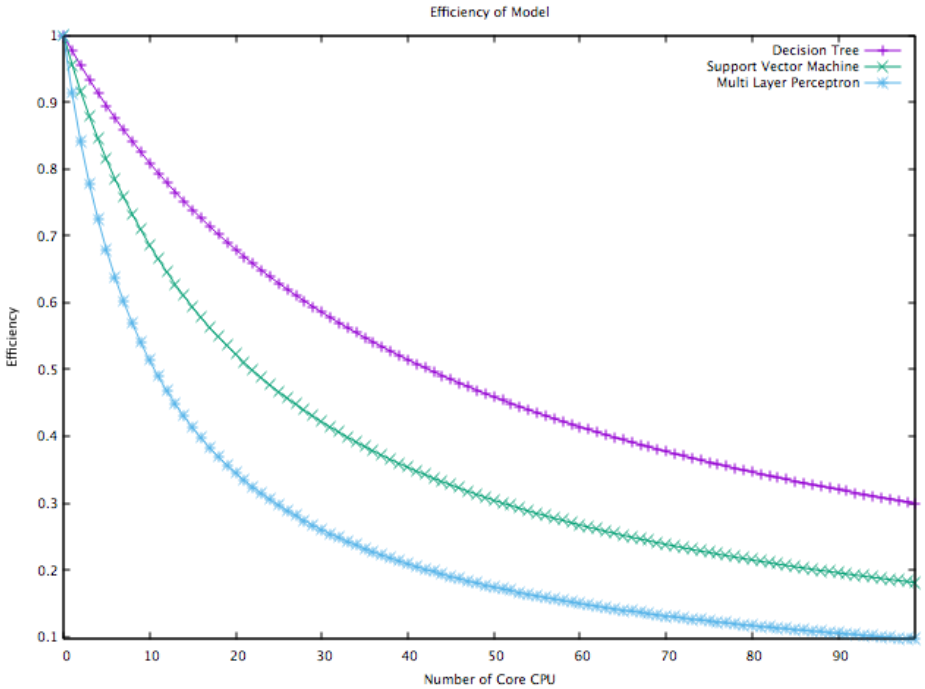
ซึ่งหากพิจารณา ค่าสมรรถนะของเวลาในการประมวลผลที่เพิ่มขึ้น (Speedup Ratio) ตามสมการที่ (13) โดยใช้ค่าสัดส่วนของงานที่สามารถประมวลผลขนานได้ (P) โดยเฉลี่ย จากตารางที่ (6) จะเห็นได้ว่า หากเพิ่มหน่วยประมวลผลมากขึ้นเรื่อย ๆ กราฟจะเริ่มมีความชันที่ลดลง การเพิ่มหน่วยประมวลผล จะไม่ได้ช่วยลดเวลาในการทำงานลงไปด้วย แสดงดังรูปที่ (30) และรายละเอียดค่าสมรรถนะของเวลาในการประมวลผลที่เพิ่มขึ้น (Speedup Ratio) ดังตารางที่ (20) โดยเทคนิคโครงข่ายประสามเทียมแบบ Multi-Layer Perceptron จะเข้าสู่สถานะดังกล่าวก่อน ตามด้วยเทคนิคซัพพอร์ตเวกเตอร์แมชชีน และ เทคนิคต้นไม้ตัดสินใจ ตามลำดับ

อีกทั้งเมื่อพิจารณาค่าประสิทธิภาพในการเพิ่มหน่วยประมวลผล (Efficiency) ตามสมการที่ (14) จะเห็นได้ว่าหากเพิ่มหน่วยประมวลผลขึ้นไปเรื่อย ๆ แสดงดังรูปที่ (31) และรายละเอียดดังตารางที่ (21) ที่จุดสังเกตุเดียวกัน เทคนิคต้นไม้ตัดสินใจ จะมีค่าประสิทธิภาพที่ดีกว่า เทคนิค ซัพพอร์ตเวกเตอร์แมชชีน และเทคนิคโครงข่ายประสามเทียมตามลำดับ อาจกล่าวตัวอย่างในสภาวะสมมติที่มีการยอมรับประสิทธิภาพไม่ต่ำกว่า 50 % ในการลงทุนเพิ่มหน่วยประมวลผล เพื่อให้ได้ซึ่งสมรรถนะของเวลา ในการประมวลผลที่เพิ่มขึ้น (Speedup Ratio) ได้ว่า เทคนิคต้นไม้ตัดสินใจจะสามารถเพิ่ม หน่วยประมวลผลได้ถึง 43 หน่วย เทคนิคซัพพอร์ตเวกเตอร์แมชชีน สามารถเพิ่มได้ถึง 22 หน่วย และสุดท้ายเทคนิคโครงข่ายประสามเทียมแบบ Multi-Layer Perceptron สามารถเพิ่มได้เพียง 11 หน่วย แสดงดังตารางที่ (20-21) โดยหากเพิ่มจำนวนหน่วยประมวลผลมากไปกว่านี้ จะไม่ได้ส่งผลให้ช่วยลดเวลาในการทำงานลงได้ตามประสิทธิภาพที่ต้องการ และอาจกล่าวอีกตัวอย่าง ในสถานการณ์ที่มีจำนวนหน่วยประมวลผล 50 หน่วย ในการลงทุนเพิ่มสมรรถนะของเวลาในการประมวลผลที่เพิ่มขึ้น (Speedup Ratio) ให้ได้มีประสิทธิภาพที่สุด แสดงดังตารางที่ (21) จึงแนะนำควรเลือก เทคนิคต้นไม้ตัดสินใจ เนื่องจากที่ค่าหน่วยประมวลผล เท่ากับ 50 จะส่งผลทำให้ค่าประสิทธิภาพในการเพิ่มหน่วยประมวลผลดีกว่าเทคนิคอื่น ๆ

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University



รูปที่ 30 สมรรถนะของเวลาในการประมวลผลที่เพิ่มขึ้น ในช่วงหน่วยประมวลผลถึง 100 หน่วย (Speedup Ratio)



รูปที่ 31 ประสิทธิภาพของการเพิ่มหน่วยประมวลผล ในช่วงหน่วยประมวลผลถึง 100 หน่วย

จากการวิเคราะห์ประสิทธิภาพของเวลาในการประมวลผลแบบขนาน จึงกล่าวได้ว่า การสร้างแบบจำลองโดยใช้เทคนิคต้นไม้ตัดสินใจ มีความคุ้มค่ามากที่สุดในภาพรวม สำหรับการลงทุนเพิ่มหน่วยประมวล เพื่อให้ได้ซึ่งสมรรถนะของเวลาในการประมวลผลที่ดีที่สุด

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

4.5. อภิปรายผลการวิจัย

รูปแบบวิธีการทดลองแต่ละวิธีที่กล่าวมานั้นใช้เวลาในการสร้างแบบจำลองของระบบตรวจจับการบุรุกที่แตกต่างกัน แบบจำลองของเวลา T_{algo} สามารถนำมาประมาณเวลาที่ใช้ในแต่ละรูปแบบวิธีได้ นอกจากนี้แบบจำลองการประมาณจำนวนตัวประมวลผลกลางที่เพียงพอและเหมาะสมสำหรับการสร้างแบบจำลองระบบตรวจสอบการบุกรุก ตามที่ได้เสนอในงานวิจัยนี้ โดยสามารถนำมาพิจารณาเพื่อเปรียบเทียบต้นทุนด้านเวลาของแต่ละรูปแบบวิธีได้ โดยผลการทดลองและวิเคราะห์สามารถสรุปได้ดังนี้

- จากผลการวิจัยพบว่าหากใช้เทคนิคโครงข่ายประสามเทียมแบบ Multi-Layer Perceptron (Feedforward Neural Network) จะใช้เวลาในการสร้างแบบจำลองมากที่สุดเสมอ เมื่อเปรียบเทียบกับเทคนิคอื่น ๆ ในกรณีทดลองเดียวกัน
- เมื่อเปรียบเทียบกับแต่ละเทคนิค ในกรณีทดลองเดียวกัน พบว่า เทคนิคต้นไม้ตัดสินใจและเทคนิคซัพพอร์ตเวกเตอร์แมชชีน ใช้เวลาในการสร้างแบบจำลองที่ใกล้เคียงกันมาก แต่หากเรามีสภาพแวดล้อมของระบบที่มีจำนวนตัวประมวลผลกลางที่จำกัด แนะนำให้เลือกใช้เทคนิคต้นไม้ตัดสินใจ เพราะถึงแม้ทั้งสองเทคนิคจะใช้เวลาที่ใกล้เคียงกันมาก แต่หากในกรณีทดลองที่ขนาดของข้อมูลมากขึ้น จะพบว่าเทคนิคต้นไม้ตัดสินใจจะใช้เวลาในการสร้างแบบจำลองที่เร็วกว่าเล็กน้อย
- จากผลการวิจัย พบว่าหากใช้เทคนิคโครงข่ายประสามเทียมแบบ Multi-Layer Perceptron (Feedforward Neural Network) และไม่ได้มีการปรับการใช้งานของหน่วยประมวลผล ผลที่ได้แสดงให้เห็นว่าจำนวนตัวประมวลผลกลาง ที่เพิ่มขึ้นแทบจะไม่ส่งผลต่อต้นทุนด้านเวลาเลย
- จากรูปที่ (10) และ (14) สำหรับเทคนิคต้นไม้ตัดสินใจ และเทคนิคซัพพอร์ตเวกเตอร์-แมชชีน ภายใต้เงื่อนไขที่ไม่ได้มีการปรับการใช้งานของหน่วยประมวลผลในการทดลอง แสดงให้เห็นว่า เวลาที่ใช้ในการสร้างแบบจำลองระบบไม่ได้ลดลงเสมอไปเมื่อใช้จำนวนตัวประมวลผลกลางในระบบมากขึ้น ซึ่งสามารถใช้แบบจำลองในการประมาณจำนวนตัวประมวลผลกลางที่เพียงพอและเหมาะสมสำหรับการสร้างระบบตรวจสอบการบุกรุกได้ ตามสมการที่ (8) โดยการประมาณจะมีความสัมพันธ์โดยตรงกับจำนวนข้อมูล
- หากมีการปรับการใช้งานของหน่วยประมวลผลในการทดลอง ให้สามารถใช้งานหน่วยประมวลผลได้ทุกหน่วยในเวลาเดียวกัน พบว่า จำนวนหน่วยประมวลผลและจำนวนข้อมูล ส่งผลโดยตรงกับเวลาที่ใช้ในการสร้างแบบจำลอง โดยจำนวนหน่วยประมวลผลจะส่งผลต่อเวลาในการสร้างแบบจำลองมากกว่าจำนวนข้อมูลในช่วงข้อมูลจำนวนมาก ๆ แต่จะส่งผลน้อยลงกับในช่วงข้อมูลจำนวนน้อย ๆ

