# การเปรียบเทียบประสิทธิภาพการตรวจจับมัลแวร์โดยใช้เทคนิคเหมืองข้อมูล

วิทยานิพนธ์

ของ

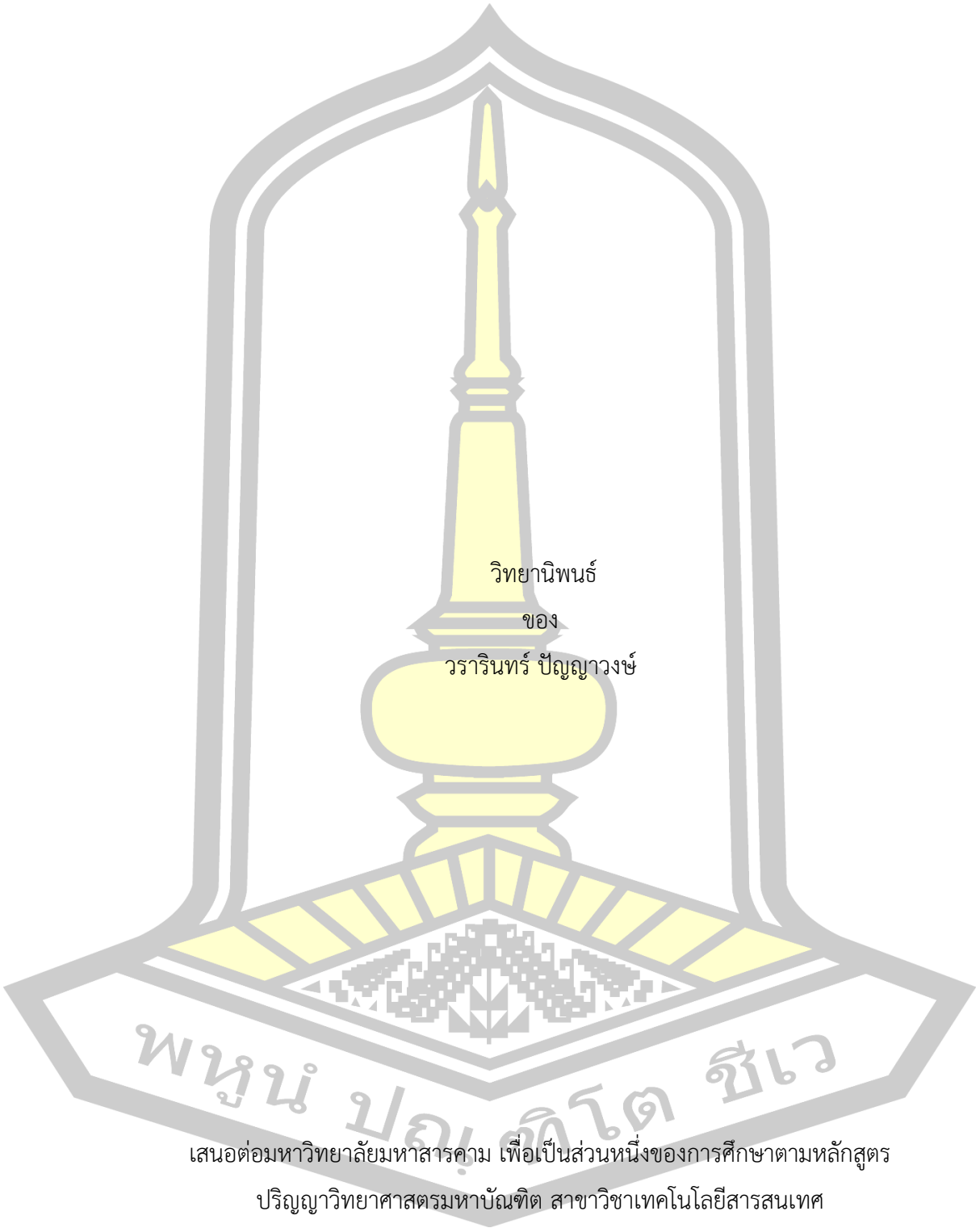
วรารินทร์ ปัญญาวงษ์

เสนอต่อมหาวิทยาลัยมหาสารคาม เพื่อเป็นส่วนหนึ่งของการศึกษาตามหลักสูตร

ปริญญาวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ

สิงหาคม 2562

# การเปรียบเทียบประสิทธิภาพการตรวจจับมัลแวร์โดยใช้เทคนิคเหมืองข้อมูล



วิทยานิพนธ์

ของ

วรารินทร์ ปัญญาวงษ์

เสนอต่อมหาวิทยาลัยมหาสารคาม เพื่อเป็นส่วนหนึ่งของการศึกษาตามหลักสูตร

ปริญญาวิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ

สิงหาคม 2562

A comparison of detecting malware effectiveness using data mining techniques

Wararin Panyawong

A Thesis Submitted in Partial Fulfillment of Requirements

for Master of Science (Information Technology)

August 2019

คณะกรรมการสอบวิทยานิพนธ์ ได้พิจารณาวิทยานิพนธ์ของนายวรารินทร์ ปัญญาวงษ์ แล้วเห็นสมควรรับเป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญา วิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ ของมหาวิทยาลัยมหาสารคาม

คณะกรรมการสอบวิทยานิพนธ์

.....ประธานกรรมการ
(ผศ. ดร. วรปภา อารีราษฎร์ )

.....อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก
(ผศ. ดร. จิรัฏฐา ภูบุญอบ )

.....กรรมการ
(ผศ. ดร. แกมกาญจน์ สมประเสริฐศรี )

.....กรรมการ
(ผศ. ดร. ฉัตรเกล้า เจริญผล )

มหาวิทยาลัยอนุมัติให้รับวิทยานิพนธ์ฉบับนี้ เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญา วิทยาศาสตรมหาบัณฑิต สาขาวิชาเทคโนโลยีสารสนเทศ ของมหาวิทยาลัยมหาสารคาม

.....
(ผศ. ศศิธร แก้วมั่น )
คณบดีคณะวิทยาการสารสนเทศ

.....
(ผศ. ดร. กริสน์ ชัยมูล )
คณบดีบัณฑิตวิทยาลัย

**ชื่อเรื่อง** การเปรียบเทียบประสิทธิภาพการตรวจจับมัลแวร์โดยใช้เทคนิคเหมืองข้อมูล
**ผู้วิจัย** วรารินทร์ ปัญญาวงษ์
**อาจารย์ที่ปรึกษา** ผู้ช่วยศาสตราจารย์ ดร. จิรัฏฐา ภูบุญอบ
**ปริญญา** วิทยาศาสตรมหาบัณฑิต **สาขาวิชา** เทคโนโลยีสารสนเทศ
**มหาวิทยาลัย** มหาวิทยาลัยมหาสารคาม **ปีที่พิมพ์** 2562

## บทคัดย่อ

งานวิจัยนี้มีวัตถุประสงค์เพื่อนำเสนอการเปรียบเทียบประสิทธิภาพในการตรวจจับมัลแวร์ประเภทแอดแวร์ (adware) โดยใช้ 3 แบบจำลองอันได้แก่ ซัพพอร์ตเวกเตอร์แมชชีน (Support Vector Machine) นาอีฟเบย์ (Naive Bayes) และเพื่อนบ้านใกล้สุด K ตัว (k-Nearest Neighbor) ข้อมูลที่ใช้ในการวิจัยเป็นข้อมูลจากเว็บไซต์ Malwaredomainlist (MDL) ปี ค.ศ. 2009 ถึง 2017 จำนวน 2,311 โดเมน ที่เกิดขึ้นบน Internet Browser ประกอบด้วย Worm, Trojan, Spyware และ virus ซึ่งผลลัพธ์จากการทดลองพบว่า ซัพพอร์ตเวกเตอร์แมชชีน มีความถูกต้อง 95.41% นาอีฟเบย์ มีความถูกต้อง 91.22% และเพื่อนบ้านใกล้สุด K ตัว มีความถูกต้องที่ 92.98% ตามลำดับ

คำสำคัญ : มัลแวร์, ซัพพอร์ตเวกเตอร์แมชชีน, นาอีฟเบย์, เพื่อนบ้านใกล้สุด K ตัว

**TITLE** A comparison of detecting malware effectiveness using data mining techniques

**AUTHOR** Wararin Panyawong

**ADVISORS** Assistant Professor Jiratta Phuboon-ob , Ph.D.

**DEGREE** Master of Science **MAJOR** Information Technology

**UNIVERSITY** Mahasarakham University **YEAR** 2019

## ABSTRACT

This research is intended to present an effective comparison of malware detection models for adware. There are 3 models including Support Vector Machine, Naive Bayes and k-Nearest Neighbor. The data used in the research is the adware malware information from the website Malwaredomainlist (MDL) 1 which are collected between year 2009 to 2017 from 2311 domains. The collected incidents on the Internet Browser are the Trojan, Worm, Spyware, virus. The results show that the Support Vector Machine is 95.41 percent accurate, the Naive Bayes is91.22 percent accurate and the k-Nearest Neighbor is 92.98 percent accurate.

Keyword : malware, Support Vector Machine, Naive Bayes, k-Nearest Neighbor

# กิตติกรรมประกาศ

งานวิจัยฉบับนี้มีจุดมุ่งหมายเพื่อ วิจัยการเปรียบเทียบคุณลักษณะที่เหมาะสมสำหรับการตรวจจับมัลแวร์ ที่ได้ให้คำปรึกษาและข้อมูลเพื่อใช้ในการประกอบการจัดทำวิจัย และการดำเนินการวิจัยมิอาจสำเร็จลุล่วงไปได้หากปราศจากความร่วมมือของอาจารย์ที่ปรึกษา ผู้ช่วยศาสตราจารย์ ดร. จิรัฏฐา ภูบุญอบ จนงานวิจัยนี้สำเร็จลุล่วงไปด้วยดี ท้ายนี้ผู้เขียนขอกราบขอบพระคุณบิดา มารดา ที่ให้การอุปการะอบรมเลี้ยงดู ตลอดจนส่งเสริมการศึกษา และให้กำลังใจเป็นอย่างดี อีกทั้งขอขอบคุณเพื่อน ๆ ที่ให้การสนับสนุนและช่วยเหลือด้วยดีเสมอมา และขอขอบพระคุณเจ้าของเอกสารและงานวิจัยทุกท่าน ที่ผู้ศึกษาค้นคว้าได้นำมาอ้างอิงในการทำวิจัย จนกระทั่งงานวิจัยฉบับนี้สำเร็จลุล่วงไปได้ด้วยดี

วรารินทร์ ปัญญาวงษ์

# สารบัญ

หน้า

# สารบัญตาราง

**หน้า**

# สารบัญภาพประกอบ

**หน้า**

# บทที่ 1
# บทนำ

## 1.1 หลักการและเหตุผล

แนวโน้มการใช้งานผ่านระบบอินเทอร์เน็ตในปัจจุบันนั้นได้มีโปรแกรมประเภทไม่หวังดีเพิ่มเป็นจำนวนมากขึ้น ซึ่งผู้ใช้งานทั่วไปอาจจะได้รับข้อมูล และโปรแกรมต่างๆ ที่มีจุดมุ่งหมายเพื่อที่จะทำลายหรือสร้างความเสียหายให้แก่ระบบคอมพิวเตอร์ ระบบเครือข่าย หรือ ทรัพย์สินและข้อมูลของผู้ใช้งาน โดยไม่รู้ตัว และโปรแกรมเหล่านี้ได้ฝังตัวลงในเครื่องคอมพิวเตอร์ของผู้ใช้งาน แล้วอาจจะเกิดอาการผิดปกติ ของเครื่องคอมพิวเตอร์ระหว่างการใช้งานอินเทอร์เน็ต เช่น การใช้งานที่ช้าลง มีการแสดงหน้าต่าง โฆษณาสินค้า หรือเว็บไซต์ลามกอนาจารแสดงอยู่ตลอดเวลา จนถึงเครื่องคอมพิวเตอร์ไม่สามารถใช้งานได้ตามปกติและต้องมีการเปิด-ปิด เครื่องคอมพิวเตอร์ใหม่อยู่บ่อยครั้งและอีกหลายกรณีเป็นต้น ซึ่งสาเหตุเหล่านี้อาจเกิดจากโปรแกรมประสงค์ร้ายประเภท มัลแวร์ (Malware) โดยทำงานในลักษณะที่เป็นไวรัส ทั้งประเภทเวิร์ม (Worm) หรือหนอน อินเทอร์เน็ต และพวกม้าโทรจัน (Trojan Horse) การแอพพลิเคชั่นดักจับในส่วนของข้อมูล (Spyware) และการดักจับข้อมูลในส่วนของข้อมูลคีย์บอร์ด (Key Logger) ของผู้ใช้งานเครื่องคอมพิวเตอร์ ตลอดจนโปรแกรมและแอพพลิเคชั่นประสงค์ร้ายประเภทใช้การขโมยข้อมูล (HTTP cookie) และทำการโจมตีด้วยโค้ดอันตราย Malicious Code Malicious Code เป็นโปรแกรมที่ถูกพัฒนาขึ้นเพื่อส่งให้เกิดผลลัพธ์ที่ไม่พึงประสงค์กับผู้ใช้งาน หรือระบบ เช่น ทำให้เกิดความขัดข้องหรือเสียหายกับระบบที่โปรแกรมนี้ติดตั้งอยู่ โดยปกติภัยคุกคามประเภทนี้ ต้องอาศัยการหลอกลวงให้ผู้ใช้งานเรียกใช้งานโปรแกรมก่อนจึงจะสามารถทำการ โจมตีได้ เช่น Virus, Trojan หรือ Spyware ต่างๆ หรือบางครั้งอาจทำการโจมตีได้ด้วยตนเอง เช่น Worm เป็นต้น ผ่านทาง Internet Browser ที่เกิดขึ้น โดยการโจมตีด้วยโค้ดอันตรายจะทำการควบคุมการทำงานโปรแกรม Internet Browser [1] ให้เป็นไปตามความต้องการของผู้ที่ไม่หวังดี เช่น การแสดงโฆษณาในลักษณะของ การ Pop-Up หน้าต่างโฆษณาออกมาเป็นระยะ เราเรียกโปรแกรมประสงค์ร้ายประเภทนี้ว่า แอดแวร์ (Adware) ซึ่งภัยเหล่านี้ในปัจจุบันได้เพิ่มขึ้นอย่างรวดเร็ว ซึ่งอาจจะเกิดผลกระทบแก่ผู้ใช้งานได้ถ้ารับโปรแกรมประสงค์ร้าย เหล่านี้ทำการโจมตีเครื่องคอมพิวเตอร์ ซึ่งนับวันอัตรายจากการถูกโจมตี จากมัลแวร์ได้เพิ่มสูงขึ้นอย่างรวดเร็วจึงจำเป็นที่จะต้องมีวิธีการในการป้องกันและตรวจจับเพื่อให้เครื่องคอมพิวเตอร์มีความมั่นคงต่อการโจมตีของมัลแวร์ต่างๆโดยวิธีการที่นิยมใช้ในการป้องกันคือการติดตั้ง โปรแกรมป้องกันไวรัสหรือโปรแกรมแอนตี้ไวรัส และอัตราการเพิ่มขึ้นทุกวันของมัลแวร์ประมาณ 390,000 ตัวอย่างของมัลแวร์ที่เป็นอันตรายจากเว็บไซต์ AV-TEST 1และปัจจุบันได้มีการศึกษาเรียนรู้เกี่ยวกับการ นำวิธีการเรียนรู้ด้วยคอมพิวเตอร์

มาประยุกต์ใช้ในการประมวลผลเพื่อตรวจจับมัลแวร์ซึ่งได้แก่การทำเหมืองข้อมูล (Data Mining) โดยวิธีการตรวจจับจากการจำแนกกลุ่ม (Classification)เพื่อระบุมัลแวร์ [1, 2] และตรวจสอบประเภทของมัลแวร์2 การทำเหมืองข้อมูลจะเกี่ยวข้องกับการวิเคราะห์ข้อมูลที่มี จำนวนมหาศาลและมีความซับซ้อนโดยส่วนใหญ่กระบวนการทำเหมืองข้อมูลนั้นจะมีข้อมูลที่ประกอบ ไปด้วยคุณลักษณะที่ไม่ตรงประเด็นและมีมิติของข้อมูลจำนวนมากซึ่งส่งผลให้การทำเหมืองข้อมูลต้องใช้ เวลาในการวิเคราะห์มากขึ้นถ้าข้อมูลมีมิติหรือตัวแปรมากจะทำให้ข้อมูลเกิดการกระจายและอาจไม่มี 2 ความสัมพันธ์กับมิติอื่น 3 ดังนั้นการคัดเลือกคุณลักษณะเพื่อลดมิติของข้อมูลจึงเป็นแนวทางหนึ่งที่ สามารถช่วยแก้ปัญหาดังกล่าว และยังสามารถช่วยให้การจำแนกข้อมูลได้แม่นยำมากขึ้น ดังนั้นเพื่อให้การตรวจจับมัลแวร์ประเภทแอดแวร์มีประสิทธิภาพและช่วยในการแก้ปัญหาการทำเหมืองข้อมูลที่มีจำนวนข้อมูลมหาศาลในงานวิจัยนี้ได้เปรียบเทียบอัลกอริทึม ซัพพอร์ตเวกเตอร์แมชชีน(Support Vector Machine) นาอีฟเบย์ (Naive Bayes) และเพื่อนบ้านใกล้สุด K ตัว (k-Nearest Neighbor)

