วิธีการเรียนรู้เชิงลึกสำหรับการคัดกรองข้อความขยะ

DEEP LEARNING METHOD FOR SPAM FILTERING

ณัชชา ศรีวิเชียร
ณัฏฐณิชา วงศ์หมอ

ปัญหาพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตร
ปริญญาวิทยาศาสตรบัณฑิต (วิทยาการคอมพิวเตอร์)
ภาควิชาวิทยาการคอมพิวเตอร์ คณะวิทยาศาสตร์
สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง
ปีการศึกษา 2560

DEEP LEARNING METHOD FOR SPAM FILTERING

NATCHA SRIWICHIEN

NUTANICHA WONGMOR

A SPECIAL PROBLEM SUBMITTED IN PARTIAL FULFILLMENT OF

THE REQUIREMENT FOR

THE DEGREE OF BACHELOR OF SCIENCE (COMPUTER SCIENCE)

DEPARTMENT OF COMPUTER SCIENCE, FACULTY OF SCIENCE

KING MONGKUT'S INSTITUTE OF TECHNOLOGY LADKRABANG

ACADEMIC YEAR 2017

หัวข้อปัญหาพิเศษ	วิธีการเรียนรู้เชิงลึกสำหรับการคัดกรองข้อความขยะ Deep Learning Method For Spam Filtering
ชื่อนักศึกษา	นางสาวณัชชา..... ศรีวิเชียร รหัสนักศึกษา 57050218 นางสาวณัฏฐณิชา... วงศ์หมอ รหัสนักศึกษา 57050219
ปริญญา	วิทยาศาสตรบัณฑิต (วิทยาการคอมพิวเตอร์)
ภาควิชา	วิทยาการคอมพิวเตอร์
ปีการศึกษา	2560
อาจารย์ที่ปรึกษา	ดร.อัคเดช อุดมชัยพร

คณะวิทยาศาสตร์ สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง (สจล.) อนุมัติให้ปัญหาพิเศษนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรบัณฑิต (วิทยาการคอมพิวเตอร์) ประจำปีการศึกษา 2560

คณะกรรมการสอบ	**ลายมือชื่อ**
ดร.อินทราพร อรัณยะนาค ประธานกรรมการ	
ผศ.ดร.สายชล ใจเย็น กรรมการ	
ดร.อัคเดช อุดมชัยพร กรรมการและอาจารย์ที่ปรึกษา	

หัวข้อปัญหาพิเศษ	วิธีการเรียนรู้เชิงลึกสำหรับการคัดกรองข้อความขยะ
ชื่อนักศึกษา	นางสาวณัชชา..... ศรีวิเชียร รหัสนักศึกษา 57050218
	นางสาวณัฏฐณิชา... วงศ์หมอ รหัสนักศึกษา 57050219
ปริญญา	วิทยาศาสตรบัณฑิต (วิทยาการคอมพิวเตอร์)
ภาควิชา	วิทยาการคอมพิวเตอร์
คณะ	วิทยาศาสตร์
มหาวิทยาลัย	สถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบัง (สจล.)
ปีการศึกษา	2560
อาจารย์ที่ปรึกษา	ดร.อัคเดช อุดมชัยพร

บทคัดย่อ

การได้รับข้อความที่ไม่ต้องการ เช่น โฆษณาต่าง ๆ การฝากร้าน ฯลฯ ข้อความเหล่านั้นเรียกว่า ข้อความขยะหรือสแปม ปัญหาพิเศษนี้จึงนำเสนอวิธีการคัดกรองข้อความขยะโดยใช้วิธีการเรียนรู้เชิงลึกซึ่งเป็นส่วนหนึ่งของการเรียนรู้แบบเครื่องที่สามารถใช้คัดกรองข้อความขยะและข้อความที่ไม่ใช่ขยะจากชุดข้อมูลอีเมลและเอสเอ็มเอส โดยจะทำการเตรียมข้อความให้พร้อมสำหรับทดลองก่อนนำไปแบ่งข้อความออกเป็นสองกลุ่ม คือ ข้อความที่ไม่ใช่ข้อความขยะและข้อความขยะ จากนั้นจะทดลองจำแนกโดยใช้วิธี Bayesian Classification, K-Nearest Neighbors (K-NN), Artificial Neural Networks (ANNs), Support Vector Machine (SVM) และ วิธีการเรียนรู้เชิงลึก เพื่อนำไปใช้ศึกษา เปรียบเทียบวิเคราะห์และหาองค์ความรู้จากวิธีการคัดกรองข้อความขยะว่าวิธีการใดให้ผลลัพธ์ที่ดีที่สุด

คำสำคัญ : ข้อความขยะ, การคัดกรองข้อความขยะ, วิธีการเรียนรู้เชิงลึก

Title	Deep Learning Method For Spam Filtering		
Students	Miss Natcha	Sriwichien	Student ID 57050218
	Miss Nutanicha	Wongmor	Student ID 57050219
Degree	Bachelor of Science (Computer Science)		
Department	Computer Science		
Faculty	Science		
University	King Mongkut's Institute of Technology Ladkrabang (KMITL)		
Academic Year	2017		
Advisor	Dr.Akadej Udomchaiporn		

Abstract

This Special problem proposes Deep Learning method, one of machine learning approaches, for spam filtering. Spam is unwanted message which could be ads or junk e-mail. In this special problem, the experiments are conducted applying Recurrent Neural Network (RNN), one of Deep Learning algorithms, to two standard datasets : Email and SMS. The experiments also apply four traditional classification methods to the dataset in order to compare Deep learning and the traditional techniques in terms of both effectiveness and efficiency.

Keyword: Spam, Spam Filtering, Deep Learning Method, Bayesian Classification, K-Nearest Neighbors (K-NN), Artificial Neural Networks (ANNs), Support Vector Machine (SVM), Accuracy

กิตติกรรมประกาศ

ปัญหาพิเศษนี้สำเร็จลุล่วงไปได้ด้วยดี ทั้งนี้ทางคณะผู้จัดทำต้องขอขอบพระคุณอาจารย์ที่ปรึกษา ดร.อัคเดช อุดมชัยพร ที่ช่วยให้คำปรึกษาและคำแนะนำที่ดี แก่คณะผู้จัดทำในการปรับปรุงปัญหาพิเศษนี้

ขอขอบพระคุณอาจารย์ผู้ควบคุมการสอบปัญหาพิเศษ ดร.อินทราพร อรัณยะนาค ประธานกรรมการ และ ผศ.ดร.สายชล ใจเย็น กรรมการ ที่ให้คำแนะนำทำให้ปัญหาพิเศษนี้ มีความสมบูรณ์มากยิ่งขึ้น

ขอขอบพระคุณสถาบันเทคโนโลยีพระจอมเกล้าเจ้าคุณทหารลาดกระบังที่มอบโอกาสให้ได้เข้าศึกษาในสถาบันแห่งนี้ ทำให้ได้พบกับคณาจารย์และบุคลากรที่มีศักยภาพ ในการช่วยพัฒนาทักษะและมอบความรู้ให้แก่นักศึกษา

ณัชชา ศรีวิเชียร
ณัฏฐณิชา วงศ์หมอ

สารบัญ

หน้า

สารบัญ (ต่อ)

หน้า

สารบัญตาราง

สารบัญภาพ

สารบัญภาพ (ต่อ)

ภาพที่ **หน้า**

บทที่ 1

บทนำ

1.1 ความเป็นมาและความสำคัญของปัญหา

สแปม (Spam) คือ การส่งข้อความที่ผู้ใช้งานไม่ต้องการไปให้ผู้ใช้งานโดยไม่ได้รับอนุญาตจากผู้รับ การสแปมส่วนใหญ่ทำเพื่อการโฆษณาเชิงพาณิชย์ มักจะเป็นสินค้าที่น่าสงสัย หรือการเสนองานที่ทำให้รายได้อย่างรวดเร็ว หรือบริการที่ก้ำกึ่งผิดกฎหมาย การได้รับข้อความขยะจะทำให้เสียเวลา แบนด์วิดธ์ และพื้นที่ในการดาวน์โหลดข้อความเหล่านี้มาอ่าน ปัจจุบันยังไม่มีวิธีที่จะกำจัดข้อความขยะเหล่านี้ได้ถึงร้อยเปอร์เซ็นต์เพราะการคัดกรองข้อความขยะนั้นทำได้ยาก เนื่องจากข้อความขยะมักจะมีคำที่คลุมเครือในเนื้อหา และมีส่วนที่คล้ายกันกับข้อความที่ไม่ใช่ข้อความขยะหลายส่วน ดังนั้นปัญหาพิเศษนี้จึงเสนอวิธีการคัดกรองข้อความขยะ โดยมุ่งเน้นไปที่วิธีการเรียนรู้เชิงลึก เนื่องจากมีงานวิจัยหลายงานนำเสนอว่าเทคนิคที่ใช้ในการคัดกรองมีผลกับประสิทธิภาพในการกำจัดข้อความขยะ เช่น Bayesian Classification, K-Nearest Neighbors (K-NN), Artificial Neural Networks (ANNs) และ Support Vector Machine (SVM) ซึ่งผลลัพธ์จากการวัดประสิทธิภาพซึ่งเป็นผลิตผลที่ได้จากปัญหาพิเศษนี้ สามารถนำไปใช้วิเคราะห์ศึกษาเพื่อหาองค์ความรู้จากวิธีการคัดกรองข้อความขยะต่อไปได้

1.2 วัตถุประสงค์ของการดำเนินงาน

1) เพื่อศึกษาและทดลองวิธีการคัดกรองข้อความขยะโดยการใช้วิธีการเรียนรู้เชิงลึกและวิธีการดั้งเดิมแบบอื่น ๆ
2) เพื่อเปรียบเทียบผลลัพธ์ของการทดลองว่าวิธีการแบบใดให้ผลลัพธ์ที่ดีที่สุด
3) เพื่อศึกษาและวิเคราะห์ผลลัพธ์ที่ได้เพื่อหาองค์ความรู้ที่ได้จากการศึกษาและทดลองวิธีการคัดกรองข้อความขยะแบบต่าง ๆ

1.3 ขอบเขตการศึกษา

1) ปัญหาพิเศษนี้ จะทดลองโดยใช้วิธีการเรียนรู้เชิงลึกและวิธีการดั้งเดิมสี่แบบ คือ Bayesian Classification, K-Nearest Neighbors (K-NN), Artificial Neural Networks (ANNs) และ Support Vector Machine (SVM)

2) ชุดข้อมูลทดลองในปัญหาพิเศษนี้ มีชุดข้อมูลที่เกี่ยวกับอีเมล และ ชุดข้อมูลที่เกี่ยวกับเอสเอ็มเอส ดังนี้

- ชุดข้อมูลเกี่ยวกับอีเมล ชื่อ PU1 Corpus ประกอบไปด้วยชุดข้อมูลสี่กลุ่ม คือ bare, lemm, lemm_stop และ stop ข้อมูลในไฟล์เป็นรูปแบบของตัวเลข จำนวน 4,396 ไฟล์
- ชุดข้อมูลเกี่ยวกับเอสเอ็มเอส
  - ชุดข้อมูล ชื่อ NUS SMS Corpus ประกอบด้วยไฟล์หนึ่งไฟล์ ที่มีข้อมูลในไฟล์เป็นรูปแบบของตัวอักษรรวมทั้งอักขระพิเศษ จำนวนทั้งหมด 5,574 ข้อความ
  - ชุดข้อมูล ชื่อ SMS Corpus ที่มีข้อมูลในไฟล์เป็นรูปแบบของตัวอักษรรวมทั้งอักขระพิเศษ จำนวนทั้งหมด 1,324 ข้อความ

1.4 ประโยชน์ที่คาดว่าจะได้รับ

1) ได้ทราบว่าวิธีการคัดกรองข้อความขยะแบบต่าง ๆ มีประสิทธิภาพและประสิทธิผลต่างกันอย่างไร
2) ได้ทราบวิธีการที่ดีที่สุดจากการทดลองเมื่อใช้กับชุดข้อมูลแบบต่าง ๆ ที่นำมาทดสอบ
3) ได้องค์ความรู้ใหม่ ๆ เกี่ยวกับการจัดการข้อความขยะ

1.5 ขั้นตอนการดำเนินงาน

1) ศึกษาวิธีการคัดกรองข้อความขยะด้วยวิธีการเรียนรู้เชิงลึกและวิธีการอื่น ๆ ได้แก่ Bayesian Classification, K-Nearest Neighbors (K-NN), Artificial Neural Networks (ANNs) และ Support Vector Machine (SVM)
2) กำหนดและรวบรวมชุดข้อมูลทดสอบ
3) เตรียมพร้อมชุดข้อมูลทดสอบให้พร้อมสำหรับการคัดกรองข้อความขยะ
4) ทำการทดลองการคัดกรองข้อความขยะด้วยวิธีการต่าง ๆ
5) ประเมินผลและปรับปรุงผลการทดลอง
6) วิเคราะห์และสรุปผลการทดลอง
7) จัดทำเอกสารประกอบปัญหาพิเศษ

บทที่ 2

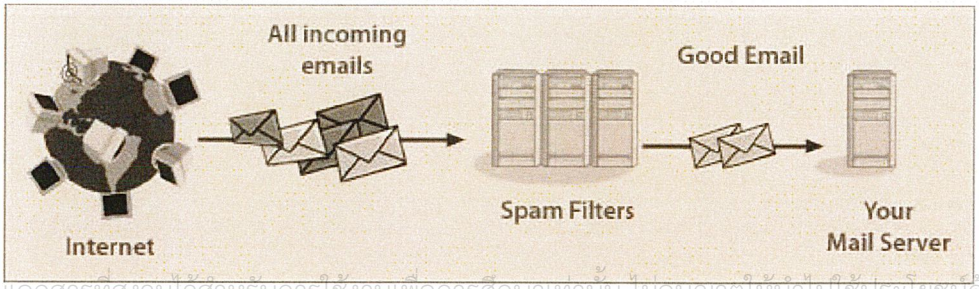
ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

ในการจัดทำปัญหาพิเศษ การคัดกรองข้อความขยะ ด้วยวิธีการเรียนรู้เชิงลึก (Deep Learning) เพื่อทำการทดลองและเปรียบเทียบผลลัพธ์ที่ดีที่สุดกับวิธีการดั้งเดิมแบบอื่น ๆ ผู้จัดทำจึงได้ศึกษาทฤษฎีและงานวิจัยที่เกี่ยวข้อง คือ Spam Filtering, การเรียนรู้เชิงลึก, โครงข่ายประสาทเทียมแบบวนซ้ำ, Bayesian Classification, K-Nearest Neighbors (K-NN), Artificial Neural Networks (ANNs), Support Vector Machine (SVM), งานวิจัยเรื่อง Machine Learning Techniques In Spam Filtering, ตัวอย่างชุดข้อมูลทดสอบ, การวัดประสิทธิภาพโดย Area Under ROC Curve (AUC) และ การวัดประสิทธิภาพโดย Accuracy โดยรายละเอียดของแต่ละหัวข้อจะกล่าวถึงในหัวข้อที่ 2.1 – 2.11 ดังต่อไปนี้

2.1 Spam Filtering

สแปม (Spam) คือ จดหมาย ข้อความ ข่าวสาร ข้อมูล ที่ถูกส่งมาโดยผู้รับไม่ได้ต้องการ ซึ่งมักจะพบเจอบ่อยตามอีเมลและข้อความเอสเอ็มเอส ซึ่งก่อให้เกิดความเสียเวลาและเปลืองพื้นที่ในการจัดเก็บ โดยสแปมที่พบเห็นบ่อยที่สุดจะเป็นข้อมูลเกี่ยวกับการโฆษณาสินค้าหรือโฆษณาเชิญชวนต่าง ๆ เป็นต้น

Spam Filter คือโปรแกรมที่มีการประมวลผลข้อความที่เข้ามาโดยอัตโนมัติ เพื่อช่วยป้องกันและคัดกรองข้อความที่ไม่พึงประสงค์ และการคัดกรองสแปมที่ดีควรเรียนรู้พฤติกรรมการใช้งานของผู้ใช้งาน เพื่อทำให้สามารถวิเคราะห์สแปมนั้นได้อย่างแม่นยำและเข้าใจความต้องการของผู้ใช้งานมากที่สุด กระบวนการคัดกรองสแปมแสดงดังภาพที่ 2.1



ภาพที่ 2.1 กระบวนการคัดกรองสแปม

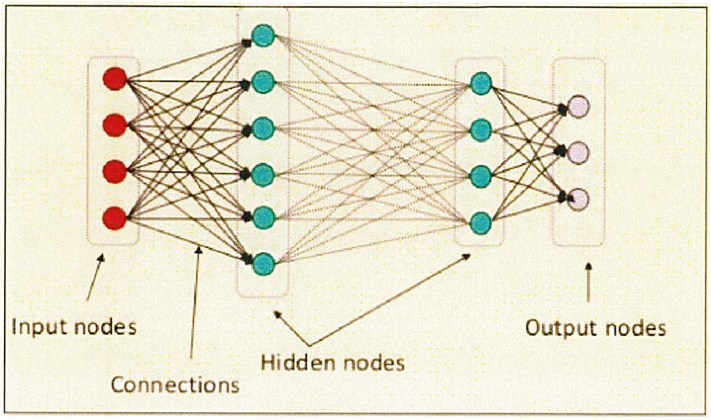
2.2 การเรียนรู้เชิงลึก (Deep Learning)

การเรียนรู้เชิงลึก มาจากแบบจำลองการเรียนรู้ของเครื่องชนิดโครงข่ายประสาทเทียม (Artificial Neural Network) ซึ่งเป็นรูปแบบขั้นสูงของปัญญาประดิษฐ์เพื่อช่วยในการกรองข้อมูล การเรียนรู้เชิงลึกเหมาะกับการประมวลผลข้อมูลที่มีจำนวนมาก ซึ่งเวลานำข้อมูลไปประมวลผล วิธีการเรียนรู้เชิงลึกจะทำการหาคุณลักษณะที่ดีที่สุดให้เลย โดยการทำงานนั้นจะมีความใกล้เคียงกับเครือข่ายประสาทสมองของมนุษย์และมักจะพยายามเลียนแบบมนุษย์ในด้านการประมวลผลของสมองที่สามารถรับรู้ผ่านทางประสาทสัมผัสได้

หลักการโดยทั่วไปของการเรียนรู้เชิงลึกคือการมีหน่วยประมวลผลหลาย ๆ ชั้น ข้อมูลขาเข้าในแต่ละชั้นได้มาจากการปฏิสัมพันธ์กับชั้นอื่น ๆ ทั้งนี้ การเรียนรู้เชิงลึกพยายามหาความสัมพันธ์ที่ซับซ้อนมากขึ้น นั่นคือ เมื่อมีจำนวนของชั้นและหน่วยประมวลผลที่อยู่ในชั้นมากขึ้น ข้อมูลในชั้นสูง ๆ ก็จะยิ่งซับซ้อนมากขึ้น

สถาปัตยกรรมโครงสร้างของการเรียนรู้เชิงลึกมักจะสร้างแบบเป็นชั้น ๆ (Layer-By-Layer) ด้วยขั้นตอนวิธีเชิงละโมบ (Greedy Method) ซึ่งการหาสิ่งที่ซับซ้อนมากขึ้นไปเรื่อย ๆ ในแต่ละชั้นนี้เองที่ทำให้การเรียนรู้เชิงลึกมีประสิทธิภาพมากกว่าวิธีการอื่น ๆ ตัวอย่างเช่น ข้อมูลในชั้นต้นอาจจะเรียนรู้ว่าภาพที่เข้ามาประกอบด้วยเส้นต่าง ๆ มาประกอบกันเป็นรูปสี่เหลี่ยม และชั้นต่อ ๆ มาคือการหาความสัมพันธ์ของเส้นสี่เหลี่ยมจนกระทั่งคอมพิวเตอร์รู้ได้ว่าภาพที่เข้ามาเป็นภาพของธงชาติ เป็นต้น

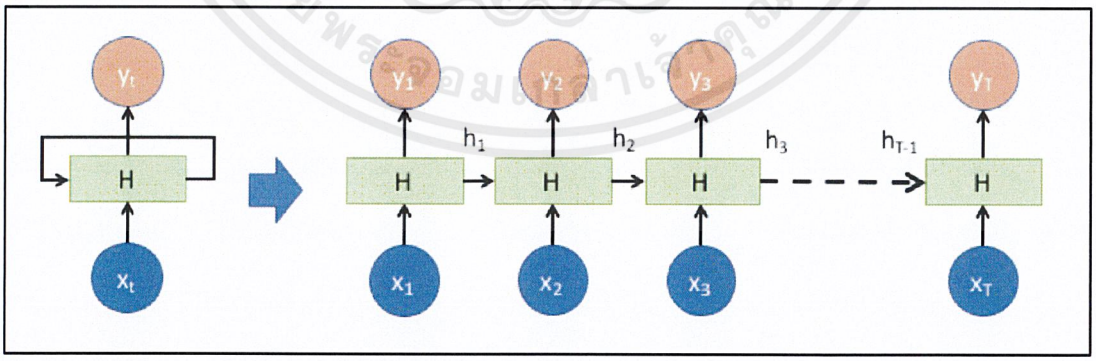
ในการเรียนรู้แบบมีผู้สอน (Supervised Learning) นั้น การเรียนรู้เชิงลึกจะช่วยลดภาระในการหาคุณลักษณะที่เกี่ยวข้อง เพราะวิธีการนี้จะแปลงข้อมูลไปสู่รูปแบบอื่นในระดับที่สูงขึ้นโดยอัตโนมัติ และให้ความสำคัญกับข้อมูลที่ซ้ำซ้อนลดลงไปด้วย นอกจากนี้ การเรียนรู้เชิงลึกยังสามารถนำไปปรับใช้กับการเรียนรู้แบบไม่มีผู้สอน (Unsupervised Learning) ได้ด้วย สถาปัตยกรรมโครงสร้างของการเรียนรู้เชิงลึกแสดงดังภาพที่ 2.2



ภาพที่ 2.2 สถาปัตยกรรมโครงสร้างของการเรียนรู้เชิงลึก

2.3 โครงข่ายประสาทเทียมแบบวนซ้ำ (Recurrent Neural Network)