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

- หากมีการปรับการใช้งานของหน่วยประมวลผลในการทดลอง ให้สามารถใช้งานหน่วยประมวลผลได้ทุกหน่วยในเวลาเดียวกัน จะพบว่า แนวโน้มของเวลาที่ใช้ในการสร้างแบบจำลองเทคนิคต้นไม้ตัดสินใจและเทคนิคโครงข่ายประสาทเทียมนั้นมีลักษณะ 2 แบบ ตามรูปที่ (11) และ (17) โดยในช่วงข้อมูลที่น้อยกว่าเท่ากับ 60,000 ของเทคนิคต้นไม้ตัดสินใจ และน้อยกว่าเท่ากับ 50,000 ของเทคนิคโครงข่ายประสาทเทียม สามารถเข้าใจได้ว่า จำนวนหน่วยประมวลผล 1 หรือ 2 หน่วย จะส่งผลต่อเวลาที่ใช้ที่ไม่ต่างกัน จึงแนะนำได้ว่า สามารถเลือกใช้หน่วยประมวลผลเพียง 1 หน่วย แทนการเลือกใช้ 2 หน่วยได้เช่นกัน
- สำหรับการประมวลผลแบบขนานที่จำนวนหน่วยประมวลไม่เกิน 4 หน่วย สมรรถนะของเวลาในการประมวลผลที่เพิ่มขึ้น (Speedup Ratio) โดยเฉลี่ยของเทคนิคซัพพอร์ต-เวกเตอร์แมชชีน ดีกว่าเทคนิคต้นไม้การสินใจ และเทคนิคโครงข่ายประสามเทียมแบบ Multi-Layer Perceptron ตามลำดับ
- สำหรับการประมวลผลแบบขนานที่จำนวนหน่วยประมวลเกิน 4 หน่วยขึ้นไป สมรรถนะของเวลาในการประมวลผลที่เพิ่มขึ้น (Speedup Ratio) โดยเฉลี่ยของเทคนิคต้นไม้การสินใจ ดีกว่าเทคนิคซัพพอร์ตเวกเตอร์แมชชีน และเทคนิคโครงข่ายประสามเทียมแบบ Multi-Layer Perceptron ตามลำดับ
- จากการวิเคราะห์ประสิทธิภาพของเวลาในการประมวลผลแบบขนาน การสร้างแบบจำลอง โดยใช้เทคนิคต้นไม้ตัดสินใจ มีความคุ้มค่ามากที่สุดในภาพรวม สำหรับการลงทุนเพิ่มหน่วยประมวลผล เพื่อให้ได้ซึ่งสมรรถนะของเวลาในการประมวลผลที่ดีที่สุด

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

บทที่ 5
บทสรุปของการวิจัย

ในบทนี้จะกล่าวถึงสิ่งที่ได้จากการวิจัย แนวทางการวิจัยต่อ และบทสรุป ดังนี้

5.1 สิ่งที่ได้จากการวิจัย (Contribution)

สิ่งที่ได้จากการวิจัยนี้ ได้แก่

1. อธิบายขั้นตอนวิธีการสร้าง และการทดลอง แบบจำลองระบบตรวจการบุกรุกแต่ละรูปแบบเทคนิคการเรียนรู้ด้วยเครื่องต่าง ๆ
2. นำเสนอการเปรียบเทียบภาพรวมของประสิทธิภาพด้านเวลาของแต่ละรูปแบบ
3. นำเสนอแบบจำลองของต้นทุนด้านเวลาที่ใช้ในการประมาณเวลาในการสร้างแบบจำลองระบบตรวจการบุกรุกแต่ละรูปแบบเทคนิคการเรียนรู้ด้วยเครื่อง ภายใต้เงื่อนไขที่ไม่ได้มีการปรับการใช้งานของหน่วยประมวลผล ตามสมการที่ (2),(6) และ (10) และมีการปรับการใช้งานของหน่วยประมวลผล ตามสมการที่ (4),(5),(9),(11) และ (12)
4. นำเสนอการประมาณจำนวนตัวประมวลผลกลางที่เพียงพอและเหมาะสมสำหรับการสร้างระบบตรวจสอบการบุกรุกในแต่ละรูปแบบเทคนิคการเรียนรู้ด้วยเครื่อง ภายใต้เงื่อนไขที่ไม่ได้มีการปรับการใช้งานของหน่วยประมวลผล ตามสมการที่ (8)
5. แนะนำการเลือกใช้เทคนิคการเรียนรู้ด้วยเครื่องต่าง ๆ ในแต่ละเงื่อนไข แสดงได้ดังรูปที่ (21-22) เพื่อเป็นแนวทางในการตัดสินใจเลือกรูปแบบเทคนิคการเรียนรู้ด้วยเครื่องที่เหมาะสมและมีประสิทธิภาพด้านต้นทุนเวลา
6. นำเสนอการเปรียบเทียบสมรรถนะของเวลาในการประมวลผลที่เพิ่มขึ้น (Speedup Ratio) ของแต่ละรูปแบบ ตามรูปที่ (25-27)
7. นำเสนอสัดส่วนของงานที่สามารถประมวลผลขนานได้(P) ของการสร้างแบบจำลองระบบแต่ละรูปแบบเทคนิคการเรียนรู้ด้วยเครื่อง ตามตารางที่ (6) เพื่อเป็นแนวทางในการตัดสินใจเลือกรูปแบบเทคนิคการเรียนรู้ด้วยเครื่องในการสร้างระบบตรวจจับการบุกรุกที่เหมาะสมสำหรับการลงทุนเพิ่มหน่วยประมวลผล เพื่อให้ได้สมรรถนะของเวลาที่ดีที่สุด

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

5.2 แนวทางการวิจัย

ในงานวิจัยนี้ได้ศึกษาและวิเคราะห์ในด้านเวลาในการสร้างแบบจำลองระบบตรวจการบุกรุก ซึ่งเป็นต้นทุนมนการสร้างระบบที่มีความสำคัญ อย่างไรก็ตามงานวิจัยนี้สามารถนำไปศึกษาต่อในส่วนของรูปแบบเทคนิคการเรียนรู้ด้วยเครื่องในรูปแบบอื่น ๆ เช่น เทคนิคการวิเคราะห์การถดถอยเชิงเส้นตรง (Linear regression analysis), การเรียนรู้แบบเบย์ (Bayesian Learning) เป็นต้น

นอกจากนี้ยังมีประเด็นที่ต้องศึกษาต่ออีกหลายประเด็นด้วยกัน เช่น จำนวนข้อมูลที่มากขึ้นอาจส่งผลให้แนวโน้มของเวลา ในการสร้างแบบจำลองนั้น มีทิศทางที่ชัดเจน หรือเปลี่ยนแปลงไป ซึ่งในสถานการณ์จริงข้อมูลที่ต้องการใช้ในการตรวจจับการบุกรุกทางด้านเครือข่าย อาจมีจำนวนมากกว่าในกรณีทดลองในงานวิจัยนี้ อีกทั้งอาจมีปัจจัยจำกัดด้านเวลาที่ต้องการผลลัพธ์การตรวจจับที่รวดเร็วมากขึ้น โดยอยู่บนสมมติฐานการเลือกพารามิเตอร์ที่เหมาะสมในการสร้างระบบตรวจจับการบุกรุก โดยสามารถนำงานวิจัยนี้ไปศึกษาต่อในกรณีนี้ได้

