การจำแนกประเภทแบบหลายฉลากของบทความในฐานข้อมูลวารสารวิชาการไทยจากบทคัดย่อ

นายจินตรัย พุทธิพรชัย

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

วิทยานิพนธ์นี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต
สาขาวิชาวิทยาศาสตร์คอมพิวเตอร์ ภาควิชาวิศวกรรมคอมพิวเตอร์
คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย
ปีการศึกษา 2565
ลิขสิทธิ์ของจุฬาลงกรณ์มหาวิทยาลัย

Multi-Label Classification for Articles in Thai Journal Database from Article's Abstract

Mr. Chintrai Puttipornchai

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

A Thesis Submitted in Partial Fulfillment of the Requirements
for the Degree of Master of Science in Computer Science
Department of Computer Engineering
FACULTY OF ENGINEERING
Chulalongkorn University
Academic Year 2022

Copyright of Chulalongkorn University

หัวข้อวิทยานิพนธ์	การจำแนกประเภทแบบหลายฉลากของบทความในฐานข้อมูลวารสารวิชาการไทยจากบทคัดย่อ
โดย	นายจินตรัย พุทธิพรชัย
สาขาวิชา	วิทยาศาสตร์คอมพิวเตอร์
อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก	อาจารย์ ดร.เนื่องวงศ์ ทวยเจริญ

คณะวิศวกรรมศาสตร์ จุฬาลงกรณ์มหาวิทยาลัย อนุมัติให้นับวิทยานิพนธ์ฉบับนี้เป็นส่วนหนึ่งของการศึกษาตามหลักสูตรปริญญาวิทยาศาสตรมหาบัณฑิต

..... คณบดีคณะวิศวกรรมศาสตร์
(ศาสตราจารย์ ดร.สุพจน์ เตชวรสินสกุล)

คณะกรรมการสอบวิทยานิพนธ์

..... ประธานกรรมการ
(ศาสตราจารย์ ดร.บุญเสริม กิจศิริกุล)

..... อาจารย์ที่ปรึกษาวิทยานิพนธ์หลัก
(อาจารย์ ดร.เนื่องวงศ์ ทวยเจริญ)

..... กรรมการ
(อาจารย์ ดร.ณรงค์เดช กีรติพรานนท์)

..... กรรมการภายนอกมหาวิทยาลัย
(อาจารย์ ดร.รัฐศิลป์ รานอกภานุวัชร์)

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

จินตรัย พุทธิพรชัย : การจำแนกประเภทแบบหลายฉลากของบทความในฐานข้อมูลวารสารวิชาการไทยจากบทคัดย่อ. (Multi-Label Classification for Articles in Thai Journal Database from Article's Abstract) อ.ที่ปรึกษาหลัก : อ. ดร.เนื่องวงศ์ ทวยเจริญ

บทความวิจัยของไทยที่มีจำนวนเพิ่มมากขึ้นทำให้การจัดหมวดหมู่เป็นหมวดหมู่ย่อยเป็นเรื่องที่ท้าทาย ซึ่งต้องใช้ผู้เชี่ยวชาญและต้องใช้เวลามากในการจัดประเภทบทความประเภทต่าง ๆ ดังนั้นงานวิจัยนี้จึงนำเสนอวิธีการและเทคนิคในการจำแนกบทความวิทยาการคอมพิวเตอร์แบบหลายฉลากในวารสารไทยและนำเสนอการเปรียบเทียบวิธีการต่าง ๆ สำหรับการจำแนกประเภทหลายฉลาก คือ Binary Relevance (BR), Classifier Chains (CC) และ Label Power-set (LP) ด้วยวิธีการตัดคำที่ใช้ตัวแยกประเภทซัพพอร์ตเวกเตอร์แมชชีน พบว่าวิธีการ CC-SVM-RBF kernel ร่วมกับวิธีการตัดคำภาษาไทย pythainlp และ TF-IDF ให้ผลลัพธ์ที่ดีที่สุดสำหรับ ตัวชี้วัดประสิทธิภาพการเลือกตอบตามตัวอย่าง และ ตัวชี้วัดประสิทธิภาพการจำแนกประเภทหลายฉลาก โดยมี ML-accuracy = 0.578, Subset accuracy = 0.300, ค่าเรียกคืน = 0.670 และ ค่าเฉลี่ยไมโครสำหรับค่าเรียกคืน = 0.670 อย่างไรก็ตามวิธีการ BR-SVM-RBF kernel ร่วมกับวิธีการตัดคำภาษาไทย pythainlp ให้ผลลัพธ์ที่ดีที่สุดสำหรับ ตัวชี้วัดประสิทธิภาพการเลือกตอบตามตัวอย่าง และ ตัวชี้วัดประสิทธิภาพการจำแนกประเภทหลายฉลาก คือ Hamming loss = 0.106, ค่าแม่นยำ = 0.735, ตัววัด F1 = 0.665, ค่าเฉลี่ยไมโครสำหรับค่าแม่นยำ = 0.586 และ ค่าเฉลี่ยไมโครสำหรับตัววัด F1 = 0.715 งานในอนาคตควรปรับปรุง Subset accuracy สำหรับแบบจำลองการจำแนกประเภทหลายฉลากในภาษาไทย

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

สาขาวิชา วิทยาศาสตร์คอมพิวเตอร์ ลายมือชื่อนิสิต .....