การเอาข้อมูลบางส่วนที่ทำนายมาใช้ในการทำนายครั้งต่อไปและนำไปใช้ในการประมวลผลข้อมูลโดยจะเอาข้อมูลที่ส่งออกจากโหนดกลับมาเป็นข้อมูลนำเข้าใหม่ โดยเลือกประเภท Long Short-Term Memory (LSTM) ทำหน้าที่แก้ปัญหาลำดับเวลาที่มีความยาวมากของโครงข่ายประสาทเทียมแบบวนซ้ำ โดยสถานะของส่วนที่เล็กที่สุด (Cell State) เป็นตัวเก็บสถานะ (State) ของหน่วยความจำที่เล็กที่สุด (Memory Cell) ใน Long Short-Term Memory และ เกท (Gate) ซึ่งมีค่าอนาล็อก (Analog) เป็นตัวควบคุมการไหลของข้อมูล หลักการทำงานของโครงข่ายประสาทเทียมแบบวนซ้ำแสดงดังภาพที่ 2.3



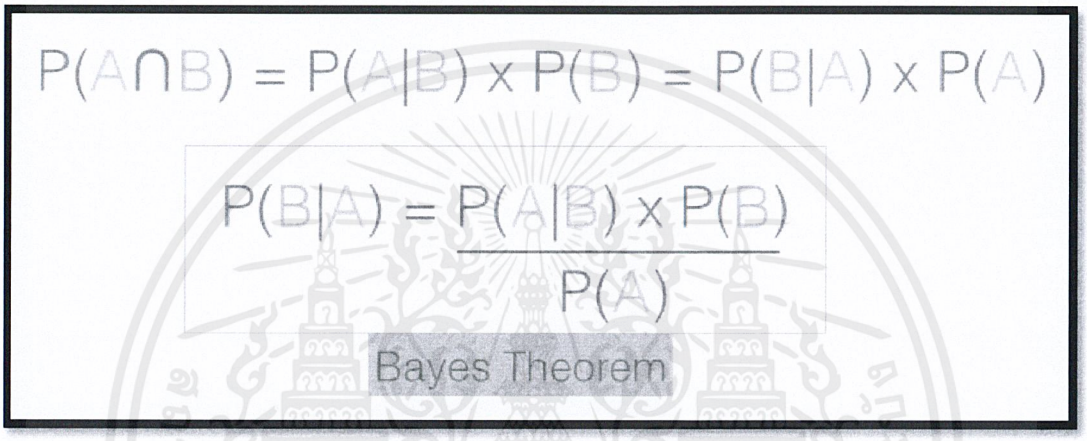
ภาพที่ 2.3 หลักการทำงานของโครงข่ายประสาทเทียมแบบวนซ้ำ

2.4 Bayesian Classification

ในหัวข้อนี้จะกล่าวถึงเรื่อง Bayesian และ Naïve Bayesian Classification ซึ่งรายละเอียดจะกล่าวถึงในหัวข้อที่ 2.4.1 และ 2.4.2

2.4.1 Bayesian

การเรียนรู้แบบเบย์ (Bayesian Learning) เป็นการจำแนกที่อาศัยหลักการของความน่าจะเป็นเข้ามาช่วยในการหาคำตอบของประเภทตัวอย่างใหม่ ซึ่งสมการทฤษฎีของเบย์ สามารถนำไปใช้งานทางด้าน Data Mining กำหนดให้ A คือ แอตทริบิวต์ (Attribute) และ C คือ ค่าคลาส (Class) สมการ Bayes Theorem แสดงดังภาพที่ 2.4 – 2.5

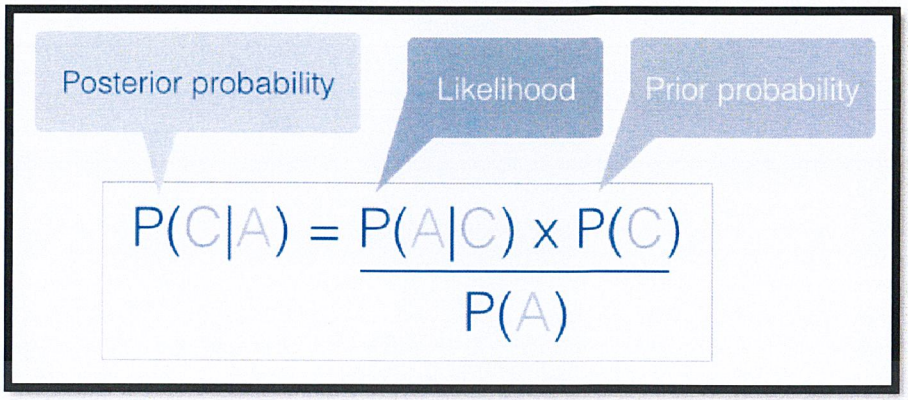


ภาพที่ 2.4 สมการ Bayes Theorem

จากสมการของ Bayes จะมีสามส่วนที่สำคัญ คือ

- Posterior Probability หรือ P(C|A) คือ ค่าความน่าจะเป็นที่ข้อมูลที่มีแอตทริบิวต์ A จะอยู่ในคลาส C
- Likelihood หรือ P(A|C) คือ ค่าความน่าจะเป็นที่ข้อมูล Training Data ที่อยู่ในคลาส C และมีแอตทริบิวต์ A โดยที่ A = a1 ∩ a2 ... ∩ aM โดยที่ M คือจำนวนแอตทริบิวต์ใน Training Data
- Prior Probability หรือ P(C) คือ ค่าความน่าจะเป็นของคลาส C

สมการความสัมพันธ์ของ Posterior Probability, Likelihood และ Prior Probability แสดงดังภาพที่ 2.5



ภาพที่ 2.5 สมการความสัมพันธ์ของ Posterior Probability, Likelihood และ Prior Probability

2.4.2 Naive Bayesian Classification

แต่การที่แอตทริบิวต์ A = a1 ⋂ a2 ... ⋂ aM ที่เกิดขึ้นใน Training Data อาจจะมีจำนวนน้อยมากหรือไม่มีรูปแบบของแอตทริบิวต์แบบนี้เกิดขึ้นเลย ดังนั้นจึงได้ใช้หลักการที่ว่าแต่ละแอตทริบิวต์เป็นอิสระต่อกันทำให้สามารถเปลี่ยนสมการ P(A|C) ได้เป็น

$$P(A|C) = P(a_1|C) \times P(a_2|C) \times \dots P(a_M|C)$$

โดยที่ A คือ แอตทริบิวต์ (Attribute) และ C คือ ค่าคลาส (Class)

2.5 K-Nearest Neighbors (K-NN)

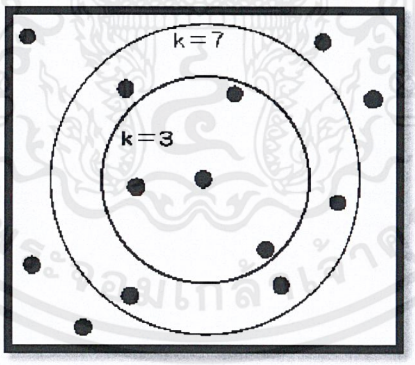
การเรียนรู้แบบ K-Nearest Neighbor (K-NN) เป็นขั้นตอนวิธีที่ใช้ในการจัดกลุ่มข้อมูล โดยการจัดข้อมูลที่อยู่ใกล้กันให้เป็นกลุ่มเดียวกันซึ่งเทคนิคนี้จะทำให้ตัดสินใจได้ว่า คลาสไหนที่จะแทนเงื่อนไขหรือกรณีใหม่ ๆ ได้บ้าง โดยการตรวจสอบจำนวน K ซึ่งถ้าหากเงื่อนไขของการตัดสินใจมีความซับซ้อน วิธีนี้สามารถสร้างโมเดลที่มีประสิทธิภาพได้ แต่ขั้นตอนวิธีการเพื่อนบ้านใกล้ที่สุดจะใช้ระยะเวลาในการคำนวณนานเมื่อข้อมูลมีความซับซ้อน อย่างเช่น ข้อมูลกราฟ หรือข้อมูลแบบลำดับเป็นต้น

การนำเทคนิคของขั้นตอนวิธีการเพื่อนบ้านใกล้ที่สุดไปใช้นั้น เป็นการหาระยะห่างระหว่างแต่ละแอตทริบิวต์ในข้อมูล จากนั้นก็คำนวณค่าออกมา ซึ่งวิธีนี้จะเหมาะสำหรับข้อมูลแบบตัวเลข แต่แอตทริบิวต์ที่เป็นค่าแบบไม่ต่อเนื่องนั้นก็สามารถทำได้ เพียงแต่ต้องการการจัดการแบบพิเศษเพิ่มขึ้น

อย่างเช่น ถ้าเป็นเรื่องของสี จะต้องกำหนดมาตรวัด วัดความแตกต่างระหว่างสีน้ำเงินกับสีเขียว ต่อจากนั้นต้องมีวิธีในการรวมค่าระยะห่างของ

แอตทริบิวต์ทุกค่าที่วัดมาได้ เมื่อสามารถคำนวณระยะห่างระหว่างเงื่อนไขหรือกรณีต่าง ๆ ได้ จึงเลือกชุดของเงื่อนไขที่ใช้จัดคลาสมาเป็นฐานสำหรับการจัดคลาสในเงื่อนไขใหม่ ๆ แล้วจึงจะตัดสินได้ว่าขอบเขตของจุดข้างเคียงที่ควรเป็นนั้น ควรมีขนาดใหญ่เท่าไร และอาจมีการตัดสินใจได้ด้วยว่าจะนับจำนวนจุดข้างเคียงได้อย่างไร โดยขั้นตอนวิธีการเพื่อนบ้านใกล้ที่สุดมีขั้นตอนโดยสรุป ดังนี้

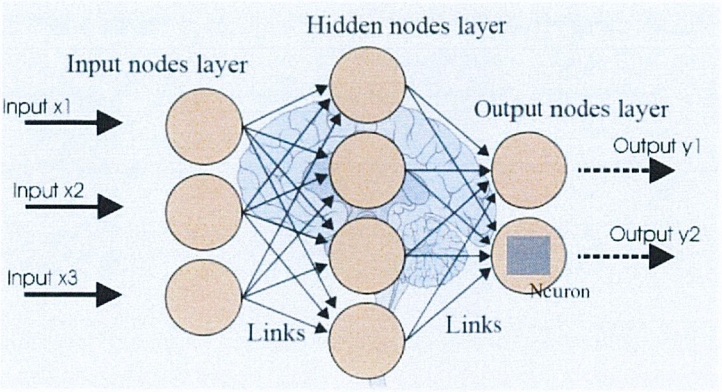
- กำหนดขนาดของ K (ควรกำหนดให้เป็นเลขคี่)
- คำนวณระยะห่าง (Distance) ของข้อมูลที่ต้องการพิจารณากับกลุ่มข้อมูลตัวอย่าง
- จัดเรียงลำดับของระยะห่าง และเลือกพิจารณาชุดข้อมูลที่ใกล้จุดที่ต้องการพิจารณาตามจำนวน K ที่กำหนดไว้
- พิจารณาข้อมูลจำนวน K ชุด และสังเกตว่ากลุ่มไหนที่ใกล้จุดที่พิจารณาเป็นจำนวนมากที่สุด กำหนดกลุ่มให้กับจุดที่พิจารณาที่ใกล้จุดพิจารณามากที่สุด ตัวอย่างการจัดข้อมูลที่อยู่ใกล้กันให้เป็นกลุ่มเดียวกันโดยวิธี K-Nearest Neighbor แสดงดังภาพที่ 2.6



ภาพที่ 2.6 ตัวอย่างการจัดข้อมูลที่อยู่ใกล้กันให้เป็นกลุ่มเดียวกัน โดยวิธี K-Nearest Neighbor (K-NN)

2.6 Artificial Neural Networks (ANNs)

การเรียนรู้แบบโครงข่ายประสาทเทียม (Artificial Neural Networks) เป็นโมเดลทางคณิตศาสตร์ สำหรับประมวลผลสารสนเทศด้วยการคำนวณแบบเชื่อมต่อเพื่อจำลองการทำงานของเครือข่ายประสาทในสมองมนุษย์ สถาปัตยกรรมโครงสร้างของการเรียนรู้แบบโครงข่ายประสาทเทียม แสดงดังภาพที่ 2.7

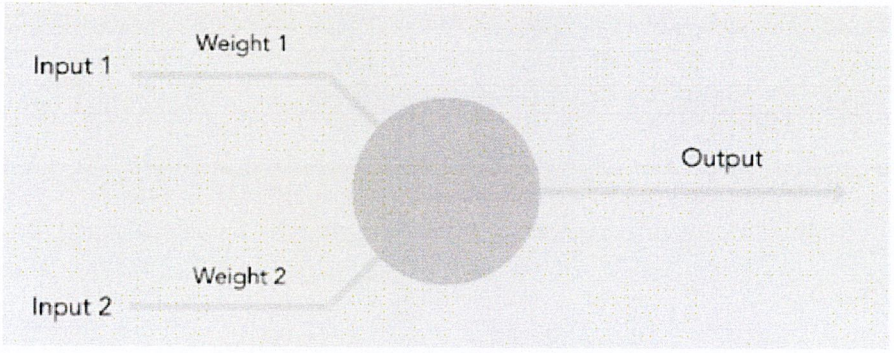


ภาพที่ 2.7 สถาปัตยกรรมโครงสร้างของการเรียนรู้แบบโครงข่ายประสาทเทียม

ในหัวข้อนี้จะกล่าวถึงเรื่อง Neuron, Perceptron และ Multilayer Perceptron ซึ่งรายละเอียดจะกล่าวถึงในหัวข้อที่ 2.6.1 - 2.6.3

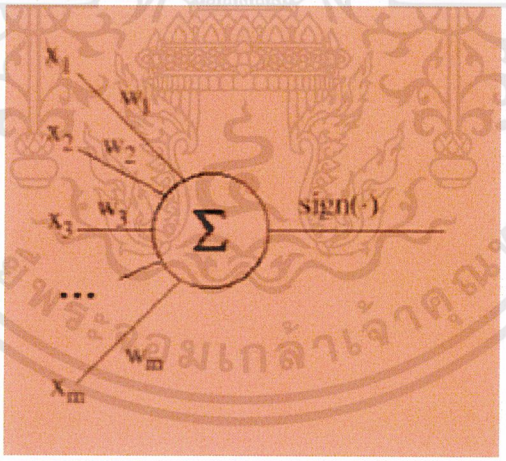
2.6.1 Neuron หรือ นิวรอนคือส่วนที่เล็กที่สุดของ Neural Network ซึ่งสังเกตได้จากภาพที่ 2.7 ซึ่งทำหน้าที่คำนวณ อินพุตที่เข้ามา เพื่อให้ได้ผลลัพธ์ออกไป กระบวนการทำงานของนิวรอนมีส่วนประกอบสำคัญดังนี้

- Input หรือค่าที่ส่งเข้ามาที่นิวรอนโดยจะมีขาที่เข้ามาได้หลายขา
- Weight เป็นการให้น้ำหนักของขาแต่ละขาที่ส่งเข้ามา โดยมีค่าระหว่าง 0 - 1 เมื่อเริ่มต้นจะเป็นการสุ่มขึ้นมา จากนั้นตัวนิวรอนเมื่อทำการเรียนรู้เรื่อย ๆ ก็จะปรับค่าน้ำหนักให้ได้คำตอบที่ใกล้เคียงที่สุด
- Bias คือค่าความโน้มเอียงที่จะช่วยเข้ามาทำให้ค่าที่เข้ามาอยู่ในระหว่าง 0 - 1 ได้ โดยจะเป็นเลขสุ่มและปรับไปเรื่อย ๆ ทุกครั้งที่เรียนรู้
- Output คือผลลัพธ์
- Back Propagation คือการที่นิวรอนนำค่าความผิดพลาดของเอาท์พุตที่ได้กับเอาท์พุตที่เราสั่งให้มันเรียนรู้ นำไปปรับค่าน้ำหนักและค่าความโน้มเอียงให้เกิดผลลัพธ์ที่ถูกต้องตามที่ได้เรียนรู้มา กระบวนการทำงานของนิวรอนแสดงดังภาพที่ 2.8



ภาพที่ 2.8 กระบวนการทำงานของนิวรอน

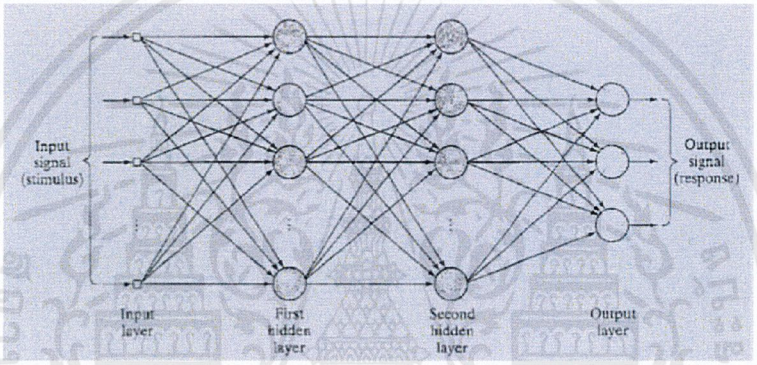
2.6.2 Perceptron หรือ เพอร์เซปตรอนประกอบด้วยเซลล์ประสาทเทียม ซึ่งอินพุตจะถูกส่งตรงไปยังเอาต์พุต โดยผ่านชุดค่าน้ำหนัก ถ้าค่าที่ได้จากการคำนวณนี้ มากกว่าขีดแบ่ง (Threshold) นิวรอนก็จะให้ค่าเอาต์พุตเท่ากับ 1 ถ้าน้อยกว่าก็จะให้ค่า -1 เพื่อให้ง่ายจึงกำหนดให้ขีดแบ่งเป็นค่าน้ำหนักตัวหนึ่งของอินพุตที่เป็นค่าคงที่ โดยฟังก์ชันค่าขีดแบ่งจะมีจุดศูนย์กลางอยู่ที่ 0 กระบวนการทำงานของเพอร์เซปตรอนแสดงดังภาพที่ 2.9



ภาพที่ 2.9 กระบวนการทำงานของเพอร์เซปตรอน

2.6.3 Multilayer Perceptron หรือ เพอร์เซปตรอนแบบหลายชั้นใช้สำหรับงานที่มีความซับซ้อนโดยใช้กระบวนการเรียนรู้แบบมีผู้สอนและใช้การส่งค่าย้อนกลับ (Back Propagation) สำหรับการเรียนรู้กระบวนการส่งค่าย้อนกลับ ประกอบด้วยสองส่วนย่อยคือ การส่งผ่านไปข้างหน้า (Forward Pass) และการส่งผ่านย้อนกลับ (Backward Pass) สำหรับการส่งผ่านไปข้างหน้า ข้อมูลจะผ่านเข้าโครงข่ายประสาทเทียมที่ชั้นข้อมูลเข้า และจะส่งผ่านจากชั้นหนึ่งไปสู่อีกชั้นหนึ่งจนกระทั่งถึง

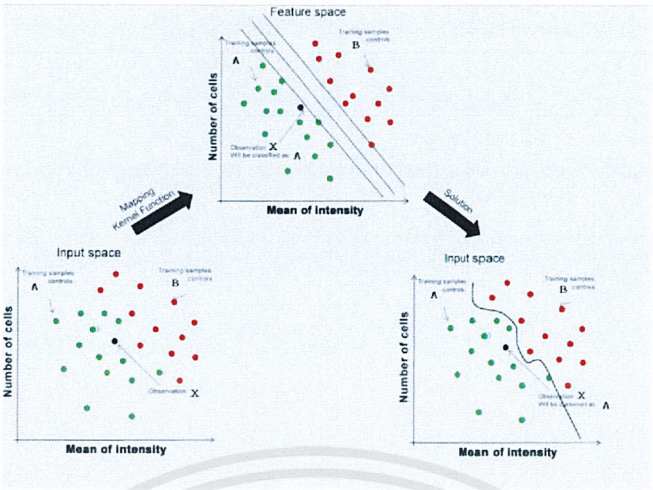
ชั้นข้อมูลออก ส่วนการส่งผ่านย้อนกลับ ค่าน้ำหนักการเชื่อมต่อจะถูกปรับเปลี่ยนให้สอดคล้องกับกฎการแก้ข้อผิดพลาด (Error-Correction) ซึ่งก็คือผลต่างของผลตอบที่แท้จริง (Actual Response) กับผลตอบเป้าหมาย (Target Response) ผลต่างนี้จะเกิดเป็นสัญญาณผิดพลาด (Error Signal) ซึ่งสัญญาณผิดพลาดนี้จะถูกส่งย้อนกลับเข้าสู่โครงข่ายประสาทเทียมในทิศทางตรงกันข้ามกับการเชื่อมต่อ และค่าน้ำหนักของการเชื่อมต่อจะถูกปรับจนกระทั่งผลตอบที่แท้จริงเข้าใกล้ผลตอบเป้าหมาย สถาปัตยกรรมโครงสร้างของการเรียนรู้โครงข่ายประสาทเทียมแบบเพอร์เซปตรอนแบบหลายชั้นแสดงดังภาพที่ 2.10



ภาพที่ 2.10 สถาปัตยกรรมโครงสร้างของการเรียนรู้โครงข่ายประสาทเทียมแบบเพอร์เซปตรอนแบบหลายชั้น

2.7 Support Vector Machine (SVM)

การเรียนรู้แบบ Support Vector Machine เป็นเทคนิคการเรียนรู้ด้วยเครื่องจักร ที่นิยมใช้เป็นอย่างมากเพราะเป็นขั้นตอนวิธีที่สามารถนำมาช่วยแก้ปัญหาการจำแนกข้อมูลโดยอาศัยหลักการของการหาสัมประสิทธิ์ของสมการเพื่อสร้างเส้นแบ่งแยกกลุ่มข้อมูลที่ถูกป้อนเข้าสู่กระบวนการสอนให้ระบบเรียนรู้ โดยเน้นไปยังเส้นแบ่งแยกแยะกลุ่มข้อมูลได้ดีที่สุด สำหรับแนวคิดนั้น เกิดจากการที่นำค่าของกลุ่มข้อมูลมาวางลงในฟีเจอร์สเปซ (Feature Space) จากนั้นจึงหาเส้นที่ใช้แบ่งข้อมูลทั้งสองออกจากกันโดยจะสร้างเส้นแบ่ง (Hyperplane) ที่เป็นเส้นตรงขึ้นมาเพื่อให้ทราบว่าเส้นตรงที่แบ่งสองกลุ่มออกจากกันนั้น เส้นตรงใดเป็นเส้นที่ดีที่สุด กระบวนการเรียนรู้แบบ Support Vector Machine แสดงดังภาพที่ 2.11