5.3 บทสรุป

$$\mathrm{T_{Algo}} = \mathrm{T(c, w)} \tag{1}$$

จากงานวิจัยทำให้สามารถประมาณต้นทุนทางด้านเวลา สำหรับการสร้างแบบจำลองระบบตรวจการบุกรุกแต่ละรูปแบบเทคนิคการเรียนรู้ด้วยเครื่อง ดังสมการที่ (1)

$$c = \left(\frac{b_1}{b_2}\right)(w - a_2) + a_1 \tag{8}$$

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

และสามารถประมาณจำนวนตัวประมวลผลกลางที่เพียงพอและเหมาะสมสำหรับการสร้างระบบตรวจสอบการบุกรุกในแต่ละรูปแบบเทคนิคการเรียนรู้ด้วยเครื่อง ภายใต้เงื่อนไขที่ไม่ได้มีการปรับการใช้งานของหน่วยประมวลผล ได้ดังสมการ (8) อีกทั้งสำหรับการสร้างระบบตรวจสอบการบุกรุกในแต่ละรูปแบบเทคนิคการเรียนรู้ด้วยเครื่อง ภายใต้เงื่อนไขที่มีการปรับการใช้งานของหน่วยประมวลผลแบบขนานได้ จึงทำให้สามารถประมาณสัดส่วนของงานที่สามารถประมวลผลขนานกันได้(P) ดังสมการที่ (13)

$$\text{Overall Speedup (c)} = \frac{1}{(1-P)+\frac{P}{c}} \tag{13}$$

สุดท้ายนี้งานวิจัยยังวิเคราะห์ถึงแนวทางเลือกรูปแบบเทคนิคการเรียนรู้ด้วยเครื่องในการสร้างระบบตรวจจับการบุกรุกที่เหมาะสมและมีประสิทธิภาพด้านเวลาตามที่ได้กล่าวในการอภิปรายผลการวิจัย

บรรณานุกรม

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ประวัติผู้เขียน

ชื่อ-สกุล	PRAIYA TUNGJATURASOPON
วัน เดือน ปี เกิด	28 June 1990
สถานที่เกิด	Bangkok
ผลงานตีพิมพ์	PERFORMANCE ANALYSIS OF MACHINE LEARNING TECHNIQUES IN INTRUSION DETECTION (ICNS2018)

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

1. Aggarwal, P. and S.K.J.P.C.S. Sharma, *Analysis of KDD dataset attributes-class wise for intrusion detection.* 2015. 57: p. 842-851.
2. Adewumi), E.P.a.A.A.C.a.A., *Efficient Feature Selection Technique for Network Intrusion Detection System Using Discrete Differential Evolution and Decision Tree.* International Journal of Network Security, 2017. 19(5): p. 660-669.
3. Hee-su Chae, S.H.C., *Feature Selection for efficient Intrusion Detection using Attribute Ratio.* INTERNATIONAL JOURNAL OF COMPUTERS AND COMMUNICATIONS, 2014. 8: p. 134-139.
4. L.Dhanabal, D.S.P.S., *A Study on NSL-KDD Dataset for Intrusion Detection System Based on Classification Algorithms.* International Journal of Advanced Research in Computer and Communication Engineering, 2015. 4(6): p. 446-452.
5. Balon-Perin, A., *Ensemble-based methods for intrusion detection*. 2012, Institutt for datateknikk og informasjonsvitenskap.
6. Sofi, I., A. Mahajan, and V.J.I. Mansotra, *Machine learning techniques used for the detection and analysis of modern types of ddos attacks.* 2017. 4(06).
7. Tantikitti, S., *Image Processing for Detecting Dengue Virus from WBC*, in *Master of Science in Information Technology*. 2016, SIAM University. p. 72.
8. PACHARAWONGSAKDA, D.E. การแบ่งข้อมูลเพื่อนำทดสอบประสิทธิภาพของโมเดล. 2014; Available from: http://dataminingtrend.com/2014/data-mining-techniques/cross-validation/.
9. Krerk Piromsopa, P.D., สถาปัตยกรรมคอมพิวเตอร์ การออกแบบและการวิเคราะห์. 2561, กรุงเทพ: โรงพิมพ์แห่งจุฬาลงกรณ์มหาวิทยาลัย. 236.
10. Devaraju, S. and S.J.I.J.o.S.C. Ramakrishnan, *PERFORMANCE COMPARISON FOR INTRUSION DETECTION SYSTEM USING NEURAL NETWORK WITH KDD DATASET.* 2014. 4(3).
11. Keshri, A., et al. *DoS attacks prevention using IDS and data mining*. in *Accessibility to Digital World (ICADW),* 2016 *International Conference on*. 2016. IEEE.

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

12. Al-Yaseen, W.L., Z.A. Othman, and M.Z.A.J.E.S.w.A. Nazri, *Multi-level hybrid support vector machine and extreme learning machine based on modified K-means for intrusion detection system.* 2017. 67: p. 296-303.

13. Rangra, K., K.J.I.j.o.a.r.i.c.s. Bansal, and s. engineering, *Comparative study of data mining tools.* 2014. 4(6).

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ภาคผนวก

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

ภาคผนวก ก.
รายละเอียดผลการทดลองทั้งหมด

ตามที่ได้แยกการทดลองเป็น 2 ประเภทตามลักษณะการใช้งานของหน่วยประมวลผล คือ การทดลองที่ไม่ได้มีการปรับการใช้งานของหน่วยประมวลผล และ การทดลองที่มีการปรับ ให้สามารถใช้งานหน่วยประมวลผลได้ทุกหน่วยในเวลาเดียวกัน จึงมีรายละเอียดผลการทดลองดังนี้

1. การทดลองที่ไม่ได้มีการปรับการใช้งานของหน่วยประมวลผล
    - เทคนิคต้นไม้ตัดสินใจ ตามตารางที่ (7-8)
    - เทคนิคซัพพอร์ตเวกเตอร์แมชชีน ตามตารางที่ (9-10)
    - เทคนิคโครงข่ายประสามเทียมแบบ Multi-Layer Perceptron ตามตารางที่ (11-12)
    - ผลการทดลองโดยเฉลี่ยของกลุ่มผลลัพธ์ เพื่อนำไปใช้ในการเปรียบเทียบและวิเคราะห์ในงานวิจัย ตามตารางที่ (13)
2. การทดลองที่มีการปรับ ให้สามารถใช้งานหน่วยประมวลผลได้ทุกหน่วยในเวลาเดียวกัน
    - เทคนิคต้นไม้ตัดสินใจ ตามตารางที่ (14)
    - เทคนิคซัพพอร์ตเวกเตอร์แมชชีน ตามตารางที่ (15)
    - เทคนิคโครงข่ายประสามเทียมแบบ Multi-Layer Perceptron ตามตารางที่ (16-17)
    - ผลการทดลองโดยเฉลี่ยของกลุ่มผลลัพธ์ เพื่อนำไปใช้ในการเปรียบเทียบและวิเคราะห์ในงานวิจัย ตามตารางที่ (18)
3. ค่าความถูกต้อง (Accuracy) ของแต่ละเทคนิคการเรียนรู้ ที่ใช้ในการทดลองงานวิจัย ตามตารางที่ (19)
4. ค่าสมรรถนะของเวลาในการประมวลผลที่เพิ่มขึ้น ในช่วงหน่วยประมวลผลถึง 100 หน่วย ตามตารางที่ (20)
5. ค่าประสิทธิภาพของการเพิ่มหน่วยประมวลผล ในช่วงหน่วยประมวลผลถึง 100 หน่วย ตามตารางที่ (21)

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

CPU / Workloads	1					2					4				
10000	1.14	1.09	1.05	1.56	1.09	0.72	0.86	1.45	1.80	1.31	0.66	0.69	0.59	0.78	0.58
20000	2.53	2.84	2.58	2.73	2.52	3.23	2.72	2.14	2.92	2.55	1.44	1.47	1.55	1.53	1.49
30000	4.44	4.20	7.28	3.58	3.66	6.31	6.50	7.13	7.38	6.14	2.56	2.91	2.67	2.77	2.38
40000	16.22	11.42	6.49	6.22	5.99	9.34	10.27	4.39	9.27	9.83	4.88	4.27	4.30	3.94	3.69
50000	9.61	11.48	10.61	8.98	8.91	13.58	14.86	14.36	14.78	14.45	6.48	6.50	6.28	6.34	6.61
60000	9.17	8.36	10.22	9.78	9.89	14.50	14.38	14.53	14.58	14.06	9.06	10.03	9.44	10.44	9.14
70000	19.19	30.17	23.34	19.31	19.08	18.97	18.94	18.97	19.08	18.83	14.42	13.48	13.66	15.44	14.48
80000	25.14	13.00	13.64	19.28	12.53	23.08	21.58	22.20	22.25	22.41	16.98	22.89	18.45	17.31	17.14
90000	27.81	22.69	20.69	30.14	21.78	26.88	27.27	27.13	26.63	27.31	21.27	28.14	27.19	25.80	25.69
100000	29.80	29.73	30.36	27.50	30.19	29.03	26.80	27.34	31.17	29.19	31.03	28.75	33.17	32.53	30.23
110000	35.50	33.61	48.45	43.53	51.30	35.14	36.83	35.78	35.25	36.48	36.50	34.23	36.61	34.95	38.19
120000	58.08	53.13	43.98	43.45	54.38	51.38	41.45	40.78	40.41	41.05	39.39	43.39	43.67	36.16	48.92
130000	65.3	68.2	63.8	70	69.8	50.1	47.1	43.1	45.5	46.9	40.1	43.2	45.9	48	49.3
140000	79.8	82.7	77.7	73.3	84.3	49.1	53.4	54	57.2	50.2	48.9	51.3	46	49.1	53.1