**1.2 วัตถุประสงค์ของการวิจัย**

1. เพื่อสร้างแบบจำลองในการตรวจจับมัลแวร์โดยใช้เทคนิคเหมืองข้อมูล
2. เพื่อเปรียบเทียบประสิทธิภาพของอัลกอทึมในการจำแนกมัลแวร์โดยใช้เทคนิคเหมืองข้อมูล

**1.3 ความสำคัญของการวิจัย**

1. สามารถคัดเลือกมัลแวร์ที่มีคุณลักษณะพิเศษหรือมัลแวร์ที่ไม่มีคุณลักษณะที่เกี่ยวข้องช่วยให้การประมวลผลมีความถูกต้องและความเร็วเพิ่มขึ้น

2. ได้วิธีการคัดเลือกแบบจำลองเพื่อใช้ในการตรวจจับมัลแวร์และจำแนกคุณลักษณะของข้อมูลมัลแวร์ที่ทำให้ความถูกต้องของการตรวจจับ

**1.4 ขอบเขตของการวิจัย**

1.ข้อมูลมัลแวร์ที่ใช้นำมาทำการทดลองการวิจัย ซึ่งเป็นข้อมูลมัลแวร์ในส่วนของ แอดแวร์ สปายแวร์ ไวรัส มัลแวร์ และม้าโทรจัน ข้อมูลจากเว็บไซต์ Malwaredomainlist

2. ประยุตใช้อัลกอริทึมซัพพอร์ตเวกเตอร์แมชชีน(Support vector machines: SVM)นาอีฟเบย์ (Naïve Bayes) เพื่อนบ้านใกล้สุด K ตัว (k-Nearest Neighbor: k-NN)

3. เปรียบเทียบประสิทธิภาพของอัลกอทึมซัพพอร์ตเวกเตอร์แมชชีน(Support vector machines: SVM)นาอีฟเบย์ (Naïve Bayes) เพื่อนบ้านใกล้สุด K ตัว (k-Nearest Neighbor: k-NN)

**1.5 นิยามศัพท์เฉพาะ**

1. มัลแวร์ (Malware) โปรแกรมคอมพิวเตอร์ที่มีจุดประสงค์ร้ายต่อระบบคอมพิวเตอร์และเครือข่าย โดยการเข้ามาบุกรุกเครื่องคอมพิวเตอร์ โดยที่ไม่รู้ตัวและสร้างความเสียหายให้กับระบบคอมพิวเตอร์และเครือข่าย

2. การตรวจจับ (detecting) กระบวนการตรวจจับหาข้อผิดพลาดเป็นการเพิ่มคุณสมบัติการจำแนกและแก้ไขข้อผิดพลาดที่เกิดขึ้นอาทิเช่น ไวรัส (Virus) การแอบดักจับข้อมูล (Spyware) แอ็ดแวร์ (Adware) ซึ่งภัยเหล่านี้ในปัจจุบันได้เพิ่มขึ้นอย่างรวดเร็ว ซึ่งอาจจะเกิดผลกระทบแก่ผู้ใช้งานได้ถ้ารับโปรแกรมเหล่านี้เข้ามาในเครื่องคอมพิวเตอร์

# บทที่ 2
# ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

ในบทนี้เป็นการอธิบายถึงทฤษฎีและงานวิจัยที่เกี่ยวข้องกับการตรวจจับมัลแวร์ประเภท แอดแวร์ สปายแวร์ ไวรัส ม้าโทรจัน โดยเทคนิคเหมืองข้อมูล ซึ่งประกอบด้วย การทำเหมืองข้อมูล อัลกอริทึม ซัพพอร์ตเวกเตอร์แมชชีน นาอีฟเบย์ เพื่อนบ้านใกล้สุด K ตัว และงานวิจัยที่เกี่ยวข้อง

## 2.1 ทฤษฎีที่เกี่ยวข้อง

สำหรับซอฟต์แวร์ที่เป็นอันตรายประกอบด้วยการเขียนโปรแกรม4 ที่ออกแบบมาเพื่อทำลายหรือปฏิเสธการดำเนินการรวบรวมข้อมูลที่นำไปสู่ การสูญเสียความเป็นส่วนตัวหรือการแสวงหาผลประโยชน์ที่ได้รับการเข้าถึงทรัพยากรของระบบ, และ อื่น ๆ ที่ไม่เหมาะสม พฤติกรรมการแสดงออกเป็นคำทั่วไปที่ใช้โดยผู้เชี่ยวชาญด้านคอมพิวเตอร์หมายถึง ความหลากหลายของรูปแบบของซอฟต์แวร์ที่เป็นมิตรล่วงล้ำหรือรำคาญหรือรหัสโปรแกรมที่เป็น ซอฟต์แวร์ที่เป็นอันตราย (Malware) ย่อมาจากคำว่า Malicious Software ซึ่งหมายถึง โปรแกรมประสงค์ร้ายต่างๆ โดยทำงานในลักษณะที่เป็นการโจมตีระบบ การทำให้ระบบเสียหาย รวมไป ถึงการโจรกรรมข้อมูล มัลแวร์ แบ่งออกได้หลากหลายประเภท อาทิเช่น ไวรัส (Virus) เวิร์ม (Worm) หรือหนอนอินเทอร์เน็ต ม้าโทรจัน (Trojan Horse) การแอบดักจับข้อมูล (Spyware) คีย์ล๊อกเกอร์ (Key Logger) บนเครื่องคอมพิวเตอร์ของผู้ใช้งาน ตลอดจนโปรแกรมประเภทขโมยข้อมูล (Cookie) และการฝัง Malicious Mobile Code (MMC) ผ่านทางช่องโหว่ของโปรแกรม Internet Browser [1]โดย โปรแกรมจะทำการควบคุมการทำงานโปรแกรม Internet Browser ให้เป็นไปตามความต้องการของผู้ที่ ไม่หวังดี เช่น การแสดงโฆษณาในลักษณะของการ Pop-Up หน้าต่างโฆษณาออกมาเป็นระยะ เราเรียก โปรแกรมประเภทนี้ว่า แอดแวร์ (Adware) ซึ่งภัยเหล่านี้ในปัจจุบันได้เพิ่มขึ้นอย่างรวดเร็ว ซึ่งอาจจะเกิด ผลกระทบแก่ผู้ใช้งานได้ ถ้ารับโปรแกรมเหล่านี้เข้ามาในเครื่องคอมพิวเตอร์

**Malware มัลแวร์** [3] คำว่ามัลแวร์ (Malware) เป็นคำที่ย่อมาจาก Malicious Software ซึ่งหมายถึงซอฟต์แวร์ชนิดหนึ่งที่ถูกออกแบบมาด้วยเจตนาที่ประสงค์ร้ายต่อผู้ใช้ ยกตัวอย่างเช่น การแอบแฝงตนเองเข้ามาในเครื่องคอมพิวเตอร์ของผู้ใช้เพื่อขโมยข้อมูลส่วนตัวของผู้ใช้ หรือสร้างความเยหายให้กับโปรแกรมหรือข้อมูลของผู้ใช้คอมพิวเตอร์เป็นต้น โดยมัลแวร์สามารถแบ่งออกได้เป็น 3 ประเภทอันได้แก่ ไวรัส (Virus) หนอนคอมพิวเตอร์ (Computer worm) และม้าโทรจัน (Trojan horse)

**Virus ไวรัส** [4-6]เป็นโปรแกรมคอมพิวเตอร์ที่ถูกออกแบบมาให้บันทึกอยู่ในโฮสต์ไฟล์ (Host file) จำพวกไฟล์กระทำการ (Executable file) โดยไวรัสไม่สามารถอยู่อย่างโดดเดี่ยวได้ (Standalone) จะต้องอยู่บน โฮสต์ไฟล์เสมอ ไวรัสสามารถแพร่พันธ์ไปยังไฟล์เป้าหมายที่อยู่ในเครื่องคอมพิวเตอร์ที่ถูกติดตั้งไวรัสไปแล้วได้ด้วยตัวของมันเอง ซึ่งข้อจำกัดของไวรัสก็คือมันไม่สามารถแพร่พันธ์ไปนอกเครื่องคอมพิวเตอร์ของผู้ใช้ได้เว้นเสียแต่ว่าผู้ใช้ทำตัวเป็นพาหะเอง เช่นผู้ใช้ส่งอีเมลหรือส่งอุปกรณ์เก็บข้อมูลที่มีไวรัสให้กับผู้ใช้คอมพิวเตอร์คนอื่นๆ

ในการทำงานนั้นไวรัสจะทำการคัดลอกโค้ตของตัวมันเองไปไว้ในโฮสต์ไฟล์เป้าหมาย โดยเมื่อระบบปฏิบัติการหรือผู้ใช้ทำการเรียกใช้โปรแกรมที่ติดไวรัสนั้นก็จะเป็นการสั่งให้โค้ตของไวรัสที่แอบแฝงเอาไว้ในโปรแกรมดังกล่าวทำงานด้วย ซึ่งก็ขึ้นอยู่กับว่าไวรัสแต่ละตัวถูกออกแบบมาให้ทำอย่างไรกับเครื่องเป้าหมายบ้าง

**Adware แอดแวร์** [3] หมายถึงแพ็กเกจซอฟต์แวร์ใดๆ ที่สามารถทำงาน แสดง หรือดาวน์โหลดสื่อโฆษณาโดยอัตโนมัติ ไปยังคอมพิวเตอร์ที่ได้รับการติดตั้งซอฟต์แวร์ชนิดนี้ไว้ หรือขณะที่โปรแกรมประยุกต์กำลังเรียกใช้ ซอฟต์แวร์โฆษณาบางประเภทเป็นซอฟต์แวร์สอดแนม (spyware)

**Spyware** [3] เป็นโปรแกรมที่แฝงมาขณะเล่นอินเตอร์เน็ตโดยจะทำการติดตั้งลงไปในเครื่องของเรา และจะทำการเก็บพฤติกรรมการใช้งานอินเตอร์เน็ต รวมถึงข้อมูลส่วนตัวหลาย ๆ อย่าง สิ่งสำคัญต่าง ๆ เช่น Password หรือ หมายเลขบัตรเครดิตของเราด้วย นอกจากนี้อาจจะมีการสำรวจโปรแกรม และไฟล์ต่าง ๆ ในเครื่องและ Spyware นี้จะทำการส่งข้อมูลดังกล่าวไปในเครื่องปลายทางที่โปรแกรมได้ระบุเอาไว้ ดังนั้นข้อมูลต่าง ๆ ในเครื่องของท่านอาจไม่เป็นความลับอีกต่อไป

**Worms หนอนคอมพิวเตอร์** [12] ถูกสร้างขึ้นโดย Robert Morris, Jr. มันคือโปรแกรมที่จะสืบพันธุ์โดยการจำลองตนเองมากขึ้นเรื่อยๆ จากระบบหนึ่ง ครอบครองทรัพยากรและทำให้ระบบช้าลง และมีคุณลักษณะที่ค่อนข้างคล้ายคลึงกับไวรัสซึ่งมีผู้ใช้งานบางส่วนนั่งสับสนระหว่างหนอนคอมพิวเตอร์และไวรัสคอมพิวเตอร์อยู่บ้างกล่าวได้ว่าหนอนคอมพิวเตอร์นั้นมีคุณลักษณะการแพร่พันธ์ด้วยตนองได้คล้างคลึงกับไวรัสแต่ในการแพร่พันธ์ของหนอนคอมพิวเตอร์นั้นไม่จำเป็นต้องมีโฮสต์ไฟล์เหมือน หนอนคอมพิวเตอร์สามารถอยู่อย่างโดดเดี่ยว และสามารถแพร่พันธ์ไปยังเครื่องคอมพิวเตอร์เครื่องหนึ่งไปยังเครื่องหนึ่งในเครือข่ายหรือระบบอินเตอร์เน็ตได้ด้วยตนเองโดยสามารถแบ่งประเภทของหนอนคอวพิวเตอร์ตามการแพร่พันธ์ของมันได้

**Trojan Horses หรือม้าโทรจัน** [19] เป็นโปรแกรมที่ถูกออกแบบมาให้ทำหน้าที่เฉพาะทาง โดยม้าโทรจันจะแตกต่างกับไวรัสคอมพิวเตอร์และหนอนคอมพิวเตอร์ตรงที่มันไม่สามารถแพร่พันธ์ตนเองไปยังไฟล์แครื่องคอมพิวเตอร์เครื่องอื่นๆได้รูปแบบการทำงานของม้าโทรจันนั้นค่อนข้างที่จะชัดเจนกล่าวได้คือม้าโทรจันจะทำหน้าที่ขโมยข้อมูลส่วนตัวของผู้ใช้ เช่น บัญชีผู้ใช้และรหัสผ่านอีเมล แล้วส่งข้อมูลไปยังผู้เขียนโปรแกรมม้าโทรจัน ม้าโทรจันมักจะอยู่ตามเว็ยไซต์ที่อันตรายรอโอกาสที่ผู้ใช้งานคอมพิวเตอร์ที่ร็เท่าไม่ถึงการณ์ดาวน์โหลดไฟล์ม้าโทรจันมาเข้าในเครื่องตนเอง

Trojan: Win32/Fuery.B!cl ซึ่งเป็นประเภทของมัลแวร์ที่ออกแบบมาเพื่อที่จะให้การเข้าถึงระบบของผู้ใช้ มันไม่สามารถจำลองตนเองเช่นไวรัส แต่ก็สามารถนำไปสู่การแพร่กระจ่ายและถูกติดตั้งบนเครื่อง มันจะถูกกระจายทั่วไปผ่านแนบมากับอีเมล์, ,Facebook ,ดาวน์โหลดฟรีแวร์, การเข้าถึงของเว็บไซต์ไม่ได้รับอนุญาตหรือจากเว็บเบราเซอร์ นอกจากนี้ยังอาจได้รับการโอนเข้าสู่ระบบผ่านทางแฟลชไดรฟ์ USB หรืออุปกรณ์ภายนอกอื่น ๆ

Trojan: Win32/Fuery.B!cl ไม่สามารถตรวจพบได้อย่างง่ายดายด้วยตัวเอง แต่เมื่อติดเข้าสู่ระบบส่งผลให้ทำงานช้าลงเนื่องจากการประมวลผลหนักและการใช้งานเครือข่าย

**การจำแนกประเภทความเสียหายที่เกิดขึ้นจากมัลแวร์ (Classified Damages)**

**1. ความหมายของความเสียหาย**

ในงานวิจัยนี้จะนิยามความเสียหายที่เกิดขึ้นกับผู้ใช้ไม่ว่าโดยความเสียนั้นมาจากความต้องการของผู้สร้างมัลแวร์ หรือมาจากผลข้างเคียงของมัลแวร์ ถือเป็นความเสียหายต่อผู้ใช้งานคอมพิวเตอร์ทั้งสิ้น ยกตัวอย่างเช่นมัลแวร์บางตัวไม่ได้สร้างความเสียหายให้กับผู้ใช้งานโดยตรงแต่การที่มัลแวร์ได้อาศัยในเครื่องของผู้ใช้งานนั้น ก็จะต้องมีการใช้ทรัพยากรของเครื่องคอมพิวเตอร์ เช่น พื้นที่ฮาร์ดดิสก์ หน่วยความจำ และยังต้องทำให้หน่วยประมวลผลกลาง (CPU) ทำงานหนักขึ้นมากกว่าเดิม ซึ่งเป็นภาระที่ผู้ใช้ไม่ควรที่จะต้องแบกไว้ จึงนับได้ว่าการคงอยู่ของมัลแวร์ในเครื่องคอมพิวเตอร์นั้นก็ถือเป็นความเสียหายอย่างหนึ่ง โดยสามารถแบ่งความเสียหายหลักๆ ได้ดังนี้

**2. ประเภทของความเสียหาย**

- **ความเสียหายด้านข้อมูล (Data Damagd)**

เป็นความเสียหายที่เกิดขึ้นจากการที่ผู้ใช้ถูกขโมยข้อมูลส่วนตัวซึ่งอาจจะเป็นบัญชีผู้ใช้และรหัสผ่านของอีเมล หรือบัญชีผู้ใช้ไม่สามารถเรียกใช้งานไฟล์เอกสาร หรือไฟล์งานของผู้ใช้ได้ตามปกติ

- **ความเสียหายด้านข้อมูลส่วนตัว (Private Information Damaged)**

เป็นความเสียหายที่เกิดขึ้นจากการที่ผู้ใช้ถูกขโมยข้อมูลส่วนตัวซึ่งอาจจะเป็นบัญชีผู้ใช้และรหัสผ่านของอีเมล หรือบัญชีผู้ใช้และรหัสผ่านสำหรับทำธุรกรรมทางการเงินของผู้ใช้ ซึ่งสร้างความเสียหายให้กับผู้ใช้เมื่อถูกขโมยข้อมูลเหล่านี้ไป

- **ความเสียหายด้านโปรมแกรม (Program Damaged)**

เป็นความเสียหายที่มัลแวร์ได้ทำการแก้ไข หรือลบไฟล์โปรแกรมบนเครื่องคอมพิวเตอร์ของผู้ใช้ ซึ่งอาจทำให้โปรแกรมต่างๆ ไม่สามารถทำงานได้ตามปกติ