ภาพที่ 2.11 กระบวนการเรียนรู้แบบ Support Vector Machine (SVM)

2.8 งานวิจัย เรื่อง Machine Learning Techniques In Spam Filtering

จากบทความ เรื่อง Machine Learning Techniques In Spam Filtering เขียนโดย Konstantin Tretyakov เป็นงานวิจัยที่เกี่ยวข้องกับการคัดกรองข้อความขยะโดยใช้วิธีการสี่แบบ คือ Bayesian Classification, K-Nearest Neighbors (K-NN), Artificial Neural Networks (ANNs) และ Support Vector Machine (SVM) ซึ่งมีการกล่าวถึงผลการวัดประสิทธิภาพจากการทดสอบด้วยวิธีต่าง ๆ และมีผลลัพธ์ดังแสดงในตารางที่ 2.1 – 2.3

Algorithm	$N_{L \rightarrow S}$	$N_{S \rightarrow L}$	P	F_L	F_S	G
Naive Bayesian (λ=1)	0	138	87.4%	0.0%	28.7%	1.56
K-NN (k=51)	68	33	90.8%	11.0%	6.9%	1.61
Perceptron	8	8	98.5%	1.3%	1.7%	1.75
SVM	10	11	98.1%	1.6%	2.3%	1.74

ตารางที่ 2.1 แสดงผลลัพธ์การวัดประสิทธิภาพของการทดสอบด้วยวิธีต่าง ๆ

ตารางที่ 2.1 แสดงผลลัพธ์การวัดประสิทธิภาพของการทดสอบด้วยวิธีต่าง ๆ ซึ่งกล่าวได้ว่า ความสามารถของวิธีการเพอร์เซปตรอน ให้ประสิทธิภาพที่ดี และถ้าเทียบกับทางทฤษฎีแล้ว SVM

ควรจะมีผลลัพธ์ที่ดีกว่านี้ ในส่วนของวิธีการ Bayesian นั้นไม่ก่อให้เกิดค่าผิดพลาดเมื่อมีแอทริบิวต์จำนวนมาก จึงเป็นข้อดีของวิธีการแบบ Bayesian แต่ก็สามารถเกิดค่าผิดพลาดได้เมื่อมีแอททริบิวต์จำนวนน้อย ซึ่งถ้าต้องการลดค่าผิดพลาดสามารถทำได้โดยการเพิ่มพารามิเตอร์เข้าไปเพื่อให้จำนวนค่าความผิดพลาดลดลง แต่หากฟีเจอร์มีจำนวนมากการปรับค่าพารามิเตอร์นั้นก็ไม่ก่อให้เกิดผลใด ๆ และวิธีการ K-NN แสดงผลลัพธ์ให้เห็นว่าเกิดค่าความผิดพลาดจำนวนมาก

Algorithm	$N_L \longrightarrow S$	$N_S \longrightarrow L$	P	F_L	F_S	G
Naive Bayesian (λ=8)	0	140	87.3%	0.0%	29.1%	1.55
L/K-NN (k=51 , l=35)	0	337	69.3%	0.0%	70.0%	1.23
SVM Soft Margin (Cost = 0.3)	0	101	90.8%	0.0%	21.0%	1.61

ตารางที่ 2.2 แสดงผลลัพธ์การวัดประสิทธิภาพของ Naive Bayesian, K-NN และ SVM

ตารางที่ 2.2 แสดงผลลัพธ์การวัดประสิทธิภาพของ Naive Bayesian, K-NN และ SVM ซึ่งกล่าวได้ว่า ความสามารถของวิธี SVM ให้ประสิทธิภาพดีที่สุด และวิธีการของ Naive Bayesian, K-NN และ SVM นั้นไม่ก่อให้เกิดค่าผิดพลาดใด ๆ

Algorithm	$N_L \longrightarrow S$	$N_S \longrightarrow L$	P	F_L	F_S	G
N.B $\cup$ SVM s.m.	0	61	94.4%	0.0%	12.7%	1.68

ตารางที่ 2.3 แสดงผลลัพธ์การวัดประสิทธิภาพของ N.B $\cup$ SVM s.m.

ตารางที่ 2.3 แสดงผลลัพธ์การวัดประสิทธิภาพของ N.B $\cup$ SVM s.m. ซึ่งกล่าวได้ว่า เมื่อ Naive Bayesian $\cup$ SVM นั้นทำให้ประสิทธิภาพนั้นดีขึ้นมากและไม่ก่อให้เกิดค่าผิดพลาดใด ๆ

ดังนั้น จากการกล่าวในบทความช่วงแรกว่า ในการคัดกรองข้อความขยะไม่ควรจะทำให้เกิดค่าความผิดพลาด ซึ่งจากผลลัพธ์ที่แสดงในตารางมีเพียงวิธีการ Bayesian ที่สอดคล้องกับคำกล่าวนั้น

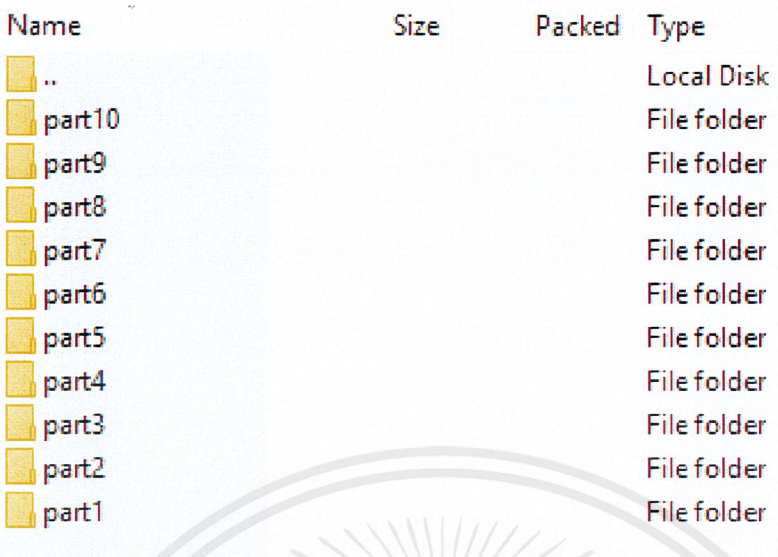
และในปัจจุบัน ยังไม่เคยมีการนำเสนอวิธีการเรียนรู้เชิงลึกสำหรับการคัดกรองข้อความขยะ นอกจากการใช้วิธีการแบบดั้งเดิม

2.9 ตัวอย่างชุดข้อมูลทดสอบ

ในหัวข้อนี้จะกล่าวถึงชุดข้อมูลทดสอบซึ่งประกอบด้วยสองชุดข้อมูลได้แก่ ชุดข้อมูลอีเมล และ ชุดข้อมูลเอสเอ็มเอส รายละเอียดของชุดข้อมูลทดสอบจะกล่าวถึงในหัวข้อที่ 2.9.1 – 2.9.2 ดังนี้

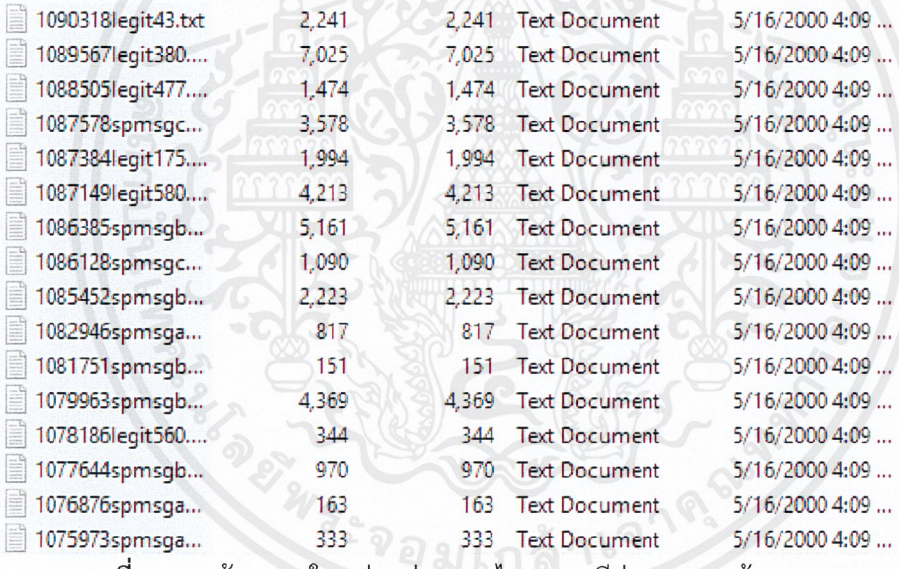
2.9.1 ชุดข้อมูลอีเมล: ใช้ชุดข้อมูลทดสอบ PU1 Corpus ประกอบไปด้วยไดเรกทอรีย่อยที่ถูกทำการแปลงรูปแบบของเนื้อหาแล้วสี่กลุ่ม คือ bare, lemm, lemm_stop, stop ข้อมูลในไฟล์เป็นรูปแบบของตัวเลข จำนวน 4,396 ไฟล์ โดยภาพที่ 2.12 แสดงกลุ่มของไดเรกทอรีย่อยสี่กลุ่มของชุดข้อมูล PU1 Corpus, ภาพที่ 2.13 แสดงไดเรกทอรีย่อยในแต่ละกลุ่มของชุดข้อมูล PU1 Corpus และภาพที่ 2.14 แสดงข้อความในแต่ละส่วนของไดเรกทอรีย่อยของชุดข้อมูล PU1 Corpus

ภาพที่ 2.12 กลุ่มของไดเรกทอรีย่อย 4 กลุ่มของชุดข้อมูล PU1 Corpus



Name	Size	Packed	Type
..			Local Disk
part10			File folder
part9			File folder
part8			File folder
part7			File folder
part6			File folder
part5			File folder
part4			File folder
part3			File folder
part2			File folder
part1			File folder

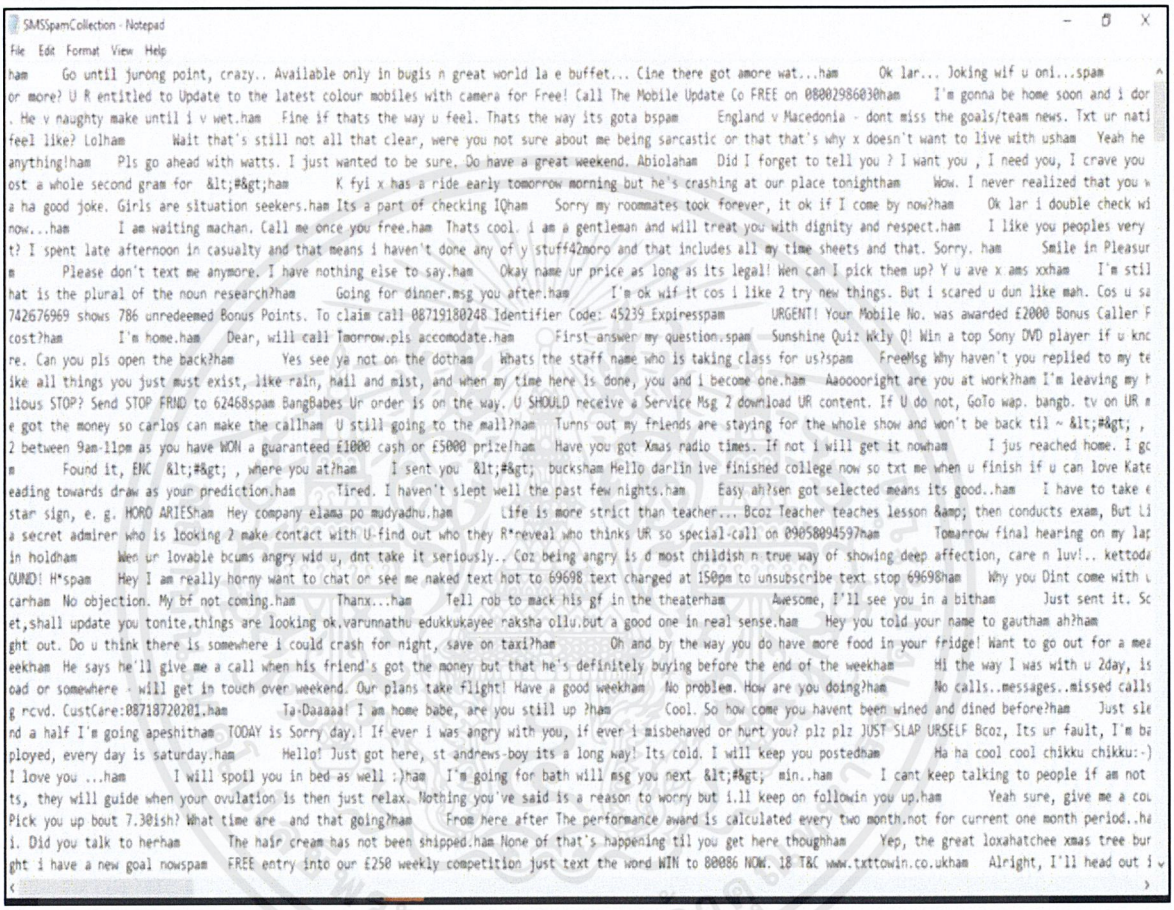
ภาพที่ 2.13 ไดเรกทอรีย่อยในแต่ละกลุ่มของชุดข้อมูล PU1 Corpus



1090318legit43.txt	2,241	2,241	Text Document	5/16/2000 4:09 ...
1089567legit380....	7,025	7,025	Text Document	5/16/2000 4:09 ...
1088505legit477....	1,474	1,474	Text Document	5/16/2000 4:09 ...
1087578spmsgc...	3,578	3,578	Text Document	5/16/2000 4:09 ...
1087384legit175....	1,994	1,994	Text Document	5/16/2000 4:09 ...
1087149legit580....	4,213	4,213	Text Document	5/16/2000 4:09 ...
1086385spmsgb...	5,161	5,161	Text Document	5/16/2000 4:09 ...
1086128spmsgc...	1,090	1,090	Text Document	5/16/2000 4:09 ...
1085452spmsgb...	2,223	2,223	Text Document	5/16/2000 4:09 ...
1082946spmsga...	817	817	Text Document	5/16/2000 4:09 ...
1081751spmsgb...	151	151	Text Document	5/16/2000 4:09 ...
1079963spmsgb...	4,369	4,369	Text Document	5/16/2000 4:09 ...
1078186legit560....	344	344	Text Document	5/16/2000 4:09 ...
1077644spmsgb...	970	970	Text Document	5/16/2000 4:09 ...
1076876spmsga...	163	163	Text Document	5/16/2000 4:09 ...
1075973spmsga...	333	333	Text Document	5/16/2000 4:09 ...

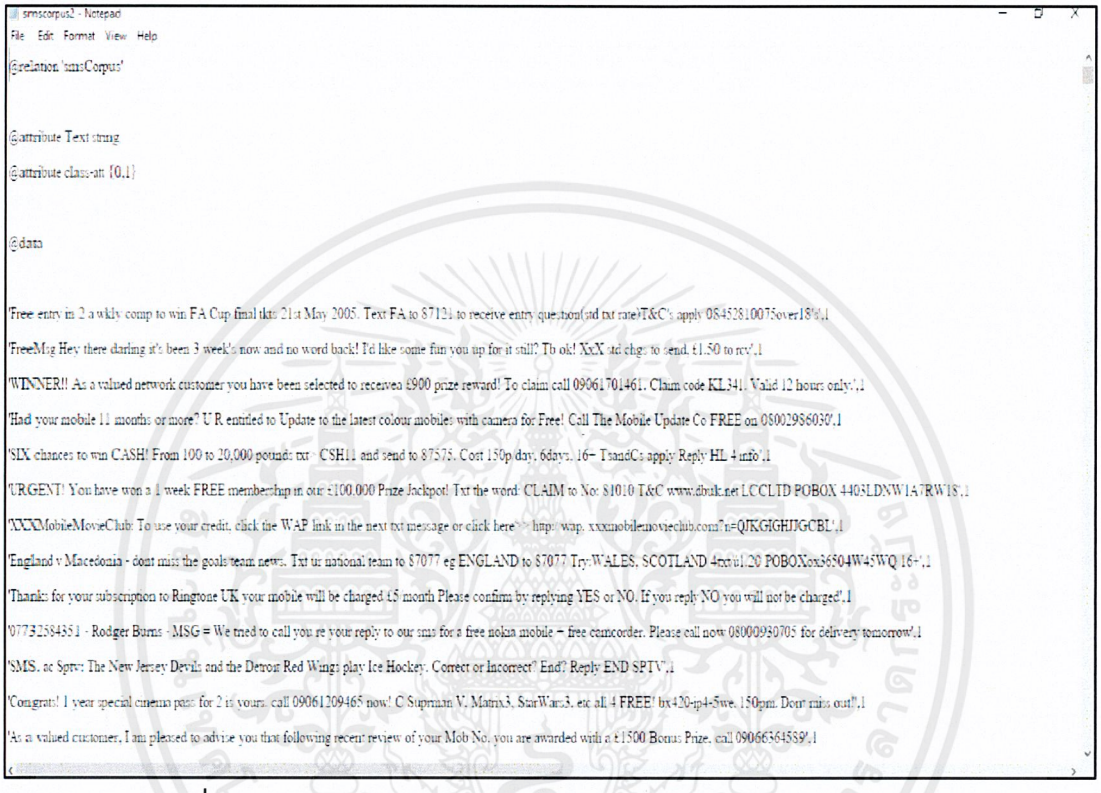
ภาพที่ 2.14 ข้อความในแต่ละส่วนของไดเรกทอรีย่อยของชุดข้อมูล PU1 Corpus

2.9.2 ชุดข้อมูลเอสเอ็มเอส: ใช้ชุดข้อมูลทดสอบ NUS SMS Corpus ประกอบด้วยไฟล์หนึ่งไฟล์ ที่มีข้อมูลเป็นรูปแบบของตัวอักษรรวมทั้งอักขระพิเศษ จำนวนทั้งหมด 5,574 ข้อความ โดยภาพที่ 2.15 แสดงตัวอย่างข้อความเอสเอ็มเอสในชุดข้อมูลทดสอบ NUS SMS Corpus



ภาพที่ 2.15 ตัวอย่างข้อความเอสเอ็มเอสในชุดข้อมูลทดสอบ NUS SMS Corpus

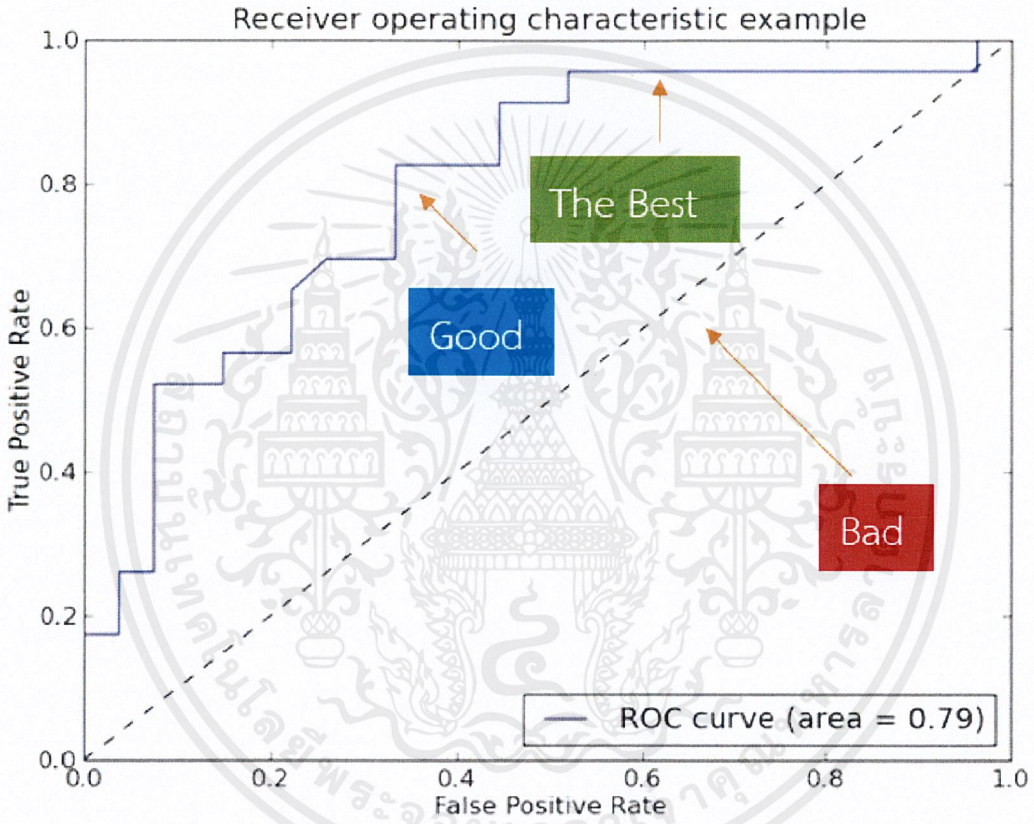
ชุดข้อมูลเอสเอ็มเอส: ใช้ชุดข้อมูลทดสอบ SMS Corpus ประกอบด้วยไฟล์หนึ่งไฟล์ที่มีข้อมูลเป็นรูปแบบของตัวอักษรรวมทั้งอักขระพิเศษ จำนวนทั้งหมด 1,324 ข้อความ โดยภาพที่ 2.16 แสดงตัวอย่างข้อความเอสเอ็มเอสในชุดข้อมูลทดสอบ SMS Corpus



ภาพที่ 2.16 ตัวอย่างข้อความเอสเอ็มเอสในชุดข้อมูลทดสอบ SMS Corpus

2.10 การวัดประสิทธิภาพโดย Area Under ROC Curve (AUC)

เป็นการวัดประสิทธิภาพของโมเดลโดย Area Under ROC Curve (AUC) นั้นใช้แสดงค่าพื้นที่ใต้กราฟและดูความสัมพันธ์ของกราฟ ค่า ROC curve (AUC) ยิ่งมีค่าเข้าใกล้ 1 จะยิ่งดีแสดงว่ามีประสิทธิภาพดีเนื่องจากมีค่า True Positive Rate เยอะ โดยภาพที่ 2.17 แสดงตัวอย่างความสัมพันธ์ของกราฟ Area Under ROC Curve (AUC)