ตารางที่ 7 ผลการทดลองของทุกกรณีในงานวิจัยที่การทดลองที่ไม่ได้มีการปรับการใช้งานของหน่วยประมวลผล (เทคนิคต้นไม้ตัดสินใจ) (1)

CPU / Workloads	8					16				
10000	0.56	0.59	0.56	0.58	0.56	0.59	0.56	0.55	0.64	0.61
20000	1.76	1.70	1.69	1.78	1.69	1.67	1.56	1.59	1.42	1.42
30000	2.81	2.95	2.91	2.72	3.03	2.61	2.50	2.58	2.42	2.34
40000	4.83	4.53	4.80	5.31	5.49	3.80	3.95	4.31	4.36	4.36
50000	5.39	6.39	6.66	7.22	7.13	7.30	7.27	6.88	7.34	6.16
60000	7.42	9.23	9.89	8.47	8.50	11.88	12.25	10.69	9.05	11.42
70000	10.95	12.41	9.88	10.78	10.17	17.19	11.47	13.86	16.55	12.53
80000	15.31	18.39	12.20	12.84	13.53	18.08	15.98	18.33	19.98	18.56
90000	20.30	18.39	19.23	15.77	21.25	25.06	21.28	24.98	26.86	26.00
100000	26.19	24.89	24.36	25.67	26.67	26.72	27.06	26.14	26.63	26.64
110000	28.92	34.36	27.84	28.58	30.14	28.53	31.77	29.30	32.41	27.05
120000	41.19	38.83	31.23	38.84	41.39	28.03	27.94	28.81	36.52	27.63
130000	38.1	33.7	41.7	35.5	32.9	39.1	35.1	37.2	29.8	33.6
140000	38.2	43.9	39.2	40.9	35.6	36.5	38.2	37.2	43.5	31.4

ตารางที่ 8 ผลการทดลองของทุกกรณีในงานวิจัยที่การทดลองที่ไม่ได้มีการปรับการใช้งานของหน่วยประมวลผล (เทคนิคต้นไม้ตัดสินใจ) (2)

CPU / Workloads	1					2					4				
10000	4.95	4.36	5.00	4.03	5.74	3.95	4.39	4.66	4.27	4.56	2.98	3.06	3.61	3.09	3.25
20000	8.31	10.86	8.53	10.11	10.41	8.75	8.98	9.42	9.20	9.16	6.59	7.06	6.81	6.80	6.64
30000	13.50	13.72	14.86	13.38	15.22	13.17	14.69	14.53	13.77	14.72	11.20	10.84	10.77	11.64	11.31
40000	21.98	34.48	34.00	19.45	18.66	16.58	17.16	24.89	19.72	19.23	15.56	16.23	15.23	14.80	15.83
50000	24.28	21.14	23.33	23.64	24.45	17.91	17.77	20.11	17.84	20.70	17.95	18.31	22.70	19.41	20.45
60000	27.73	29.92	28.70	27.75	29.08	32.48	37.28	35.61	37.42	36.41	26.17	25.05	25.02	25.58	24.97
70000	37.77	32.28	38.78	41.87	47.23	30.50	32.33	35.69	36.13	37.39	34.42	33.61	33.28	32.53	33.72
80000	44.00	45.86	60.94	46.38	42.56	33.08	33.16	35.14	31.61	33.72	43.64	41.77	40.23	41.14	44.91
90000	57.63	49.05	52.95	49.42	50.08	40.02	41.27	42.42	44.00	43.00	45.83	46.31	46.53	48.17	49.47
100000	64.67	74.72	53.91	63.11	73.77	51.33	58.44	53.33	47.92	48.89	64.86	72.44	76.28	66.25	72.13
110000	68.42	72.08	69.86	115.84	77.17	56.41	53.94	55.25	55.72	56.06	62.88	68.75	61.55	62.75	65.23
120000	91.66	222.91	205.31	105.67	122.97	64.63	86.08	79.52	77.06	81.48	72.27	71.88	74.44	77.84	75.88

ตารางที่ 9 ผลการทดลองของทุกกรณีในงานวิจัยที่การทดลองที่ไม่ได้มีการปรับการใช้งานของหน่วยประมวลผล (เทคนิคซัพพอร์ตเวกเตอร์แมชชีน) (1)

CPU / Workloads	8					16				
10000	3.16	3.19	2.86	2.97	3.06	2.84	3.02	2.81	3.32	3.41
20000	6.64	6.97	6.72	6.81	6.67	6.92	7.83	6.84	7.31	7.45
30000	11.67	11.20	11.55	11.28	11.58	11.30	10.89	10.45	11.56	11.77
40000	16.08	16.11	17.52	15.48	15.80	15.39	14.97	15.31	15.44	17.09
50000	20.91	20.55	21.36	20.16	20.41	22.36	20.47	22.03	20.97	21.98
60000	23.75	22.78	22.11	22.89	22.27	25.98	24.38	24.83	27.05	25.91
70000	30.41	30.66	29.83	29.63	29.17	33.92	34.78	34.59	34.33	36.11
80000	40.44	38.80	37.84	38.56	38.97	44.88	39.13	45.06	43.75	44.63
90000	49.33	47.20	48.66	48.36	48.25	52.69	51.02	53.23	52.41	52.91
100000	60.72	64.19	54.13	59.17	60.36	55.55	54.94	53.81	57.63	53.11
110000	70.41	66.44	66.23	66.13	66.53	72.45	70.34	70.92	73.30	74.66
120000	69.47	65.61	65.38	67.08	74.34	72.61	81.30	70.66	69.63	74.77

ตารางที่ 10 ผลการทดลองของทุกกรณีในงานวิจัยที่การทดลองที่ไม่ได้มีการปรับการใช้งานของหน่วยประมวลผล (เทคนิคซัพพอร์ตเวกเตอร์แมชชีน) (2)

CPU / Workloads	1					2					4				
10000	784.86	729.61	713.73	758.58	748.01	903.48	1186.80	598.78	575.33	763.76	742.20	950.78	881.28	843.13	735.16
20000	1258.37	1315.03	1570.58	3430.03	1421.30	1715.50	1471.31	1302.88	1226.16	1707.78	1563.64	1573.64	1574.77	1557.23	1516.02
30000	1938.69	1984.77	2007.44	1989.27	2047.66	2022.30	2016.50	1999.98	1854.50	2931.25	2471.75	2494.44	2335.03	2033.09	2705.76
40000	2613.81	2843.55	2644.97	2754.97	4545.47	2675.66	3344.26	2518.47	2562.64	3135.14	3512.45	3517.30	2666.24	3184.64	2993.53
50000	3968.69	3887.66	4067.58	4019.22	3868.13	3685.88	4039.66	4013.03	4029.49	4137.09	3939.33	5091.31	5110.73	5121.70	4540.77
60000	4154.91	4090.06	4068.73	4388.52	4063.83	4815.17	4746.31	3903.74	3934.32	4140.83	4606.66	4622.83	4364.81	5355.14	5012.78
70000	4972.80	4919.70	4989.12	4923.42	4965.80	6374.52	5299.78	4656.13	5496.34	4804.78	5841.48	5668.92	5330.56	5506.86	5805.39
80000	6372.97	11002.36	5676.56	5687.22	5916.39	6412.97	6155.23	5583.23	6409.11	6692.47	6551.73	6108.87	7361.69	5774.00	5858.81
90000	6372.97	6252.88	6448.91	6329.22	8747.22	6412.97	6149.48	12705.70	6963.94	7388.22	6551.73	6910.05	8111.47	7286.39	7667.97
100000	6269.53	8485.91	8566.83	8092.03	8072.20	6253.78	8090.12	8229.34	10394.40	9711.33	7278.95	7730.56	8660.77	10437.80	7845.33
110000	7951.25	8435.77	7570.19	7919.33	7616.67	7704.81	9137.17	9421.78	7328.59	10864.38	9287.12	8188.42	8784.20	11562.24	9717.42
120000	8398.20	10605.08	9527.38	8402.36	8180.38	9540.56	13128.98	9144.36	9559.66	9595.11	8956.30	9652.56	9379.64	9511.72	10652.28

ตารางที่ 11 ผลการทดลองของทุกกรณีในงานวิจัยที่การทดลองที่ไม่ได้มีการปรับการใช้งานของหน่วยประมวลผล (เทคนิคโครงข่ายประสามเทียมแบบ Multi-Layer Perceptron) (1)

Core CPU / Workloads	8					16				
10000	806.91	733.55	705.39	804.48	749.80	716.17	728.78	720.16	721.05	711.61
20000	1441.28	1451.61	1588.03	1337.61	1522.22	1529.98	1537.33	1724.47	1556.88	1536.34
30000	2276.08	2306.21	2197.23	2117.20	2230.03	2384.67	2004.59	2692.56	2418.11	2351.84
40000	3016.47	3102.17	3178.16	3025.64	2781.16	3617.03	3696.03	3297.20	3525.61	3588.31
50000	4188.45	4248.72	4815.17	5081.72	4122.87	3565.19	4050.53	4175.38	4249.44	4206.67
60000	4728.92	5347.65	4520.98	4633.70	4215.27	5767.08	4907.06	5332.88	4079.39	4987.09
70000	5733.30	6545.81	6667.19	5461.36	6875.14	6899.98	5723.45	5132.92	5864.02	6670.30
80000	5811.00	7004.47	7683.17	7412.72	7177.78	5502.41	7402.66	7146.67	5839.47	5603.19
90000	6915.45	7279.64	7509.06	7362.86	6858.11	6471.00	8462.72	6229.88	6439.58	8562.28
100000	8300.81	8314.70	8851.67	6763.66	9914.88	8175.40	7654.90	8413.69	7532.75	8307.27
110000	9625.42	10801.91	7849.14	9836.44	7869.83	7738.70	8332.41	10187.80	7772.78	9423.50
120000	10533.70	11283.41	10841.25	10272.05	9947.81	10342.95	8603.50	9635.81	8748.97	9942.84