ปีการศึกษา 2565 ลายมือชื่อ อ.ที่ปรึกษาหลัก .....

6370043021 : MAJOR COMPUTER SCIENCE
KEYWORD: multi-label classification, Binary Relevance, Classifier Chains, Label Power-set

Chintrai Puttipornchai : Multi-Label Classification for Articles in Thai Journal Database from Article's Abstract. Advisor: NUENGWONG TUAYCHAROEN, Ph.D.

The increasing number of Thai research articles makes it challenging to classify them into sub-categories. This task requires specialists and a lot of time to classify the different types of articles. Therefore, this research presents methods and techniques for multi-label classification of computer science articles in Thai journals. We present a comparison of different methods for multi-label classification, including Binary Relevance (BR), Classifier Chains (CC), and Label Power-set (LP) with a word segmentation method that uses a Support Vector Machine (SVM) classifier. We found that the CC-SVM-RBF kernel method combined with pythainlp word segmentation and TF-IDF produces the best results for both example-based and label-based metrics, with ML-accuracy is 0.578, Subset accuracy is 0.300, Recall is 0.670 and Micro-average recall is 0.670 On the other hand, BR-SVM-RBF combined with pythainlp word segmentation and TF-IDF produces the best results for both example-based and label-based metrics with Hamming loss is 0.106, Precision is 0.735, F-measure is 0.655, Micro-average precision is 0.586 and Micro-average F-Measure is 0.715. In Future work, Subset accuracy should be improved for the multi-label classification model in the Thai language.

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

Field of Study:	Computer Science	Student's Signature .....
Academic Year:	2022	Advisor's Signature .....

กิตติกรรมประกาศ

วิทยานิพนธ์ฉบับนี้ สำเร็จลุล่วงได้เพราะความกรุณาอย่างยิ่งจากผู้เชี่ยวชาญและผู้ให้คำปรึกษาทุกท่าน ได้ให้คำแนะนำ ช่วยเหลือและให้คำปรึกษาอย่างดียิ่ง ผู้วิจัยขอขอบพระคุณอย่างสูงมา ณ โอกาสนี้

ขอขอบพระคุณอาจารย์ที่ปรึกษา อาจาย์ ดร. เนื่องวงศ์ ทวยเจริญ ที่ได้กรุณาปรับปรุงแก้ไขข้อบกพร่องและให้คำแนะนำ ชี้แนะแนวทางต่าง ๆ ในการทำงานวิจัย และ ขอขอบพระคุณ คุณสภาจรรยาชัชวาล ที่ได้รวบรวมข้อมูลบทความวิทยาการคอมพิวจากทาง NECTEC เพื่อใช้ในการทำงานวิจัย ซึ่งเป็นประโยชน์อย่างยิ่งในการทำงานวิจัย

สุดท้ายนี้ขอขอบพระคุณ คุณพ่อ คุณแม่ และพี่ชาย ที่ค่อยช่วยเหลือ เป็นกำลังใจและให้คำแนะนำแก่ผู้วิจัยตลอดมา

จินตรัย พุทธิพรชัย

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

สารบัญ

หน้า

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

Chulalongkorn University

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

สารบัญตาราง

หน้า

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

สารบัญรูปภาพ

หน้า

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

บทที่ 1
บทนำ

1.1 ที่มาและความสำคัญของปัญหา

ปี 2553 ศูนย์ดัชนีการอ้างอิงวารสารไทย (TCI), สำนักงานกองทุนสนับสนุนการวิจัย, มหาวิทยาลัยเทคโนโลยีพระจอมเกล้าธนบุรี และมหาวิทยาลัยธรรมศาสตร์ ได้พัฒนาและปรับปรุงซอฟต์แวร์โอเพนซอร์ซ (Open source software) ชื่อว่า Open Journal System (OJS) มาให้บริการชื่อว่า Thai Journal Online หรือ ThaiJO โปรแกรม OJS เป็นระบบการจัดการวารสารและการตีพิมพ์ในรูปแบบ Electronic journal หรือ e-journals ซึ่งโปรแกรม OJS เปิดให้ผู้พัฒนาหรือผู้สนใจสามารถดาวน์โหลดไปใช้งานโดยไม่มีค่าใช้จ่าย แต่สงวนลิขสิทธิ์ซอฟต์แวร์อยู่ภายใต้เงื่อนไขสัญญาอนุญาตแบบสาธารณะ โดย ThaiJO เปิดให้วารสารไทยเข้ามาใช้งานระบบ มีวารสารไทยที่ต้องการจัดทำวารสารในรูปแบบ e-journals มีความสนใจเข้าใจงานเป็นจำนวนมาก โดยทาง TCI ได้จัดอบรบการใช้งานให้คำแนะนำและช่วยเหลือสนับสนุนวารสารไทยให้ก้าวไปสู่การมีคุณภาพและมาตรฐานทัดเทียมกับวารสารวิชาการอื่น ๆ ในระดับสากล [1]