ภาพที่ 2.17 ตัวอย่างความสัมพันธ์ของกราฟ Area Under ROC Curve (AUC)

2.11 การวัดประสิทธิภาพจาก Accuracy

เป็นการวัดประสิทธิภาพจำนวนข้อมูลที่ทำนายถูกของทุกคลาสโดยพิจารณาจากตาราง Confusion Matrix สำหรับการจำแนกข้อมูล ดังแสดงในตารางที่ 2.4

ค่าที่ทำนายได้ / ค่าจริง	A (+)	B (-)
A (+)	TP	FN
B (-)	FP	TN

ตารางที่ 2.4 Confusion Matrix สำหรับการจำแนกข้อมูล

โดยจากตารางแทนค่าได้ดังนี้

- ค่าจริงเป็น + ทำนาย เป็น + คือ TP (True Positive)
- ค่าจริงเป็น - ทำนาย เป็น + คือ FP (False Positive)
- ค่าจริงเป็น - ทำนาย เป็น – คือ TN (True Negative)
- ค่าจริงเป็น + ทำนาย เป็น – คือ FN (False Negative)

โดย Accuracy มีสูตรการคำนวนดังนี้

$$\text{Accuracy} = \frac{\mathbf{TP + TN}}{\mathbf{TP + FP + FN + TN}} \times 100\ \%$$

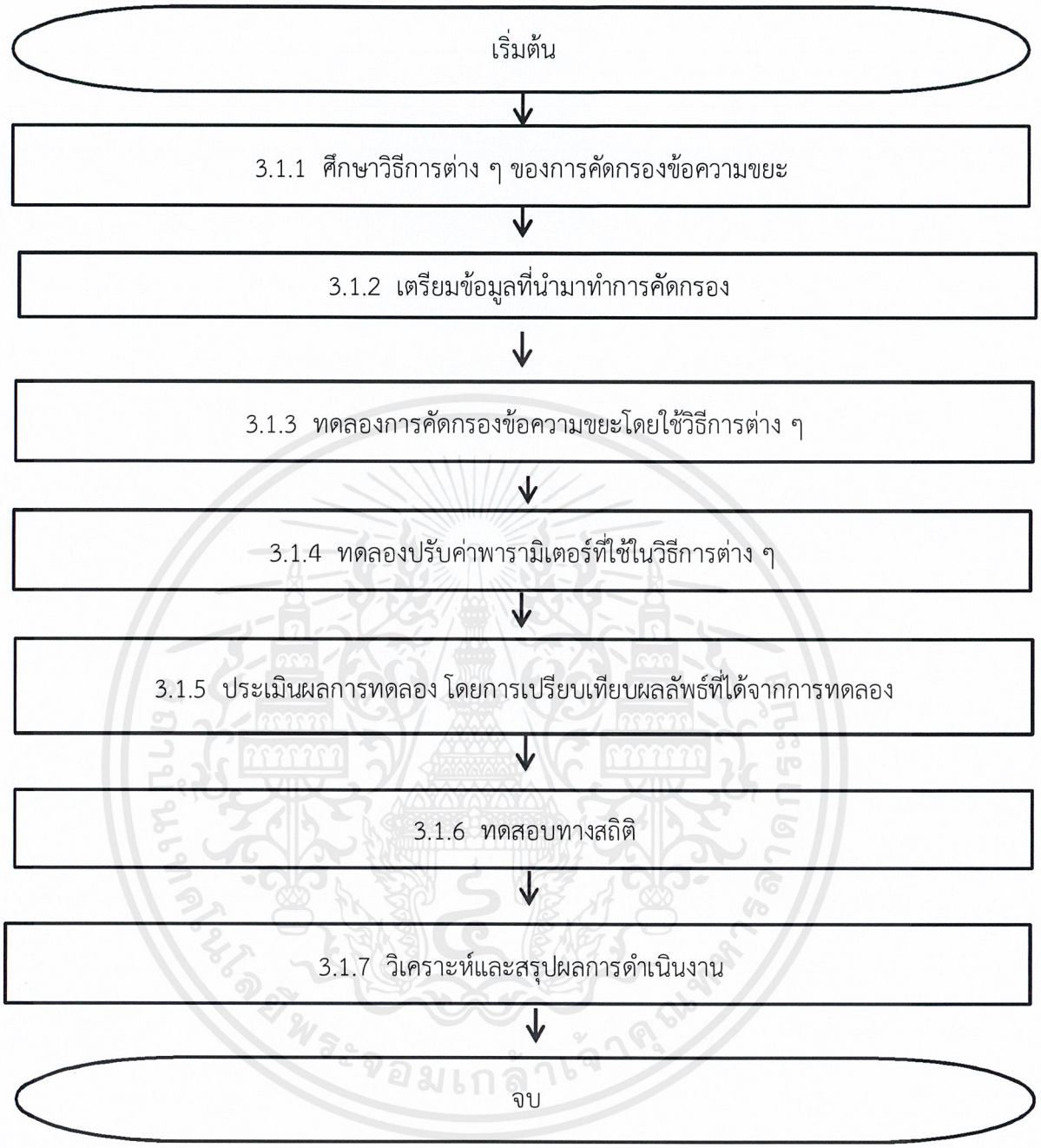
บทที่ 3

วิธีการดำเนินงาน

ในบทนี้จะกล่าวถึงการดำเนินงาน เรื่องการคัดกรองข้อความขยะโดยการใช้วิธีการเรียนรู้เชิงลึก (Deep Learning) และวิธีการดั้งเดิมแบบอื่น ๆ สี่แบบคือ Bayesian Classification, K-Nearest Neighbors (K-NN), Artificial Neural Networks (ANNs) และ Support Vector Machine (SVM) เนื้อหาในบทนี้ประกอบด้วยระเบียบวิธีการดำเนินงาน และขั้นตอนวิธีการคัดกรองข้อความขยะ โดยรายละเอียดของแต่ละหัวข้อได้ถูกอธิบายในหัวข้อที่ 3.1 – 3.2 ดังต่อไปนี้

3.1 ระเบียบวิธีการดำเนินงาน

การศึกษาวิธีการต่าง ๆ ของการคัดกรองข้อความขยะ มีระเบียบวิธีการดำเนินงาน แสดงดังภาพที่ 3.1



ภาพที่ 3.1 ระเบียบวิธีการดำเนินงานของการคัดกรองข้อความขยะ

3.1.1 ศึกษาวิธีการต่าง ๆ ของการคัดกรองข้อความขยะ

ศึกษาวิธีการคัดกรองข้อความขยะแบบต่าง ๆ ได้แก่ วิธีการเรียนรู้เชิงลึก, Bayesian Classification, K-Nearest Neighbors (K-NN), Artificial Neural Networks (ANNs) และ Support Vector Machine (SVM)

3.1.2 เตรียมข้อมูลที่นำมาทำการคัดกรอง

รวบรวมข้อมูลและทำการตัดอักขระพิเศษในข้อความ

3.1.3 ทดลองการคัดกรองข้อความขยะโดยใช้วิธีการต่าง ๆ

ทดลองทำการคัดกรองข้อความขยะตามวิธีการที่ได้ศึกษามา

3.1.4 ทดลองปรับค่าพารามิเตอร์ที่ใช้ในวิธีการต่าง ๆ

ทดลองปรับค่าพารามิเตอร์ของวิธีการคัดกรองข้อความขยะ เพื่อทดสอบว่าชุดของพารามิเตอร์ชุดใดให้ผลลัพธ์ที่ดีที่สุดสำหรับวิธีการแบบต่าง ๆ

3.1.5 ประเมินผลการทดลอง

เปรียบเทียบผลลัพธ์ที่ได้จากการทดลอง และผลลัพธ์ที่ได้จากงานวิจัยที่เกี่ยวข้อง

3.1.6 ทดสอบทางสถิติ

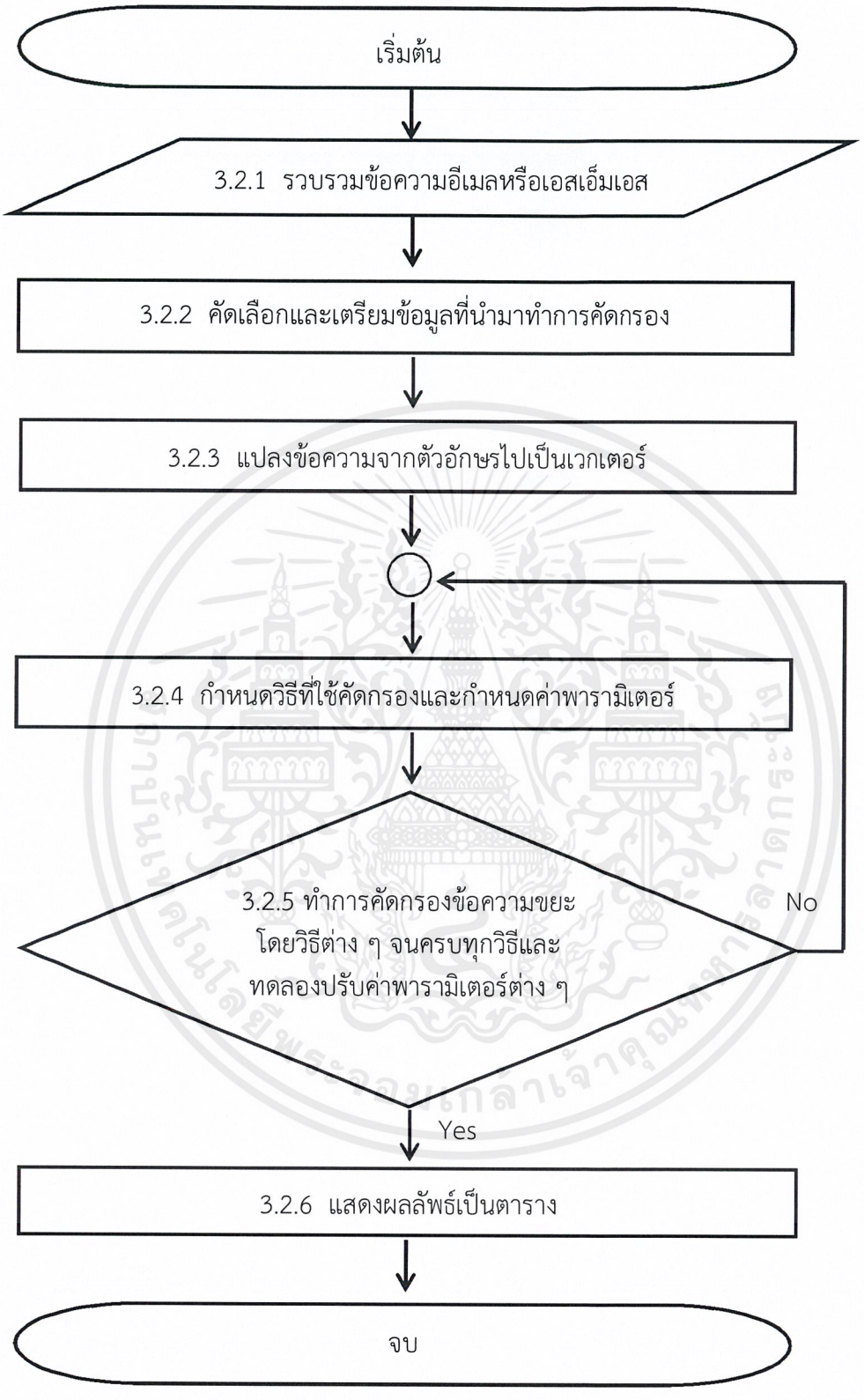
เปรียบเทียบผลลัพธ์ที่ได้จากการทดลอง และผลลัพธ์ที่ได้จากงานวิจัยที่เกี่ยวข้องด้วยวิธีการทางสถิติ เพื่อทดสอบว่ามีความแตกต่างกันอย่างมีนัยสำคัญหรือไม่

3.1.7 วิเคราะห์และสรุปผลการดำเนินงาน

นำผลลัพธ์ที่ได้ทั้งหมดจากการดำเนินงานมาวิเคราะห์ รวบรวม และสรุปผล เพื่อหาองค์ความรู้ที่ได้จากการทดลอง

3.2 ขั้นตอนวิธีการคัดกรองข้อความขยะโดยใช้วิธีการต่าง ๆ

ขั้นตอนการคัดกรองข้อความขยะโดยใช้วิธีการต่าง ๆ นั้น เริ่มจากการรับข้อความอีเมลหรือเอสเอ็มเอส แล้วนำไปคัดเลือกและเตรียมข้อมูลของส่วนที่จะนำมาทำการคัดกรอง เพื่อควบคุมคุณภาพของเนื้อหาที่ต้องการ และเตรียมความพร้อมสำหรับนำไปใช้ เมื่อเลือกและเตรียมข้อมูลส่วนที่จะทำการคัดกรองได้แล้วจะมีการ นำข้อความอักษรไปแปลงเป็นเวกเตอร์ เพื่อที่จะนำผลลัพธ์ที่ได้ไปประมวลผลในขั้นตอนต่อไป คือการกำหนดวิธีที่ใช้คัดกรองและค่าพารามิเตอร์เบื้องต้น เพื่อที่จะทำการคัดกรองข้อความขยะ โดยวิธีการต่าง ๆ และปรับพารามิเตอร์ เพื่อให้ได้ผลลัพธ์ที่ดีที่สุดต่อไป ขั้นตอนการทำงานของวิธีการคัดกรองข้อความขยะแสดงดังภาพที่ 3.2



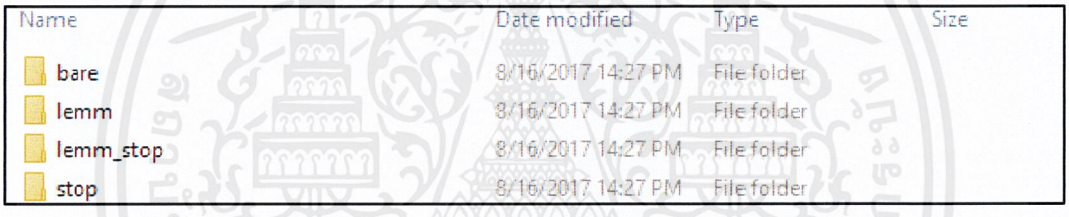
ภาพที่ 3.2 ขั้นตอนการทำงานของวิธีการคัดกรองข้อความขยะ

แต่ละขั้นตอนการทำงานของวิธีการคัดกรองข้อความขยะจะถูกกล่าวถึงในหัวข้อที่ 3.2.1 – 3.2.6 ดังต่อไปนี้

3.2.1 รวบรวมข้อความอีเมลหรือเอสเอ็มเอส

ในปัญหาพิเศษนี้จะทดสอบวิธีการที่นำเสนอกับชุดข้อมูลทดสอบสองชุด ได้แก่ ชุดข้อมูลอีเมลและชุดข้อมูลเอสเอ็มเอส โดยที่แต่ละชุดข้อมูลมีรายละเอียดดังต่อไปนี้

ชุดข้อมูลอีเมล จากการอ้างอิงจากงานวิจัยที่เกี่ยวข้อง เรื่อง Machine Learning Techniques In Spam Filtering โดยใช้ข้อมูลทดสอบ PU1 Corpus ประกอบด้วยไดเรกทอรีย่อยที่ถูกทำการแปลงรูปแบบของเนื้อหาแล้ว กลุ่มของไดเรกทอรีย่อยสี่กลุ่มของชุดข้อมูล PU1 Corpus แสดงดังภาพที่ 3.3

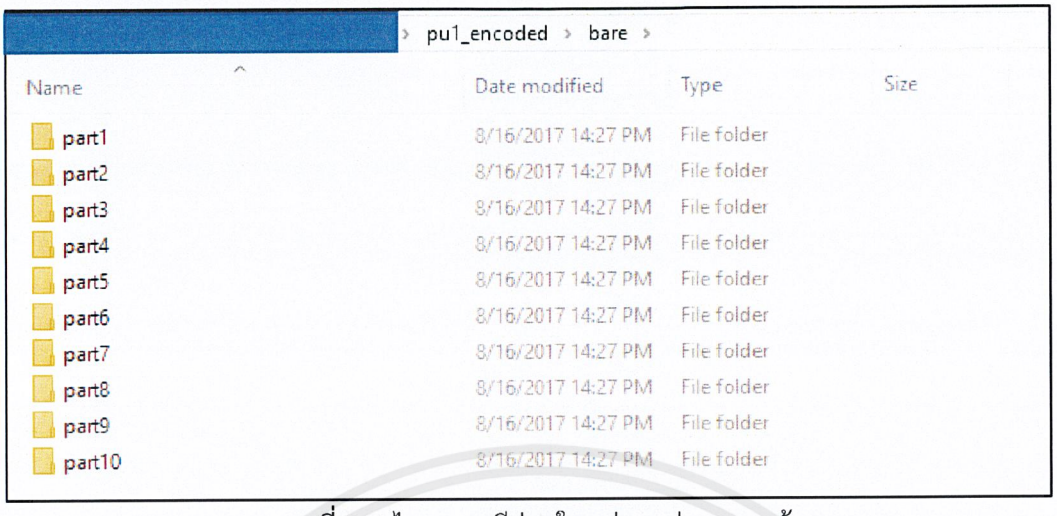


ภาพที่ 3.3 กลุ่มของไดเรกทอรีย่อยสี่กลุ่มของชุดข้อมูล PU1 Corpus

จากภาพที่ 3.3 สามารถกล่าวได้ว่าแต่ละกลุ่มมีความแตกต่างกันดังต่อไปนี้

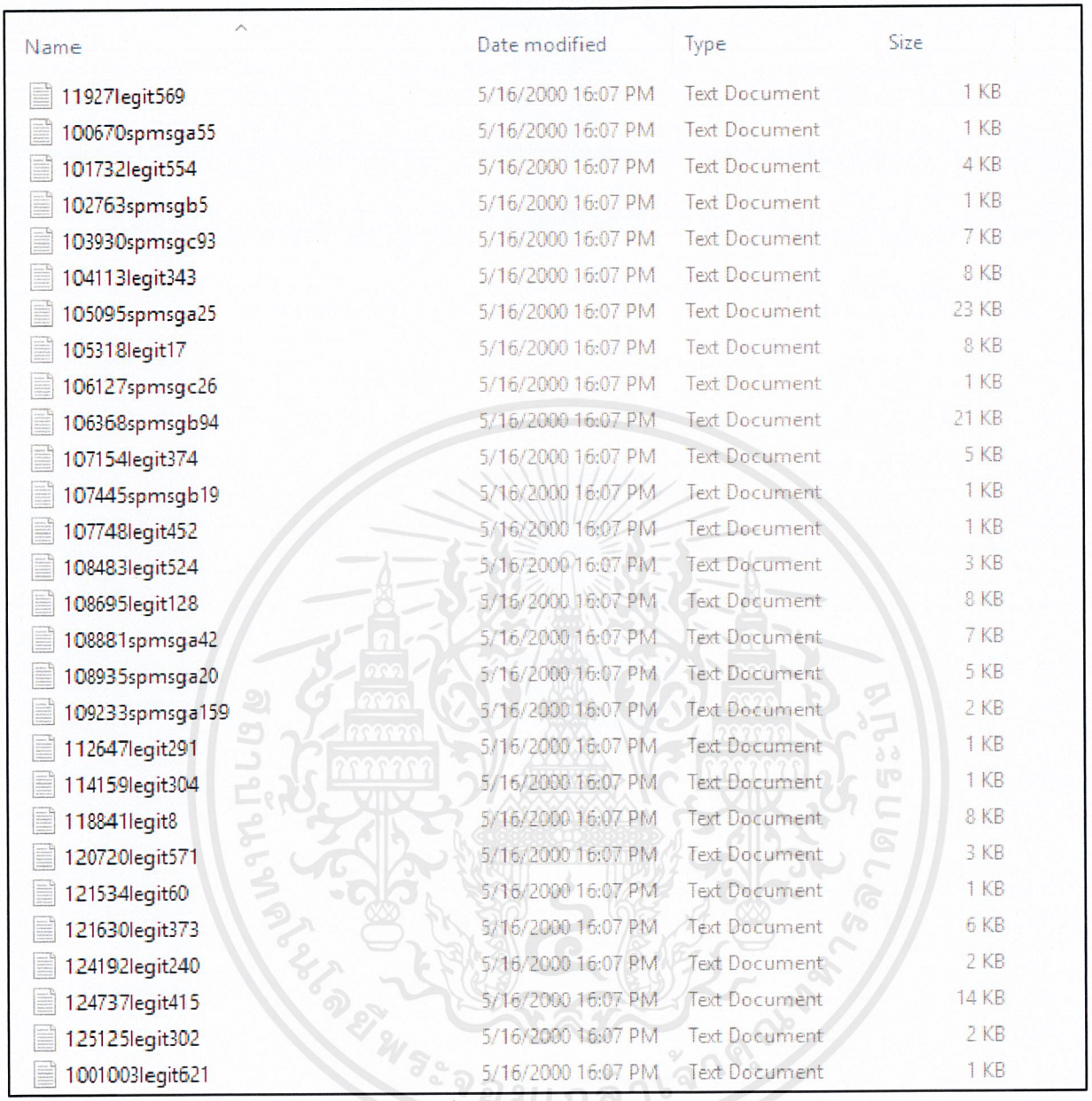
- bare: ไม่มีการใช้งาน Lemmatiser และ Stop-List
- lemm: มีการใช้งาน Lemmatiser แต่ไม่มีการ Stop-List
- lemm_stop: มีการใช้งาน Lemmatiser และ Stop-List
- stop: มีการใช้งาน Lemmatiser แต่ไม่มีการใช้งาน Stop-List

โดยที่ Lemmatiser คือ การตัดส่วนประกอบของคำให้อยู่ในรูปแบบรากศัพท์ (Stem) เช่น Connects, Connecting หรือ Connected จะถูกตัดเหลือแค่ Connect ส่วน Stop-List คือรายการคำที่ไม่สื่อความหมาย เช่น In, On, At เป็นต้น