ตารางที่ 12 ผลการทดลองของทุกกรณีในงานวิจัยที่การทดลองที่ไม่ได้มีการปรับการใช้งานของหน่วยประมวลผล (เทคนิคโครงข่ายประสามเทียมแบบ Multi-Layer Perceptron) (2)

Algorithm	DT					SVM					MLP				
Core CPU / Workloads	1	2	4	8	16	1	2	4	8	16	1	2	4	8	16
10000	1.11	1.37	0.61	0.58	0.60	4.77	4.20	3.10	3.06	2.99	745.40	755.34	819.86	748.62	721.70
20000	2.61	2.73	1.48	1.75	1.55	9.68	8.96	6.83	6.68	7.53	1416.75	1333.45	1545.63	1471.70	1541.07
30000	4.10	6.61	2.78	2.81	2.50	14.03	14.33	11.26	11.34	11.04	1993.83	2012.92	2333.09	2181.49	2371.75
40000	6.17	9.48	4.09	4.88	4.02	20.03	17.82	15.54	16.00	15.24	2747.83	2585.59	3341.09	3048.09	3536.75
50000	4.02	14.66	6.48	6.42	6.91	23.02	19.58	19.39	20.64	22.12	3917.78	4027.39	4914.27	4417.45	4168.88
60000	9.76	14.49	9.62	8.37	11.85	28.78	24.83	25.01	22.65	25.78	4104.57	4488.40	4996.92	4903.42	5059.78
70000	16.19	18.96	14.13	10.63	15.87	37.31	35.28	33.77	30.08	34.57	4958.21	5200.30	5660.39	6315.43	6478.10
80000	19.35	22.06	19.55	12.86	18.01	45.41	38.28	41.05	38.54	42.50	5760.06	6418.94	6674.10	7024.55	6796.27
90000	24.09	27.02	27.01	19.31	24.37	49.52	42.56	47.00	48.07	52.67	6383.70	6921.71	7043.25	7383.85	7077.25
100000	30.12	28.52	30.50	24.97	26.78	79.14	52.89	58.62	61.76	65.43	7615.82	8292.51	8520.69	7793.06	8045.86
110000	42.49	35.39	36.34	28.85	28.29	83.04	54.97	64.35	66.30	71.52	7829.08	8754.59	8753.25	8448.13	8509.56
120000	55.20	41.09	39.74	30.47	31.09	106.77	77.40	74.72	67.39	72.68	9326.98	9414.86	9538.85	9690.82	8996.09
130000	67.79	46.50	45.68	35.78	35.31										
140000	80.09	52.52	49.76	39.43	37.29										

ตารางที่ 13 ผลการทดลองโดยเฉลี่ยของทุกกรณีในงานวิจัยที่ไม่มีการปรับการใช้งานของหน่วย

CPU / Workloads	1					2					4					8					16				
10000	5	4	3	3	3	4	3	2	3	4	3	2	2	3	1	2	1	1	0	0	0	0	0	0	0
20000	11	9	10	9	10	9	8	8	7	8	5	4	4	5	4	2	2	2	2	2	1	1	1	1	1
30000	20	19	19	20	20	17	16	16	17	17	8	7	8	8	7	3	3	3	3	3	3	2	2	2	2
40000	39	30	33	25	28	29	27	26	26	28	11	11	10	11	11	5	5	5	5	5	2	3	3	3	3
50000	63	51	51	48	53	37	46	49	43	48	16	16	17	17	17	9	8	9	8	8	4	5	4	5	4
60000	78	72	73	65	91	84	76	60	63	68	29	27	29	30	32	14	12	13	11	12	6	7	7	5	6
70000	108	126	132	95	111	73	82	79	85	88	42	44	46	57	55	19	18	16	16	16	8	9	8	8	8
80000	153	133	154	205	134	110	105	102	117	109	60	60	55	54	53	28	21	21	23	23	12	11	12	11	12
90000	249	271	176	221	231	142	143	127	120	126	66	71	66	55	55	31	32	30	31	35	17	17	14	17	15
100000	219	234	279	353	286	129	125	130	125	125	76	72	68	74	75	39	38	37	38	41	19	18	18	18	19
110000	251	243	255	295	256	150	152	141	149	156	85	85	79	113	110	50	56	62	50	45	23	22	20	21	21
120000	356	308	343	298	319	200	200	197	199	162	96	115	121	110	108	66	60	46	47	59	20	27	27	25	25

ตารางที่ 14 ผลการทดลองของทุกกรณีในงานวิจัยที่มีการปรับการใช้งานของหน่วยประมวลผล (เทคนิคต้นไม้ตัดสินใจ)

CPU / Workloads	1					2					4					8					16				
10000	33	24	20	21	20	14	13	13	13	13	11	8	8	8	8	4	4	4	4	4	2	2	2	2	2
20000	54	54	54	53	54	34	33	33	33	32	21	18	17	17	18	11	10	11	11	11	5	5	6	6	5
30000	92	93	92	94	91	56	55	54	55	56	29	29	29	29	29	18	18	18	18	18	9	10	9	9	9
40000	135	134	135	137	135	84	82	83	80	79	43	44	44	44	43	27	27	27	27	26	14	13	13	13	13
50000	186	190	191	190	189	105	104	106	106	103	58	58	58	56	57	43	38	37	39	38	18	18	18	18	19
60000	255	239	231	232	234	148	135	137	135	137	72	70	78	71	70	55	47	46	47	47	24	23	23	23	23
70000	334	317	309	328	320	158	159	170	172	172	97	96	97	97	98	66	66	68	65	67	41	33	34	33	33
80000	376	383	402	393	392	199	197	190	191	195	143	143	132	137	129	85	83	82	85	83	38	37	38	38	39
90000	463	462	467	457	473	232	232	232	233	232	152	158	154	151	152	101	111	102	98	97	46	47	47	45	45
100000	570	580	575	569	566	281	280	274	277	274	188	231	206	196	197	128	126	124	124	126	59	59	58	61	58
110000	660	656	610	628	670	315	324	311	313	310	217	203	201	197	204	139	138	133	134	142	124	74	66	68	77
120000	760	751	748	724	645	409	376	401	379	374	231	228	235	236	230	205	189	175	200	182	82	77	83	78	84

ตารางที่ 15 ผลการทดลองของทุกกรณีในงานวิจัยที่มีการปรับการใช้งานของหน่วย (เทคนิคซัพพอร์ตเวกเตอร์แมชชีน)

CPU / Workloads	1					2					4				
10000	776	737	606	605	603	510	627	508	554	549	276	262	261	269	270
20000	1614	1290	1289	1300	1280	1138	1071	1198	1187	1184	584	568	587	583	580
30000	1938	1934	1938	1934	1938	1829	1805	1807	1820	1620	1041	1056	901	893	880
40000	2593	2600	2665	2656	2885	1904	1925	2541	2527	2490	1196	1193	1190	1177	1190
50000	3264	3268	3397	3234	3502	2938	2833	2967	2936	2684	1482	1475	1478	1492	1487
60000	4428	4223	4497	4145	4135	3269	3310	3316	3281	2989	1863	1898	1879	1827	1818
70000	4967	5041	5035	5015	4996	3716	3756	3394	3362	3894	2225	2217	2228	2218	2215
80000	5773	5629	5724	5874	2490	3953	4185	4362	4241	4233	2492	1929	2724	2701	2666
90000	6407	6659	6659	6408	6377	4721	4863	4407	4756	4553	2613	2814	2710	2737	2764
100000	7761	7070	6975	7158	7321	5429	5041	5248	5053	5900	3840	3216	3597	3544	2930
110000	8562	7128	6805	7888	7812	5198	6067	5487	5188	8783	3173	3815	3077	3542	3079
120000	9341	8520	9537	12026	8010	6384	6726	6242	7391	7578	3574	4061	3200	4231	3831

ตารางที่ 16 ผลการทดลองของทุกกรณีในงานวิจัยที่มีการปรับการใช้งานของหน่วย (เทคนิคโครงข่ายประสามเทียมแบบ Multi-Layer Perceptron) (1)

CPU / Workloads	8					16				
10000	170	167	190	193	192	86	85	84	86	85
20000	401	396	399	412	401	181	187	181	182	183
30000	553	614	612	614	548	288	475	277	274	276
40000	750	748	739	742	754	389	384	390	385	387
50000	922	931	946	972	925	454	461	452	450	451
60000	1113	1190	1186	1196	988	543	561	559	552	552
70000	1324	1334	1322	1350	1322	673	668	668	655	646
80000	1630	1602	1612	1608	1639	835	780	775	764	776
90000	1849	1843	1797	1831	1826	763	898	759	923	834
100000	1896	2174	1834	1938	1933	865	901	1066	911	899
110000	2278	2462	2135	2531	1887	1004	1133	1039	1325	1063
120000	2527	2322	2050	2051	2308	1118	1505	1073	1237	1270