- **ความเสียหายด้านทรัพยากรเครือข่าย (Network Resource Damaged)**

เป็นความเสียหายที่เกิดจากการใช้ทรัพนากรในเครือข่ายซึ่งอาจจะเป็นความจงใจของมัลแวร์เองในการโจมตีระบบเครือข่าย หรือเพื่อสนับสนุนการทำงานของมัลแวร์ ซึ่งมีทั้งการแพร่กระจายตนเองผ่านระบบเครือข่าย รวมถึงการโจมตีแบบ DoS (Denial of Services) ซึ่งจะต้องทำให้สิ้นเปลืองแบนด์วิธ (Bandwidth) จำนวนหนึ่งไปกับการทำงานของมัลแวร์

- **ความเสียหายด้านไฟล์ของระบบ (System File Damaged)**

เป็นความเสียหายที่เกิดขึ้นกับไฟล์ของระบบปฏิบัติการในเครื่องคอมพิวเตอร์ซึ่งมัลแวร์ได้ทำการลบ หรือแก้ไขไฟล์ของระบบ ซึ่งส่งผลให้ระบบไม่สามารถทำงานได้หรืออาจแก้ไขไฟล์ดังกล่าวนั้นเพื่ออำนวยความสะดวกให้กับการทำงานของตัวมัลแวร์เอง

- **ความเสียหายด้านความปลอดภัย (Security Damaged)**

เป็นความเสียหายที่เกิดจากเจตนาของมัลแวร์ที่จะทำให้เครื่องของผู้ใช้มีความเสี่ยงด้านความปลอดภัย ยกตัวอย่างเช่น มัลแวร์ทำการลบไฟล์สำคัญของโปรแกรมกำจัดมัลแวร์ที่ออกไปจากเครื่องของผู้ใช้เพื่อไม่ให้ผู้ใช้สามารถตรวจพบมัลแวร์ได้ หรือมัลแวร์ทำการปรับเปลี่ยนแก้ไขค่าบางอย่างในไฟล์โปรแกรมนั้นๆ อนุญาตให้มัลแวร์ดังกล่าวทำงานได้ เป็นต้น

- **ความเสียหายด้านอินเตอร์เฟส (Interface Damaged)**

เป็นความเสียหายที่เกิดขึ้นกับผู้ใช้โดยผ่านอุปกรณ์อินเตอร์เฟสต่างๆ เช่น จอแสดงผล เมาท์ คีย์บอร์ด เป็นต้น ยกตัวอย่างเช่น มัลแวร์บางตัวจะทำการเปลี่ยนรูปแบบอักษรบนจอแสดงผล หรือทำให้คีย์บอร์ดไม่สามารถพิมพ์ได้ตามปกติ

**การทำเหมืองข้อมูล (Data Mining)** [16] คือกระบวนการที่กระทำกับข้อมูลจำนวนมากเพื่อค้นหารูปแบบและความสัมพันธ์ ที่ซ่อนอยู่ในชุดข้อมูลนั้น ในปัจจุบันการทำเหมืองข้อมูลได้ถูกนำไป ประยุกต์ใช้ในงานหลายประเภท ทั้งในด้านธุรกิจที่ช่วยในการตัดสินใจของผู้บริหาร ในด้านวิทยาศาสตร์ และการแพทย์รวมทั้งในด้านเศรษฐกิจและสังคมการทำเหมืองข้อมูลเปรียบเสมือนวิวัฒนาการหนึ่งใน การจัดเก็บและตีความหมาย ข้อมูล จากเดิมที่มีการจัดเก็บข้อมูลอย่างง่ายๆ มาสู่

การจัดเก็บในรูป ฐานข้อมูลที่สามารถดึงข้อมูลสารสนเทศมาใช้จนถึงการทำ เหมืองข้อมูลที่สามารถค้นพบความรู้ที่ซ่อนการทำเหมืองขอมูลได้ถูกนำไปประยุกต์ใช้ในงานหลายประเภท ทั้งในด้านธุรกิจที่ช่วยในการตัดสินใจของผู้บริหาร ในด้านวิทยาศาสตร์และการแพทย์รวมทั้งในด้านเศรษฐกิจและสังคม การทำเหมืองข้อมูลเปรียบเสมือนวิวัฒนาการหนึ่งในการจัดเก็บและตีความหมาย ข้อมูลจากเดิมที่มีการจัดเก็บข้อมูลอย่างง่ายๆ มาสู่การจัดเก็บในรูปฐานข้อมูลที่สามารถดึงข้อมูลสารสนเทศมาใช้จนถึงการทำ เหมืองข้อมูลที่สามารถค้นพบความรู้ที่ซ่อนอยู่ในข้อมูล [16]

**วิวัฒนาการของการทำเหมืองข้อมูล**

- ปี 1960 Data Collection คือ การนำข้อมูลมาจัดเก็บอย่างเหมาะสมในอุปกรณ์ที่น่าเชื่อถือและป้องกันการสูญ หายได้เป็นอย่างดี
- ปี 1980 Data Access คือ การนำข้อมูลที่จัดเก็บมาสร้างความสัมพันธ์ต่อกันในข้อมูลเพื่อประโยชน์ใน การนำไปวิเคราะห์ และการตัดสินใจอย่างมีคุณภาพ
- ปี 1990 Data Warehouse & Decision Support คือ การรวบรวมข้อมูลมาจัดเก็บลงไปในฐานข้อมูลขนาดใหญ่โดยครอบคลุมทุกด้านของ องค์กร เพื่อช่วยสนับสนุนการตัดสินใจ
- ปี 2000 Data Mining คือ การนำข้อมูลจากฐานข้อมูลมาวิเคราะห์และประมวลผล โดยการสร้างแบบจำลองและความสัมพันธ์ทางสถิติ

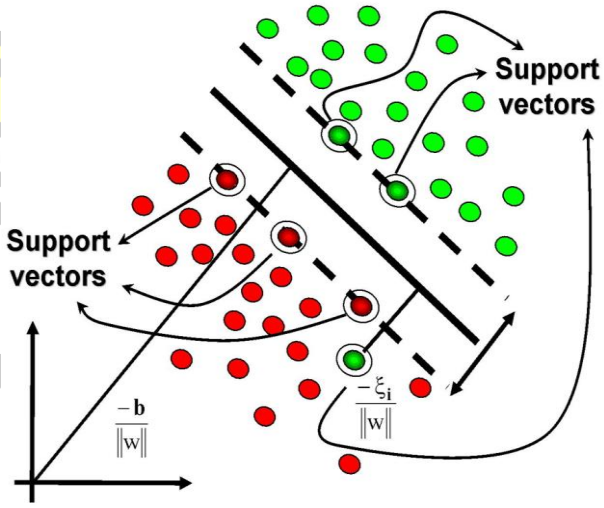
**ขั้นตอนการทำเหมืองข้อมูล**

ประกอบด้วยขั้นตอนการทำงานย่อยที่จะเปลี่ยนข้อมูลดิบให้กลายเป็นความรู้ ประกอบด้วยขั้นตอนดังนี้

- Data Cleaning เป็นขั้นตอนสำหรับการคัดข้อมูลที่ไม่เกี่ยวข้องออกไป
- Data Integration เป็นขั้นตอนการรวมข้อมูลที่มีหลายแหล่งให้เป็นข้อมูลชุดเดียวกัน
- Data Selection เป็นขั้นตอนการดึงข้อมูลสำหรับการวิเคราะห์จากแหล่งที่บันทึกไว้
- Data Transformation เป็นขั้นตอนการแปลงข้อมูลให้เหมาะสมสำหรับการใช้งาน
- Data Mining เป็นขั้นตอนการค้นหารูปแบบที่เป็นประโยชน์จากข้อมูลที่มีอยู่
- Pattern Evaluation เป็นขั้นตอนการประเมินรูปแบบที่ได้จากการทำเหมืองข้อมูล
- Knowledge Representation เป็นขั้นตอนการนำเสนอความรู้ที่ค้นพบ โดยใช้เทคนิคในการนำเสนอเพื่อให้เข้าใจ

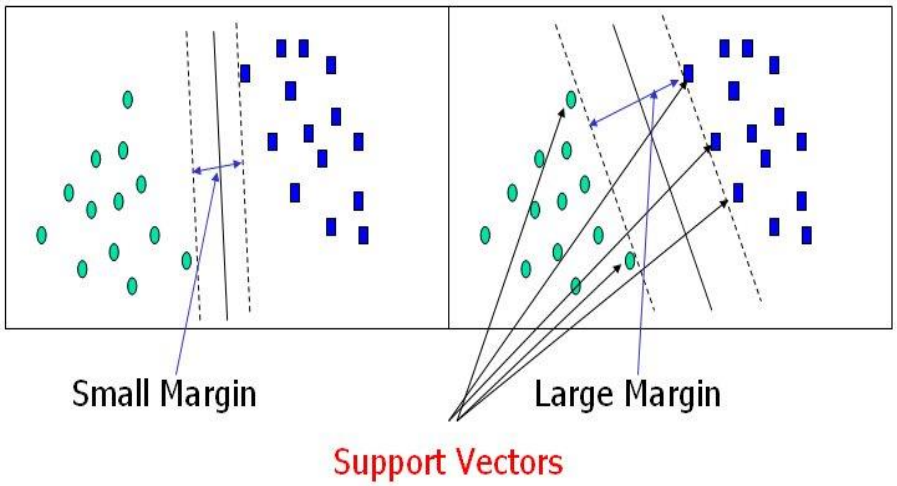
**อัลกอริทึม ซัพพอร์ตเวกเตอร์แมชชีน (Support Vector Machine)** [16] คือ กระบวนการสอนเครื่องแบบมีผู้สอน (supervised learning) เพื่อให้สามารถสร้างตัวจำแนกข้อมูล (classifier) ที่มีความทั่วไป (generalize) สูง นั้นคือสามารถทำงานได้ดีกับตัวอย่างที่ไม่รู้จัก (unknown database) ด้วยกระบวนการปรับรูปแบบ ข้อมูลจากข้อมูลที่มีมิติต่ำ (low dimension dataset) บนพื้นที่ข้อมูล นำเข้า (input space) ให้อยู่ใน รูปแบบของข้อมูลที่มีมิติสูง (high dimension dataset) บนพื้นที่ข้อมูลคุณลักษณะ (feature space) โดยใช้ฟังก์ชันในการปรับรูปแบบข้อมูลเรียกว่าฟังก์ชัน เคอร์เนล (kernel function) ซึ่งความสามารถ ดังกล่าวช่วยให้การสร้างตัวจำแนกข้อมูลด้วยสมการกาลังสอง (quadratic equation) บนพื้นที่ข้อมูล [เพิ่มความถูกต้องของตัวแบบ ซัพพอร์ตเวกเตอร์แมทชีนแบบค่ากำลังสองน้อยที่สุด ด้วยขั้นตอนวิธีการค้นหาแบบนกดุเหว่า Accuracy Improvement of Least Squares Support Vector Machinesusing Cuckoo Search Algorithm

SVM เป็นอัลกอริทึมในการคัดแยกที่มีการนำมาใช้กันอย่างกว้างขวางในด้านการประมวลผลเป็นภาพดิจิตอล หลักการของ SVM คือการให้อินพุทที่ใช้ฝึกเป็นเวคเตอร์ในสเปซ N มิติ เช่นถ้าในกรณีของ 2 มิติ และ 3 มิติ จะเป็นจุดที่อยู่ในระนาบ xy และสเปซ xyz ตามลำดับ จากนั้นทำการสร้างไฮเปอร์เพลน (Hyperplane) ที่จะแยกกลุ่มของเวคเตอร์อินพุทออกเป็นประเภทต่างๆ ในกรณีที่เป็น 2 มิติ และ 3 มิติ ไฮเปอร์เพลนจะ คือเส้นตรงและระนาบตามลำดับ ข้อเด่นของ SVM จะทำการเก็บแมพ (Map) เวคเตอร์ในสเปซอินพุทให้เข้าสู่ Feature Space โดยใช้ฟังก์ชั่นหรือเรียกว่าเคอร์นัล (kernel) ชนิดต่างๆ เช่น โพลีโนเมียล (Polynomial) เรเดียล (Radial) เป็นต้น ใน Feature Space ดังกล่าเวคเตอร์อินพุท สามารถแยกประเภทได้โดยไฮเปอร์เพลน ดังตัวอย่างภาพประกอบที่ 1



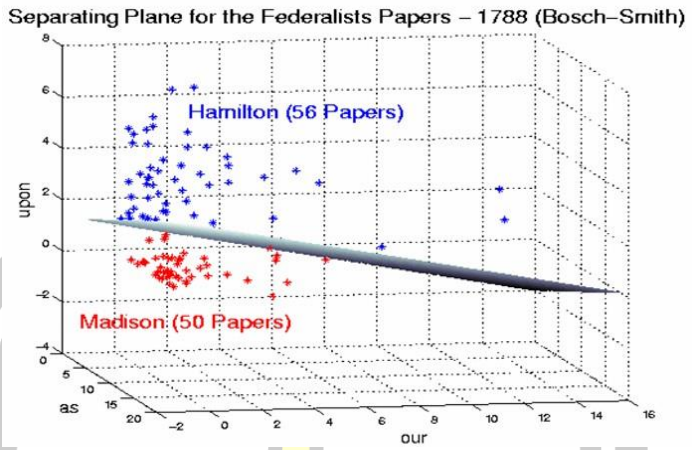

ภาพประกอบที่ 1 ตัวอย่าง SVM ใน 2 มิติ

เครือข่ายปัญญาประดิษฐ์ กล่าวคือ SVM ที่ใช้ฟังก์ชั่นซิกมอยด์ในการแมพ เทียบเท่ากับเครือข่ายปัญญาประดิษฐ์แบบ Feedforward ที่มี 2 ชั้น มีข้อแตกต่างจากเครือข่ายปัญญาประดิษฐ์ก็คือ การแก้สมาการหาค่าน้ำหนักใช้ในการแก้สมาการ Quadratic ที่มีข้อบังคับเชิงเส้น (Linear Constrained) แทนที่จะเป็นการหาค่าต่ำสุด (minimization) อย่างในกรณีของเครือข่ายปัญญาประดิษฐ์ ดังตัวอย่างภาพประกอบที่ 2




ภาพประกอบที่ 2 ตัวอย่าง SVM ใน 2 มิติ

สมมติว่าเราต้องการคัดแยกอินพุทออกเป็น 2 กลุ่ม โดยใช้ไฮเปอร์เพลน ที่เป็นเส้นตรง จะเห็นว่ามีเส้นตรงจำนวนมากที่สามารถคัดแยกได้ แต่เส้นตรงเส้นไหนที่ดีที่สุด (Optimal Line) รูปที่ 2.16 แสดงตัวอย่างของ 2 เส้นตรง เราจะนิยาม Margin เป็นผลรวมระยะห่างของเส้นตรงที่เป็นไฮเปอร์เพลน (เส้นทึบในรูปที่ 2) ถึงเส้นตรงที่ผ่านอินพุทที่ ใกล้ที่สุดและขนานกับไฮเปอร์เพลน ของทั้งสองกลุ่ม (เส้นทึบในรูปที่ 2) ระยะดังกล่าวอาจมองเป็นเวคเตอร์และมีชื่อว่า ซัพพอร์ตเวกเตอร์แมชชีน (Support Vector Machine) อัลกอริทึม SVM จะเลือกไฮเปอร์เพลนที่ให้ค่า Margin มีค่าสูงสุดดังแสดงในรูปที่ 2 กรณีของ 3 มิติ จะเป็นทำนองเดียวกัน อัลกอริทึม SVM ใน 3 มิติดังตัวอย่างภาพประกอบที่ 3