ภาพที่ 3.4 ไดเรกทอรีย่อยในแต่ละกลุ่มของชุดข้อมูล

จากภาพที่ 3.4 ไดเร็กทอรีทั้งสี่กลุ่มประกอบด้วยไดเรกทอรีย่อยสิบส่วน (part1, ..., part10) ทั้งสิบส่วนย่อยถูกใช้ทำการทดลองสิบครั้ง โดยที่หนึ่งส่วนใช้สำหรับการทดสอบและอีกเก้าส่วนถูกใช้สำหรับการสร้างโมเดล แล้วทำซ้ำจนทุก ๆ ส่วนได้ถูกใช้สำหรับการทดสอบ (Ten-Fold Cross Validation)



Name	Date modified	Type	Size
11927legit569	5/16/2000 16:07 PM	Text Document	1 KB
100670spmsga55	5/16/2000 16:07 PM	Text Document	1 KB
101732legit554	5/16/2000 16:07 PM	Text Document	4 KB
102763spmsgb5	5/16/2000 16:07 PM	Text Document	1 KB
103930spmsgc93	5/16/2000 16:07 PM	Text Document	7 KB
104113legit343	5/16/2000 16:07 PM	Text Document	8 KB
105095spmsga25	5/16/2000 16:07 PM	Text Document	23 KB
105318legit17	5/16/2000 16:07 PM	Text Document	8 KB
106127spmsgc26	5/16/2000 16:07 PM	Text Document	1 KB
106368spmsgb94	5/16/2000 16:07 PM	Text Document	21 KB
107154legit374	5/16/2000 16:07 PM	Text Document	5 KB
107445spmsgb19	5/16/2000 16:07 PM	Text Document	1 KB
107748legit452	5/16/2000 16:07 PM	Text Document	1 KB
108483legit524	5/16/2000 16:07 PM	Text Document	3 KB
108695legit128	5/16/2000 16:07 PM	Text Document	8 KB
108881spmsga42	5/16/2000 16:07 PM	Text Document	7 KB
108935spmsga20	5/16/2000 16:07 PM	Text Document	5 KB
109233spmsga159	5/16/2000 16:07 PM	Text Document	2 KB
112647legit291	5/16/2000 16:07 PM	Text Document	1 KB
114159legit304	5/16/2000 16:07 PM	Text Document	1 KB
118841legit8	5/16/2000 16:07 PM	Text Document	8 KB
120720legit571	5/16/2000 16:07 PM	Text Document	3 KB
121534legit60	5/16/2000 16:07 PM	Text Document	1 KB
121630legit373	5/16/2000 16:07 PM	Text Document	6 KB
124192legit240	5/16/2000 16:07 PM	Text Document	2 KB
124737legit415	5/16/2000 16:07 PM	Text Document	14 KB
125125legit302	5/16/2000 16:07 PM	Text Document	2 KB
1001003legit621	5/16/2000 16:07 PM	Text Document	1 KB

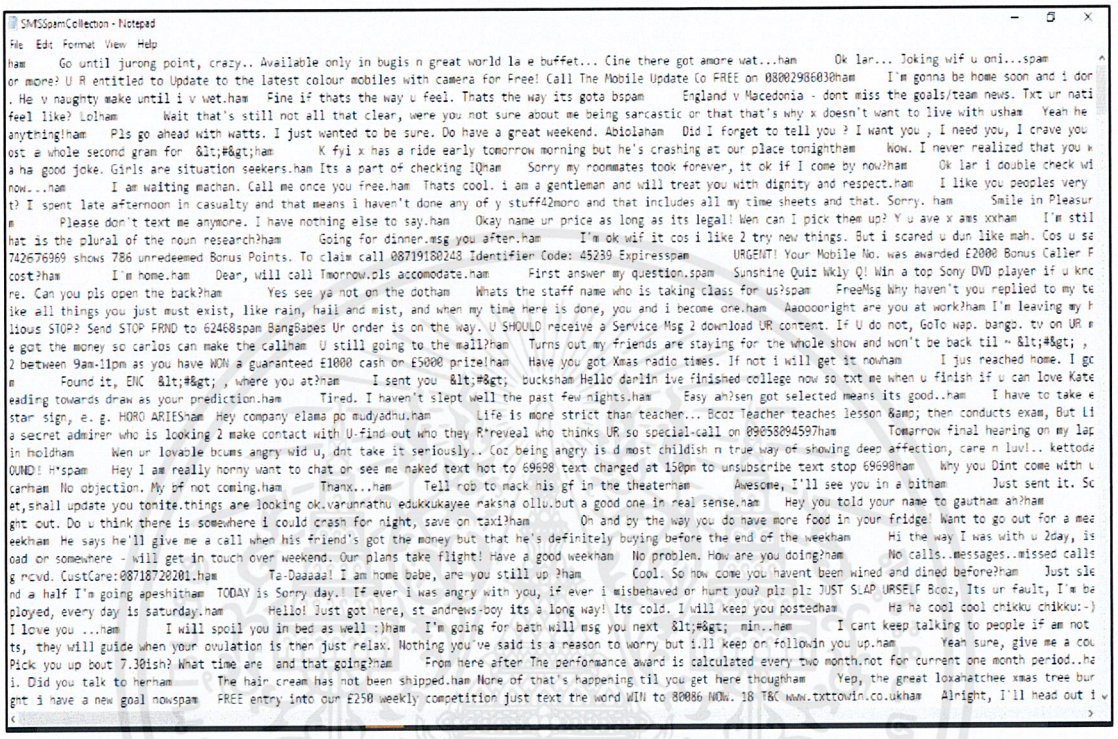
ภาพที่ 3.5 ข้อความที่อยู่ในไดเรกทอรีย่อยของชุดข้อมูล PU1 Corpus

ภาพที่ 3.5 แสดงข้อความที่อยู่ในไดเรกทอรีย่อยของชุดข้อมูล PU1 Corpus และสังเกตได้ว่าในแต่ละไดเรกทอรีย่อยทั้งสิบส่วน ประกอบด้วย ข้อความขยะและข้อความที่ไม่ใช่ข้อความขยะ ซึ่งสามารถบอกได้จากลักษณะดังต่อไปนี้

- ไฟล์ที่มีชื่อมีรูปแบบ * spmsg * .txt เป็นข้อความขยะ มีจำนวน 1,920 ข้อความ
- ไฟล์ที่มีชื่อมีรูปแบบ * legit * .txt เป็นข้อความที่ไม่ใช่ข้อความขยะ มีจำนวน 2,476 ข้อความ

ชุดข้อมูลเอสเอ็มเอส

- ใช้ข้อมูลทดสอบ NUS SMS Corpus ซึ่งตัวอย่างของข้อมูลเอสเอ็มเอสแสดงดังภาพที่ 3.6

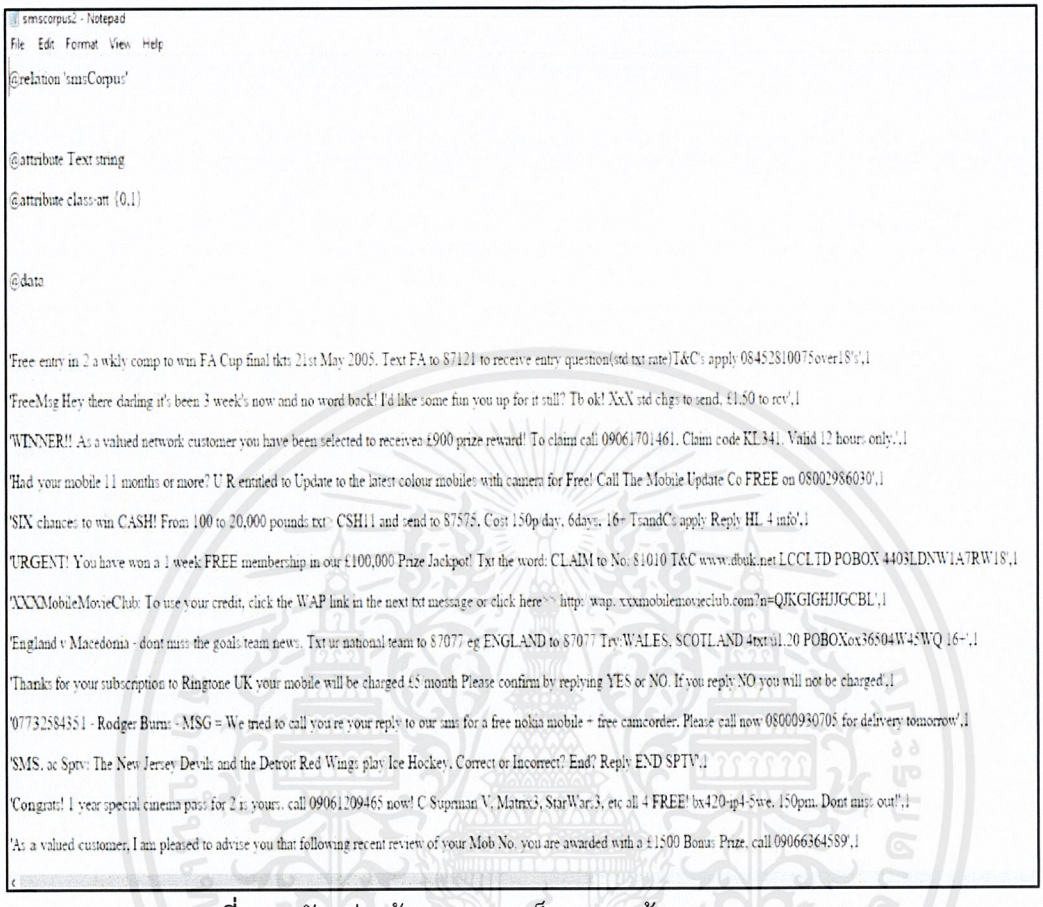


ภาพที่ 3.6 ตัวอย่างข้อความเอสเอ็มเอสชุดข้อมูลทดสอบ NUS SMS Corpus

จากภาพที่ 3.6 ข้อมูลทดสอบของชุดข้อมูล NUS SMS Corpus ประกอบด้วย ข้อความขยะและข้อความที่ไม่ใช่ข้อความขยะ ซึ่งสามารถบอกได้จากลักษณะดังต่อไปนี้

- ข้อความที่มีรูปแบบ “Spam .....” เป็นข้อความขยะ มีจำนวน 747 ข้อความ
- ข้อความที่มีรูปแบบ “Ham .....” เป็นข้อความที่ไม่ใช่ข้อความขยะ มีจำนวน 4,827 ข้อความ

- ใช้ข้อมูลทดสอบ SMS Corpus ซึ่งตัวอย่างของข้อมูลเอสเอ็มเอสแสดงดังภาพที่ 3.7



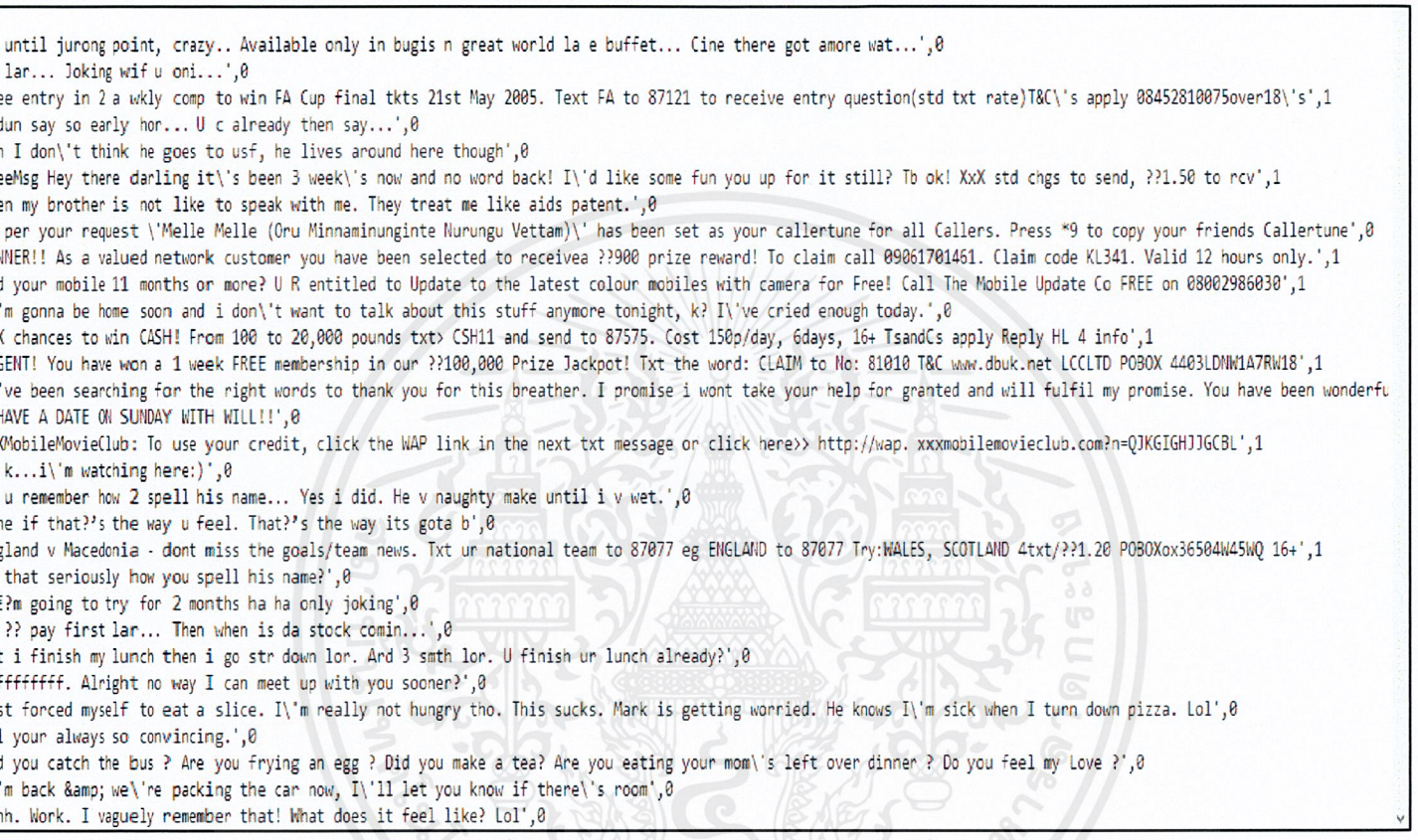
ภาพที่ 3.7 ตัวอย่างข้อความเอสเอ็มเอสชุดข้อมูลทดสอบ SMS Corpus

จากภาพที่ 3.7 ข้อมูลทดสอบของชุดข้อมูล SMS Corpus ประกอบด้วย ข้อความขยะและข้อความที่ไม่ใช่ข้อความขยะ ซึ่งสามารถบอกได้จากลักษณะดังต่อไปนี้

- ข้อความที่มีรูปแบบ “Spam .....” เป็นข้อความขยะ มีจำนวน 1,353 ข้อความ
- ข้อความที่มีรูปแบบ “Ham .....” เป็นข้อความที่ไม่ใช่ข้อความขยะ มีจำนวน 0 ข้อความ

3.2.2 คัดเลือกและเตรียมข้อมูลที่นำมาทำการคัดกรอง

ในขั้นตอนนี้จะทำการคัดเลือกและเตรียมข้อมูลที่จะนำมาทำการคัดกรองให้อยู่ในรูปแบบของไฟล์ Text ตัวอย่างของข้อความที่นำมาคัดกรองในรูปแบบไฟล์ Text แสดงดังภาพที่ 3.8



```
 until jurong point, crazy.. Available only in bugis n great world la e buffet... Cine there got amore wat...',0
 lar... Joking wif u oni...',0
ee entry in 2 a wkly comp to win FA Cup final tkts 21st May 2005. Text FA to 87121 to receive entry question(std txt rate)T&C\'s apply 08452810075over18\'s',1
dun say so early hor... U c already then say...',0
 I don\'t think he goes to usf, he lives around here though',0
eeMsg Hey there darling it\'s been 3 week\'s now and no word back! I\'d like some fun you up for it still? Tb ok! XxX std chgs to send, ??1.50 to rcv',1
en my brother is not like to speak with me. They treat me like aids patent.',0
 per your request \'Melle Melle (Oru Minnaminunginte Nurungu Vettam)\' has been set as your callertune for all Callers. Press *9 to copy your friends Callertune',0
NNER!! As a valued network customer you have been selected to receivea ??900 prize reward! To claim call 09061701461. Claim code KL341. Valid 12 hours only.',1
d your mobile 11 months or more? U R entitled to Update to the latest colour mobiles with camera for Free! Call The Mobile Update Co FREE on 08002986030',1
'm gonna be home soon and i don\'t want to talk about this stuff anymore tonight, k? I\'ve cried enough today.',0
X chances to win CASH! From 100 to 20,000 pounds txt> CSH11 and send to 87575. Cost 150p/day, 6days, 16+ TsandCs apply Reply HL 4 info',1
GENT! You have won a 1 week FREE membership in our ??100,000 Prize Jackpot! Txt the word: CLAIM to No: 81010 T&C www.dbuk.net LCCLTD POBOX 4403LDNW1A7RW18',1
've been searching for the right words to thank you for this breather. I promise i wont take your help for granted and will fulfil my promise. You have been wonderfu
HAVE A DATE ON SUNDAY WITH WILL!!',0
XMobileMovieClub: To use your credit, click the WAP link in the next txt message or click here>> http://wap. xxxmobilemovieclub.com?n=QJKGIGHJJGCBL',1
 k...i\'m watching here:)',0
 u remember how 2 spell his name... Yes i did. He v naughty make until i v wet.',0
he if that?'s the way u feel. That?'s the way its gota b',0
gland v Macedonia - dont miss the goals/team news. Txt ur national team to 87077 eg ENGLAND to 87077 Try:WALES, SCOTLAND 4txt/??1.20 POBOXox36504W45WQ 16+',1
 that seriously how you spell his name?',0
?m going to try for 2 months ha ha only joking',0
 ?? pay first lar... Then when is da stock comin...',0
 i finish my lunch then i go str down lor. Ard 3 smth lor. U finish ur lunch already?',0
ffffffff. Alright no way I can meet up with you sooner?',0
st forced myself to eat a slice. I\'m really not hungry tho. This sucks. Mark is getting worried. He knows I\'m sick when I turn down pizza. Lol',0
l your always so convincing.',0
d you catch the bus ? Are you frying an egg ? Did you make a tea? Are you eating your mom\'s left over dinner ? Do you feel my Love ?',0
'm back &amp;amp; we\'re packing the car now, I\'ll let you know if there\'s room',0
hh. Work. I vaguely remember that! What does it feel like? Lol',0
```

ภาพที่ 3.8 ข้อความที่นำมาคัดกรองในรูปแบบไฟล์ Text

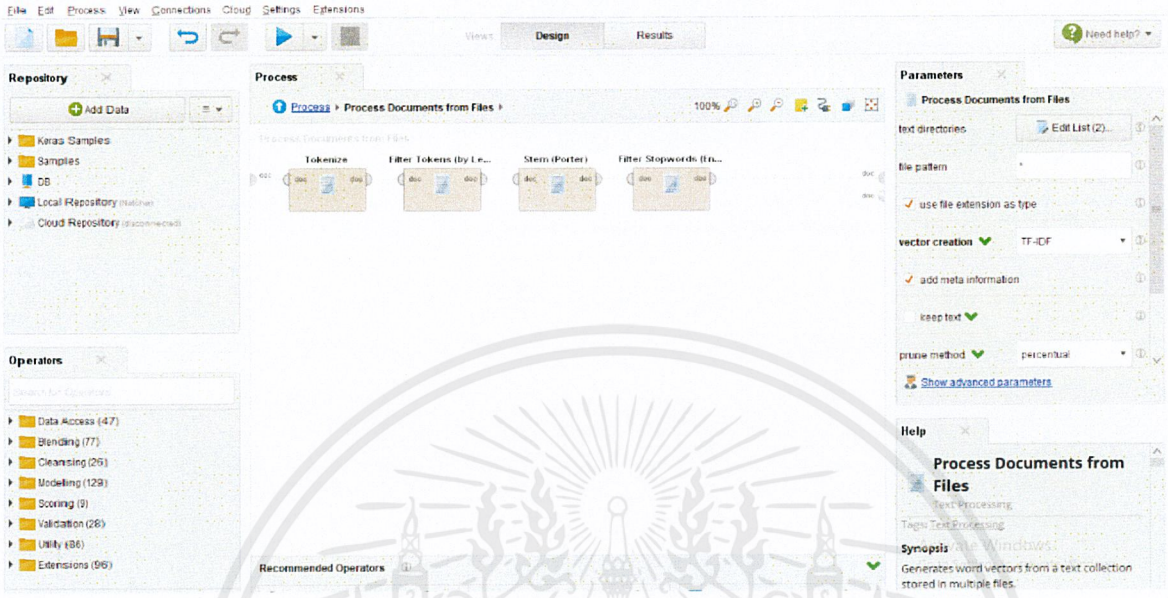
จากภาพที่ 3.8 ไฟล์ Text เป็นไฟล์ที่โปรแกรม RapidMiner สามารถประยุกต์ใช้ในการประมวลผลได้โดยภายในไฟล์มีการแบ่งประเภทข้อมูลเป็นสองส่วน ดังต่อไปนี้

- 1 หมายถึง ข้อความขยะ
- 0 หมายถึง ข้อความที่ไม่ใช่ข้อความขยะ

ซึ่งข้อมูลในแต่ละแอตทริบิวต์ที่ต้องการใช้ในการวิเคราะห์ ซึ่งต้องทำการระบุว่าข้อความที่นำมาใช้ในการวิเคราะห์เป็นข้อความขยะหรือไม่ใช่ข้อความขยะ ซึ่งสามารถพิจารณาได้ดังนี้

- ข้อความที่มีรูปแบบ “....., 1” เป็นข้อความขยะ
- ข้อความที่มีรูปแบบ “....., 0” เป็นข้อความที่ไม่ใช่ข้อความขยะ