ปี 2560 ศูนย์เทคโนโลยีอิเล็กทรอนิกส์และคอมพิวเตอร์แห่งชาติ (NECTEC) เข้ามาร่วมเป็นส่วนหนึ่งของ TCI และเข้ามาช่วยดูแลปรับปรุงระบบ ตั้งชื่อเป็น ThaiJO เวอร์ชั่น 2 หรือ ThaiJO 2.0 ทำให้ระบบเว็บไซต์วารสารมีความทันสมัย ง่ายต่อการใช้งานมากขึ้น รวมถึงเข้ามาช่วยในการพัฒนาระบบการค้นหาบทความ การจัดเก็บทำดัชนีข้อมูลเอกสารของวารสาร และได้พัฒนาระบบตรวจความซ้ำซ้อนของบทความ (Plagiarism checking system) เพื่อให้บรรณาธิการสามารถตรวจสอบความซ้ำซ้อนของบทความว่า มีความซ้ำซ้อนกับเอกสารอื่น ๆ ที่เผยแพร่แล้วในระบบ ThaiJO 2.0 ระบบนี้ช่วยให้บรรณาธิการวารสารทำงานได้สะดวก รวดเร็วยิ่งขึ้นในการบริหารจัดการต้นฉบับ [1]

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ในปัจจุบัน Thai Journal Online (ThaiJO) เป็นแพลตฟอร์มออนไลน์ที่รวบรวมวารสารทางวิชาการของประเทศไทย ณ วันที่ 10 ตุลาคม 2565 มีจำนวนวารสาร 1,141 จำนวน , บทความจำนวน 226,571 ซึ่งมีจำนวนเพิ่มขึ้นเรื่อย ๆ ทำให้ยากต่อการจำแนกประเภทวารสารหรือบทความเพื่อให้บรรณาธิการวารสารหรือผู้เชี่ยวชาญเฉพาะด้านตรวจสอบหรือประเมินคุณภาพบทความในสาขานั้น ๆ เช่น สังคมศาสตร์ มนุษย์ศาสตร์ เป็นต้น ส่งผลให้ใช้เวลานานในการจำแนกประเภทก่อนจะส่งต่อบทความให้ผู้เชี่ยวชาญเฉพาะด้านตรวจสอบสอบความถูกต้องซึ่งการจำแนกประเภทวารสารหรือบทความจึงมีประโยชน์

งานวิจัยนี้นำเสนอวิธีการจำแนกประเภทแบบหลายฉลากของบทความภาษาไทยที่เกี่ยวข้องกับสาขาวิทยาการคอมพิวเตอร์ (Computer Science) ซึ่งประกอบด้วย 12 สาขาย่อย ด้วยวิธีสกัดคำสำคัญจากบทคัดย่อ และใช้วิธีการตัดคำ Dictionary based, Custom Dictionary Based และ Deepcut จากนั้นคำนวณความถี่ของคำที่เกิดขึ้น (TF-IDF) แต่ละเอกสาร นำมาจำแนกประเภท ด้วยขั้นตอนวิธี Support Vector Machine ร่วมกับวิธีการ Problem Transformation คือ Binary Relevance, Classifier Chains และ Label Power-set

การวัดประสิทธิภาพของการจำแนกประเภทแบบหลายฉลากโดยวัดประสิทธิภาพจาก 2 วิธีหลักคือ ตัวชี้วัดประสิทธิภาพการเลือกตอบตามตัวอย่าง (Example-based Metrics) และ ตัวชี้วัดประสิทธิภาพการจำแนกประเภทหลายฉลาก (Label-based Metrics) โดย Example-based Metrics ประกอบด้วย Hamming loss, ML-accuracy, Subset accuracy, ค่าแม่นยำ (Precision), ค่าเรียกคืน (Recall) และ ตัววัด F1 (F-Measure) ซึ่ง Label-based Metrics การวัดค่าเฉลี่ยระดับไมโคร ประกอบด้วย ค่าเฉลี่ยไมโครสำหรับค่าแม่นยำ (Micro-average precision), ค่าเฉลี่ยไมโครสำหรับค่าเรียกคืน (Micro-average recall) และ ค่าเฉลี่ยไมโครสำหรับตัววัด F1 (Micro-average F-measure) โดยการวัดประสิทธิภาพทั้ง 2 วิธีจะเฉลี่ยค่า K รอบจากขั้นตอนวิธีการตรวจสอบไขว้ (K-fold Cross Validation)