ภาพประกอบที่ 3 ตัวอย่าง SVM ใน 3 มิติ

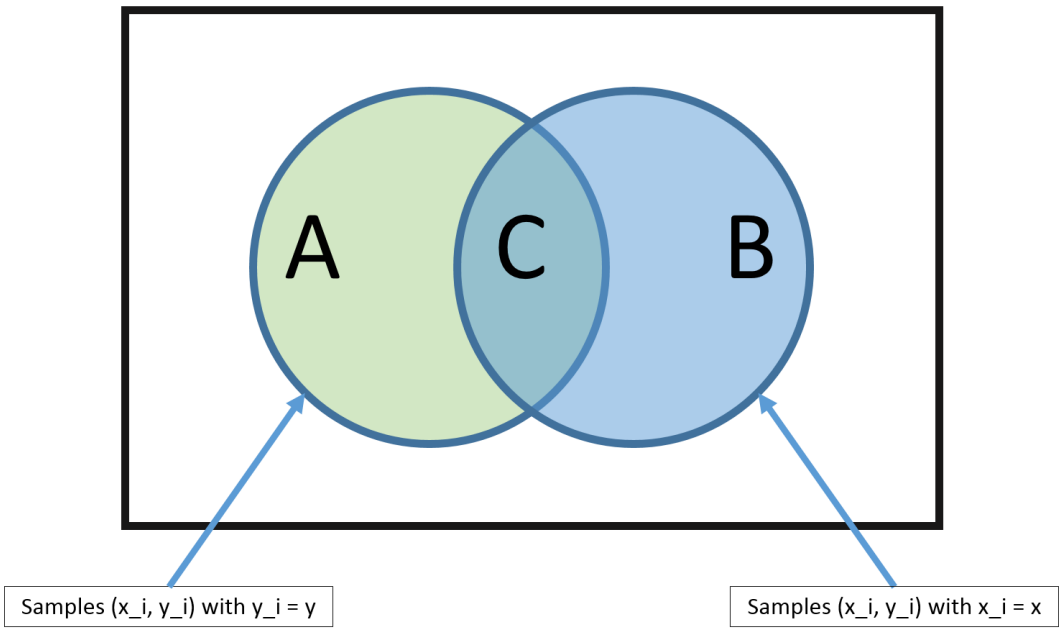
**อัลกอริทึม นาอีฟเบย์ (Naive Bayes)** [13] เป็นขั้นตอนวิธีที่ได้รับความนิยมและถูกนำมาใช้อย่างแพร่หลายในงานจำแนกหมวดหมู่เอกสาร เนื่องจากความเรียบง่ายของขั้นตอนวิธีและให้ประสิทธิภาพการจำแนกที่ดี นาอีฟเบย์สเป็นขั้นตอนวิธีที่มีพื้นฐานมาจากทฤษฎีเบย์ส (Bayes' Theorem) ซึ่งอาศัยหลักความน่าจะเป็นในการทำนายผลลัพธ์ โดยการวิเคราะห์ความสัมพันธ์ระหว่างตัวแปรเพื่อใช้ในการสร้างเงื่อนไขความน่าจะเป็นสำหรับแต่ละความสัมพันธ์ ใช้การคำนวณความน่าจะเป็นซึ่งถูกใช้ในการทำนายผล จัดเป็นเทคนิคในการแก้ปัญหาแบบ classification ที่สามารถคาดการณ์ ผลลัพธ์ ได้และสามารถอธิบายได้ด้วย มันจะทำการวิเคราะห์ความสัมพันธ์ระหว่างตัวแปรเพื่อใช้ในการสร้างเงื่อนไขความน่าจะเป็นสำหรับแต่ละความสัมพันธ์ การเรียนรู้อย่างงายเป็นวิธีจำแนกประเภทข้อมูลที่มีประสิทธิภาพวิธีหนึ่งที่การทำงานที่ไม่ซับซอน เหมาะกับกรณีของเซตตัวอย่างมีจำนวนมากและคุณสมบัติ (Attribute) ของตัวอย่างไมขึ้นตอกัน โดยกำหนดให้ความนาจะเป็นของข้อมูลที่จะเป็น [16] ดังสมการ

$$P(a_1, a_2, \dots, a_n|v_j) = \prod_{i=1}^{n} p\,(a_i|v_j)$$

กลุ่ม $v_j$ สำหรับข้อมูลที่มีคุณสมบัติ n ตัว $x = \{a_1, a_2, \dots a_n\}$ หรือใช้ลักษณ์ว่า $P(a_1,\ a_2, \dots a_n|v_j)$ โดยที่ $\prod$ หมายถึง ผลคูณของค่า $P\,(a_i|v_j)$ ทั้งหมด $i = 1,2,3 \dots, n$ และ $j = 1, 2, 3, \dots, n$ ดังนั้นเราจะได้วิธีการจำแนกประเภทแบบนาอีเบย์อย่างง่าย ดังสมการ

$$v_{\mathrm{NB}} = \underset{v,\in V}{argmax}\, P(v_j) \times \prod_{i=1}^{n} P(a_i|v_j)$$

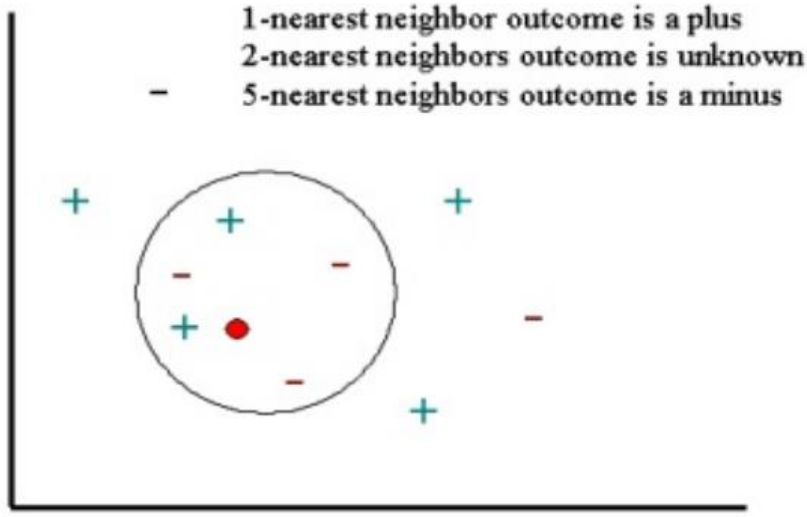
ดังตัวอย่างภาพประกอบที่ 4




ภาพประกอบที่ 4 นาอีฟเบย์

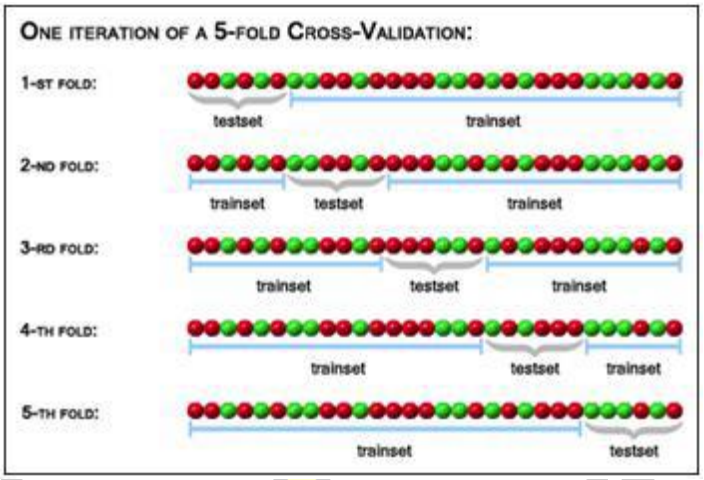
**อัลกอริทึม เพื่อนบ้านใกล้สุด K ตัว (k-Nearest Neighbor)** [13] คือ วิธีการในการจัดแบ่งคลาส เทคนิคนี้จะตัดสินใจ ว่าคลาสใดที่จะแทนเงื่อนไขหรือกรณีใหม่ๆ ได้บ้าง โดยการตรวจสอบจำนวนบางจำนวน (“K” ใน K-nearest neighbor) ของกรณีหรือเงื่อนไขที่เหมือนกันหรือใกล้เคียงกันมากที่สุด โดยจะหาผลรวม (Count Up) ของจำนวนเงื่อนไข หรือกรณีต่างๆสำหรับแต่ละคลาส และกำหนดเงื่อนไขใหม่ๆ ให้คลาสที่เหมือนกันกับคลาสที่ใกล้เคียงกันมากที่สุด จะจำแนก ประเภทข้อมูลโดยขึ้นกับข้อมูลที่มีคุณสมบัติใกล้เคียงที่สุด K ตัวจากข้อมูลบนชุดข้อมูลตัวอย่างทำงานโดยขึ้นกับระยะทางน้อยสุดจากสมาชิกใหม่ หรือข้อมูลที่ป้อนถาม (input query instance) กับข้อมูลตัวอย่างฝึกฝน จะคำนวณหาเพื่อนบ้านที่ใกล้ที่สุด K ตัว หลังจากนั้นรวบรวม สมาชิกที่ใกล้เคียงที่สุด K ตัวแล้วเลือกคลาสที่ สมาชิกสวนใหญ่ ที่ในกลุ่ม K ดังกล่าวสังกัดอยู่มาก ที่สุดให้กับสมาชิกใหม่ ข้อมูลการจำแนกโดยใช้ข้อมูลข้างเคียง K ตัว ประกอบด้วยแอททริบิวต์ หลายตัวแปร Xi ซึ่งจะนำมาใช้ในการแบ่งกลุ่ม Yi โดยระบุค่าตัวเลขจำนวนเต็มบวกให้กับ K ซึ่งค่า นี้จะเป็นตัวบอกจำนวนของกรณี (case) ที่จะต้องค้นหาในการทำนายกรณีใหม่อัลกอริทึมแบบ KNN ได้แก่ 1-NN , 2-NN ,

3-NN , ..... K-NN ตัวอย่าง 2-KNN หมายถึงอัลกอริทึมนี้ จะค้นหา 2 กรณีที่มีลักษณะใกล้เคียงกับกรณีใหม่ ( 2 Nearest Cases ) การนำระยะทางที่หาได้จาก สมาชิกใน ข้อมูลตัวอย่างฝึกฝน มาเรียงลำดับจากน้อยไปหามากแล้วเลือกสมาชิกที่มีระยะทาง (Distance) ใกล้เคียงที่สุดออกมา K ตัว โดยใช้การวัดระยะทางแบบ Euclidean distance มี หลักการคือ การวัดระยะทางระหว่างสองวัตถุ ถ้าวัตถุห่างกันมากแสดงว่าวัตถุนั้นมีความคล้ายกัน น้อย ถ้ามีค่าน้อยก็แสดงว่ามีความคล้ายคลึงกันมาก [13] ดังตัวอย่างภาพประกอบที่ 5




ภาพประกอบที่ 5 K-Nearest

**k-fold cross-validation** วิธีการวิเคราะห์ความแม่นตรงของโมเดล(models) k-fold cross-validation การตรวจสอบ ไขว้กัน (Cross-Validation) [22] เป็นวิธีการตรวจสอบค่าความผิดพลาดในการคาดการของโมเดล โดย พื้นฐานวิธีการ ตรวจสอบการไขว้กัน k-Fold Cross-validation เป็นวิธีการที่แบ่งข้อมูลออกเป็นกลุ่ม จำนวน k กลุ่ม (k-Fold) ในตอนแรกเลือกข้อมูลกลุ่มที่ 1 เป็นข้อมูลชุดทดสอบ และข้อมูลชุดที่เหลือจะเป็นข้อมูล ชุดสอนนำข้อมูลไปจัดหมวดหมู่ จากนั้นจะสับข้อมูล กลุ่มที่ 2 มาเป็นชุดทดสอบและข้อมูลกลุ่มอื่นๆที่ เหลือเป็นชุดทดสอบ สลับอย่างนี้ ไปเรื่อยๆจนครบ k กลุ่ม ในขั้นตอนสุดท้ายจะหาค่าเฉลี่ยของค่าความ ถูกต้องในแต่ละกลุ่ม วิธีการนี้ข้อมูลทุกตัวอย่างจะได้เป็นทั้งชุดทดสอบ และชุดสอน เช่น K - Fold Cross Validation ( K = 10) ชุดข้อมูลหลังจากทำการแบ่งออกเป็น 10 ชุด ข้อมูล ย่อยเท่าๆกัน โดยแต่ละกล่องคือ ชุดข้อมูลย่อย 1 ชุด ดังตัวอย่างภาพประกอบที่ 6




ภาพประกอบที่ 6 K - fold Cross Validation ( K = 5)

จากวิธีการข้างต้นนั้น ดัง ภาพประกอบที่ 6 จะได้ค่าความผิดพลาดของแต่ละรอบการคำนวณซึ่ง ประกอบด้วย e1, e2, e3, e4 และ e5 โดยปรกติแล้วนั้น การหาค่าเฉลี่ยความผิดพลาดและใช้ค่านั้น เป็นตัวแทน ของความผิดพลาดของโมเดลหรือวิธีการที่นำเสนอ ซึ่ง สามารถแสดงได้ดังสมการต่อไปนี้ Average Error = ( e1 + e2 + .... + eK )/ K จากตัวอย่างในข้างต้นนั้น ข้อดีของวิธีการนี้คือข้อมูลในแต่ละ ชุดที่ทำการแบ่งจะถูกทดสอบอย่างน้อย 1 ครัง้ และถูกเรียนรู้ทัง้ หมด K - 1 ครัง้ โดยในขัน้ ตอน เหล่านี้เราสามารถกำหนดได้ว่าต้องกาขนาดข้อมูลขนาดใด และต้องการทำการคำนวณเป็น จำนวนรอบเท่าใด แต่อย่างไรก็ตามเมื่อมองในมุมกลับกันวิธีการนี้ใช้เวลาในการคำนวณเป็น K เท่า ซึ่ง ในความเห็นส่วนตัวเวลานั้น ไม่เป็นปัจจัยสำคัญต่อการวัดผลเมื่อเทียบกับกับความถูกต้อง ของการวัดผล

## 2.2 งานวิจัยที่เกี่ยวข้อง

Marcus A. Maloof [6] ได้นำเสนอวิธีการสกัดคุณลักษณะข้อมูลด้วยเทคนิค เอ็นแกรม (N-Grams) ในรูปแบบของลำดับไบต์ (Byte Sequence) ซึ่งข้อมูลที่ใช้ในการทดสอบ จะประกอบด้วย Backdoor, Virus และ Mass Mailer โดยมีการเลือกใช้วิธีการจำแนก (Classifier) ที่ แตกต่างกันทั้งหมด7 แบบ ได้แก่ Boosted SVM, IBK, SVM, Boosted Naïve Bayes, Boosted J48, J48และ Naïve Bayesจากผลการทดสอบจะได้วิธีการ Boosted J48 ที่ให้ผลลัพธ์ค่าความถูกต้องดี ที่สุดโดยได้ ค่าเฉลี่ยของพื้นที่ใต้โค้งอยู่ที่ 98%

รองศาสตราจารย์ ดร.สมชาย ปราการเจริญ [7] ได้นำเสนอการวิเคราะห์รูปแบบ ข้อมูลสำหรับสร้างกฎในการตรวจจับ มัลแวร์โดยใช้ต้นไม้ตัดสินใจ (Decision Tree)และกฎความสัมพันธ์

(Association Rule)จึงทำให้ได้กฎที่เกิดจากรูปแบบข้อมูลที่เกิดขึ้นร่วมกัน (Common Pattern) และรูปแบบข้อมูลเฉพาะ (SpecialPattern) ที่เกิดขึ้นเฉพาะใน มัลแวร์นั้น ซึ่งรูปแบบข้อมูล จะแสดงคุณลักษณะด้วยข้อมูลแบบเอ็นแกรม (N-Gram)ข้อมูลที่ใช้ในการวิจัย เป็นข้อมูลมัลแวร์ 15 ประเภท ม้าโทรจันที่เกิดขึ้นบนระบบปฏิบัติการ Windows รุ่น 32 บิต จำนวน 90 ไฟล์ ที่ประกอบด้วย มัลแวร์ 3 ตระกูล คือ Trojan.AntiAV, Trojan.BHO และ Trojan.Dialer และผลลัพธ์จากการทดสอบ กฎในการตรวจ จับมัลแวร์ มีค่าความถูกต้องที่ 96.67 เปอร์เซ็นต์

KonradRieck, Thorsten Holz, CarstenWillems, Patrick Dusselและ PavelLaskov [8] ได้นำเสนอการจำแนกกลุ่มประเภทของข้อมูลมัลแวร์ ซึ่งได้แก่ Worm, Backdoor และ Trojan horse โดยจะวิเคราะห์คุณลักษณะจากพฤติกรรมการทำงาน ซึ่งจะเป็นชุดคำสั่ง API function call แล้วนำข้อมูลที่ได้มาจำแนกประเภทของมัลแวร์ด้วย Support Vector Machine จาก การนำคุณลักษณะที่ได้มาวิเคราะห์และจำแนกข้อมูล ซึ่งได้ผลลัพธ์ความถูกต้องแม่นยำอยู่ที่ 70% แต่การสกัด คุณลักษณะข้อมูลจำเป็นต้องใช้ระยะเวลา เนื่องจากชุดข้อมูล API function call ค่อนข้างมีความ ซับซ้อน

Ahmad Azab, Robert Layton, MamounAlazabและ Jonathan Oliver [9] ได้ กล่าวไว้ว่าอาชญากรรมยังคงเป็นความท้าทายและเป็นที่กำลังเติบโตมัลแวร์เป็นหนึ่งในภัยคุกคามความปลอดภัยที่ร้ายแรงที่สุดบนโลกอินเทอร์เน็ตในทุกวันโดยใช้ขั้นตอนวิธีไบนารีแบ่งกลุ่มที่อยู่ในตัวแปรเดียวกันเข้าด้วยกันโดยใช้อัลกอริทึม K-NN สองสายพันธุ์ซุสได้รับการทดสอบและ TSPY_ZBOT MAL_ZBOT ผลของเราแสดงให้เห็นว่า TLSH และ SDHASH ให้ผลลัพธ์ที่ถูกต้องสูงสุดในการให้คะแนน F-ตัวชี้วัดของ 0.989 และ 0.999 ตามลำดับ

Igor Santos, Yoseba K. Penya, Jaime Devesaและ Pablo G. Bringas [10] ได้นำเทคนิควิธีการ N-Gram มาสร้างข้อมูลซิกเนเจอร์ เพื่อใช้ในการตรวจจับมัลแวร์ที่ไม่รู้จัก โดยจะใช้K-Nearest Neighbour Algorithm ในการจำแนกข้อมูลว่าไฟล์ที่ไม่รู้จักนั้นเป็น Malware หรือBenign จากการทดสอบประสิทธิภาพแสดงให้เห็นว่าวิธีการนี้มีความแม่นยำในการจำแนกข้อมูลเนื่องจากได้ค่า False Positive Ratio เป็น 0% และยังสามารถตรวจจับมัลแวร์ได้ดี โดยมีค่าDetection Ratio เป็น 74.37%