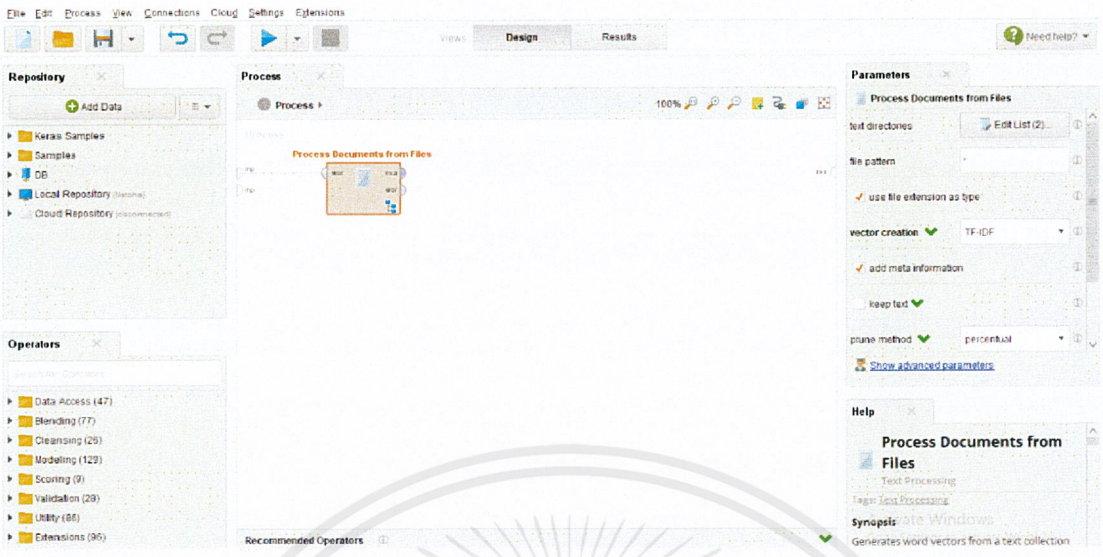
หน้าจอโปรแกรม RapidMiner เมื่อมีการอ่านไฟล์รูปแบบ Text และคำนวณค่า Term representation แสดงดังภาพที่ 3.9



ภาพที่ 3.9 หน้าจอโปรแกรม RapidMiner เมื่อมีการอ่านไฟล์รูปแบบ Text

3.2.3 แปลงข้อความจากตัวอักษรไปเป็นเวกเตอร์

การแปลงข้อความจากตัวอักษรไปเป็นเวกเตอร์ สามารถทำได้โดยใช้โปรแกรม RapidMiner หน้าจอโปรแกรม RapidMiner เมื่อต้องการแปลงข้อความจากตัวอักษรไปเป็นเวกเตอร์และ หน้าจอโปรแกรม RapidMiner หลังจากแปลงข้อความไปเป็นเวกเตอร์แสดงดังภาพที่ 3.10 - 3.13



ภาพที่ 3.10 หน้าจอโปรแกรม RapidMiner เมื่อต้องการแปลงข้อความจากตัวอักษรไปเป็นเวกเตอร์

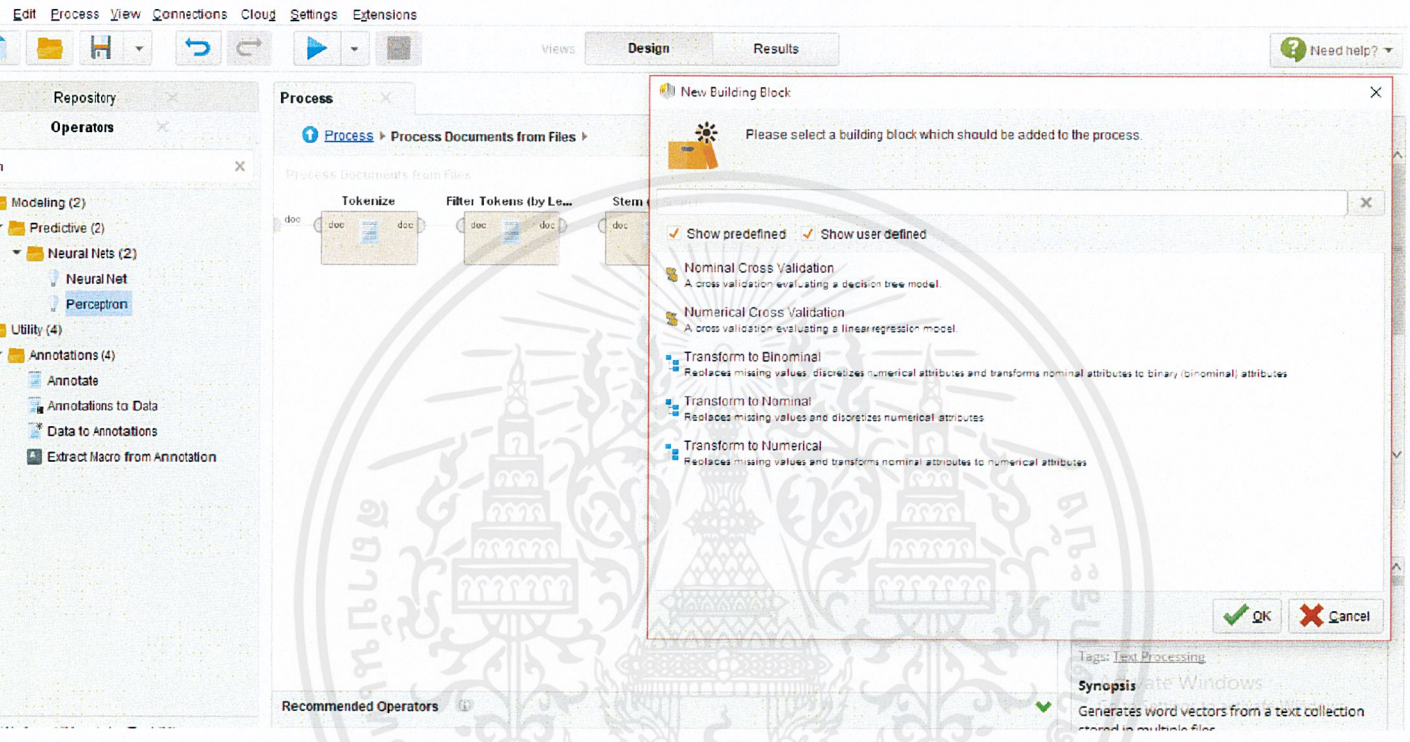
จากภาพที่ 3.10 สามารถอธิบายว่าโปรแกรม RapidMiner ใช้วิธีการต่าง ๆ ดังนี้

- Tokenizer ตัดข้อความ Text ออกเป็นคำศัพท์ต่าง ๆ (Term) เพื่อแปลงจากข้อความไปเป็นคำที่ถูกตัดแต่งแล้วหรือตัดช่องว่างออกแล้ว (Token) เช่น ตัวอย่างข้อความ: “The quick brown fox” จะถูกแปลงเป็น 4 คำ ดังนี้

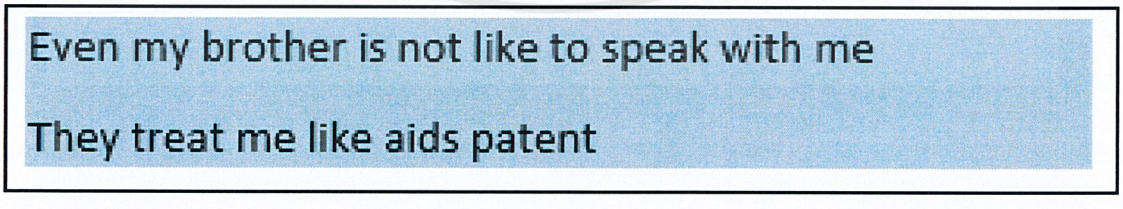
  (sentence
  (word The)
  (word quick)
  (word brown)
  (word fox))

- Filter Tokens (by length) กรองคำศัพท์ที่มีความยาวน้อยกว่าหรือมากกว่าที่กำหนด
- Stem (Porter) แปลงคำให้อยู่ในรากศัพท์เพื่อลดรูปคำให้อยู่ในรูปแบบคำปกติ เช่น วิธีการนี้จะลดรูปคำจาก: "stems", "stemmer", "stemming" และ "stemmed" ให้กลายเป็นคำปกติ คือ "stem"
- Filter Stopwords (English) ตัดคำเชื่อม หรือ คำที่ไม่จำเป็นทิ้ง เช่น the, is, at, which และ on

จากภาพที่ 3.10 หลังจากกำหนดใช้วิธีการต่าง ๆในการแปลงเพื่อทำการตัดแบ่ง อักขระ ตัวเลข แล้วย้อนกลับมาที่ Process เลือกเมนู Edit > Insert Building Block ระบุว่าจะทำการแปลงไปเป็นชนิดข้อความแบบใด และ ส่วนของหน้าจอโปรแกรม RapidMiner เพื่อทำการระบุการแปลงชนิดข้อความ แสดงดังภาพที่ 3.11



ภาพที่ 3.11 หน้าจอโปรแกรม RapidMiner เพื่อทำการระบุการแปลงชนิดข้อความ



ภาพที่ 3.12 ตัวอย่างข้อความที่ไม่ใช่ข้อความขยะก่อนแปลงจากตัวอักษรไปเป็นเวกเตอร์

Term	Count
Even	1
my	1
brother	1
is	1
not	1
like	2
to	1
speak	1
with	1
me	2
They	1
treat	1
aids	1
patent	1

{ 1, 1, 1, 1, 1, 2, 1, 1, 1, 2, 1, 1, 1, 1 }

ภาพที่ 3.13 ตัวอย่างข้อความที่ไม่ใช่ข้อความขยะหลังจากแปลงจากตัวอักษรไปเป็นเวกเตอร์

3.2.4 กำหนดวิธีที่ใช้คัดกรองและกำหนดค่าพารามิเตอร์

การกำหนดวิธีที่ใช้คัดกรองโดยเลือกทดลองทีละหนึ่งวิธีจากทั้งหมดห้าวิธีการคือ วิธีการเรียนรู้เชิงลึก, Bayesian Classification, K-Nearest Neighbors (K-NN), Artificial Neural Networks (ANNs) และ Support Vector Machine (SVM) จากนั้นทำการกำหนดค่าพารามิเตอร์เบื้องต้นของแต่ละวิธีการ

- วิธีการเรียนรู้เชิงลึก

ในปัญหาพิเศษนี้ จะใช้วิธีการเรียนรู้โครงข่ายประสาทเทียมแบบวนซ้ำ (Recurrent Neural Network) คือ การเอาข้อมูลบางส่วนที่ทำนายมาใช้ในการทำนายครั้งต่อไปและนำไปใช้ในการประมวลผลข้อมูลโดยจะเอาข้อมูลที่ส่งออกจากโหนดกลับมาเป็นข้อมูลนำเข้าใหม่ โดยเลือกประเภท Long Short-Term Memory (LSTM) ซึ่งทำหน้าที่แก้ปัญหาลำดับเวลาที่มีความยาวมากของโครงข่ายประสาทเทียมแบบวนซ้ำ โดยสถานะของส่วนที่เล็กที่สุด (Cell State) เป็นตัวเก็บสถานะ (State) ของหน่วยความจำที่เล็กที่สุด (Memory Cell) ใน Long Short-Term Memory และ เกท (Gate) ซึ่งมีค่าอนาล็อก (Analog) เป็นตัวควบคุมการไหลของข้อมูล

อินพุต (Input)

- Word Ids: เวกเตอร์ของคำ ซึ่งแสดงคุณลักษณะ (Feature) ของคำ ก่อนส่งไปยัง Long Short-Term Memory
- Embedding Matrix: เป็นอาเรย์ (Array) ของขนาดคำ (vocabulary_size) และขนาดของคุณลักษณะ (embedding size)

กระบวนการจัดเก็บเวกเตอร์ของคำ แสดงดังภาพที่ 3.14

```
# embedding_matrix is a tensor of shape [vocabulary_size, embedding size]
word_embeddings = tf.nn.embedding_lookup(embedding_matrix, word_ids)
```

ภาพที่ 3.14 กระบวนการจัดเก็บเวกเตอร์ของคำ

ค่าพารามิเตอร์

- จำนวนชั้นของ Long Short-Term Memory (number_of_layers) โดยเอาท์พุต (Output) ของชั้นแรกจะเป็นอินพุตของชั้นถัดไป

กระบวนการทำงานของคลาส ซึ่งเป็นคลาสที่ทำหน้าที่ MultiRNNCell แสดงดังภาพที่ 3.15

```
def lstm_cell():
  return tf.contrib.rnn.BasicLSTMCell(lstm_size)
stacked_lstm = tf.contrib.rnn.MultiRNNCell(
    [lstm_cell() for _ in range(number_of_layers)])

initial_state = state = stacked_lstm.zero_state(batch_size, tf.float32)
for i in range(num_steps):
    # The value of state is updated after processing each batch of words.
    output, state = stacked_lstm(words[:, i], state)

    # The rest of the code.
    # ...

final_state = state
```

ภาพที่ 3.15 คลาส MultiRNNCell ใน Long Short-Term Memory

- Bayesian Classification

อินพุต

- Value: คำที่ถูกตัดแต่งแล้ว (Token)

เมธอด setTokenizer แสดงดังภาพที่ 3.16

```
public void setTokenizer(Tokenizer value)
```

ภาพที่ 3.16 เมธอด setTokenizer

เมธอด getProbability แสดงดังภาพที่ 3.17

```
public double getProbability(int iNode,
                             int iParent,
                             int iValue)
```

ภาพที่ 3.17 เมธอด getProbability

- K-Nearest Neighbors (K-NN)

อินพุต

- Value: คำที่ถูกตัดแต่งแล้ว (Token)

เมธอด setTokenizer แสดงดังภาพที่ 3.18

```
public void setTokenizer(Tokenizer value)
```

ภาพที่ 3.18 เมธอด setTokenizer

ค่าพารามิเตอร์

- K: จำนวนข้อมูลใกล้เคียงที่ถูกนำมาพิจารณา

เมธอด setKNN แสดงดังภาพที่ 3.19

```
public void setKNN(int k)
```

ภาพที่ 3.19 เมธอด setKNN

- Artificial Neural Networks (ANNs)

อินพุต

- Value: คำที่ถูกตัดแต่งแล้ว (Token)

เมธอด setTokenizer แสดงดังภาพที่ 3.20

```
public void setTokenizer(Tokenizer value)
```

ภาพที่ 3.20 เมธอด: setTokenizer

ค่าพารามิเตอร์

- h: โหนดที่อยู่ในเลเยอร์ชั้นซ่อน

เมธอด setHiddenLayers แสดงดังภาพที่ 3.21

```
public void setHiddenLayers(java.lang.String h)
```

ภาพที่ 3.21 เมธอด setHiddenLayers

- Support Vector Machine (SVM)

อินพุต

- Value: คำที่ถูกตัดแต่งแล้ว (Token)

เมธอด setTokenizer แสดงดังภาพที่ 3.22

```
public void setTokenizer(Tokenizer value)
```

ภาพที่ 3.22 เมธอด setTokenizer

ค่าพารามิเตอร์

- Value: ประเภทของ SVM

เมธอด setSVMType แสดงดังภาพที่ 3.23

```
public void setSVMType(SelectedTag value)
```

ภาพที่ 3.23 เมธอด setSVMType

บทที่ 4

ผลการดำเนินงานและการอภิปรายผล

ในบทนี้จะกล่าวถึงผลการดำเนินงาน การอภิปรายผลการดำเนินงาน และปัญหาที่พบในการดำเนินงาน

4.1 ผลการดำเนินงาน

จากวิธีการที่ได้นำเสนอในบทที่สาม ผลลัพธ์หลังจากที่ทำการคัดกรองข้อความขยะของชุดข้อมูลทดสอบซึ่งประกอบด้วยชุดข้อมูลสองประเภทได้แก่ ชุดข้อมูลเอสเอ็มเอส และชุดข้อมูลอีเมล โดยรายละเอียดจะกล่าวถึงในหัวข้อที่ 4.1.1 – 4.1.2 ดังต่อไปนี้

4.1.1 ผลลัพธ์หลังจากที่ทำการคัดกรองข้อความขยะของชุดข้อมูลเอสเอ็มเอสจำนวน 2 ชุดข้อมูล ซึ่งผลลัพธ์ดังกล่าวได้ถูกประมวลผลด้วยซอฟต์แวร์ RapidMiner โดยซอฟต์แวร์ RapidMiner จะรับไฟล์ที่บันทึกคำตอบประเภทข้อความ และประโยคที่ใช้คัดกรองเป็นข้อมูลนำเข้า แล้วแสดงค่าผลลัพธ์ที่ได้จากการปรับค่าพารามิเตอร์ของวิธีการเรียนรู้เชิงลึก (Deep Learning) และวิธีการดั้งเดิมแบบอื่น ๆ สี่แบบคือ Bayesian Classification, K-Nearest Neighbors (K-NN), Artificial Neural Networks (ANNs) และ Support Vector Machine (SVM) ในรูปแบบของตารางและแผนภูมิ

- ผลลัพธ์ของชุดข้อมูลทดสอบ SMS Corpus สามารถแสดงได้ดังตารางที่ 4.1 - 4.5 และภาพที่ 4.1 โดยตารางที่ 4.1 คือ ตารางผลลัพธ์ของวิธี Artificial Neural Networks (ANNs) ตารางที่ 4.2 คือ ตารางผลลัพธ์ของวิธี Bayesian Classification ตารางที่ 4.3 คือ ตารางผลลัพธ์ของวิธี Support Vector Machine (SVM) ตารางที่ 4.4 คือ ตารางผลลัพธ์ของวิธี K-Nearest Neighbor (K-NN) ตารางที่ 4.5 คือ ตารางผลลัพธ์ของวิธีการเรียนรู้เชิงลึก และภาพที่ 4.1 คือ ภาพแผนภูมิแสดงความสัมพันธ์ระหว่าง Accuracy กับ Hidden Node(s) ของวิธีการเรียนรู้เชิงลึก

พารามิเตอร์ที่เกี่ยวข้อง	Hidden Node(s)	Accuracy	AUC	Runtime
1.อัตราความเร็วการเรียนรู้ (Learning Rate) = 0.5 2.อัตราการควมคุมความเหนี่ยวนำ (Momentum) = 0.2 3.ค่าคลาดเคลื่อน (Error epsilon) = 2.0E-16	1	95.73%	0.969	9 นาที
	2	95.98%	0.967	12 นาที
	4	95.73%	0.969	9 นาที
	8	96.23%	0.965	13 นาที
	16	95.98%	0.970	22 นาที
	32	96.48%	0.970	39 นาที

ตารางที่ 4.1 ผลลัพธ์ของวิธี Artificial Neural Networks (ANNs) กับชุดข้อมูลทดสอบ SMS Corpus

Accuracy	AUC	Runtime
95.73%	0.963	3 นาที

ตารางที่ 4.2 ผลลัพธ์ของวิธี Bayesian Classification กับชุดข้อมูลทดสอบ SMS Corpus

เคอร์เนลฟังก์ชัน (Kernel Function)	Accuracy	AUC	Runtime
เคอร์เนลเชิงเส้น (Linear Kernel)	94.22%	0.964	2 นาที
พหุนาม (Polynomial)	95.23%	0.965	4 นาที
Radial Basis Function (RBF)	84.67%	0.872	4 นาที

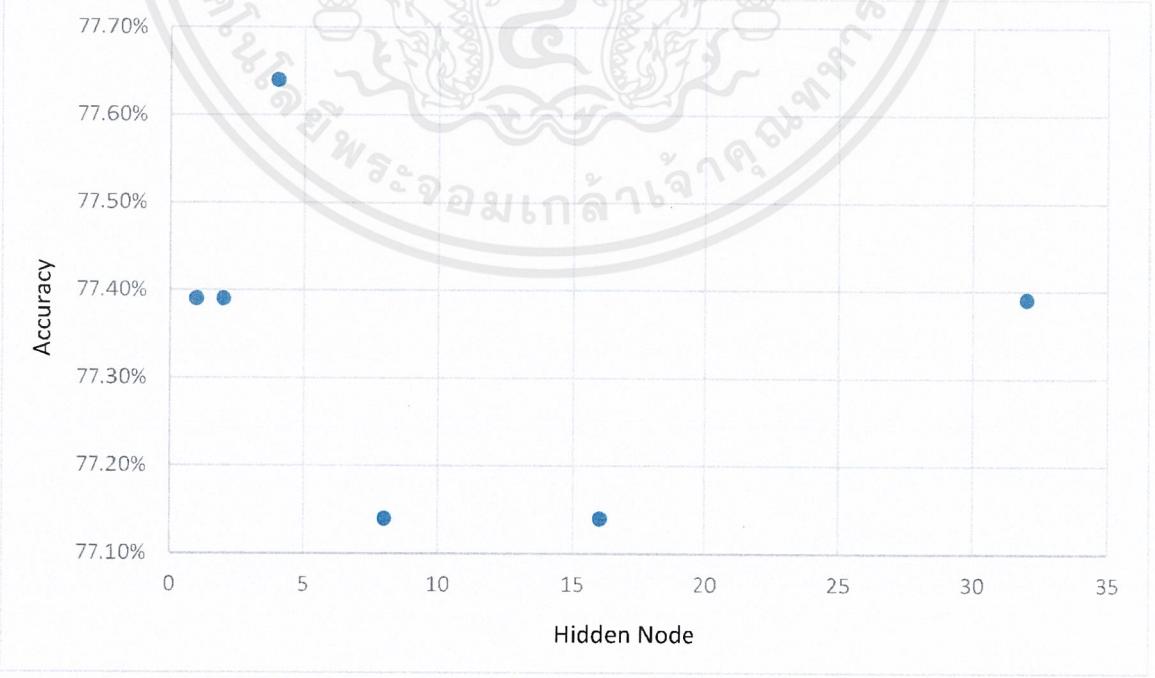
ตารางที่ 4.3 ผลลัพธ์ของวิธี Support Vector Machine (SVM) กับชุดข้อมูลทดสอบ SMS Corpus

จำนวน K	Accuracy	AUC	Runtime
1	74.12%	0.500	1 นาที
3	72.36%	0.771	2 นาที
5	71.86%	0.922	2 นาที
7	94.47%	0.940	1 นาที
9	83.67%	0.942	1 นาที
11	75.63%	0.944	1 นาที

ตารางที่ 4.4 ผลลัพธ์ของวิธี K-Nearest Neighbor (K-NN) กับชุดข้อมูลทดสอบ SMS Corpus

พารามิเตอร์ที่เกี่ยวข้อง	Hidden Node(s)	Accuracy	AUC	Runtime
1.ค่าคลาดเคลื่อน = 2.0E-16	1	77.39%	0.638	10 นาที
	2	77.39%	0.638	10 นาที
	4	77.64%	0.639	12 นาที
	8	77.14%	0.640	16 นาที
	16	77.14%	0.643	40 นาที
	32	77.39%	0.648	47 นาที