1.2 วัตถุประสงค์ของงานวิจัย

เพื่อสร้างแบบจำลองจำแนกประเภทแบบหลายฉลากของบทความในสาขาวิทยาการคอมพิวเตอร์ ที่ได้รับการตีพิมพ์เผยแพร่ในวารสารวิชาการไทยในแพลตฟอร์ม ThaiJO โดยใช้เทคนิคการแบ่งคำภาษาไทยจากบทคัดย่อโดยใช้เครื่องมือบนภาษา Python คือ pythainlp (Maximum Matching), Custom Dictionary pythainlp (Maximum Matching) และ Deepcut แล้วจึงใช้เทคนิค TF-IDF สกัดคำสำคัญ เพื่อนำไปจำแนกประเภทของบทความ ด้วยขั้นตอนวิธี Support Vector Machine และ Problem Transformation คือ Binary Relevance, Classifier Chains และ Label Power-set

1.3 ขอบเขตงานวิจัย

1. งานวิจัยนี้รองรับการจำแนกประเภทของบทความที่เป็นภาษาไทยเท่านั้น
2. งานวิจัยนี้สร้างแบบจำลองจำแนกประเภทข้อมูลบทความที่แผยแพร่บนแพลตฟอร์ม ThaiJO เท่านั้น
3. ประเภทบทความที่นำมาวิเคราะห์ คือบทความสาขาวิทยาการคอมพิวเตอร์ (Computer Science) ซึ่งประกอบด้วยสาขาย่อยจำนวน 12 สาขาย่อย คือ

    3.1 General Computer Science (GCS)

    3.2 Artificial Intelligence (AI)

    3.3 Computational Theory and Mathematics (CTM)

    3.4 Computer Graphics and Computer-Aided Design (CG)

    3.5 Computer Networks and -Communications (CNC)

    3.6 Computer Science Applications (CSA)

    3.7 Computer Vision and Pattern – Recognition (CV)

    3.8 Hardware and Architecture (HA)

    3.9 Human-Computer Interaction (HCI)

    3.10 Information Systems (IS)

    3.11 Signal Processing (SP)

    3.12 Software (SW)

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

1.4 ขั้นตอนและวิธีการดำเนินงานวิจัย

1. ศึกษางานวิจัยที่เกี่ยวข้องและเครื่องมือที่ใช้ในการตัดคำภาษาไทย
2. ศึกษาทฤษฎีที่เกี่ยวของกับ TF-IDF
3. ศึกษาทฤษฎีและเรียนรู้การใช้เครื่องมือในการสร้างแบบจำลอง
4. เก็บข้อมูลบทความจากวารสารจากแพลตฟอร์ม ThaiJO
5. กำหนดสาขาย่อยด้านวิทยาการคอมพิวเตอร์ให้กับบทความ
6. ออกแบบวิธีการตัดคำของภาษาไทย และสกัดคำสำคัญโดยใช้เครื่องมือบนภาษา Python
7. สร้างแบบจำลองการจำแนกประเภท

8. วิเคราะห์ผลและวัดประสิทธิภาพของแบบจำลองการจำแนกประเภท
9. วิเคราะห์และสรุปผลการทดลอง

1.5 ประโยชน์ที่คาดว่าจะได้รับจากงานวิจัย

1. แบบจำลองสามารถจำแนกประเภทของบทความได้
2. สามารถเลือกวิธีการแบ่งคำและสร้างแบบจำลองที่มีประสิทธิภาพสำหรับการจำแนกประเภทบทความ
3. ช่วยให้บรรณาธิการวารสารส่งต่อบทความให้ผู้ประเมินได้เร็วขึ้นและตรงตามสาขางานวิจัย

1.6 งานวิจัยที่ได้รับการตีพิมพ์

งานวิจัยนี้ได้รับการคัดเลือกและตีพิมพ์เป็นบทความวิชาการเรื่อง “Multi-Label Classification for Articles in Thai Journal Database from Article's Abstract” โดย นายจินตรัย พุทธิพรชัย คุณสภา จรรยาชัชวาล และ อาจารย์.ดร. เนื่องวงศ์ ทวยเจริญ ซึ่งได้ไปนำเสนอ ผลงานในงานประชุมวิชาการ “The 19th International Joint Conference on Computer Science and Software Engineering (JCSSE 2022)” ซึ่งจัดขึ้นที่มหาวิทยาลัยศิลปากร วังท่าพระ ระหว่างวันที่ 22-25 มิถุนายน 2565