ฉัตรชัย เลี้ยงบุญประคอง [11]ใช้สกัดคุณลักษณะข้อมูลด้วยวิธีการ N-grams แล้วนำคุณลักษณะข้อมูลที่ได้มาลดจำนวนข้อมูลด้วยวิธีการ Sequential Floating Forward Selection หลังจากนั้นจะจำแนกประเภทของมัลแวร์แฟมิลี่ด้วยเทคนิค C4.5, Multilayer Perceptron และ Support Vector Machine จากการจำแนกประเภทของมัลแวร์แฟมิลี่ ผลลัพธ์ที่ได้จะมีประสิทธิภาพใน การจำแนกข้อมูลที่มีความถูกต้องแม่นยำสูง โดยมีค่าความถูกต้อง (Accurracy) เป็น 96.64% ซึ่งได้จาก การจำแนกข้อมูลด้วยเทคนิค Support Vector Machine

N. R. Rosyid, M. Ohrui, H. Kikuchi, P. Sooraksa and M. Terada [5] “ A discovery of sequential attack patterns of malware in botnets,” IEEE International Conference on System Man and Cybermetics(SMC)งานวิจัยเรื่องนี้มีเป้าหมายในการค้นหารูปแบบของการโจมตีแบบใหม่ๆ ของมัลแวร์ ซึ่งไม่ใช่เรื่องง่ายเพราะเลือกข้อมูลมีเป็นจำนวนมาก โดยการแก้ปัญหาของงานวิจัยนี้จะใช้ วิธีการ PrefixSpanเพื่อวิเคราะห์รูปแบบการโจมตีของมัลแวร์และใช้ข้อมูล CCC Dataset ปี ค.ศ.2009

Kotsiantis และคณะ [12]ได้เสนองานวิจัยที่เปรียบเทียบประสิทธิภาพของอัลกอริธึมเพื่อพยากรณ์ประสิทธิภาพของนักศึกษาใน ระบบการศึกษาทางไกลด้วยอัลกอริธึม C4.5 อัลกอริธึม Naïve Bayes อัลกอริธึม Ripper และ อัลกอริธึม k-Nearest Neighbor ผลการวิจัยแสดงให้เห็นว่าอัลกอริธึม Naïve Bayes ให้ค่าประสิทธิภาพ 74.70% สูงกว่าอัลกอริธึมอื่นๆ

Cheewaprakobkit [13]ได้เสนองานวิจัยที่ใช้เทคนิคต้นไม้ตัดสินใจด้วยอัลกอริธึม C4.5 สำหรับเปรียบเทียบประสิทธิภาพกับ โครงข่ายประสาทเทียมเพื่อแยกประเภทของปัจจัยที่ส่งผลต่อผลสัมฤทธิ์ทางการเรียนของนักศึกษาระดับปริญญาตรีในหลักสูตร นานาชาติ จากงานวิจัยแสดงให้เห็นว่า C4.5 ให้ความแม่นยำในการพยากรณ์ที่ 85.13% มากกว่าโครงข่ายประสาทเทียมที่ได้ผลเป็น 83.87%

Pansumret และคณะ [14]ได้เสนองานวิจัยที่เปรียบเทียบอัลกอริธึม C4.5 อัลกอริธึม Naïve Bayes และ อัลกอริธึม k-Nearest Neighbor สำหรับวิเคราะห์ปจจัยที่ส่งผลต่อระดับผลการเรียนของนักศึกษาโดยงานวิจัยแสดงให้เห็นว่าอัลกอริธึม C4.5 ให้ค่า ประสิทธิภาพ 73.55% สูงกว่าอัลกอริธึมอื่นๆ

Muntham และ Ingsrisawang [15]ได้เสนองานวิจัยที่ใช้ต้นไม้ตัดสินใจด้วยอัลกอริธึม C4.5 เพื่อวินิจฉัยโรคระบบการหายใจโดย ใช้ข้อมูลจากเวชระเบียนจำนวน 7,327 ราย โดยแบ่งออกเป็นโรคติดเชื้อทางเดินหายใจส่วนบนแบบเฉียบพลันพบว่าใช้ตัวแปรที่คัดเลือก 7 ตัวแปรกับชุดข้อมูลเรียนรู้ต่อชุดข้อมูลทดสอบ 70:30 ได้ค่าความถูกต้องของการจำแนกเท่ากับ 92.32% โรคปอดอักเสบพบว่าใช้ตัว แปรที่คัดเลือก 8 ตัวแปรกับชุดข้อมูลเรียนรู้ต่อชุดข้อมูลทดสอบ 70:30 ได้ค่าความถูกต้องของการจำแนกเท่ากับ 94.70% และโรคโพรง อากาศข้างจมูกอักเสบเฉียบพลันพบว่าใช้ตัวแปรที่คัดเลือก 7 ตัวแปรกับชุดข้อมูลเรียนรู้ต่อชุดข้อมูลทดสอบ 50:50 ได้ค่าความถูกต้อง ของการจำแนกเท่ากับ 94.69%

มณีรัตน์ ภารนันท์ [16]ได้กล่าวถึง WEKA ย่อมาจาก Waikato Environment for Knowledge Analysis ซึ่งเป็นชื่อมหาลัยแห่งหนึ่ง WEKA เป็นโปรแกรมที่พัฒนาขึ้นบนพื้นฐานของภาษาจาวา (Java)สามารถรัน (run) ได้หลายระบบปฏิบัติการ และสามารถพัฒนาต่อยอดโปรแกรมได้ เป็นเครื่องมือที่ ใช้ทำงานในด้านการทำดาต้าไมนิ่งที่รวบรวมแนวคิดอัลกอริทึมมากมาย ซึ่ง

อัลกอริทึมสามารถเลือกใช้งาน โดยตรงได้จาก 2 ทางคือจากชุดเครื่องมือที่มีอัลกอริทึมมาให้ หรือเลือกใช้จากอัลกอริทึมที่ได้เขียนเป็น โปรแกรมลงไปเป็นชุดเครื่องมือเพิ่มเติม และชุดเครื่องมือมีฟังก์ชันสำหรับการทำงานร่วมกับข้อมูล ได้แก่ Pre-Processing, Classification, Regression, Clustering, Association rules, Selection และ Visualization

นิเวศ จิระวิชิตชัย [17]ได้ทำการวิจัยเรื่องการค้นหาเทคนิคเหมืองข้อมูลเพื่อสร้างโมเดลการวิเคราะห์โรคอัตโนมัติ งานวิจัยนี้มีวัตถุประสงค์เพื่อค้นหาเทคนิคด้านเหมืองข้อมูล เพื่อสร้างโมเดลการวิเคราะห์ อัตโนมัติทดสอบประสิทธิภาพในการจำแนก (Classification) สำหรับข้อมูลทางการแพทย์ โดย ทดลองกับ 7 อัลกอริทึม ซึ่งประกอบด้วย Naïve Bayes,Multilayer Perceptron,Radial Basis Function Network, Support Vector Machine, K-Nearest Neighbor, Decision Tree, Ripper ทำการศึกษาเปรียบเทียบ วิธีลดคุณลักษณะที่เหมาะสมด้วยวิธี Correlation-based Feature Subset Selection (CFS) และวิธี Feature selection method based on correlation measure and relevance & redundancy analysis (FCBF) รวมถึงทดสอบอัลกอริทึมประเภท Single learning และ Multiple learning และทำการเพิ่มประสิทธิภาพการจำแนกด้วยวิธี Bagging และ Boosting

ณิชาพร สุระ [18]ทำการวิจัยเรื่องการจำแนกหมวดหมู่เอกสารภาษาไทยอัตโนมัติโดยใช้อัลกอริทึม FPTC มีวัตถุประสงค์เพื่อศึกษาเทคนิค พัฒนาระบบ และทดสอบประสิทธิภาพ อัลกอริทึม FPTC สำหรับการจำแนกหมวดหมู่เอกสารข้อความ โดยปัจจุบันเอกสารในรูปแบบ อิเล็กทรอนิกส์มีปริมาณและเนื้อหาทีหลากหลายมากขึ้น การสืบค้นและการจัดการเอกสารจะง่าย และเป็นไปตามความต้องการ ต้องอาศัยการจัดแบ่งเอกสารเป็นกลุ่มหรือหมวดหมู่ ให้สอดคล้อง และตรงกับดัชนีเพื่อให้จัดเก็บและค้นคืนเอกสารได้อย่างรวดเร็ว และมีประสิทธิภาพ งานวิจัยนี้ ประสงค์เพื่อพัฒนาเครื่องมือในการจำแนกหมวดหมู่เอกสารข้อความภาษาไทยด้วยอัลกอริทึม 38 Feature Projection Text Categorization (FPTC) ซึ่งเป็นอัลกอริทึมที่ปรับมาจาก k-Nearest Neighbor ลักษณะเด่นของ FPTC คือ การแทนคุณลักษณะในแบบภาพฉายของแต่ละคุณลักษณะ การจำแนกหมวดหมู่จะใช้วิธีการเปรียบเทียบความคล้ายของคำที่ปรากฏในเอกสารที่ใช้ทดสอบ กับคำที่ปรากฏในเอกสารที่ใช้ในกระบวนการเรียนรู้เพื่อหาเอกสารที่คล้ายกับเอกสารทดสอบมาก ที่สุด และกำหนดหมวดหมู่ของเอกสารนั้นให้กับเอกสารทดสอบ โดยจะใช้เอกสารข่าวภาษาไทย จากหนังสือพิมพ์ออนไลน์เป็นกรณีศึกษา จากผลการทดสอบพบว่า การจำแนกหมวดหมู่ด้วย อัลกอริทึม FPTC สามารถจำแนกหมวดหมู่เอกสารภาษาไทยได้อย่างมีประสิทธิผลดีสำหรับ ข้อมูลที่มีการกระจายตัวของหมวดหมู่เท่ากัน

# บทที่ 3
# วิธีดำเนินการวิจัย

การดำเนินงานวิจัยการเปรียบเทียบวิธีการเลือกแบบจำลองเพื่อการตรวจจับมัลแวร์โดยใช้เทคนิคการทำเหมืองข้อมูลโดยซึ่งประกอบด้วย ขั้นตอนในการวิจัยได้แบ่งออก 3 ขั้นตอน คือ การเตรียมข้อมูล การสร้างแบบจำลอง การวัดประสิทธิภาพแบบการสร้างจำลอง

## 3.1 การเตรียมข้อมูล

ทำการจัดเตรียมข้อมูลจากเว็บไซต์ Malwaredomainlist (MDL)13 ปีค.ศ. 2009 ถึง 2017 เพื่อใช้ในการวิเคราะห์การตรวจจับมัลแวร์ซึ่งเก็บข้อมูลเกี่ยวกับ วันที่เกิดมัลแวร์ โดเมน ไอพี ลักษณะคลาด การแปลงข้อมูลที่ได้ทำการรวบรวมมาให้เป็นข้อมูลที่สามารถนำไปวิเคราะห์ในการสร้างแบบจำลองโดยการ วิธี Malware File Detection จำนวน 2,311 โดเมน และผู้วิจัยได้คัดเลือกข้อมูลจำนวน 1,916 เรคคอร์ด 6 แอททริบิว คลาสผลลัพธ์จำนวน 6 คลาส ดังตารางที่ 1 และตารางที่ 2

ตารางที่ 1 แอททริบิวของข้อมูล

| ลำดับ | ชื่อแอททริบิว | สัญลักษณ์แอททริบิว | ชนิดแอททริบิว | ความหมาย |
|---|---|---|---|---|
| 1 | Date | date | Nominal | วันเกิด |
| 2 | Domain | domain | Nominal | ชื่อที่ใช้ระบุลงในคอมพิวเตอร์ |
| 3 | IP | ip address | Nominal | หมายเลขที่ใช้สำหรับระบุตัวตนของเครื่องคอมพิวเตอร์ที่เชื่อมต่ออยู่บนเครื่อข่าย |
| 4 | Reverse Lookup | reverse Lookup | Nominal | |
| 5 | Description | description | Nominal | คลาสชนิดของมัลแวร์ |
| 6 | Class | label | Nominal | คลาสประเภทมัลแวร์ |

จากตารางที่ 1 แสดง ชื่อแอททริบิว สัญลักษณ์แอททริบิว ชนิดแอททริบิว และความหมายของแอททริบิว

ตารางที่ 2 คลาส

| ลำดับ | ชื่อคลาส | จำนวนเรคคอร์ด |
|---|---|---|
| 1 | Trojan | 1,184 |
| 2 | Malware | 587 |
| 3 | No | 75 |
| 4 | virus | 30 |
| 5 | trojan | 33 |
| 6 | Spyware | 28 |

จากตารางที่ 2 แสดงชื่อคลาส และจำนวนเรคคอร์ด ได้ดังนี้ คลาส Trojan มีจำนวน 1,184 เรคคอร์ด คลาส Malware มีจำนวน 587 เรคคอร์ด คลาสไม่เป็นมัลแวร์มีจำนวน 75 เรคคอร์ด คลาส Virus มีจำนวน 30 เรคคอร์ด คลาส trojan มีจำนวน 33 เรคคอร์ด และคลาส Spyware มีจำนวน 28 เรคคอร์ด

**3.2 การสร้างแบบจำลอง**

การวิจัยครั้งนี้ได้นำข้อมูลจากขั้นตอนการเตรียมข้อมูล มาทำการวิเคราะห์โดยจะเลือกใช้อัลกอริทึมที่ได้ทำการเลือกมาเปรียบเทียบประสิทธิภาพในการตรวจจับมัลแวร์ ทั้ง 3 อัลกอริทึมเพื่อหาตัวแบบที่ให้ผลที่ดีที่สุดเพื่อวัดผลการในวิจัยครั้งนี้ โดยวัดจากค่าความถูกต้องของแต่ล่ะอัลกอริทึม และเทนิคในการทำเหมืองข้อมูลซึ่งได้แก่ อัลกอริทึม ซัพพอร์ตเวกเตอร์แมชชีน (Support Vector Machine) อัลกอริทึม นาอีฟเบย์ (Naive Bayes) และอัลกอริทึม เพื่อนบ้านใกล้สุด K ตัว (k-Nearest Neighbor) ในการวิจัยครั้งนี้ได้ใช้โปรแกรม Weka เป็นโปรแกรมช่วยในการทดสอบอัลกอริทึม

**3.3 วัดประสิทธิภาพแบบจำลอง**

การวัดประสิทธิภาพแบบจำลองในงานวิจัยนี้ผู้วิจัยได้นำวิธี 10-fold cross validationโมเดลมาใช้ เพื่อให้ข้อมูลทุกชุดเป็นทั้งชุดการสอนและชุดการทดสอบ โดยใช้ข้อมูล 9 ชุดเพื่อเป็นข้อมูลในการสอน (Training fold) และใช้ข้อมูลอีก 1 ชุด เป็นข้อมูลในการทดสอบ (Testing fold) หลังจากนั้นทำการสลับข้อมูลและทำซ้ำจนครบ 10 รอบ ดังแสดงในรูปที่ 6


ภาพประกอบที่ 7 10 fold cross validation

การวัดประสิทธิภาพของแบบจำลอง ค่าที่ใช้ในการวัดประสิทธิภาพ ได้แก่

1. ค่าความแม่นยำ (Precision)
2. ค่าเรียกคืน (Recall)
3. ค่า F-Measure

# บทที่ 4
# ผลการวิจัย

ผลการวิจัยข้อมูลเพื่อเปรียบเทียบแบบจำลองสำหรับการตรวจจับมัลแวร์ที่รวบรวมข้อมูลจากเว็บไซต์ Malwaredomainlist (MDL)13 ปีค.ศ. 2009 ถึง 2017 จำนวน 2,311 โดเมน และผู้วิจัยได้คัดเลือกข้อมูลจำนวน 1,916 เรคคอร์ด 6 แอททริบิว คลาสผลลัพธ์จำนวน 8 คลาส โดยใช้เทคนิคเหมืองข้อมูลจำนวน 3 อัลกอริทึม ได้แก่

1. ซัพพอร์ตเวกเตอร์แมชชีน(Support vector machines: SVM)
2. นาอีฟเบย์ (Naïve Bayes)
3. เพื่อนบ้านใกล้สุด K ตัว (k-Nearest Neighbor: k-NN)

ในการทดลองครั้งนี้งานวิจัยนี้ผู้วิจัยได้นำวิธี 10-fold cross validation โมเดลในการสอน Training ข้อมูล และ Testing ข้อมูล สามารถแสดงประสิทธิภาพของการสร้างแบบจำลองและการสร้างแบบจำลองดังต่อไปนี้

## 4.1 ผลการสร้างแบบจำลอง

4.1.1 ผลการสร้างแบบจำลองในการตรวจจับมัลแวร์โดยใช้เทคนิค ซัพพอร์ตเวกเตอร์แมชชีน Support vector machines จากผลการทดลองสร้างแบบจำลองในการตรวจจับมัลแวร์โดยใช้เทคนิค ซัพพอร์ตเวกเตอร์แมชชีน Support vector machines ได้การจำแนกที่ถูกต้องจำนวน 1,848 ตัวอย่าง ได้ผลการทดลองค่า Precision Recall F-measure และ Accuracy ได้ดังตารางที่ 3