ตารางที่ 4.5 ผลลัพธ์ของวิธีการเรียนรู้เชิงลึก กับชุดข้อมูลทดสอบ SMS Corpus



ภาพที่ 4.1 แผนภูมิแสดงความสัมพันธ์ระหว่าง Accuracy และจำนวน Hidden Node(s) ของวิธีการเรียนรู้เชิงลึกกับชุดข้อมูลทดสอบ SMS Corpus

- ผลลัพธ์ของชุดข้อมูลทดสอบ NUS SMS Corpus สามารถแสดงได้ดังตารางที่ 4.6 - 4.10 และภาพที่ 4.2 โดยตารางที่ 4.6 คือ ตารางผลลัพธ์ของวิธี Artificial Neural Networks (ANNs) ตารางที่ 4.7 คือ ตารางผลลัพธ์ของวิธี Bayesian Classification ตารางที่ 4.8 คือ ตารางผลลัพธ์ของวิธี Support Vector Machine (SVM) ตารางที่ 4.9 คือ ตารางผลลัพธ์ของวิธี K-Nearest Neighbor (K-NN) ตารางที่ 4.10 คือ ตารางผลลัพธ์ของวิธีการเรียนรู้เชิงลึก และภาพที่ 4.2 คือ ภาพแผนภูมิแสดงความสัมพันธ์ระหว่าง Accuracy กับ Hidden Node(s) ของวิธีการเรียนรู้เชิงลึก

พารามิเตอร์ที่เกี่ยวข้อง	Hidden Node(s)	Accuracy	AUC	Runtime
1.อัตราความเร็วการเรียนรู้ = 0.5 2.อัตราการควมคุมความเหนี่ยวนำ = 0.2 3.ค่าคลาดเคลื่อน = 2.0E-16	1	94.09%	0.914	48 นาที
	2	94.51%	0.932	30 นาที
	4	94.03%	0.927	36 นาที
	8	93.67%	0.925	51 นาที
	16	94.21%	0.927	1 ชั่วโมง
	32	89.67%	0.927	1 ชั่วโมง 21 นาที

ตารางที่ 4.6 ผลลัพธ์ของวิธี Artificial Neural Networks (ANNs) กับชุดข้อมูลทดสอบ NUS SMS Corpus

Accuracy	AUC	Runtime
94.81%	0.926	2 นาที

ตารางที่ 4.7 ผลลัพธ์ของวิธี Bayesian Classification กับชุดข้อมูลทดสอบ NUS SMS Corpus

Kernel Function	Accuracy	AUC	Runtime
Linear Kernel	94.69%	0.933	11 นาที
Polynomial	95.10%	0.930	8 นาที
Radial Basis Function	92.30%	0.855	10 นาที

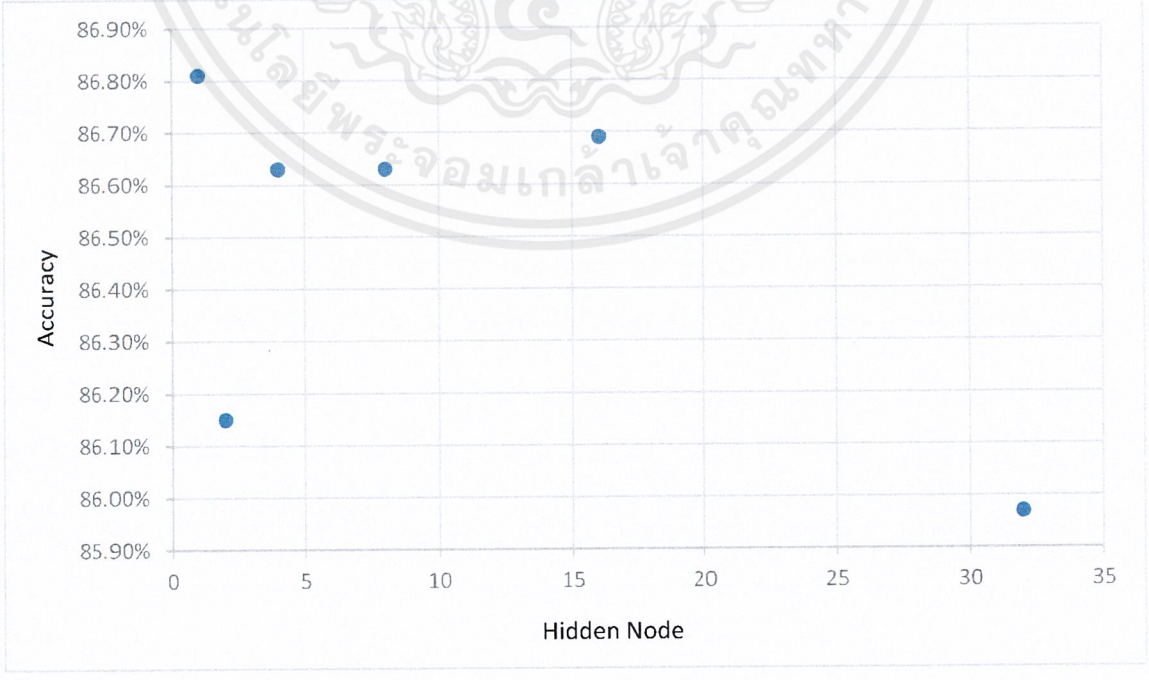
ตารางที่ 4.8 ผลลัพธ์ของวิธี Support Vector Machine (SVM) กับชุดข้อมูลทดสอบ NUS SMS Corpus

จำนวน K	Accuracy	AUC	Runtime
1	62.63%	0.500	9 นาที
3	61.79%	0.715	9 นาที
5	67.10%	0.722	8 นาที
7	70.69%	0.721	8 นาที
9	71.10%	0.712	8 นาที
11	69.55%	0.792	9 นาที

ตารางที่ 4.9 ผลลัพธ์ของวิธี K-Nearest Neighbor (K-NN) กับชุดข้อมูลทดสอบ NUS SMS Corpus

พารามิเตอร์ที่เกี่ยวข้อง	Hidden Node(s)	Accuracy	AUC	Runtime
1.ค่าคลาดเคลื่อน = 2.0E-16	1	86.81%	0.497	29 นาที
	2	86.15%	0.517	18 นาที
	4	86.63%	0.456	18 นาที
	8	86.63%	0.488	27 นาที
	16	86.69%	0.513	40 นาที
	32	85.97%	0.574	33 นาที

ตารางที่ 4.10 ผลลัพธ์ของวิธีการเรียนรู้เชิงลึกกับชุดข้อมูลทดสอบ NUS SMS Corpus



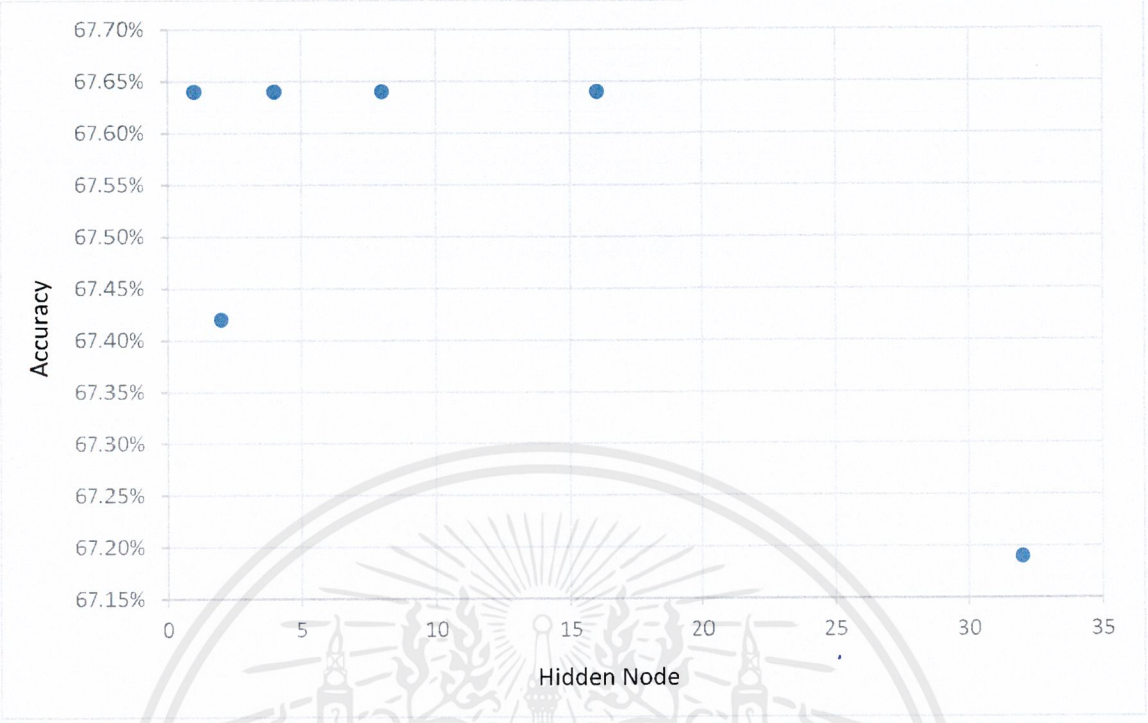
ภาพที่ 4.2 แผนภูมิแสดงความสัมพันธ์ระหว่าง Accuracy และจำนวน Hidden Node(s) ของวิธีการเรียนรู้เชิงลึกกับชุดข้อมูลทดสอบ NUS SMS Corpus

4.1.2 ผลลัพธ์หลังจากที่ทำการคัดกรองข้อความขยะของชุดข้อมูลอีเมลจำนวน 1 ชุดข้อมูล ซึ่งผลลัพธ์ดังกล่าวได้ถูกประมวลผลด้วยซอฟต์แวร์ RapidMiner โดยซอฟต์แวร์ RapidMiner จะรับไฟล์ที่บันทึกคำตอบประเภทข้อความ และประโยคที่ใช้คัดกรองเป็นข้อมูลนำเข้า แล้วแสดงค่าผลลัพธ์ที่ได้จากการปรับค่าพารามิเตอร์ต่าง ๆ ของวิธีการเรียนรู้เชิงลึก (Deep Learning) ในรูปแบบของตารางและแผนภูมิ

- ผลลัพธ์ของชุดข้อมูลทดสอบ PU1 Corpus สามารถแสดงได้ดังตารางที่ 4.11 และภาพที่ 4.3 โดยตารางที่ 4.11 คือ ตารางผลลัพธ์ของวิธีการเรียนรู้เชิงลึกและภาพที่ 4.3 คือ ภาพแผนภูมิแสดงความสัมพันธ์ระหว่าง Accuracy กับ Hidden Node(s) ของวิธีการเรียนรู้เชิงลึก

พารามิเตอร์ที่เกี่ยวข้อง	Hidden Node(s)	Accuracy	AUC	Runtime
1.ค่าคลาดเคลื่อน = 2.0E-16	1	67.64%	0.626	18 นาที
	2	67.42%	0.420	11 นาที
	4	67.64%	0.447	20 นาที
	8	67.64%	0.510	20 นาที
	16	67.64%	0.537	30 นาที
	32	67.19%	0.543	50 นาที

ตารางที่ 4.11 ผลลัพธ์ของวิธีการเรียนรู้เชิงลึกกับชุดข้อมูลทดสอบ PU1 Corpus



ภาพที่ 4.3 แผนภูมิแสดงความสัมพันธ์ระหว่าง Accuracy และจำนวน Hidden Node(s) ของวิธีการเรียนรู้เชิงลึกกับชุดข้อมูลทดสอบ PU1 Corpus

4.2 การประเมินผลการดำเนินงาน

จากผลการดำเนินงานในหัวข้อ 4.1 สามารถนำผลลัพธ์ที่ได้จากวิธีการที่นำเสนอ เปรียบเทียบกันระหว่างข้อมูลแต่ละชุด ซึ่งแสดงดังตารางที่ 4.12

ชุดข้อมูล	วิธีการที่ใช้คัดกรองข้อความขยะ								
	Support Vector Machine (SVM)			K-Nearest Neighbors (K-NN)			Artificial Neural Networks (ANNs)		
	Accuracy (%)	AUC	Runtime (Minutes)	Accuracy (%)	AUC	Runtime (Minutes)	Accuracy (%)	AUC	Runtime (Minutes)
SMS Corpus	95.23	0.965	4	94.47	0.940	1	96.48	0.970	39
NUS SMS Corpus	95.10	0.930	8	69.55	0.792	9	94.51	0.932	30
PU1 Corpus	98.1	N/A	N/A	90.8	N/A	N/A	98.5	N/A	N/A

ตารางที่ 4.12 การเปรียบเทียบผลลัพธ์ที่ได้จากวิธีการและชุดข้อมูลต่าง ๆ

ชื่อชุดข้อมูล	ชื่อวิธีการที่ใช้คัดกรองข้อความขยะ					
	Bayesian Classification			วิธีการเรียนรู้เชิงลึก (Deep Learning)		
	Accuracy (%)	AUC	Runtime (Minutes)	Accuracy (%)	AUC	Runtime (Minutes)
SMS Corpus	95.73	0.963	3	77.39	0.648	47
NUS SMS Corpus	94.81	0.926	2	85.97	0.574	33
PU1 Corpus	87.4	N/A	N/A	67.19	0.543	50

ตารางที่ 4.12 การเปรียบเทียบผลลัพธ์ที่ได้จากวิธีการและชุดข้อมูลต่าง ๆ (ต่อ)

วิธีการที่นำเสนอให้ค่า Accuracy, Area Under Curve (AUC) และเวลาที่ใช้ (Runtime) ซึ่งจากผลการดำเนินงานที่ได้สามารถนำมาทดสอบทางสถิติโดยใช้ค่าเฉลี่ยเลขคณิต ซึ่งเป็นค่ากลางทางสถิติ โดยมีสมมติฐานดังนี้

เมื่อชุดข้อมูลทดสอบ รวมถึงขั้นตอนวิธีการต่าง ๆ ที่ใช้ ส่งผลต่อประสิทธิภาพของการคัดกรองข้อความขยะ ดังนั้น เมื่อกล่าวถึงค่า AUC จะเป็นตัวบอกภาพรวม (Overall Test Accuracy) นั้น ๆ โดยมีการแบ่งค่าที่ออกมาได้ดังนี้

0.9-1.0 : Very Good, Excellent

08-0.9 : Good

0.7-0.8 : Fair

0.6-0.7 : Poor

0.5-0.6 : Fail

<0.5: Worthless Test

กล่าวได้ว่า ยิ่งค่า AUC เข้าใกล้ 1 หรือ ค่า Accuracy เข้าใกล้ 100% ยิ่งแสดงถึง ความถูกต้องจากการทดสอบชุดข้อมูลที่มาก เพราะแสดงถึงความน่าจะเป็นที่ชุดข้อมูลทดสอบนั้น ๆ จะ

ให้ผลถูกต้องในการคัดกรองข้อความขยะตามความเป็นจริง สามารถนำค่าที่ได้มาทำการเปรียบเทียบดังต่อไปนี้

- สำหรับชุดข้อมูล SMS Corpus
    - วิธีการที่ดีที่สุดในแง่ประสิทธิผล (Effectiveness) คือ Artificial Neural Networks (ANNs) ได้ค่า Accuracy = 96.48 % และค่า AUC = 0.970
    - วิธีการที่ดีที่สุดในแง่ประสิทธิภาพ (Efficiency) คือ K-Nearest Neighbors (K-NN) โดยมี Runtime คือ 1 นาที
    - วิธีการที่แย่ที่สุดในแง่ประสิทธิผล (Effectiveness) คือ Deep Learning ได้ค่า Accuracy = 77.39 % และค่า AUC = 0.648
    - วิธีการที่แย่ที่สุดในแง่ประสิทธิภาพ (Efficiency) คือ Deep Learning โดยมี Runtime คือ 47 นาที
- สำหรับชุดข้อมูล NUS SMS Corpus
    - วิธีการที่ดีที่สุดในแง่ประสิทธิผล (Effectiveness) คือ Support Vector Machine (SVM) ได้ค่า Accuracy = 95.10 % และ Artificial Neural Networks (ANNs) ได้ค่า AUC = 0.930
    - วิธีการที่ดีที่สุดในแง่ประสิทธิภาพ (Efficiency) คือ Bayesian Classification โดยมี Runtime คือ 2 นาที
    - วิธีการที่แย่ที่สุดในแง่ประสิทธิผล (Effectiveness) คือ K-Nearest Neighbors (K-NN) ได้ค่า Accuracy = 69.55 % และ Deep Learning ได้ค่า AUC = 0.574
    - วิธีการที่แย่ที่สุดในแง่ประสิทธิภาพ (Efficiency) คือ Deep Learning โดยมี Runtime คือ 33 นาที
- สำหรับชุดข้อมูล PU1 Corpus
    - วิธีการที่ดีที่สุดในแง่ประสิทธิผล (Effectiveness) คือ Artificial Neural Networks (ANNs) ได้ค่า Accuracy = 98.5 %
    - วิธีการที่แย่ที่สุดในแง่ประสิทธิผล (Effectiveness) คือ Deep Learning ได้ค่า Accuracy = 67.19 %

4.3 การวิเคราะห์ผลการดำเนินงาน

4.3.1 ผลลัพธ์ของชุดข้อมูลทดสอบ SMS Corpus

- **ผลลัพธ์ของวิธี Artificial Neural Networks (ANNs)**

- กำหนดให้อัตราความเร็วการเรียนรู้เป็น 0.5
- กำหนดอัตราการควบคุมความเหนี่ยวนำเป็น 0.2
- กำหนดค่าความคลาดเคลื่อนเป็น 2.0E-16
- กำหนดจำนวน Hidden Nodes เป็น log n ฐาน 2

จำนวน Hidden Nodes เป็น 1 ทำให้ได้ค่า Accuracy เป็น 95.73% และ มีค่า AUC เป็น 0.969 โดยใช้เวลาในการประมวลผล 9 นาที หลังจากนั้นทำการปรับจำนวน Hidden Nodes เป็น 2 ทำให้เห็นแนวโน้มว่าค่า Accuracy เพิ่มขึ้นเป็น 95.98% และ มีค่า AUC ลดลงเป็น 0.967 โดยใช้เวลาในการประมวลผลเพิ่มขึ้นเป็น 12 นาที หลังจากนั้นทำการปรับจำนวน Hidden Nodes เป็น 4 ผลลัพธ์ของ Accuracy มีค่าเป็น 95.73% และ มีค่า AUC เป็น 0.969 โดยใช้เวลาในการประมวลผล 9 นาที ทำให้เห็นได้ว่าจำนวน Hidden Nodes เป็น 4 มีผลลัพธ์เท่ากับจำนวน Hidden Nodes เป็น 1 หลังจากนั้นทำการปรับจำนวน Hidden Nodes เป็น 8 ทำให้ได้ค่า Accuracy เป็น 96.23% และ มีค่า AUC เป็น 0.965 โดยใช้เวลา 13 นาที ทำให้เห็นได้ว่าค่า Accuracy เพิ่มขึ้น ค่า AUC ลดลง และ ใช้เวลาการประมวลผลเพิ่มขึ้น หลังจากนั้นปรับจำนวน Hidden Nodes เป็น 16 มีค่า Accuracy เป็น 95.98% ซึ่งมีค่าเท่ากับจำนวน Hidden Nodes เป็น 2 และ มีค่า AUC เพิ่มขึ้นเป็น 0.970 โดยใช้เวลาในการประมวลผลเพิ่มขึ้นเป็น 22 นาที หลังจากนั้นทำการปรับจำนวน Hidden Nodes เป็น 32 ซึ่งให้ผลลัพธ์ดีที่สุดและใช้เวลาในการประมวลผลมากที่สุด ทำให้ได้ค่า Accuracy เป็น 96.48% และมีค่า AUC เป็น 0.970 โดยใช้ระยะเวลาในการประมวลผล 39 นาที

- **ผลลัพธ์ของวิธี Bayesian Classification**

ได้ค่า Accuracy เป็น 95.73% และ AUC มีค่าเป็น 0.963 โดยใช้เวลาในการประมวลผล 3 นาที

- **ผลลัพธ์ของวิธี Support Vector Machine (SVM)**

- กำหนดเคอร์เนลฟังก์ชัน

เคอร์เนลเชิงเส้นทำให้ได้ค่า Accuracy เป็น 94.22% และ AUC เป็น 0.964 โดยใช้เวลาในการประมลวผล 2 นาที หลังจากนั้นปรับเคอร์เนลฟังก์ชันเป็นแบบ Radial Basis Function ทำให้เห็นแนวโน้มว่าค่า Accuracy ลดลงเป็น 84.67% และ AUC ลดลงเป็น 0.872 โดยใช้เวลาในการประมวลผลเพิ่มขึ้นเป็น 4 นาที หลังจากนั้นปรับเคอร์เนลฟังก์ชันเป็นแบบพหุนาม ซึ่งให้ผลลัพธ์ดีที่สุดและใช้เวลาในการประมวลผลมากที่สุด ทำให้ได้ค่า Accuracy เป็น 95.23% และมีค่า AUC เป็น 0.965 โดยใช้ระยะเวลาในการประมวลผล 4 นาที

- **ผลลัพธ์ของวิธี K-Nearest Neighbor (K-NN)**

จำนวนค่า K เป็น 1 ทำให้ได้ค่า Accuracy เป็น 74.12% และ มีค่า AUC เป็น 0.500 โดยใช้เวลาในการประมวลผล 1 นาที หลังจากนั้นทำการปรับจำนวนค่า K เป็น 3 ทำให้เห็นแนวโน้มว่าค่า Accuracy ลดลงเป็น 72.36% และ มีค่า AUC เพิ่มขึ้นเป็น 0.771 โดยใช้เวลาในการประมวลผลเพิ่มขึ้นเป็น 2 นาที หลังจากนั้นทำการปรับจำนวนค่า K เป็น 5 ผลลัพธ์ของ Accuracy มีค่าน้อยสุดเป็น 71.86% และค่า AUC เพิ่มขึ้นจากค่า K ก่อนหน้าเป็น 0.922 โดยใช้เวลาประมวลผล 2 นาที หลังจากนั้นทำการปรับจำนวนค่า K เป็น 11 ทำให้เห็นแนวโน้มว่าค่า Accuracy เพิ่มขึ้นเป็น 75.63% และมีค่า AUC มากสุด คือ 0.944 โดยใช้เวลาในการประมวลผล 1 นาที หลังจากนั้นทำการปรับจำนวนค่า K เป็น 9 ทำให้ได้ว่าค่า Accuracy เพิ่มขึ้นเป็น 83.67% และ AUC มีค่าเป็น 0.942 โดยใช้เวลาในการประมวลผล 1 นาที หลังจากนั้นทำการปรับจำนวนค่า K เป็น 7 ซึ่งให้ผลลัพธ์ดีที่สุด ทำให้ได้ค่า Accuracy เป็น 94.47% และมีค่า AUC เป็น 0.940 โดยใช้ระยะเวลาในการประมวลผล 1 นาที