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

บทที่ 2
ทฤษฎีและงานวิจัยที่เกี่ยวข้อง

ทฤษฎีที่เกี่ยวข้อง

ทฤษฎีที่เกี่ยวข้องในงานวิจัยนี้ ได้แก่

1. การตัดคำภาษาไทย
    - Dictionary Based (Maximal Matching)
    - Machine Learning
2. การสกัดคำสำคัญ
    - TF-IDF
3. การจำแนกประเภทของข้อความ
    - ซัพพอร์ตเวกเตอร์แมชชีน
4. วิธีการจำแนกประเภทแบบหลายฉลาก
    - Binary Relevance
    - Classifier Chains
    - Label Power-set
5. ขั้นตอนวิธีการตรวจสอบไขว้ (K-fold Cross Validation)
6. การวัดประสิทธิภาพของแบบจำลอง

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

2.1 การตัดคำภาษาไทย

การตัดคำเป็นหนึ่งในงานที่สำคัญของการประมวลผลภาษาธรรมชาติ งานส่วนใหญ่ที่ใช้งานเกี่ยวกับการประมวลผลภาษาธรรมชาติ จะต้องทำการตัดคำก่อนที่จะดำเนินการในส่วนอื่นต่อไป เช่น การแปลภาษา จะต้องทำการตัดคำก่อนที่จะดำเนินการวิเคราะห์ตามหลักไวยากรณ์และแปลเป็นภาษาอื่นได้ แต่ในการตัดคำในภาษาไทย จีน ญี่ปุ่น จะไม่ง่ายเหมือนกับ ภาษาอังกฤษ ภาษาสเปน เพราะ คำภาษาไทยไม่ได้ถูกแบ่งส่วนเหมือนในภาษาอังกฤษ เช่น การเว้นวรรค, การจุลภาค เป็นต้น วิธีการตัดคำซึ่งแบ่งออกเป็น 2 ประเภทที่แตกต่างกัน [2] ได้แก่

2.1.1 การตัดคำโดยใช้พจนานุกรม (Dictionary-based)

วิธีการตัดคำโดยใช้ จะใช้ชุดข้อมูลจากพจนานุกรมสำหรับการวิเคราะห์และตัดคำ วิธีการนี้จะเข้าไปค้นหาลำดับของอักขระในพจนานุกรมเพื่อจับคู่คำที่ถูกต้อง แต่การตัดคำโดยใช้พจนานุกรมจะมีปัญหาคำที่ไม่มีให้พจนานุกรมจะไม่สามารถตัดคำได้ ประสิทธิภาพ

ของวิธีการตัดคำโดยใช้พจนานุกรมขึ้นอยู่กับคุณภาพและขนาดของพจนานุกรม และสามารถปรับปรุงประสิทธิภาพโดยการเพิ่มคำใหม่หรือคำเฉพาะที่ไม่มีในพจนานุกรมลงในพจนานุกรมที่ใช้สำหรับกระบวนการตัดคำได้ ยกตัวอย่างการตัดคำจากพจนานุกรม [2] เช่น

Maximal matching

การตัดคำโดยใช้วิธีการ Dictionary-based ขั้นตอนวิธี Maximal matching วิธีนี้จะพิจารณาข้อความทั้งหมดทีละคำจากซ้ายไปขวา การตัดคำจะเข้าไปค้นหาคำในพจนานุกรม ถ้าคำที่ค้นหาเจอในพจนานุกรมให้เลือกจำนวนคำที่ตัดได้จำนวนน้อยที่สุด ยกตัวอย่างเช่น “ไปหามเหสี” ดังแสดงในรูปที่ 1มีพจนานุกรมตัวอย่างคือ [1 า, ม, เ, ส, ี ,ไป,หา,หาม,เห,สี,มเหสี] อธิบายตามรูปที่ 1 การเข้าไปค้นหาในพจนานุกรมจะเริ่มด้วยตำแหน่งที่ 1 แถวที่ 1 ตัว คือ “ไ” ถูกพบในพจนานุกรมจะถูกระบุเป็น 1 ในตำแหน่ง 2 แถวที่ 1 คือ “ป” เมื่อรวมกับตำแหน่งที่ 1 และ ตำแหน่งที่ 2 จะเป็นคำว่า “ไป” พบในพจนานุกรมจะถูกระบุเป็น 1 ในตำแหน่งที่ 3 คือคำว่า “ไปห” ซึ่งไม่มีคำนี้ในพจนานุกรมจึงถูกระบุเป็น “inf” หมายถึงไม่พบคำนี้ในพจนานุกรม ต่อมาในแถวที่ 2 จะเริ่มต้นด้วยตำแหน่งที่ 2 แต่จะดูค่าย้อนกลับไปในตำแหน่งที่ 1 แถวที่ 1 แล้วเลือกค่าที่น้อยที่สุดซึ่งคือ 1 เป็นตัวอักษร “ไ” เมื่อนำมารวมกันได้เป็น {“ไ”, “ป”} 2 คำนี้เจอในพจนานุกรมในตำแหน่งที่ 2 แถวที่ 2 จึงถูกระบุเป็น 2 ในตำแหน่งต่อมาจะเป็น {“ไ”, “ปห”} ซึ่ง “ปห” ไม่มีในพจนานุกรมจึงถูกระบุเป็น “inf” โดยจะเข้าไปค้นหาจนครบทุกตำแหน่ง เช่น คำในตำแหน่งที่ 3 เริ่มที่แถวที่ 3 โดยผลลัพธ์จะเป็น {“ไป”, “ห”} , {“ไป”, “หา”}, {“ไป”, “หาม”} ทั้งหมดถูกพบในพจนานุกรมจึงถูกระบุเป็น 2 ในตำแหน่งที่ 3, 4, 5 ตามลำดับ