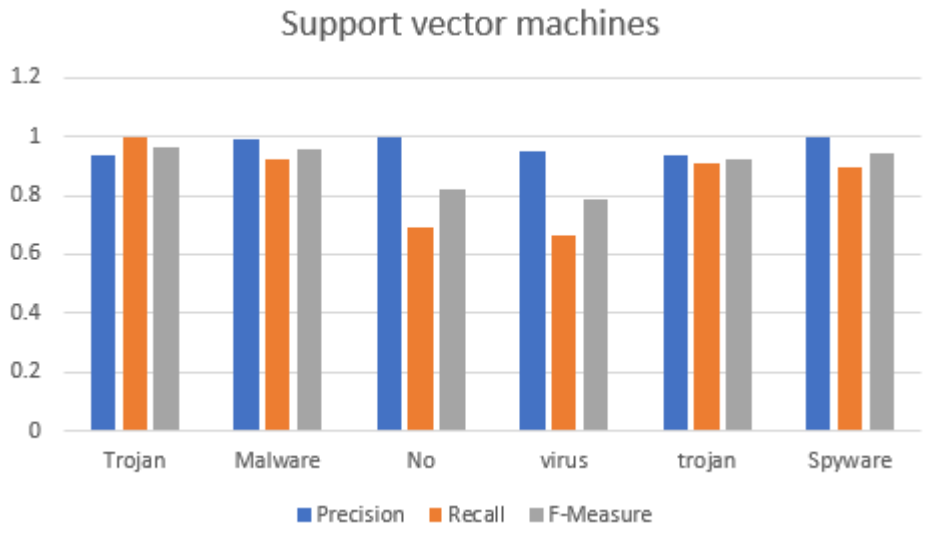
ตารางที่ 3 ผลการสร้างแบบจำลองโดยใช้เทคนิค ซัพพอร์ตเวกเตอร์แมชชีน Support vector machine

| Class | Precision | Recall | F-Measure |
|---|---|---|---|
| Trojan | 0.935 | 0.997 | 0.965 |
| Malware | 0.993 | 0.922 | 0.956 |
| No | 1 | 0.693 | 0.819 |
| virus | 0.952 | 0.667 | 0.784 |
| trojan | 0.938 | 0.909 | 0.923 |
| Spyware | 1 | 0.893 | 0.943 |

ตารางที่ 4 ผลการสร้างแบบจำลองโดยใช้เทคนิค ซัพพอร์ตเวกเตอร์แมชชีน Support vector machine (ต่อ)

| Class | Precision | Recall | F-Measure |
|---|---|---|---|
| Trojan | 0.935 | 0.997 | 0.965 |
| Malware | 0.993 | 0.922 | 0.956 |
| **Average** | 96.97% | 84.68% | 89.83% |

จากตารางที่ 3 แสดงผลการทดลองการสร้างแบบจำลองในการตรวจจับมัลแวร์โดยใช้เทคนิค ซัพพอร์ตเวกเตอร์แมชชีน Support vector machines พบว่าค่า Precision มี Average อยู่ที่ 96.97% ค่า Recall มี Average อยู่ที่ 84.68% และค่า F-measure มี Average อยู่ที่ 89.83%




ภาพประกอบที่ 8 กราฟแสดงการสร้างแบบจำลองในการตรวจจับมัลแวร์โดยใช้เทคนิค ซัพพอร์ตเวกเตอร์แมชชีน Support vector machines

จากภาพประกอบที่ 8 แสดงกราฟการสร้างแบบจำลองในการตรวจจับมัลแวร์โดยใช้เทคนิค ซัพพอร์ตเวกเตอร์แมชชีน Support vector machines ได้ดังนี้

Trojan พบว่า Precision มีค่าอยู่ที่ 0.935 Recall มีค่าอยู่ที่ 0.997 และ F-measure มีค่าอยู่ที่ 0.965

Malware พบว่า Precision มีค่าอยู่ที่ 0.993 Recall มีค่าอยู่ที่ 0.922 และ F-measure มีค่าอยู่ที่ 0.956

ไม่เป็นมัลแวร์ พบว่า Precision มีค่าอยู่ที่ 1 Recall มีค่าอยู่ที่ 0.693 และ F-measure มีค่าอยู่ที่ 0.819

Virus พบว่า Precision มีค่าอยู่ที่ 0.952 Recall มีค่าอยู่ที่ 0.667 และ F-measure มีค่าอยู่ที่ 0.784

trojan พบว่า Precision มีค่าอยู่ที่ 0.938 Recall มีค่าอยู่ที่ 0.909 และ F-measure มีค่าอยู่ที่ 0.923

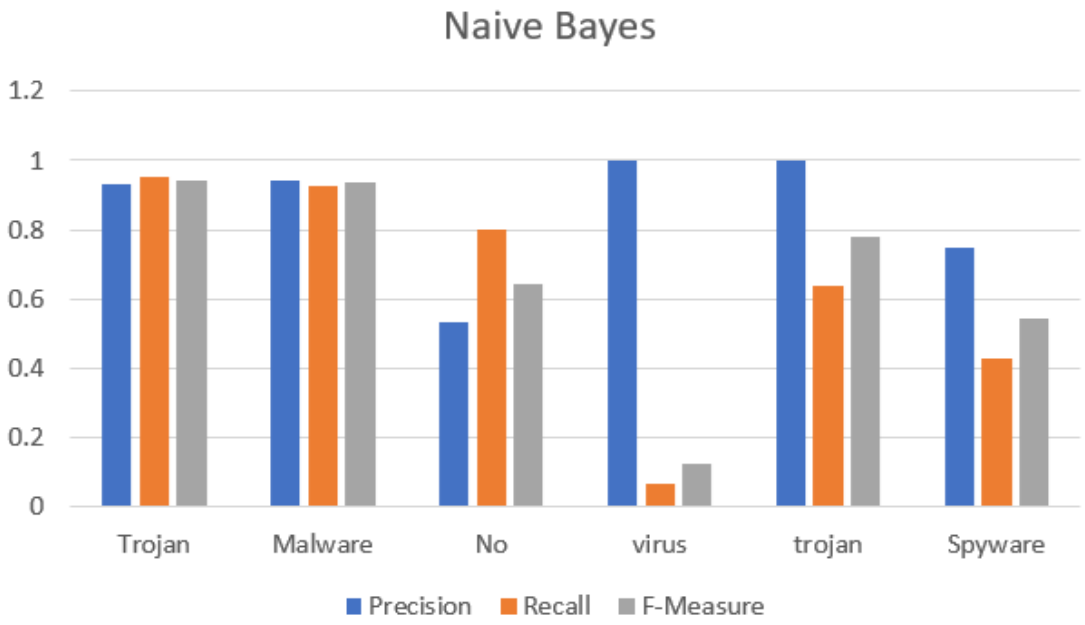
Spyware พบว่า Precision มีค่าอยู่ที่ 1 Recall มีค่าอยู่ที่ 0.893 และ F-measure มีค่าอยู่ที่ 0.943

4.1.2 ผลการสร้างแบบจำลองในการตรวจจับมัลแวร์โดยใช้นาอีฟเบย์ Naïve Bayes จากผลการทดลองสร้างแบบจำลองในการตรวจจับมัลแวร์โดยใช้เทคนิค นาอีฟเบย์ Naïve Bayes ได้การจำแนกที่ถูกต้องจำนวน 1,767 ตัวอย่าง ได้ผลการทดลองค่า Precision Recall และ F-measure ได้ดังตารางที่ 4

ตารางที่ 5 ผลการสร้างแบบจำลองโดยใช้เทคนิค นาอีฟเบย์ Naive Bayes

| Class | Precision | Recall | F-Measure |
|---|---|---|---|
| Trojan | 0.932 | 0.954 | 0.943 |
| Malware | 0.944 | 0.925 | 0.935 |
| No | 0.536 | 0.8 | 0.642 |
| virus | 1 | 0.067 | 0.125 |
| trojan | 1 | 0.636 | 0.778 |
| Spyware | 0.75 | 0.429 | 0.545 |
| **Average** | 86.03% | 63.52% | 66.13% |

จากตารางที่ 4 แสดงผลการทดลองการสร้างแบบจำลองในการตรวจจับมัลแวร์โดยใช้เทคนิค นาอีฟเบย์ Naïve Bayes พบว่าค่า Precision มี Average อยู่ที่ 86.03% ค่า Recall มี Average อยู่ที่ 63.52% และค่า F-measure มี Average อยู่ที่ 66.13%




ภาพประกอบที่ 9 กราฟแสดงการสร้างแบบจำลองในการตรวจจับมัลแวร์โดยใช้เทคนิค นาอีฟเบย์ Naive Bayes

จากภาพประกอบที่ 9 แสดงกราฟการสร้างแบบจำลองในการตรวจจับมัลแวร์โดยใช้เทคนิค นาอีฟเบย์ Naive Bayes ได้ดังนี้

Trojan พบว่า Precision มีค่าอยู่ที่ 0.932 Recall มีค่าอยู่ที่ 0.954 และ F-measure มีค่าอยู่ที่ 0.943

Malware พบว่า Precision มีค่าอยู่ที่ 0.944 Recall มีค่าอยู่ที่ 0.925 และ F-measure มีค่าอยู่ที่ 0.935

ไม่เป็นมัลแวร์ พบว่า Precision มีค่าอยู่ที่ 0.536 Recall มีค่าอยู่ที่ 0.8 และ F-measure มีค่าอยู่ที่ 0.8642

Virus พบว่า Precision มีค่าอยู่ที่ 1 Recall มีค่าอยู่ที่ 0.067 และ F-measure มีค่าอยู่ที่ 0.125

trojan พบว่า Precision มีค่าอยู่ที่ 1 Recall มีค่าอยู่ที่ 0.636 และ F-measure มีค่าอยู่ที่ 0.778

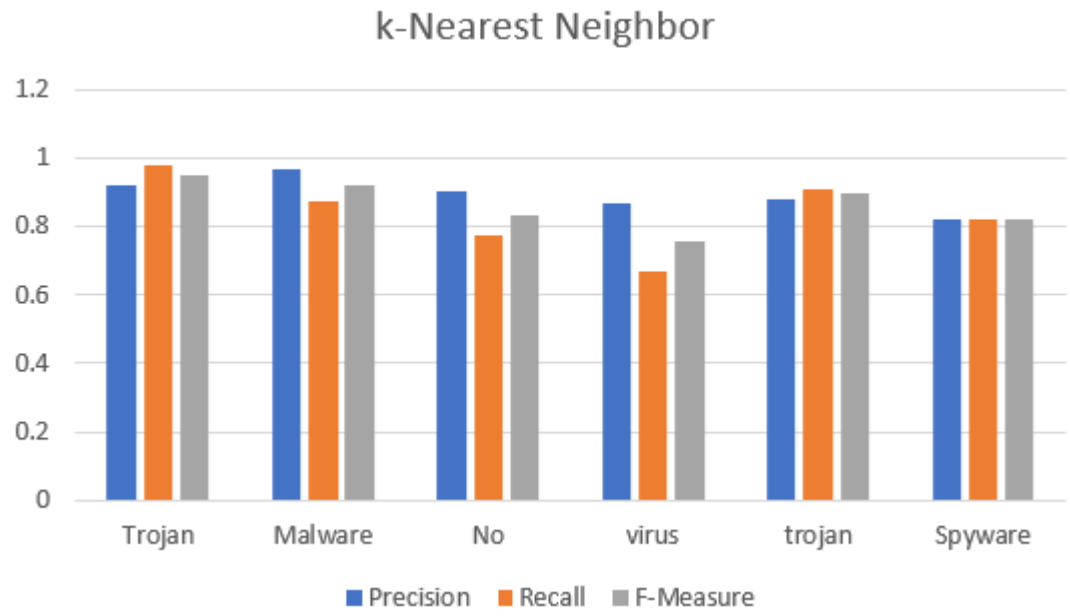
Spyware พบว่า Precision มีค่าอยู่ที่ 0.75 Recall มีค่าอยู่ที่ 0.429 และ F-measure มีค่าอยู่ที่ 0.545

4.1.3 ผลการสร้างแบบจำลองในการตรวจจับมัลแวร์โดยใช้ เพื่อนบ้านใกล้สุด K ตัว k-Nearest Neighbor จากผลการทดลองสร้างแบบจำลองในการตรวจจับมัลแวร์โดยใช้เทคนิค เพื่อนบ้านใกล้สุด K ตัว k-Nearest Neighbor ได้การจำแนกที่ถูกต้องจำนวน 1,801 ตัวอย่าง ได้ผลการทดลองค่า Precision Recall และ F-measure ได้ดังตารางที่ 5

ตารางที่ 6 ผลการสร้างแบบจำลองโดยใช้เทคนิค เพื่อนบ้านใกล้สุด K ตัว k-Nearest Neighbor

| Class | Precision | Recall | F-Measure |
|---|---|---|---|
| Trojan | 0.92 | 0.978 | 0.948 |
| Malware | 0.968 | 0.872 | 0.918 |
| No | 0.906 | 0.773 | 0.835 |
| virus | 0.87 | 0.667 | 0.755 |
| trojan | 0.882 | 0.909 | 0.896 |
| Spyware | 0.821 | 0.821 | 0.821 |
| **Average** | 89.45% | 83.67% | 86.22% |

จากตารางที่ 5 แสดงผลการทดลองการสร้างแบบจำลองในการตรวจจับมัลแวร์โดยใช้เทคนิค เพื่อนบ้านใกล้สุด K ตัว k-Nearest Neighbor พบว่าค่า Precision มี Average อยู่ที่ 89.45% ค่า Recall มี Average อยู่ที่ 83.67% และค่า F-measure มี Average อยู่ที่ 86.22%




ภาพประกอบที่ 10 กราฟแสดงการสร้างแบบจำลองในการตรวจจับมัลแวร์โดยใช้เทคนิค เพื่อนบ้านใกล้สุด K ตัว k-Nearest Neighbor

จากภาพประกอบที่ 10 แสดงกราฟการสร้างแบบจำลองในการตรวจจับมัลแวร์โดยใช้เทคนิคเพื่อนบ้านใกล้สุด K ตัว k-Nearest Neighbor ได้ดังนี้

Trojan พบว่า Precision มีค่าอยู่ที่ 0.92 Recall มีค่าอยู่ที่ 0.978 และ F-measure มีค่าอยู่ที่ 0.948

Malware พบว่า Precision มีค่าอยู่ที่ 0.968 Recall มีค่าอยู่ที่ 0.872 และ F-measure มีค่าอยู่ที่ 0.918

ไม่เป็นมัลแวร์ พบว่า Precision มีค่าอยู่ที่ 0.906 Recall มีค่าอยู่ที่ 0.773 และ F-measure มีค่าอยู่ที่ 0.835

Virus พบว่า Precision มีค่าอยู่ที่ 0.87 Recall มีค่าอยู่ที่ 0.667 และ F-measure มีค่าอยู่ที่ 0.755

trojan พบว่า Precision มีค่าอยู่ที่ 0.882 Recall มีค่าอยู่ที่ 0.909 และ F-measure มีค่าอยู่ที่ 0.755

Spyware พบว่า Precision มีค่าอยู่ที่ 0.821 Recall มีค่าอยู่ที่ 0.821 และ F-measure มีค่าอยู่ที่ 0.821

**4.2 การวิเคราะห์ประสิทธิภาพของโมเดลการสร้างแบบจำลอง**

ในงานวิจัยนี้ผู้วิจัยได้วัดประสิทธิภาพของโมเดลการสร้างแบบจำลองโดยใช้ค่า Confusion Matrix [19]

```
=== Confusion Matrix ===

    a    b    c    d    e    f   <-- classified as
 1180    2    0    1    1    0 |    a = trojan
   43  543    0    0    1    0 |    b = malware
   22    1   52    0    0    0 |    c = No
    9    1    0   20    0    0 |    d = virus
    3    0    0    0   30    0 |    e = trojan
    3    0    0    0    0   25 |    f = Spyware
```

ภาพประกอบที่ 11 ผลการวัดประสิทธิภาพของโมเดลการสร้างแบบจำลองโดยใช้เทคนิคซัพพอร์ตเวกเตอร์แมชชีน Support vector machine

จากภาพประกอบที่ 12 แสดงผลการวัดวัดประสิทธิภาพของโมเดลการสร้างแบบจำลองโดยใช้เทคนิค ซัพพอร์ตเวกเตอร์แมชชีน Support vector machines และผลการทดสอบแสดงในรูปแบบ Confusion Matrix จากข้อมูลจำนวน 1,916 เรคคอร์ด 6 แอททริบิว คลาสผลลัพธ์จำนวน 8 คลาส พบว่าข้อมูลจำนวน 1,916 เรคคอร์ด

- ข้อมูลที่เป็น Trojan ได้การจำแนกที่ถูกต้องจำนวน 1,180 เรคคอร์ด คิดเป็น 61.59% ได้การจำแนกที่ไม่ถูกต้องซึ่งเป็น malware 43 เรคคอร์ด คิดเป็น 2.24% ไม่เป็นมัลแวร์ 22 เรคคอร์ด คิดเป็น 1.15% เป็น virus 9 เรคคอร์ด คิดเป็น 0.47% เป็น trojan 3 เรคคอร์ด คิดเป็น 0.16% และเป็น Spyware 3 เรคคอร์ด คิดเป็น 0.16%