ตารางที่ 17 ผลการทดลองของทุกกรณีในงานวิจัยที่มีการปรับการใช้งานของหน่วย (เทคนิคโครงข่ายประสามเทียมแบบ Multi-Layer Perceptron) (2)

Algorithm	DT					SVM					MLP				
CPU / Workloads	1	2	4	8	16	1	2	4	8	16	1	2	4	8	16
10000	3.33	3.33	2.33	0.67	0.00	21.67	13.00	8.00	4.00	2.00	649.33	537.67	267.00	184.00	85.33
20000	9.67	8.00	4.33	2.00	1.00	54.00	33.00	17.67	11.00	5.33	1293.00	1169.67	582.33	400.33	182.00
30000	19.67	16.67	7.67	3.00	2.00	92.33	55.33	29.00	18.00	9.00	1936.67	1810.67	945.00	593.00	280.33
40000	30.33	27.00	11.00	5.00	3.00	135.00	81.67	43.67	27.00	13.00	2640.33	2314.00	1191.00	746.67	387.00
50000	51.67	45.67	16.67	8.33	4.33	189.67	105.00	57.67	38.33	18.00	3309.67	2951.33	1482.33	934.00	452.33
60000	74.33	69.00	29.33	12.33	6.33	235.00	136.33	71.00	47.00	23.00	4265.33	3286.67	1856.33	1163.00	554.33
70000	115.00	82.00	48.33	16.67	8.00	321.67	167.00	97.00	66.33	33.33	5015.33	3622.00	2220.00	1326.67	663.67
80000	147.00	108.00	56.33	22.33	11.67	389.33	194.33	137.33	83.67	38.00	5708.67	4219.67	2619.67	1616.67	777.00
90000	233.67	131.67	62.33	31.33	16.33	464.00	232.00	152.67	100.33	46.00	6491.33	4676.67	2737.00	1833.33	831.67
100000	266.33	126.33	73.67	38.33	18.33	571.33	277.00	199.67	125.33	58.67	7183.00	5243.33	3452.33	1922.33	903.67
110000	254.00	150.33	93.33	52.00	21.33	648.00	313.00	202.67	137.00	73.00	7609.33	5584.00	3264.67	2291.67	1078.33
120000	323.33	198.67	111.00	55.33	25.67	741.00	385.33	232.00	190.33	80.67	9132.67	6833.67	3822.00	2227.00	1208.33
120000 (ปรับค่าเอนเอียง)	227.00	143.00	102.67	41.67	22.00	757.33	484.67	315.00	200.33	94.33	7514.00	6945.33	3545.67	2057.33	1142.00

ตารางที่ 18 ผลการทดลองโดยเฉลี่ยของทุกกรณีในงานวิจัยที่มีการปรับการใช้งานของหน่วย

Algorithm / Workloads	DT	SVM	MLP
10000	99.2	98.1	92.5
20000	99.4	98.2	94.3
30000	99.5	98.4	93.5
40000	99.5	98.4	95.2
50000	99.6	98.5	92.9
60000	99.6	98.6	95.3
70000	99.7	98.8	94.9
80000	99.7	98.8	94.4
90000	99.7	98.8	94.3
100000	99.7	98.9	93.5
110000	99.7	98.9	95.7
120000	99.7	99	94.1
120000 (ปรับค่าโน้มเอียงข้องข้อมูล)	98.31	98.49	80.08

ตารางที่ 19 เปอร์เซ็นต์ค่าความถูกต้อง (Accuracy) ของแต่ละเทคนิคการเรียนรู้

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

Algorithm / CPU	DT	SVM	MLP
1	1	1	1
2	1.953897782	1.912521277	1.826775969
3	2.864809633	2.748561586	2.521750095
4	3.73557717	3.517349326	3.11411277
5	4.568796944	4.226685602	3.625027188
6	5.366846307	4.883209897	4.070211142
7	6.131906059	5.492608518	4.461582587
8	6.865980355	6.059779727	4.808342474
9	7.57091423	6.588965679	5.117707267
10	8.248409088	7.08385872	5.395416594
11	8.900036409	7.547687663	5.646092647
12	9.52724992	7.983288316	5.873499504
13	10.13139642	8.39316151	6.08073343
14	10.71372543	8.77952114	6.270364689
15	11.27539783	9.144334171	6.444544693
16	11.81749358	9.489354131	6.605087982
17	12.34101864	9.816149298	6.753535675
18	12.84691121	10.12612653	6.891205099
19	13.33604734	10.42055152	7.019228993
20	13.80924598	10.70056606	7.138586738
21	14.26727359	10.96720284	7.250129467
22	14.71084825	11.22139819	7.354600386
23	15.14064341	11.46400303	7.452651362
24	15.55729135	11.69579244	7.54485653
25	15.96138621	11.91747388	7.631723548
26	16.35348691	12.12969443	7.713702948
27	16.73411963	12.33304708	7.791195949
28	17.10378024	12.52807631	7.864561024
29	17.46293642	12.71528289	7.93411945
30	17.81202965	12.89512822	8.000160003
31	18.151477	13.06803811	8.062942974
32	18.48167282	13.23440615	8.122703591
33	18.80299025	13.3945967	8.179654967
34	19.11578261	13.54894757	8.23399064
35	19.42038475	13.69777235	8.285886767

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

36	19.71711418	13.84136261	8.335504038
37	20.00627224	13.97998972	8.382989329
38	20.28814505	14.11390666	8.428477163
39	20.56300452	14.24334945	8.472090977
40	20.83110918	14.36853865	8.513944244
41	21.09270501	14.48968052	8.554141456
42	21.3480262	14.60696822	8.592778997
43	21.59729582	14.7205828	8.629945912
44	21.8407265	14.83069414	8.665724597
45	22.07852103	14.93746183	8.700191404
46	22.31087291	15.04103587	8.733417187
47	22.53796688	15.14155745	8.765467787
48	22.75997942	15.23915956	8.79640447
49	22.97707919	15.33396755	8.826284314
50	23.18942748	15.42609973	8.855160566
51	23.39717858	15.51566778	8.883082952
52	23.60048018	15.60277729	8.910097968
53	23.79947372	15.68752812	8.936249136
54	23.99429469	15.77001478	8.961577238
55	24.18507297	15.8503268	8.986120528
56	24.37193311	15.92854908	9.009914929
57	24.55499457	16.00476212	9.0329942
58	24.73437203	16.07904235	9.055390104
59	24.9101756	16.15146239	9.077132549
60	25.08251101	16.22209124	9.098249724
61	25.2514799	16.29099455	9.118768219
62	25.41717996	16.35823479	9.138713137
63	25.57970514	16.42387145	9.158108197
64	25.73914582	16.48796121	9.176975828
65	25.89558899	16.55055814	9.195337257
66	26.04911838	16.61171378	9.213212584
67	26.19981465	16.67147734	9.230620863
68	26.3477555	16.72989583	9.24758016
69	26.49301583	16.78701415	9.264107625
70	26.63566782	16.84287522	9.280219543
71	26.77578112	16.89752011	9.295931393
72	26.91342288	16.9509881	9.311257893

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

73	27.04865794	17.00331681	9.326213046
74	27.18154887	17.05454227	9.340810189
75	27.31215609	17.104699	9.355062024
76	27.44053798	17.15382011	9.368980661
77	27.56675092	17.20193734	9.38257765
78	27.69084942	17.24908115	9.395864013
79	27.81288617	17.2952808	9.408850274
80	27.93291213	17.34056435	9.421546488
81	28.05097659	17.38495879	9.433962264
82	28.16712724	17.42849006	9.446106793
83	28.28141025	17.47118307	9.457988867
84	28.3938703	17.51306183	9.469616903
85	28.50455067	17.55414939	9.480998963
86	28.61349326	17.59446797	9.492142768
87	28.72073868	17.63403897	9.503055724
88	28.8263263	17.67288297	9.513744929
89	28.93029424	17.71101984	9.524217195
90	29.03267951	17.7484687	9.53447906
91	29.13351795	17.78524802	9.544536802
92	29.23284437	17.82137558	9.55439645
93	29.33069252	17.85686856	9.564063802
94	29.42709515	17.89174353	9.573544427
95	29.52208407	17.9260165	9.582843684
96	29.61569015	17.95970292	9.591966728
97	29.70794335	17.99281771	9.600918521
98	29.79887281	18.02537531	9.60970384
99	29.88850681	18.05738967	9.618327285
100	29.97687284	18.08887426	9.626793291

ตารางที่ 20 ค่าสมรรถนะของเวลาในการประมวลผลที่เพิ่มขึ้น ในช่วงหน่วยประมวลผลถึง 100 หน่วย