Chulalongkorn University

เมื่อเข้าไปค้นหาคำทั้งหมดในพจนานุกรมแล้ว ต้องทำการคำนวณย้อนกลับโดยเริ่มต้นที่ ตำแหน่งที่ 9 เลือกค่าที่น้อยที่สุดที่ถูกพบในพจนานุกรม คือ 3 อยู่ในแถวที่ 5 เป็นจุดเริ่มต้น ย้อนกลับไปจนไม่พบค่าในตำแหน่งนั้นหรือไม่เป็นค่า “inf” คือในตำแหน่งที่ 5 รวมกันเป็นคำว่า “มเหสี” ต่อมาเลือกจุดเริ่มต้นที่ไม่พบค่าในตำแหน่งนั้น คือ ตำแหน่งที่ 4 โดยจะคำนวณย้อนกลับไปจนเสร็จสิ้นจะได้คำว่า “หา”, “ไป” เมื่อนำมารวมกัน {“ไป”, “หา”, “มเหสี”} โดยสรุปว่า “ไปหามเหสี” จะถูกตัดคำเป็น ไป|หา|มเหสี

	ไ	ป	ท	า	ม	เ	ท	ส	ี
	1	2	3	4	5	6	7	8	9
1	1	1	inf	inf	inf	inf	inf	inf	inf
2		2	inf	inf	inf	inf	inf	inf	inf
3			2	2	2	inf	inf	inf	inf
4				3	inf	inf	inf	inf	inf
5					3	inf	inf	inf	3
6						3	3	inf	inf
7							4	inf	inf
8								4	4
9									5

รูปที่ 1 แสดงวิธีการ Maximal matching

2.1.2 การตัดคำโดยใช้เทคนิคการเรียนรู้เครื่อง (Machine learning-based)

การตัดคำโดยใช้ Machine learning-based จะอาศัยแบบจำลองทางสถิติจากเทคนิคการเรียนรู้ด้วยเครื่อง โดยเทคนิคนี้ใช้การติดแท็กคำโดยมีการระบุอักขระที่เริ่มต้นของคำ และอักขระภายในคำ ข้อดีของ Machine learning-based ไม่ต้องมีพจนานุกรมรองรับในการตัดคำ แต่จะขึ้นอยู่คลังข้อมูลที่นำมาฝึกฝนแบบจำลองและปัญหาคำที่ไม่รู้จักและคำที่กำกวมจะได้รับการจัดการโดยการเตรียมชุดตัวอย่างการฝึกฝนของแบบจำลองมากเพียงพอเพื่อให้การตัดคำได้แม่นยำ [2]

2.2 การสกัดคำสำคัญจากเอกสาร

เมื่อข้อความผ่านการเตรียมข้อมูลโดยจัดการกับข้อความ เช่น ตัดคำภาษาไทย, ลบ Stop word ออกจากเอกสารนั้น ๆ เป็นต้น เรียบร้อยแล้ว ขั้นตอนต่อมาจะเป็นการสกัดคำสำคัญจากเอกสาร ซึ่งเป็นส่วนที่สำคัญในการจัดประเภทของเอกสารรวมถึงงานที่เกี่ยวข้องกับการประมวลผลภาษาธรรมชาติ ในงานวิจัยนี้จะใช้วิธีการ Term frequency-inverse document frequency (TF-IDF) ดังจะอธิบายในหัวข้อต่อไป

Term frequency-inverse document frequency (TF-IDF)

การคูณของ Term frequency กับ Inverse document frequency ซึ่ง Term Frequency หมายถึง การนับคำศัพท์ที่เกิดขึ้นในเอกสารเปรียบเทียบกับคำทั้งหมดในเอกสารนั้น และ Inverse document frequency หมายถึง การกำหนดน้ำหนักของคำที่พบเจอในเอกสารทั้งหมด ค่า IDF ต่ำ จะหมายถึง คำนั้นเป็นคำทั่วไปที่ไม่ใช่คำสำคัญของเอกสารนั้น และ ค่า IDF สูงจะหมายถึง คำนั้นเป็นคำที่ถูกพบไม่บ่อยในเอกสารทั้งหมดอาจระบุเป็นคียเวิร์ดในเอกสารนั้นได้ ซึ่ง TF-IDF สามารถคำนวณได้ด้วยสมการ

$$\text{TF-IDF} = \text{TF} \times \log\left(\frac{\text{จำนวนเอกสารทั้งหมด}}{\text{จำนวนเอกสารที่มีคำนั้นปรากฏอยู่}}\right)$$