- **ผลลัพธ์ของวิธีการเรียนรู้เชิงลึก**

- กำหนดค่าความคลาดเคลื่อนเป็น 2.0E-16
- กำหนดจำนวน Hidden Nodes เป็น log n ฐาน 2

จำนวน Hidden Nodes เป็น 1 ทำให้ได้ค่า Accuracy เป็น 77.39% และ มีค่า AUC เป็น 0.638 โดยใช้เวลาในการประมวลผล 10 นาที หลังจากนั้นทำการปรับจำนวน Hidden Nodes เป็น 2 ทำให้เห็นว่าค่า Accuracy, AUC และเวลาที่ใช้ในการประมวลผลมีผลลัพธ์เท่ากับจำนวน Hidden Nodes เป็น 1 หลังจากนั้นทำการปรับจำนวน Hidden Nodes เป็น 8 ทำให้เห็นว่าผลลัพธ์ของ Accuracy ลดลงทำให้มีค่าเป็น 77.14% และ มีค่า AUC เพิ่มขึ้นเป็น 0.640 โดยใช้

เวลาในการประมวลผล 16 นาที หลังจากนั้นทำการปรับจำนวน Hidden Nodes เป็น 16 จะได้ค่า Accuracy เป็น 77.14% ซึ่งมีผลลัพธ์เท่ากับจำนวน Hidden Nodes เป็น 8 และมีค่า AUC เพิ่มขึ้นเป็น 0.643 โดยใช้เวลาในการประมวลผลเพิ่มขึ้นเป็น 40 นาที หลังจากนั้นทำการปรับจำนวน Hidden Nodes เป็น 32 จะได้ค่า Accuracy เป็น 77.39% ซึ่งมีผลลัพธ์เท่ากับจำนวนชั้นซ่อนเป็น 1, 2 และ 32 และ มีค่า AUC เป็น 0.648 โดยใช้เวลาในการประมวลผล 47 นาที หลังจากนั้นทำการปรับจำนวน Hidden Nodes เป็น 4 ซึ่งให้ผลลัพธ์ดีที่สุด ทำให้ได้ค่า Accuracy เป็น 77.64% และมีค่า AUC เป็น 0.639 โดยใช้ระยะเวลาในการประมวลผล 12 นาที

4.3.2 ผลลัพธ์ของชุดข้อมูลทดสอบ NUS SMS Corpus

- **ผลลัพธ์ของวิธี Artificial Neural Networks (ANNs)**

- กำหนดให้อัตราความเร็วการเรียนรู้เป็น 0.5
- กำหนดอัตราการควบคุมความเหนี่ยวนำเป็น 0.2
- กำหนดค่าความคลาดเคลื่อนเป็น 2.0E-16
- กำหนดจำนวน Hidden Nodes เป็น log n ฐาน 2

จำนวน Hidden Nodes เป็น 1 ทำให้ได้ค่า Accuracy เป็น 94.09% และ มีค่า AUC เป็น 0.914 โดยใช้เวลาในการประมวลผล 48 นาที หลังจากนั้นทำการปรับจำนวน Hidden Nodes เป็น 4 ทำให้เห็นแนวโน้มว่าค่า Accuracy ลดลงเป็น 94.03% และ มีค่า AUC เพิ่มขึ้นเป็น 0.927 โดยใช้เวลาในการประมวลผลเพิ่มขึ้นเป็น 36 นาที หลังจากนั้นทำการปรับจำนวน Hidden Nodes เป็น 8 ผลลัพธ์ของ Accuracy มีค่าเป็นลดลง 93.67% และ มีค่า AUC เป็น 0.925 โดยใช้เวลาในการประมวลผล 51 นาที 1 หลังจากนั้นทำการปรับจำนวนชั้นซ่อนเป็น 16 ทำให้ได้ค่า Accuracy เพิ่มขึ้นเป็น 94.21% และ มีค่า AUC เป็น 0.927 โดยใช้เวลา 1 ชั่วโมง หลังจากนั้นปรับจำนวน Hidden Nodes เป็น 32 มีค่า Accuracy ลดลงต่ำสุดเป็น 89.67% และ มีค่า AUC เป็น 0.927 ซึ่งมีผลลัพธ์เท่ากับจำนวน Hidden Nodes เป็น 4 และ 16 โดยใช้เวลาในการประมวลผลเพิ่มขึ้นเป็น 1 ชั่วโมง 21 นาที หลังจากนั้นทำการปรับจำนวน Hidden Nodes เป็น 2 ซึ่งให้ผลลัพธ์ดีที่สุดทำให้ได้ค่า Accuracy เป็น 94.51% และมีค่า AUC เพิ่มขึ้นเป็น 0.932 โดยใช้ระยะเวลาในการประมวลผล 30 นาที

- **ผลลัพธ์ของวิธี Bayesian Classification**

ได้ค่า Accuracy เป็น 94.81% และ AUC มีค่าเป็น 0.926 โดยใช้เวลาในการประมวลผล 2 นาที

- **ผลลัพธ์ของวิธี Support Vector Machine (SVM)**

- กำหนดเคอร์เนลฟังก์ชัน

เคอร์เนลเชิงเส้นทำให้ได้ค่า Accuracy เป็น 94.69% และ AUC เป็น 0.933 โดยใช้เวลาในการประมลวผล 11 นาที หลังจากนั้นปรับเคอร์เนลฟังก์ชันเป็นแบบ Radial Basis Function ทำให้เห็นแนวโน้มว่าค่า Accuracy ลดลงเป็น 92.30% และ AUC ลดลงเป็น 0.855 โดยใช้เวลาในการประมวลผลเพิ่มขึ้นเป็น 8 นาที หลังจากนั้นปรับเคอร์เนลฟังก์ชันเป็นแบบพหุนาม ซึ่งให้ผลลัพธ์ดีที่สุดและใช้เวลาในการประมวลผลมากที่สุด ทำให้ได้ค่า Accuracy เป็น 94.69% และมีค่า AUC เป็น 0.933 โดยใช้ระยะเวลาในการประมวลผล 11 นาที

- **ผลลัพธ์ของวิธี K-Nearest Neighbor (K-NN)**

จำนวนค่า K เป็น 1 ทำให้ได้ค่า Accuracy เป็น 62.63% และ มีค่า AUC เป็น 0.500 โดยใช้เวลาในการประมวลผล 9 นาที หลังจากนั้นทำการปรับจำนวนค่า K เป็น 3 ทำให้เห็นแนวโน้มว่าค่า Accuracy ลดลงเป็น 61.79% และ มีค่า AUC เพิ่มขึ้นเป็น 0.715 โดยใช้เวลาในการประมวลผล 9 นาที หลังจากนั้นทำการปรับจำนวนค่า K เป็น 5 ผลลัพธ์ของ Accuracy มีค่าเพิ่มขึ้นเป็น 67.10% และค่า AUC เพิ่มขึ้นจากค่า K ก่อนหน้าเป็น 0.722 โดยใช้เวลาประมวลผล 8 นาที หลังจากนั้นทำการปรับจำนวนค่า K เป็น 11 ทำให้เห็นแนวโน้มว่าค่า Accuracy เพิ่มขึ้นเป็น 69.55% และมีค่า AUC มากสุด คือ 0.792 โดยใช้เวลาในการประมวลผล 9 นาที หลังจากนั้นทำการปรับจำนวนค่า K เป็น 9 ทำให้ได้ว่าค่า Accuracy เพิ่มขึ้นเป็น 71.10% และ AUC มีค่าเป็น 0.712 โดยใช้เวลาในการประมวลผล 8 นาที หลังจากนั้นทำการปรับจำนวนค่า K เป็น 7 ซึ่งให้ผลลัพธ์ดีที่สุด ทำให้ได้ค่า Accuracy เป็น 70.69% และมีค่า AUC เป็น 0.721 โดยใช้ระยะเวลาในการประมวลผล 8 นาที

- **ผลลัพธ์ของวิธีการเรียนรู้เชิงลึก**

- กำหนดค่าความคลาดเคลื่อนเป็น 2.0E-16

- กำหนดจำนวน Hidden Nodes เป็น log n ฐาน 2

จำนวน Hidden Nodes เป็น 1 ทำให้ได้ค่า Accuracy ที่ให้ผลลัพธ์ดีที่สุดมีค่าเป็น 86.81% และ มีค่า AUC เป็น 0.497 โดยใช้เวลาในการประมวลผล 29 นาที หลังจากนั้นทำการปรับจำนวน Hidden Nodes เป็น 2 ทำให้เห็นว่าค่า Accuracy ลดลงเป็น 86.81% และ AUC เพิ่มขึ้นเป็น 0.517 โดยใช้เวลาในการประมวลผล 18 นาที หลังจากนั้นทำการปรับจำนวน Hidden Nodes เป็น 4 ทำให้เห็นว่าผลลัพธ์ของ Accuracy เพิ่มขึ้นทำให้มีค่าเป็น 86.63% และ มีค่า AUC ลดลงเป็น 0.456 โดยใช้เวลาในการประมวลผล 18 นาที หลังจากนั้นทำการปรับจำนวนชั้นซ่อนเป็น 8 จะได้ค่า Accuracy เป็น 86.63% ซึ่งมีผลลัพธ์เท่ากับจำนวนชั้นซ่อนเป็น 4 และมีค่า AUC เพิ่มขึ้นเป็น 0.488 โดยใช้เวลาในการประมวลผลเพิ่มขึ้นเป็น 27 นาที หลังจากนั้นทำการปรับจำนวน Hidden Nodes เป็น 16 จะได้ค่า Accuracy เป็น 86.69% และ มีค่า AUC เป็น 0.513 โดยใช้เวลาในการประมวลผล 40 นาที หลังจากนั้นทำการปรับจำนวน Hidden Nodes เป็น 32 ซึ่งให้ผลลัพธ์น้อยที่สุด ทำให้ได้ค่า Accuracy เป็น 85.97% และมีค่า AUC เป็น 0.574 โดยใช้ระยะเวลาในการประมวลผล 33 นาที

4.3.3 ผลลัพธ์ของชุดข้อมูลทดสอบ PU1 Corpus

- กำหนดค่าความคลาดเคลื่อนเป็น 2.0E-16

- กำหนดจำนวน Hidden Nodes เป็น log n ฐาน 2

ปรับจำนวนชั้นซ่อนเป็น 1, 4, 8, 16 ทำให้ได้ค่า Accuracy ที่ให้ผลลัพธ์ดีที่สุดเท่ากัน มีค่าเป็น 67.64% แต่ มีค่า AUC และ เวลาต่างกัน โดยจำนวน Hidden Nodes เป็น 1 ให้ค่า AUC เป็น 0.626 และใช้เวลาในการประมวลผล 18 นาที จำนวน Hidden Nodes เป็น 4 ให้ค่า AUC เป็น 0.447 และใช้เวลาในการประมวลผล 20 นาที จำนวน Hidden Nodes เป็น 8 ให้ค่า AUC เป็น 0.510 และใช้เวลาในการประมวลผล 20 นาที จำนวน Hidden Nodes 16 ให้ค่า AUC เป็น 0.537 และใช้เวลาในการประมวลผล 30 นาที หลังจากนั้นทำการปรับจำนวน Hidden Nodes เป็น 2 ทำให้ได้ค่า Accuracy ลดลงเป็น 67.42% และค่า AUC เป็น 0.420 โดยใช้เวลาในการประมวลผล 11 นาที หลังจากนั้นทำการปรับจำนวน Hidden Nodes เป็น 32 ซึ่งให้ผลลัพธ์น้อย

ที่สุด ทำให้ได้ค่า Accuracy เป็น 67.19% และมีค่า AUC เป็น 0.543 โดยใช้ระยะเวลาในการประมวลผลนานสุด 33 นาที

4.4 ปัญหาที่พบในการดำเนินงาน

4.4.1 ชุดข้อมูล PU Corpus1 ซึ่งเป็นข้อมูลอีเมลมีปัญหา

เนื่องจากข้อมูลเป็นจำนวนตัวเลขทั้งหมดและมี ข้อความขยะ และ ข้อความที่ไม่ใช่ขยะ รวมกันจึงต้องใช้การโปรแกรมในการแยกข้อความ อีกทั้งข้อมูลในไฟล์ Text มีหัวข้ออีเมลและเนื้อหาอีเมลรวมกัน จึงต้องทำการเขียนโปรแกรมแยกไฟล์ใหม่ทั้งหมด

4.4.2 ข้อมูลบางชุดของ ชุดข้อมูล SMS ใช้ไม่ได้

เมื่อทำการคัดกรองไฟล์ข้อความขยะ กับ ข้อความที่ไม่ใช่ขยะ ออกมา ทำให้พบว่ามีชุดข้อมูลบางชุดใช้ไม่ได้ เพราะ ตัดอักขระแล้วทำให้จำนวนไฟล์ในชุดข้อมูลลดลงและไม่สามารถนำมาใช้ในการวัดผลได้ จึงต้องทำการตัดชุดข้อมูลบางชุดออกไป

4.4.3 ข้อมูลเกิด Missing attributes

ข้อมูลที่นำเข้ามาประมวลผลโดยซอฟต์แวร์ Rapid Miner จะต้องเป็นไฟล์ CSV ที่สามารถใช้โอเปอร์เรเตอร์ Read CSV โหลดเข้ามาใช้งาน โดยการอ่านไฟล์ CSV ทุกครั้ง เมื่อไฟล์มีการอัพเดต ข้อมูลจะเปลี่ยนตามได้แต่ข้อมูลที่ใช้นำเข้าเป็นไฟล์ Text ทำให้เมื่อไฟล์มีการอัพเดตข้อมูลหรือแก้ไขข้อมูลจะไม่เปลี่ยนตาม ซึ่งข้อมูลที่ได้รับการแก้ไขนั้น Rapid Miner อาจจะหา Attribute ไม่เจอจึงทำให้เกิด Missing attributes จึงต้องมีการแก้ไขข้อมูลในไฟล์และอัพโหลดใหม่ก่อนที่จะนำไฟล์ Text มาดำเนินงานในขั้นต่อไป

4.4.4 ข้อมูลที่นำเข้าเมื่อทำการ Testing เกิดค่า Error

ข้อมูลที่จะนำเข้ามาใช้ในส่วนของ Validation นั้นจะต้องเลือกประเภทในการใช้งานและการปรับค่าพารามิเตอร์ให้สอดคล้องกับวิธีการและประเภทไฟล์ข้อมูล ถ้าเลือกประเภทและการใช้งานไม่สอดคล้องกับวิธีการและข้อมูลที่ใช้ เมื่อทำการ Testing จะเกิดค่า Error ซึ่งการปรับค่าพารามิเตอร์จะมีผลกระทบต่อความแม่นยำในการ Testing ข้อมูล

บทที่ 5

สรุปผลการดำเนินงานและข้อเสนอแนะ

ในบทนี้จะกล่าวถึงการสรุปผลการดำเนินงานและข้อเสนอแนะ ซึ่งการสรุปผลการดำเนินงานจะถูกนำเสนอในหัวข้อ 5.1 และข้อเสนอแนะจะถูกนำเสนอในหัวข้อ 5.2

5.1 สรุปผลการดำเนินงาน

ปัญหาพิเศษนี้นำเสนอวิธีการคัดกรองข้อความขยะ และ ข้อความที่ไม่ใช่ขยะ โดยนำเอาเทคนิคการเรียนรู้ของเครื่องเข้ามาประยุกต์ใช้ จากการทดสอบโดยนำชุดข้อมูลทดสอบที่เกี่ยวกับอีเมลและเอสเอ็มเอสมาทำการทดลองโดยใช้วิธีการเรียนรู้เชิงลึกและวิธีการดั้งเดิมสี่แบบ คือ Bayesian Classification, K-Nearest Neighbors (K-NN), Artificial Neural Networks (ANNs) และ Support Vector Machine (SVM) ผลการทดลองพบว่าวิธีการที่นำเสนอซึ่งก็คือวิธีการเรียนรู้เชิงลึก ได้ผลลัพธ์เป็นที่น่าพอใจแต่ยังไม่ถือว่าดีที่สุด ซึ่งอาจเกิดขึ้นด้วยปัจจัยหลายอย่างเช่น จำนวนชุดข้อมูลหรือทรัพยากรที่ใช้ เป็นต้น ซึ่งเมื่อทำการปรับค่าพารามิเตอร์ในส่วนของจำนวน Hidden Node(s) พบว่ามีผลกระทบกับผลลัพธ์ที่ได้ ในช่วงเวลานั้น ๆ และเมื่อจำนวนคุณลักษณะที่ใช้ในการทดลองมีน้อยทำให้ค่าความถูกต้องแปรผันตรงตามกัน ดังนั้น ในส่วนของการทดลองนั้นต้องเลือก ประเภทพารามิเตอร์และปรับค่าพารามิเตอร์ให้สอดคล้องกับแต่ละวิธีการและชุดข้อมูล ซึ่งถ้าทำได้อย่างเหมาะสมจะช่วยให้ผลลัพธ์ที่ได้ในแต่ละวิธีการมีประสิทธิภาพและประสิทธิผลมากยิ่งขึ้น

5.2 ข้อเสนอแนะ

การคัดกรองข้อความนั้นจะทำการแบ่งข้อความออกเป็นสองส่วน คือ ข้อความขยะ และข้อความที่ไม่ใช่ขยะ โดยในปัญหาพิเศษนี้ การแบ่งข้อความได้ใช้วิธีการเขียนโปรแกรมติดต่อกับไฟล์และเขียนกำหนดรูปแบบหรือกลุ่มคำ (Regular Expressions) เพื่อตัดอักขระที่ไม่ได้นำมาใช้ รวมถึงใช้ในการแยกประเภทข้อความสำหรับกำหนดเซตคำตอบของแต่ละคลาสเพื่อประหยัดเวลาและทรัพยากรในการทำงานเมื่อต้องทำการแยกประเภทข้อความจำนวนมาก ซึ่งถ้างานวิจัยในอนาคตมีการแยกชุดข้อมูลอย่างชัดเจนจะทำให้ขั้นตอนก่อนนำข้อมูลมาทดลองนั้นสะดวกมากยิ่งขึ้น ในกรณีที่ชุดข้อมูลทดสอบมีไม่เพียงพอ ถ้างานวิจัยในอนาคตมีการค้นคว้าหาชุดข้อมูลทดสอบที่มีผู้ทำการสร้างขึ้นใหม่ได้จำนวนมากจะทำให้เครื่องสามารถเรียนรู้จากตัวอย่างที่ใช้สอนได้มากยิ่งขึ้น และในกรณีที่การ

นำเข้าชุดข้อมูลมีความคลาดเคลื่อน ในอนาคตถ้ามีการอัพเดตข้อมูลแล้วต้องตรวจสอบข้อมูลให้ละเอียดก่อน จะทำให้ขั้นตอนการฝึกสอนนั้นมีคุณภาพมากยิ่งขึ้น ในการปรับค่าพารามิเตอร์ต้องสอดคล้องกับวิธีการเรียนรู้เชิงลึกและวิธีการดั้งเดิมสี่แบบ คือ Bayesian Classification, K-Nearest Neighbors (K-NN), Artificial Neural Networks (ANNs) และ Support Vector Machine (SVM) โดยต้องทำการศึกษาค้นคว้าถึงความสัมพันธ์ของพารามิเตอร์แต่ละตัว เพื่อให้ผลลัพธ์มีประสิทธิภาพและประสิทธิผลมากยิ่งขึ้น

บรรณานุกรม

Konstantin Tretyakov, kt@ut.ee Institute of Computer Science, University of Tartu, " MachineLearning Techniques in Spam Filtering ", Data Mining Problem-oriented Seminar, MTAT.03.177,May 2004, pp. 60-79.

Abha Tewari Student, ME VESIT, Smita Jangale Associate Professor VESIT, " Spam Filtering Methods and machine Learning Algorithm - A Survey ", International Journal of Computer Applications (0975 – 8887) Volume 154 – No.6, November 2016

" Machine Learning Techniques in Spam Filtering" Konstantin Tretyakov,kt@ut.ee Institute of Computer Science, University of Tartu Data Mining Problem oriented Seminar, MTAT.03.177,

"A Performance Evaluation of Machine Learning-Based Streaming Spam Tweets Detection" Chao Chen, Jun Zhang, Member, IEEE, Yi Xie, Yang Senior Member, IEEE,@2015

W.A. Awad Math.&Comp.Sci.Dept., Science faculty, Port Said University, S.M. ELseuofi Information. System Dept., Ras El Bar High inst, " Machine Learning methods for E-mail Classification ", International Journal of Computer
Applications (0975 – 8887) Volume 16– No.1, February 2011

V.Christina, S.Karpagavalli, G.Suganya M.Phil Research scholar Department of Computer Science(PG) P.S.G.R Krishnammal College for Women Senior Lecturer GR Govindarajulu School of Appiled Computer Technology, "Email Spam Filtering using

Supervised Machine Learning Techniques", V. Christina et al. / (IJCSE) International Journal on Computer Science and Engineering Vol. 02, No. 09, 2010, 3126-3129

Omar Saad , Ashraf Darwish and Ramadan Faraj, " A survey of machine learning techniques for Spam filtering ", IJCSNS International Journal of Computer Science and Network Security, VOL.12 No.2, February 2012

Neelam Choudhary and Ankit Kumar Jain, " Towards Filtering of SMS Spam Messages Using Machine Learning Based Technique "

AlexyBhownick, Shyamanta. M. Hazarika, reviewed article "Machine Learning for E-mail Spam Filtering: Review, Techniques and Trends,"June 3 2016

Pingchuan Liu and Teng-Sheng Moh"Content Based Spam E-mail Filtering", International Conference on Collaboration Technologies and Systems.,IEEE © 2016