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

Algorithm / CPU	DT	SVM	MLP
1	1	1	1
2	0.976948891	0.956260638	0.913387984
3	0.954936544	0.916187195	0.840583365
4	0.933894293	0.879337331	0.778528192
5	0.913759389	0.84533712	0.725005438
6	0.894474384	0.813868316	0.678368524
7	0.87598658	0.78465836	0.637368941
8	0.858247544	0.757472466	0.601042809
9	0.841212692	0.732107298	0.568634141
10	0.824840909	0.708385872	0.539541659
11	0.809094219	0.686153424	0.51328115
12	0.793937493	0.665274026	0.489458292
13	0.779338186	0.645627808	0.467748725
14	0.765266102	0.627108653	0.447883192
15	0.751693189	0.609622278	0.429636313
16	0.738593349	0.593084633	0.412817999
17	0.725942273	0.577420547	0.397266804
18	0.713717289	0.562562585	0.382844728
19	0.701897228	0.54845008	0.369433105
20	0.690462299	0.535028303	0.356929337
21	0.679393981	0.522247754	0.34524426
22	0.66867492	0.510063554	0.334300018
23	0.658288844	0.498434914	0.32402832
24	0.648220473	0.487324685	0.314369022
25	0.638455449	0.476698955	0.305268942
26	0.628980266	0.466526709	0.296680883
27	0.619782209	0.456779522	0.288562813
28	0.610849294	0.447431297	0.280877179
29	0.602170221	0.438458031	0.273590326
30	0.593734322	0.429837607	0.266672
31	0.585531516	0.421549616	0.260094935
32	0.577552276	0.413575192	0.253834487
33	0.569787583	0.40589687	0.247868332
34	0.5622289	0.398498458	0.242176195
35	0.554868136	0.391364924	0.236739622

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

36	0.547697616	0.384482295	0.231541779
37	0.54071006	0.37783756	0.226567279
38	0.533898554	0.371418596	0.221802031
39	0.527256526	0.365214088	0.217233102
40	0.520777729	0.359213466	0.212848606
41	0.51445622	0.353406842	0.208637596
42	0.508286338	0.347784958	0.204589976
43	0.502262693	0.342339135	0.200696417
44	0.496380148	0.337061231	0.196948286
45	0.490633801	0.331943596	0.193337587
46	0.485018976	0.326979041	0.189856895
47	0.47953121	0.322160797	0.186499315
48	0.474166238	0.317482491	0.183258426
49	0.468919983	0.312938113	0.180128251
50	0.46378855	0.308521995	0.177103211
51	0.458768207	0.30422878	0.174178097
52	0.453855388	0.30005341	0.171348038
53	0.449046674	0.295991097	0.168608474
54	0.444338791	0.292037311	0.165955134
55	0.4397286	0.28818776	0.16338401
56	0.435213091	0.284438376	0.160891338
57	0.430789378	0.2807853	0.158473582
58	0.42645469	0.277224868	0.156127416
59	0.422206366	0.2737536	0.153849704
60	0.41804185	0.270368187	0.151637495
61	0.413958687	0.267065484	0.149488004
62	0.409954516	0.263842497	0.147398599
63	0.406027066	0.260696372	0.145366797
64	0.402174153	0.257624394	0.143390247
65	0.398393677	0.254623971	0.141466727
66	0.394683612	0.251692633	0.13959413
67	0.39104201	0.24882802	0.137770461
68	0.387466993	0.24602788	0.135993826
69	0.383956751	0.24329006	0.134262429
70	0.38050954	0.240612503	0.132574565
71	0.377123678	0.237993241	0.130928611
72	0.37379754	0.23543039	0.129323026

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

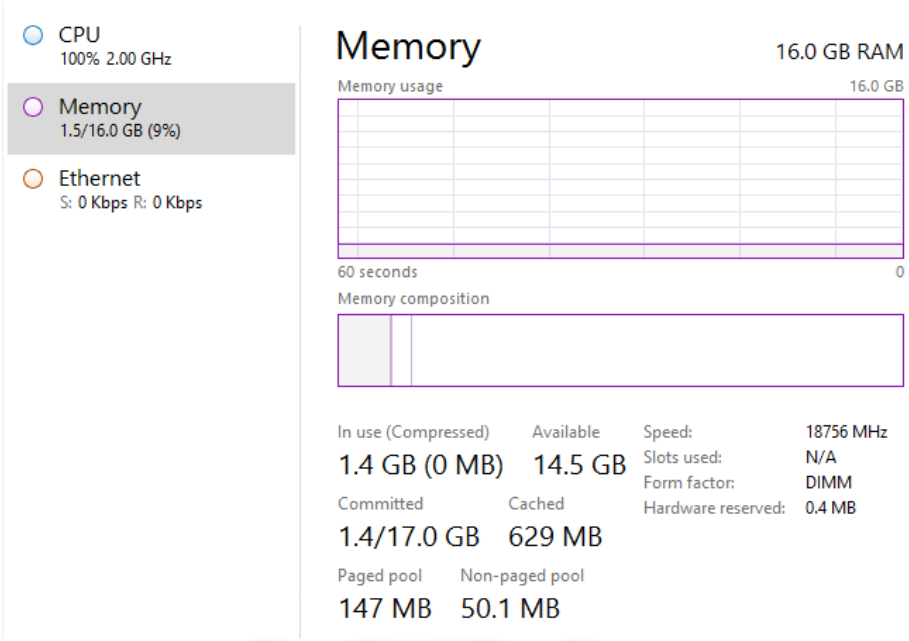
73	0.370529561	0.232922148	0.127756343
74	0.367318228	0.230466787	0.126227165
75	0.364162081	0.228062653	0.12473416
76	0.36105971	0.225708159	0.123276061
77	0.358009752	0.223401784	0.121851658
78	0.35501089	0.221142066	0.120459795
79	0.35206185	0.218927605	0.119099371
80	0.349161402	0.216757054	0.117769331
81	0.346308353	0.214629121	0.11646867
82	0.343501552	0.212542562	0.115196424
83	0.340739883	0.210496182	0.113951673
84	0.338022266	0.208488831	0.112733535
85	0.335347655	0.206519405	0.111541164
86	0.332715038	0.204586837	0.110373753
87	0.330123433	0.202690103	0.109230526
88	0.32757189	0.200828216	0.108110738
89	0.325059486	0.199000223	0.107013676
90	0.322585328	0.197205208	0.105938656
91	0.320148549	0.195442286	0.10488502
92	0.317748308	0.193710604	0.103852135
93	0.315383791	0.192009339	0.102839396
94	0.313054204	0.190337697	0.101846217
95	0.31075878	0.188694911	0.100872039
96	0.308496772	0.187080239	0.09991632
97	0.306267457	0.185492966	0.098978541
98	0.304070131	0.183932401	0.098058202
99	0.301904109	0.182397875	0.097154821
100	0.299768728	0.180888743	0.096267933

ตารางที่ 21 ค่าประสิทธิภาพของการเพิ่มหน่วยประมวลผล ในช่วงหน่วยประมวลผลถึง 100 หน่วย

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

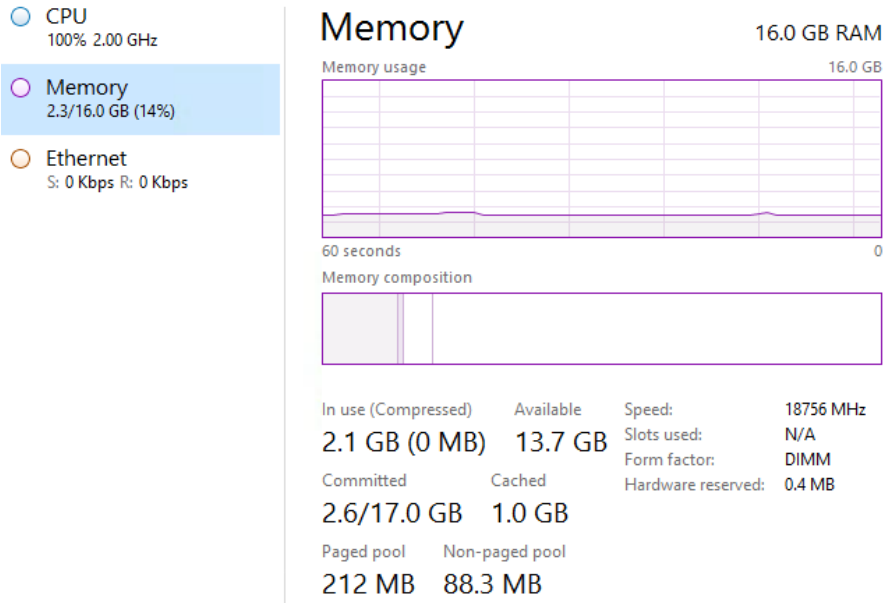
ภาคผนวก ข.
การใช้งานหน่วยความจำของกรณีทดลองในงานวิจัย

การทดลองทุกกรณีในงานวิจัย ได้ใช้เครื่องที่ใช้หน่วยความจำ 16 กิกะไบต์ โดยจากการทดลองพบว่า ปริมาณหน่วยความจำนั้นเพียงพอต่อการทดลองในทุกกรณีศึกษา แสดงได้ตามรูปที่ (32-36)