เพื่อแสดงตัวอย่างการสกัดคำด้วยวิธี TF-IDF กำหนดให้ 2 เอกสาร ซึ่งแต่ละเอกสารผ่านขั้นตอนการตัดคำ และ Text Preprocessing

- เอกสารที่ 1: **การศึกษา|วิธี|เซิร์ฟเวอร์|วัตถุประสงค์|ติดตั้ง|ทดสอบ|ระบบควบคุม|ซอฟต์แวร์|เซิร์ฟเวอร์|ซอฟต์แวร์|**
- เอกสารที่ 2: **การศึกษา|เสนอ|วิธี|การรู้จำ|เสียงพูด|ภาษาไทย|ทนทาน|สัญญาณรบกวน|สภาพแวดล้อม|วิธี|การรู้จำ|เสียง|ช่องสัญญาณ|จำนวน|**

เอกสารที่ 1 ประกอบด้วยคำศัพท์ทั้งหมด 10 คำ ซึ่งสามารถคำนวณ Term frequency ได้ดังต่อไปนี้

คำศัพท์	**จำนวนคำ**	**TF**
การศึกษา	1	1/10 = 0.1
วิธี	1	1/10 = 0.1
วัตถุประสงค์	1	1/10 = 0.1
ติดตั้ง	1	1/10 = 0.1
ทดสอบ	1	1/10 = 0.1
ระบบควบคุม	1	1/10 = 0.1
ซอฟต์แวร์	2	2/10 = 0.2
เซิร์ฟเวอร์	2	2/10 = 0.2

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

เพื่อแสดงการคำนวณ Inverse document frequency โดยพิจารณาในเอกสารที่ 1 พบว่ามีคำ การศึกษา วิธี เกิดขึ้นในเอกสารที่ 1 และ 2 ซึ่งสามารถคำนวณได้ดังต่อไปนี้

คำศัพท์	จำนวนเอกสารคำนั้นปรากฏ	IDF
การศึกษา	2	$\log\frac{2}{2} = 0$
วิธี	2	$\log\frac{2}{2} = 0$
วัตถุประสงค์	1	$\log\frac{2}{1} = 0.3$
ติดตั้ง	1	$\log\frac{2}{1} = 0.3$
ทดสอบ	1	$\log\frac{2}{1} = 0.3$
ระบบควบคุม	1	$\log\frac{2}{1} = 0.3$
ซอฟต์แวร์	1	$\log\frac{2}{1} = 0.3$
เซิร์ฟเวอร์	1	$\log\frac{2}{1} = 0.3$

ดังนั้นค่า TF-IDF ในเอกสารที่ 1 คำนวณได้ดังต่อไปนี้

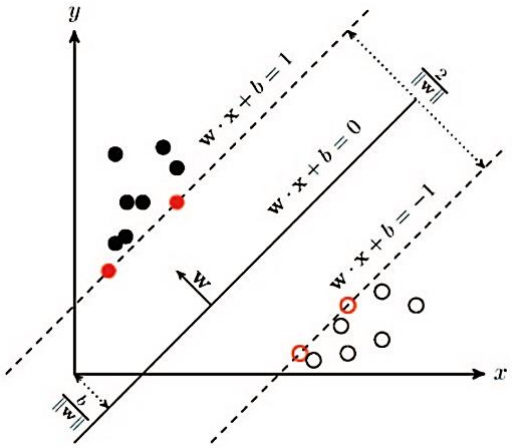
คำศัพท์	TF	IDF	TF-IDF
การศึกษา	1/10 = 0.1	$\log\frac{2}{2} = 0$	0
วิธี	1/10 = 0.1	$\log\frac{2}{2} = 0$	0
วัตถุประสงค์	1/10 = 0.1	$\log\frac{2}{1} = 0.3$	0.03
ติดตั้ง	1/10 = 0.1	$\log\frac{2}{1} = 0.3$	0.03
ทดสอบ	1/10 = 0.1	$\log\frac{2}{1} = 0.3$	0.03
ระบบควบคุม	1/10 = 0.1	$\log\frac{2}{1} = 0.3$	0.03
ซอฟต์แวร์	2/10 = 0.2	$\log\frac{2}{1} = 0.3$	0.06
เซิร์ฟเวอร์	2/10 = 0.2	$\log\frac{2}{1} = 0.3$	0.06