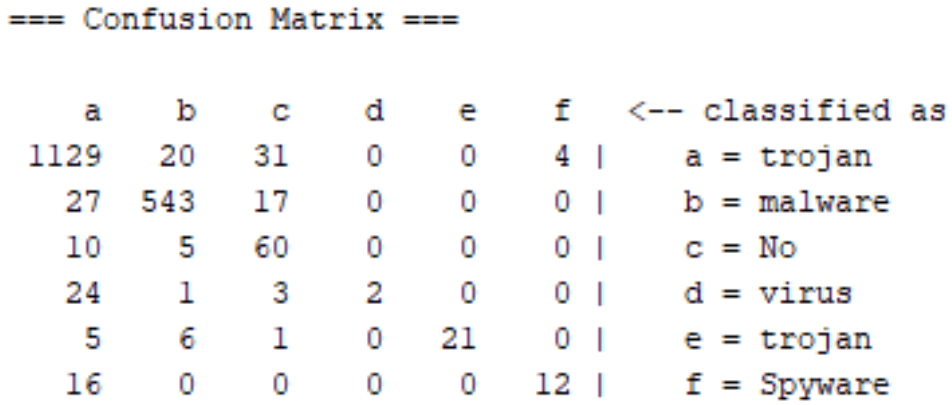
- ข้อมูลที่เป็น malware ได้การจำแนกที่ถูกต้องจำนวน 543 เรคคอร์ด คิดเป็น 28.34% ได้การจำแนกที่ไม่ถูกต้องซึ่งเป็น Trojan 2 เรคคอร์ด คิดเป็น 0.17% ไม่เป็นมัลแวร์ 1 เรคคอร์ด คิดเป็น 0.05% และเป็น virus 1 เรคคอร์ด คิดเป็น 0.05%

- ข้อมูลที่เป็น ไม่เป็นมัลแวร์ ได้การจำแนกที่ถูกต้องจำนวน 52 เรคคอร์ด คิดเป็น 2.71%

- ข้อมูลที่เป็น virus ได้การจำแนกที่ถูกต้องจำนวน 20 เรคคอร์ด คิดเป็น 1.04% การจำแนกที่ไม่ถูกต้องซึ่งเป็น Trojan 1 เรคคอร์ด คิดเป็น 0.05%

- ข้อมูลที่เป็น trojan ได้การจำแนกที่ถูกต้องจำนวน 30 เรคคอร์ด คิดเป็น 1.57% การจำแนกที่ไม่ถูกต้องซึ่งเป็น Trojan 1 เรคคอร์ด คิดเป็น 0.05% และเป็น malware 1 เรคคอร์ด คิดเป็น 0.05%

- ข้อมูลที่เป็น Spyware ได้การจำแนกที่ถูกต้องจำนวน 25 เรคคอร์ด คิดเป็น 1.30%



```
=== Confusion Matrix ===

    a    b    c    d    e    f   <-- classified as
 1129   20   31    0    0    4 |    a = trojan
   27  543   17    0    0    0 |    b = malware
   10    5   60    0    0    0 |    c = No
   24    1    3    2    0    0 |    d = virus
    5    6    1    0   21    0 |    e = trojan
   16    0    0    0    0   12 |    f = Spyware
```

ภาพประกอบที่ 12 ผลการวัดประสิทธิภาพของโมเดลการสร้างแบบจำลองโดยใช้เทคนิคนาอีฟเบย์ Naive Bayes

จากภาพประกอบที่ 13 แสดงผลการวัดวัดประสิทธิภาพของโมเดลการสร้างแบบจำลองโดยใช้เทคนิค นาอีฟเบย์ Naive Bayes และผลการทดสอบแสดงในรูปแบบ Confusion Matrix จากข้อมูลจำนวน 1,916 เรคคอร์ด 6 แอททริบิว คลาสผลลัพธ์จำนวน 8 คลาส พบว่าข้อมูลจำนวน 1,916 เรคคอร์ด

- ข้อมูลที่เป็น Trojan ได้การจำแนกที่ถูกต้องจำนวน 1,129 เรคคอร์ด คิดเป็น 58.92% ได้การจำแนกที่ไม่ถูกต้องซึ่งเป็น malware 27 เรคคอร์ด คิดเป็น 1.41% ไม่เป็นมัลแวร์ 10 เรคคอร์ด คิดเป็น 0.52% เป็น virus 24 เรคคอร์ด คิดเป็น 1.25% เป็น trojan 5 เรคคอร์ด คิดเป็น 0.26% และเป็น Spyware 16 เรคคอร์ด คิดเป็น 0.84%

- ข้อมูลที่เป็น malware ได้การจำแนกที่ถูกต้องจำนวน 543 เรคคอร์ด คิดเป็น 28.34% ได้การจำแนกที่ไม่ถูกต้องซึ่งเป็น Trojan 20 เรคคอร์ด คิดเป็น 1.69% ไม่เป็นมัลแวร์ 5 เรคคอร์ด คิดเป็น 0.26% เป็น virus 1 เรคคอร์ด คิดเป็น 0.05% และเป็น trojan 6 เรคคอร์ด คิดเป็น 0.31%

- ข้อมูลที่เป็น ไม่เป็นมัลแวร์ ได้การจำแนกที่ถูกต้องจำนวน 60 เรคคอร์ด คิดเป็น 3.13% การจำแนกที่ไม่ถูกต้องซึ่งเป็น Trojan 31 เรคคอร์ด คิดเป็น 1.69% เป็น malware 17 เรคคอร์ด คิดเป็น 0.89% เป็น virus 3 เรคคอร์ด คิดเป็น 0.16% และเป็น trojan 1 เรคคอร์ด คิดเป็น 0.05%

- ข้อมูลที่เป็น virus ได้การจำแนกที่ถูกต้องจำนวน 2 เรคคอร์ด คิดเป็น 1.10%

- ข้อมูลที่เป็น trojan ได้การจำแนกที่ถูกต้องจำนวน 21 เรคคอร์ด คิดเป็น 1.10%

- ข้อมูลที่เป็น Spyware ได้การจำแนกที่ถูกต้องจำนวน 12 เรคคอร์ด คิดเป็น 0.63% การจำแนกที่ไม่ถูกต้องซึ่งเป็น Trojan 4 เรคคอร์ด คิดเป็น 0.21

```
=== Confusion Matrix ===

    a    b    c    d    e    f   <-- classified as
 1158   15    2    3    3    3 |    a = trojan
   70  513    2    0    1    1 |    b = malware
   16    1   58    0    0    0 |    c = No
    6    1    2   20    0    1 |    d = virus
    3    0    0    0   30    0 |    e = trojan
    5    0    0    0    0   23 |    f = Spyware
```

ภาพประกอบที่ 13 ผลการวัดประสิทธิภาพของโมเดลการสร้างแบบจำลองโดยใช้เทคนิคเพื่อนบ้านใกล้สุด K ตัว k-Nearest Neighbor

จากภาพประกอบที่ 14 แสดงผลการวัดวัดประสิทธิภาพของโมเดลการสร้างแบบจำลองโดยใช้เทคนิคเพื่อนบ้านใกล้สุด K ตัว k-Nearest Neighbor และผลการทดสอบแสดงในรูปแบบ Confusion Matrix จากข้อมูลจำนวน 1,916 เรคคอร์ด 6 แอททริบิว คลาสผลลัพธ์จำนวน 8 คลาส พบว่าข้อมูลจำนวน 1,916 เรคคอร์ด

- ข้อมูลที่เป็น Trojan ได้การจำแนกที่ถูกต้องจำนวน 1,156 เรคคอร์ด คิดเป็น 60.44% ได้การจำแนกที่ไม่ถูกต้องซึ่งเป็น malware 70 เรคคอร์ด คิดเป็น 3.65% ไม่เป็นมัลแวร์ 16 เรคคอร์ด คิดเป็น 0.84% เป็น virus 6 เรคคอร์ด คิดเป็น 0.31% เป็น trojan 3 เรคคอร์ด คิดเป็น 0.16% และเป็น Spyware 5 เรคคอร์ด คิดเป็น 0.26%

- ข้อมูลที่เป็น malware ได้การจำแนกที่ถูกต้องจำนวน 513 เรคคอร์ด คิดเป็น 26.77% ได้การจำแนกที่ไม่ถูกต้องซึ่งเป็น Trojan 15 เรคคอร์ด คิดเป็น 1.27% ไม่เป็นมัลแวร์ 1 เรคคอร์ด คิดเป็น 0.05% และเป็น virus 1 เรคคอร์ด คิดเป็น 0.05%

- ข้อมูลที่เป็น ไม่เป็นมัลแวร์ ได้การจำแนกที่ถูกต้องจำนวน 58 เรคคอร์ด คิดเป็น 3.03% การจำแนกที่ไม่ถูกต้องซึ่งเป็น Trojan 2 เรคคอร์ด คิดเป็น 0.10% เป็น malware 2 เรคคอร์ด คิดเป็น 0.10% และเป็น virus 2 เรคคอร์ด คิดเป็น 0.10%

- ข้อมูลที่เป็น virus ได้การจำแนกที่ถูกต้องจำนวน 0 เรคคอร์ด คิดเป็น 0%

- ข้อมูลที่เป็น trojan ได้การจำแนกที่ถูกต้องจำนวน 30 เรคคอร์ด คิดเป็น 1.57%

- ข้อมูลที่เป็น Spyware ได้การจำแนกที่ถูกต้องจำนวน 23 เรคคอร์ด คิดเป็น 1.20%

**4.3 ผลการสร้างแบบจำลอง**

ในงานวิจัยนี้ผู้วิจัยได้วัดประสิทธิภาพโดยใช้ค่า Precision, Recall, F-Measure และ Accuracy ดังตารางที่ 8

ตารางที่ 8 ผลการวิเคราะห์ประสิทธิภาพของการสร้างแบบจำลอง

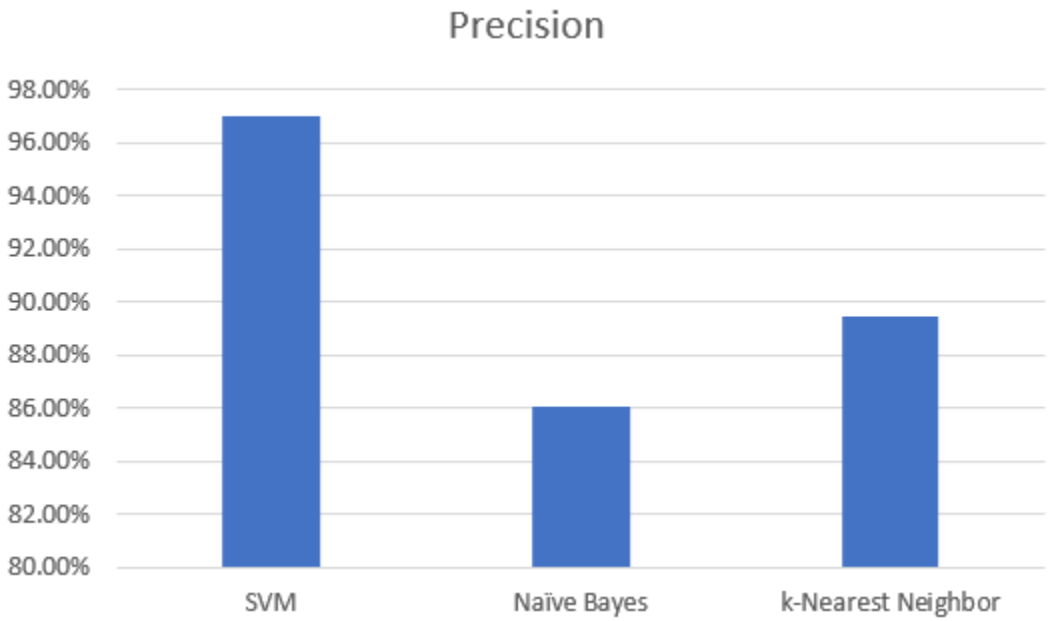
| แบบจำลอง | Precision | Recall | F-Measure | Accuracy |
|---|---|---|---|---|
| SVM | 96.97% | 84.68% | 89.83% | 95.41% |
| Naïve Bayes | 86.03% | 63.52% | 66.13% | 91.22% |
| k-Nearest Neighbor | 89.45% | 83.67% | 86.22% | 92.98% |

จากตารางที่ 8 แสดงผลการทดลองการสร้างแบบจำลองในการตรวจจับมัลแวร์ได้ดังนี้

ซัพพอร์ตเวกเตอร์แมชชีน Support vector machines พบว่าค่า Precision มี Average อยู่ที่ 96.97% ค่า Recall มี Average อยู่ที่ 84.68% ค่า F-measure มี Average อยู่ที่ 89.83% และค่า Accuracy มี Average อยู่ที่ 95.41%

เทคนิค นาอีฟเบย์ Naïve Bayes พบว่าค่า Precision มี Average อยู่ที่ 86.03% ค่า Recall มี Average อยู่ที่ 63.52% ค่า F-measure มี Average อยู่ที่ 66.13% และค่า Accuracy มี Average อยู่ที่ 91.22%

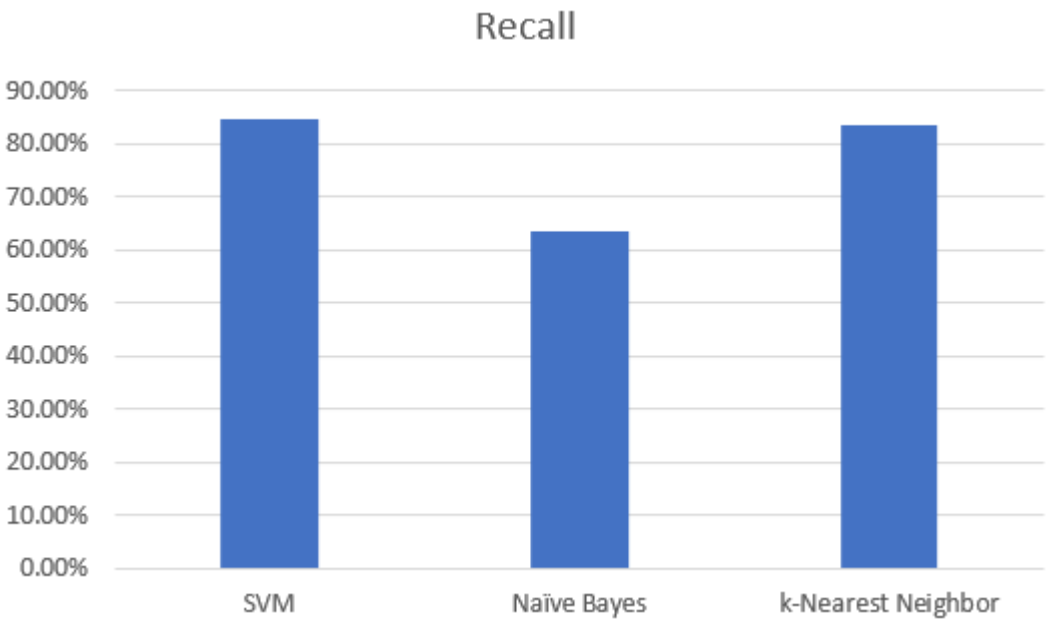
เพื่อนบ้านใกล้สุด K ตัว k-Nearest Neighbor พบว่าค่า Precision มี Average อยู่ที่ 89.45% ค่า Recall มี Average อยู่ที่ 83.67% ค่า F-measure มี Average อยู่ที่ 86.22% และค่า Accuracy มี Average อยู่ที่ 92.98%




ภาพประกอบที่ 14 Comparison Precision of Model

จากภาพประกอบที่ 11 แสดงกราฟการสร้างแบบจำลองในการตรวจจับมัลแวร์โดยมีค่า Precision ได้ดังนี้

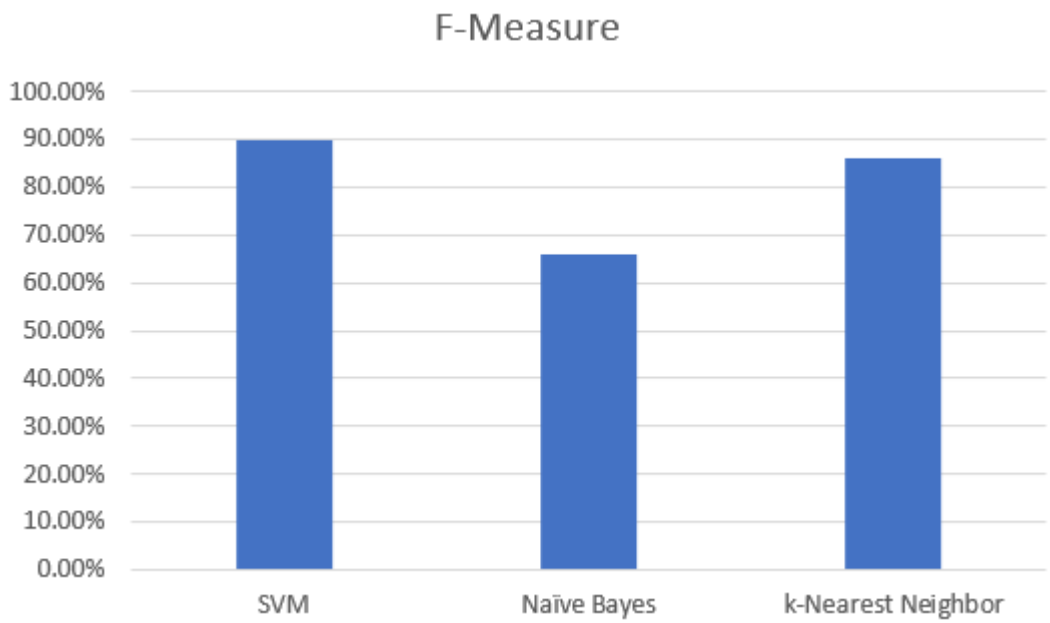
- Support vector machines มีค่าอยู่ที่ 96.97%
- Naïve Bayes มีค่าอยู่ที่ 91.22%
- k-Nearest Neighbor มีค่าอยู่ที่ 92.98%