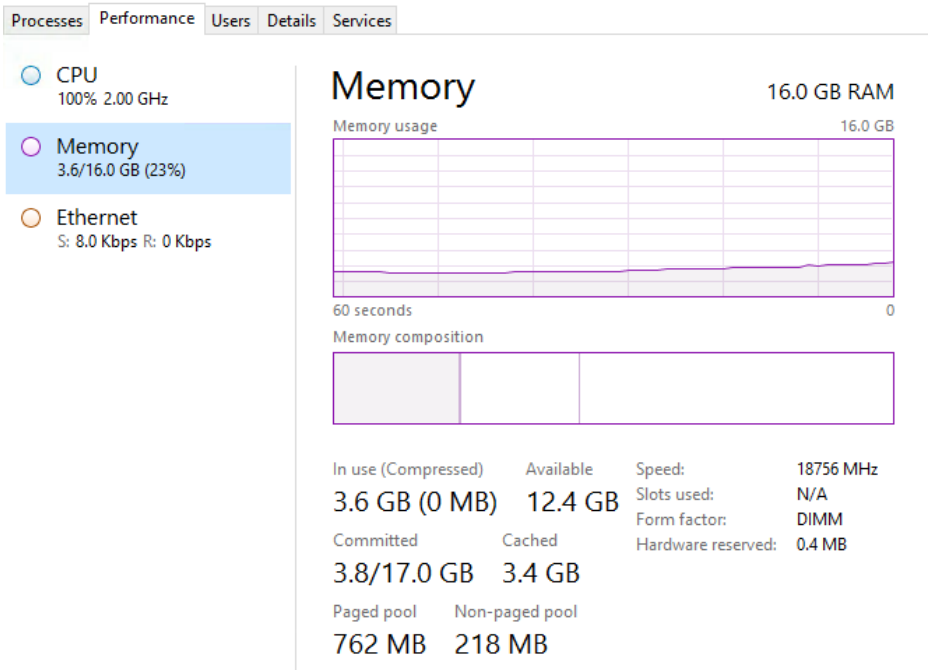


รูปที่ 32 การใช้งานหน่วยความจำ ในกรณีการทดลองหน่วยประมวลผล 1 หน่วย

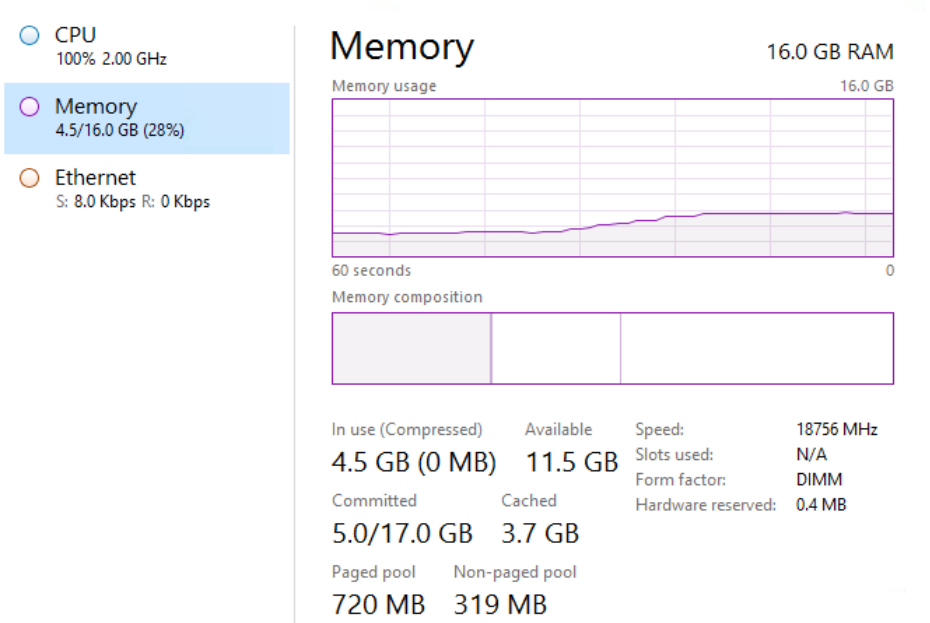
จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY



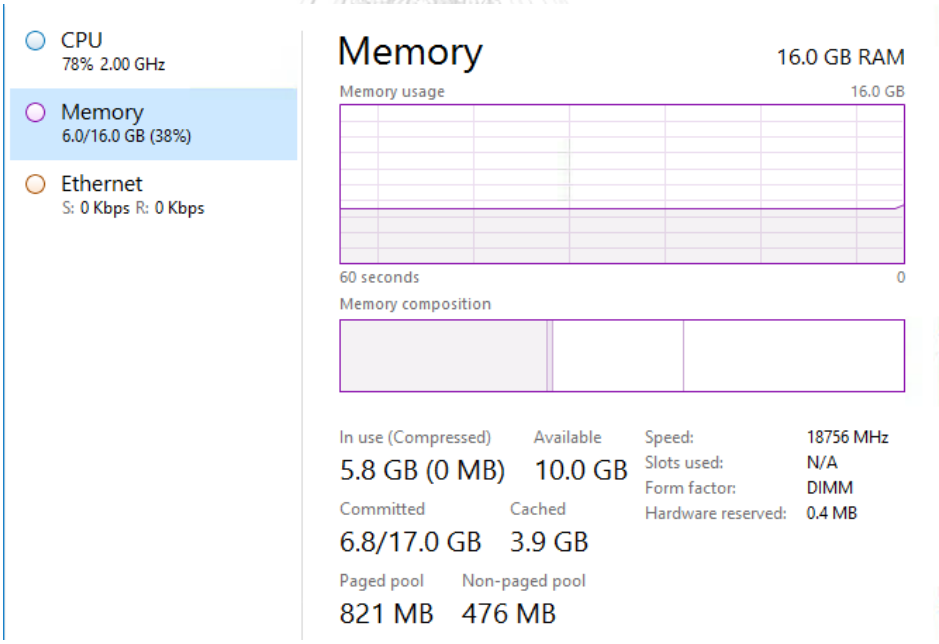
รูปที่ 33 การใช้งานหน่วยความจำ ในกรณีการทดลองหน่วยประมวลผล 2 หน่วย



รูปที่ 34 การใช้งานหน่วยความจำ ในกรณีการทดลองหน่วยประมวลผล 4 หน่วย



รูปที่ 35 การใช้งานหน่วยความจำ ในกรณีการทดลองหน่วยประมวลผล 8 หน่วย



รูปที่ 36 การใช้งานหน่วยความจำ ในกรณีการทดลองหน่วยประมวลผล 16 หน่วย

ภาคผนวก ค.
การใช้งานเครื่องมือต่างๆในงานวิจัย

1. การเตรียมข้อมูล (Data Preprocessing) โดยนำชุดข้อมูลมาคัดเลือกอย่างสุ่ม (Random function) .ให้ได้ตามจำนวนที่ต้องการของแต่ละกรณีทดลอง จากนั้นจึงปรับค่า attribute ให้อยู่ในรูปแบบที่เหมาะสม และบันทึกอยู่ในรูปแบบนามสกุล .arff เพื่อใช้กับโปรแกรม Weka ในเครื่องอื่นๆต่อไปได้ ตามรายละเอียดดังนี้

@attribute duration numeric

@attribute protocol_type {tcp,udp,icmp}

@attribute service

{ftp_data,other,private,http,remote_job,name,netbios_ns,eco_i,mtp,telnet,finger,domain_u,supdup,uucp_path,Z39_50,smtp,csnet_ns,uucp,netbios_dgm,urp_i,auth,domain,ftp,bgp,ldap,ecr_i,gopher,vmnet,systat,http_443,efs,whois,imap4,iso_tsap,echo,klogin,link,sunrpc,login,kshell,sql_net,time,hostnames,exec,ntp_u,discard,nntp,courier,ctf,ssh,daytime,shell,netstat,pop_3,nnsp,IRC,pop_2,printer,tim_i,pm_dump,red_i,netbios_ssn,rje,X11,urh_i,http_8001}

@attribute flag {SF,S0,REJ,RSTR,SH,RSTO,S1,RSTOS0,S3,S2,OTH}

@attribute src_bytes numeric

@attribute dst_bytes numeric

@attribute land_binarized {0,1}

@attribute wrong_fragment numeric

@attribute urgent numeric

@attribute hot numeric

@attribute num_failed_logins numeric

@attribute logged_in_binarized {0,1}

@attribute num_compromised numeric

@attribute root_shell_binarized {0,1}

@attribute su_attempted_binarized {0,1}

@attribute num_root numeric

@attribute num_file_creations numeric

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

@attribute num_shells numeric

@attribute num_access_files numeric

@attribute num_outbound_cmds numeric

@attribute is_host_login_binarized {0,1}

@attribute is_guest_login_binarized {0,1}

@attribute count numeric

@attribute srv_count numeric

@attribute serror_rate numeric

@attribute srv_serror_rate numeric

@attribute rerror_rate numeric

@attribute srv_rerror_rate numeric

@attribute same_srv_rate numeric

@attribute diff_srv_rate numeric

@attribute srv_diff_host_rate numeric

@attribute dst_host_count numeric

@attribute dst_host_srv_count numeric

@attribute dst_host_same_srv_rate numeric

@attribute dst_host_diff_srv_rate numeric

@attribute dst_host_same_src_port_rate numeric

@attribute dst_host_srv_diff_host_rate numeric

@attribute dst_host_serror_rate numeric

@attribute dst_host_srv_serror_rate numeric

@attribute dst_host_rerror_rate numeric

@attribute dst_host_srv_rerror_rate numeric

@attribute Class

{normal,neptune,warezclient,ipsweep,portsweep,teardrop,nmap,satan,smurf,pod,back,guess_passwd,ftp_write,multihop,rootkit,buffer_overflow,imap,warezmaster,phf,land,loadmodule}

@data

.....

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University