ภาพประกอบที่ 15 Comparison Recall of Model

จากภาพประกอบที่ 12 แสดงกราฟการสร้างแบบจำลองในการตรวจจับมัลแวร์โดยมีค่า Recall ได้ดังนี้

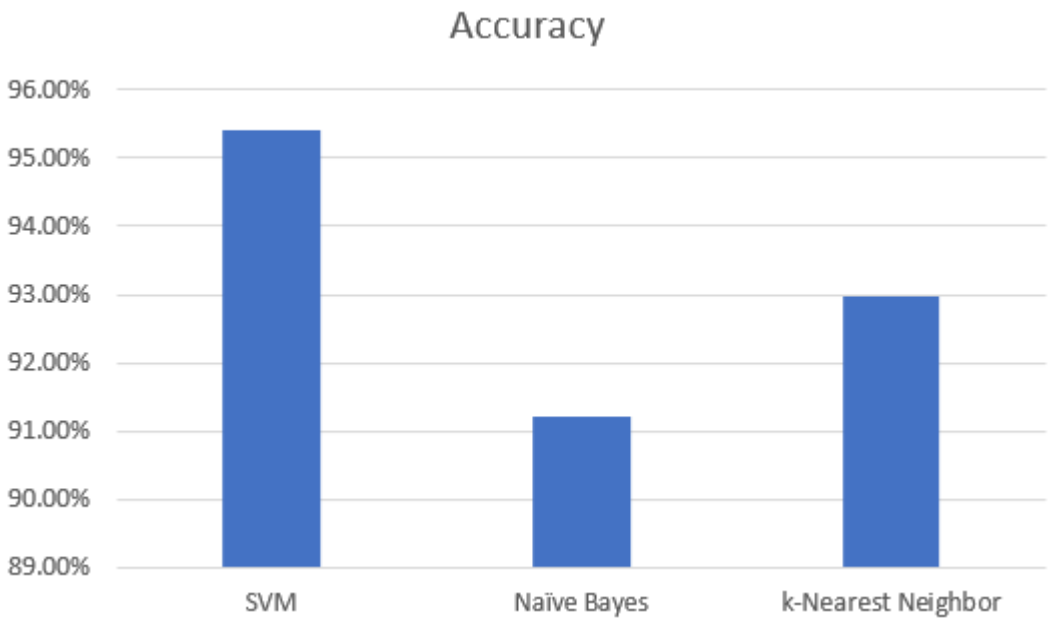
- Support vector machines มีค่าอยู่ที่ 84.68%
- Naïve Bayes มีค่าอยู่ที่ 63.52%
- k-Nearest Neighbor มีค่าอยู่ที่ 83.67%




ภาพประกอบที่ 16 Comparison F-Measure of Model

จากภาพประกอบที่ 12 แสดงกราฟการสร้างแบบจำลองในการตรวจจับมัลแวร์โดยมีค่า F-Measure ได้ดังนี้

- Support vector machines มีค่าอยู่ที่ 84.68%
- Naïve Bayes มีค่าอยู่ที่ 63.52%
- k-Nearest Neighbor มีค่าอยู่ที่ 83.67%




ภาพประกอบที่ 17 ภาพประกอบที่ 18 Comparison Accuracy of Model

จากภาพประกอบที่ 12 แสดงกราฟการสร้างแบบจำลองในการตรวจจับมัลแวร์โดยมีค่า Accuracy ได้ดังนี้

- Support vector machines มีค่าอยู่ที่ 95.41%
- Naïve Bayes มีค่าอยู่ที่ 91.22%
- k-Nearest Neighbor มีค่าอยู่ที่ 92.98%

# บทที่ 5
# สรุปอภิปรายผลการวิจัยและข้อเสนอแนะ

## 5.1 สรุปผลการวิจัย

การตรวจจับมัลแวร์ด้วยเทคนิคการจำแนกในการทำเหมืองข้อมูล โดยการเปรียบเทียบประสิทธิภาพการจำแนกแบบจำลองสำหรับการตรวจจับมัลแวร์ที่บุกรุกเครื่องคอมพิวเตอร์โดยใช้อัลกอริทึม 3 อัลกอริทึมด้วยกันคือ ซัพพอร์ตเวกเตอร์แมชชีน (Support vector machines) อีฟเบย์ (Naïve Bayes) และ เพื่อนบ้านใกล้สุด K ตัว (k-Nearest Neighbor) พบว่าแบบจำลองที่ใช้อัลกอริทึม ซัพพอร์ตเวกเตอร์แมชชีน นั้นมีความถูกต้องมากที่สุดตามด้วยอัลกอริทึม เพื่อนบ้านใกล้สุด K ตัว และ นาอีฟเบย์ ตามลำดับแต่อย่างไรก็ตามแบบจำลองของอัลกอริทึม ซัพพอร์ตเวกเตอร์แมชชีน และแบบจำลองของอัลกอริทึม เพื่อนบ้านใกล้สุด K ตัว นั้นก็มีความแตกต่างกันเพียงเล็กน้อย จึงทำให้การสร้างแบบจำลองที่ใช้อัลกอริทึม ซัพพอร์ตเวกเตอร์แมชชีน และ เพื่อนบ้านใกล้สุด K ตัว นั้นเหมาะที่จะนำไปใช้ในการตรวจจับมัลแวร์บุกรุกเครื่องคอมพิวเตอร์ เพราะสามารถเพิ่มความปลอดภัยและประสิทธิภาพให้กับระบบเครือข่ายได้ ส่วนแบบจำลองที่ใช้อัลกอริทึม Naïve Bayes นั้นมีความเหมาะสมน้อยที่สุดที่จะนำไปใช้ในการตรวจจับมัลแวร์ที่บุกรุกเครื่องคอมพิวเตอร์

## 5.2 อภิปรายผลการวิจัย

จากการทดลองงานวิจัยโดยเปรียบเทียบประสิทธิภาพการจำแนกรูปแบบแบบจำลองสำหรับการตรวจจับมัลแวร์ที่บุกรุกเครื่องคอมพิวเตอร์ ซึ่งที่รวบรวมข้อมูลจากเว็บไซต์ Malwaredomainlist (MDL)13 ปีค.ศ. 2009 ถึง 2017 จำนวน 2,311 โดเมน และผู้วิจัยได้คัดเลือกข้อมูลจำนวน 1,916 เรคคอร์ด 6 แอททริบิว คลาสผลลัพธ์จำนวน 6 คลาส เป็นชุดข้อมูลที่ใช้ในทดลองงานวิจัยและใช้วิธีการ 10-fold validation ในการสร้างแบบจำลองโดยใช้อัลกอริทึม 3 อัลกอริทึม คือ ซัพพอร์ตเวกเตอร์แมชชีน (Support vector machines) นาอีฟเบย์ (Naïve Bayes) และ เพื่อนบ้านใกล้สุด K ตัว (k-Nearest Neighbor) จากผลการทดลองงานวิจัยนั้นแบบจำลองที่สร้างจากอัลกอริทึม ซัพพอร์ตเวกเตอร์แมชชีน (Support vector machines) นั้นมีค่าความถูกต้อง (Accuracy) เท่ากับ 95.41% มีค่าความแม่นยำ (Precision) เท่ากับ 96.97% มีค่าระลึก (Recall) เท่ากับ 84.68% และค่า F-Measure เท่ากับ 89.83% แบบจำลองที่สร้างจากอัลกอริทึม นาอีฟเบย์ (Naïve Bayes) นั้นมีค่าความถูกต้อง (Accuracy) เท่ากับ 91.22% มีค่าความแม่นยำ (Precision) เท่ากับ 86.03% มีค่า

ระลึก (Recall) เท่ากับ 63.52% และค่า F-Measure 66.13% และแบบจำลองที่สร้างจากอัลกอริทึมเพื่อนบ้านใกล้สุด K ตัว (k-Nearest Neighbor) นั้นมีค่าความถูกต้อง (Accuracy) เท่ากับ 92.98% มีค่าความแม่นยำ (Precision) เท่ากับ 89.45% มีค่าระลึก (Recall) เท่ากับ 83.67% และค่า F-Measure เท่ากับ 86.22% ผลจากค่าทางสถิติต่างๆจะเห็นได้ว่าแบบจำลองที่สร้างจากอัลกอริทึมทั้ง 3 อัลกอริทึม มีประสิทธิภาพค่อนข้างสูงแต่แบบจำลองที่มีประสิทธิภาพมากที่สุดคือแบบจำลองที่จากอัลกอริทึม ซัพพอร์ตเวกเตอร์แมชชีน นั้นมีความถูกต้องมากที่สุดตามด้วยอัลกอริทึม เพื่อนบ้านใกล้สุด K ตัว และ นาอีฟเบย์ ตามลำดับ

**5.3 ข้อเสนอแนะการวิจัย**

จากการทดลองงานวิจัยการตรวจจับมัลแวร์ที่บุกรุกเครื่องคอมพิวเตอร์ ด้วยเทคนิคการจำแนกทั้ง 3 อัลกอริทึม ถึงแม้แบบจำลองนั้นมีความแม่นยำค่อนข้างสูงแต่ในปัจจุบันนั้นมีรูปแบบการบุกรุกเครื่องคอมพิวเตอร์มีหลากและซับซ้อนขึ้น การนำระบบตรวจจับมัลแวร์ไปใช้ในระบบเครื่องคอมพิวเตอร์อาจช่วยทำให้ระบบเครื่องคอมพิวเตอร์มีความปลอดภัยมากขึ้นแต่ก็ยังไม่สามารถตรวจจับมัลแวร์ในรูปแบบการบุกรุกเครื่องคอมพิวเตอร์ ทั้งหมดได้ อีกทั้งข้อมูลจากเว็บไซต์ Malwaredomainlist (MDL) ที่ได้ใช้ในงานวิจัยครั้งนี้เป็นข้อมูลที่ถูกสร้างขึ้นนานหลายปีแล้ว ปัจจุบันนั้นอาจมีรูปแบบการโจมตีและการบุกรุกเครื่องคอมพิวเตอร์ รูปแบบใหม่ๆเกิดขึ้น อาจทำให้แบบจำลองของระบบตรวจจับมัลแวร์ทในงานวิจัยนี้ไม่สามารถตรวจจับมัลแวร์ทดังกล่าวได้ แต่อย่างไรก็ตามถึงแม้เครื่องคอมพิวเตอร์จะมีซอฟต์แวร์หรือฮาร์ดแวร์ด้านความปลอดภัยเข้ามาช่วยเพิ่มความปลอดภัยมากเพียงใด ก็ยังมีช่องโหว่ให้ผู้ที่ไม่หวังดีเข้าโจมตีหรือบุกรุกเครื่องคอมพิวเตอร์ได้เสมอ และผู้ใช้เองก็ต้องระมัดระวังข้อมูลส่วนตัวเมื่อต้องใช้เครื่องคอมพิวเตอร์ที่รู้สึกว่าไม่มีความปลอดภัย

# บรรณานุกรม

# บรรณานุกรม


[1] R. S. Pirscoveanu, S. S. Hansen, T. M. T. Larsen, M. Stevanovic, J. M. Pedersen, and A. Czech, "*Analysis of Malware behavior: Type classification using machine learning*," in *2015 International Conference on Cyber Situational Awareness, Data Analytics and Assessment (CyberSA)*, 2015, pp. 1-7.

[2] C. Fan, H. Hsiao, C. Chou, and Y. Tseng, "*Malware Detection Systems Based on API Log Data Mining*," in *2015 IEEE 39th Annual Computer Software and Applications Conference*, 2015, pp. 255-260.

[3] วิภาวรรณ บัวทอง, "*การเปรียบเทียบประสิทธิภาพของเทคนิคการลดมิติข้อมูลด้วยวิธีการจัดอันดับ แบบ Information Gain , Gain Ratio และ Linear SVM Weights*," in การเปรียบเทียบประสิทธิภาพของเทคนิคการลดมิติข้อมูลด้วยวิธีการจัดอันดับ แบบ *Information Gain* 2555.

[4] นายศุภกร สระบัว, "*การตรวจจับมัลแวร์ประเภทม้าโทรจันอย่างรวดเร็วโดยเทคนิคเหมืองข้อมูล*," 2557.

[5] N. R. Rosyid, M. Ohrui, H. Kikuchi, P. Sooraksa, and M. Terada, "*A discovery of sequential attack patterns of malware in botnets*," in *2010 IEEE International Conference on Systems, Man and Cybernetics*, 2010, pp. 2564-2570.

[6] นางสาวภวิสาณัชช์ ศรศิริวงศ์, "*การพัฒนาโปรแกรมประยุกต์สำหรับการประเมินผลการฝึกประสบการณ์*," 2557.

[7] รองศาสตราจารย์ ดร.สมชาย ปราการเจริญ, "*การวิเคราะห์มัลแวร์เบื้องต้น*," 2557.

[8] S. Kotsiantis, C. Pierrakeas, and P. Pintelas, "PREDICTING STUDENTS' PERFORMANCE IN DISTANCE LEARNING USING MACHINE LEARNING TECHNIQUES," *Applied Artificial Intelligence,* vol. 18, pp. 411-426, 2004/05/01 2004.

[9] A. Azab, "*Fifth Cybercrime and Trustworthy Computing Workshop*," in *3rd Cybercrime and Trustworthy Computing Workshop (CTC-2012)* SIR Ranking of United States, 2014.

[10] Igor Santos, Yoseba K. Penya, Jaime Devesa, and P. G. Bringas, "*N-Grams-based file signatures for malware detection*," in *Proceedings of the 2009*

*International Conference on Enterprise Information Systems (ICEIS)*, 2009, pp. 317-320.

[11] ฉัตรชัย เลี้ยงบุญประคอง, "*การจำแนกประเภทมัลแวร์แฟมิลี่ด้วยคุณลักษณะรูปแบบลำดับเอ็นแกรม*," 2012.

[12] K. Rieck, "*Learning and Classification of Malware Behavior*," 2008.

[13] P. Cheewaprakobkit, "*Study of Factors Analysis Affecting Academic Achievement of Undergraduate Students in International Program*," IMECS, Hong Kong, 2013, pp. 1-8.

[14] Y. Pansumret, "*On Comparison of Data Mining Algorithms for Analysis of Factors Affecting the Academic Performance of Students*," Technol MSU, 2013, pp. 281-289.

[15] J. Z. Kolter and M. A. Maloof, "*Learning to Detect and Classify Malicious Executables in the Wild*," *J. Mach. Learn. Res.,* vol. 7, pp. 2721-2744, 2006.

[16] มณีรัตน์ ภารนันท์, "*โปรแกรมWEKA*," 2555.

[17] นายนิเวศ จิระวิชิตชัย, "*การค้นหาเทคนิคเหมืองข้อมูลเพื่อสร้างโมเดลการวิเคราะห์โรคอัตโนมัติ*," 2553.

[18] ณิชาพร สุระ., "*การจำแนกหมวดหมู่เอกสารภาษาไทยอัตโนมัติโดยใช้อัลกอริทึม FPTC*," สถาบันเทคโนโลยีพระจอมเกล้าพระนคร เหนือ, 2549.

[19] D. Simon, "*Analytic Confusion Matrix Bounds for Fault Detection and Isolation Using a Sum-of-Squared-Residuals Approach*," in *IEEE Transactions*, pp. 287-296.

# ประวัติผู้เขียน

| | |
|---|---|
| **ชื่อ** | นายวรารินทร์ ปัญญาวงษ์ |
| **วันเกิด** | วันที่ 19 พฤษภาคม พ.ศ. 2531 |
| **สถานที่เกิด** | อำเภอนิคมคำสร้อย จังหวัดมุกดาหาร |
| **สถานที่อยู่ปัจจุบัน** | บ้านเลขที่ 141 หมู่ 2 อำเภอ นิคมคำสร้อย จังหวัดมุกดาหาร รหัสไปรษณีย์ 49130 |
| **ตำแหน่งหน้าที่การงาน** | นักวิชาการคอมพิวเตอร์ |
| **สถานที่ทำงานปัจจุบัน** | คณะสถาปัตยกรรมศาสตร์ ผังเมืองและนฤมิตศิลป์ มหาวิทยาลัยมหาสารคาม ตำบลขามเรียง อำเภอกันทรวิชัย จังหวัดมหาสารคาม 44150 |
| **ประวัติการศึกษา** | พ.ศ. 2549 มัธยมศึกษาตอนปลาย โรงเรียนมุกดาหาร จังหวัดมุกดาหาร<br>พ.ศ. 2552 วิทยาลัยการอาชีพนวมินทราชินีมุกดาหาร จังหวัดมุกดาหาร<br>พ.ศ. 2554 ปริญญาวิทยาศาสตรบัณฑิต (วท.บ.) สาขาวิชาเทคโนโลยีสารสนเทศและการสื่อสาร มหาวิทยาลัยมหาสารคาม<br>พ.ศ. 2562 ปริญญาวิทยาศาสตรมหาบัณฑต (วท.ม.) สาขาวิชาเทคโนโลยีสารสนเทศ มหาวิทยาลัยมหาสารคาม |
| **ผลงานวิจัย** | มหาวิทยาลัยมหาสารคาม |