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

2.3 การจำแนกประเภท

2.3.1 ซัพพอร์ตเวกเตอร์แมชชีน

การจำแนกประเภทซัพพอร์ตเวกเตอร์แมชชีน เป็นประเภทขั้นตอนวิธีการเรียนรู้แบบมีผู้สอน (supervised learning) ค่อนข้างมีประสิทธิภาพในการแก้ปัญหาด้วยข้อมูลหลายมิติ วัตถุประสงค์ของขั้นตอนวิธีซัพพอร์ตเวกเตอร์แมชชีน คือการค้นหาระยะขอบสูงสุด (Maximum margin) เพื่อให้ได้ผลลัพธ์การจัดหมวดหมู่ที่ดีที่สุด [3]



รูปที่ 2 ซัพพอร์ตเวกเตอร์แมชชีน [3]

การจำแนกประเภทด้วยขั้นตอนวิธีซัพพอร์ตเวกเตอร์แมชชีน ถูกกำหนดโดยฟังก์ชัน $f(x) = sign(w \cdot x + b)$ โดยมีเงื่อนไข คือ

$$f(x) = 1, w \cdot x + b \geq 0$$

$$f(x) = 0, w \cdot x + b < 0$$

w คือ น้ำหนักของเวกเตอร์

x คือ เวกเตอร์ของตัวอย่าง

b คือ ไบแอส

ฟังก์ชันเคอร์เนลเป็นวิธีทางคณิตศาสตร์ที่ช่วยให้ซัพพอร์ตเวกเตอร์แมชชีนทำการจำแนกชุดข้อมูลสองมิติของชุดข้อมูลหนึ่งมิติเดิม โดยทั่วไป ฟังก์ชันเคอร์เนลจะฉายข้อมูลจากสเปซที่มีมิติต่ำไปยังสเปซที่มีมิติสูงกว่า กำหนดให้ $K(x_i,x_j)$ เป็น ฟังก์ชัน kernel [17]

- Linear kernel เป็นฟังก์ชันเชิงเส้นตามสูตรดังนี้

$$K(x, x_i) = x \cdot x^T$$

- Polynomial kernel เป็นฟังก์ชันที่มีทิศทางขึ้นอยู่กับทิศทางของเวกเตอร์สองตัวในพื้นที่ ตามสูตรดังนี้

$$K(x, x_i) = (1 + x \cdot x_i^T)^d$$, d = *ดีกรีของฟังก์ชัน*

- Radial Basis Function (RBF)

$$K(x, x_i) = e^{-\gamma \|x - x_i\|^2}, \gamma > 0$$

2.4 วิธีการจำแนกประเภทแบบหลายฉลาก

การจำแนกประเภทคือการทำนายฉลากเป้าหมายแต่ละฉลากได้อย่างแม่นยำ ซึ่งการจำแนกประเภทแบ่งออกเป็น 2 ประเภท คือ การจำแนกประเภทแบบฉลากเดียว (Single-label classification) และ การจำแนกประเภทแบบหลายฉลาก (Multi-label classification) ซึ่งการจำแนกประเภทแบบฉลากเดียว คือ การที่ตัวอย่างการฝึกฝนเชื่อมโยงกับฉลากเดียวแค่ฉลากเดียว แต่ในงานจำแนกประเภทต่าง ๆ เช่น การจำแนกประเภทข้อความ การจัดหมวดหมู่เพลง และการจัดประเภทโรคในผู้ป่วย เป็นต้น อาจมีฉลากที่มากกว่า 1 ฉลากต่อตัวอย่างการฝึกฝน จึงจำเป็นต้องใช้การจำแนกประเภทแบบหลายฉลากมาช่วยในงานประเภทนี้ [4]

2.4.1 Binary Relevance

Binary Relevance คือ วิธีการเรียนรู้การแยกประเภทไบนารี q (q = |B| ซึ่ง B คือ จำนวนคลาส ทั้งหมดในชุดข้อมูล) โดย Binary Relevance จะแปลงชุดข้อมูลต้นฉบับไปเป็น q ชุดข้อมูล ซึ่งชุดข้อมูลแต่ละชุดมีตัวอย่างทั้งหมดของต้นฉบับและทำการฝึกฝนแบบจำลองการจำแนกประเภทในแต่ละชุดข้อมูลที่ฉลากเป็นอิสระต่อกัน ข้อจำกัดข้อ Binary Relevance ไม่สามารถใช้ได้กับฉลากที่มีความสัมพันธ์กันในข้อมูลเพราะ แต่ละ Label เป็นอิสระต่อกันในการฝึกฝนของแบบจำลอง [4] ตามตัวอย่างในรูปที่ 3

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

X	Class1	Class2	Class3
X1	0	0	1
X2	0	0	0
X3	1	0	1

X	Class1
X1	0
X2	0
X3	1

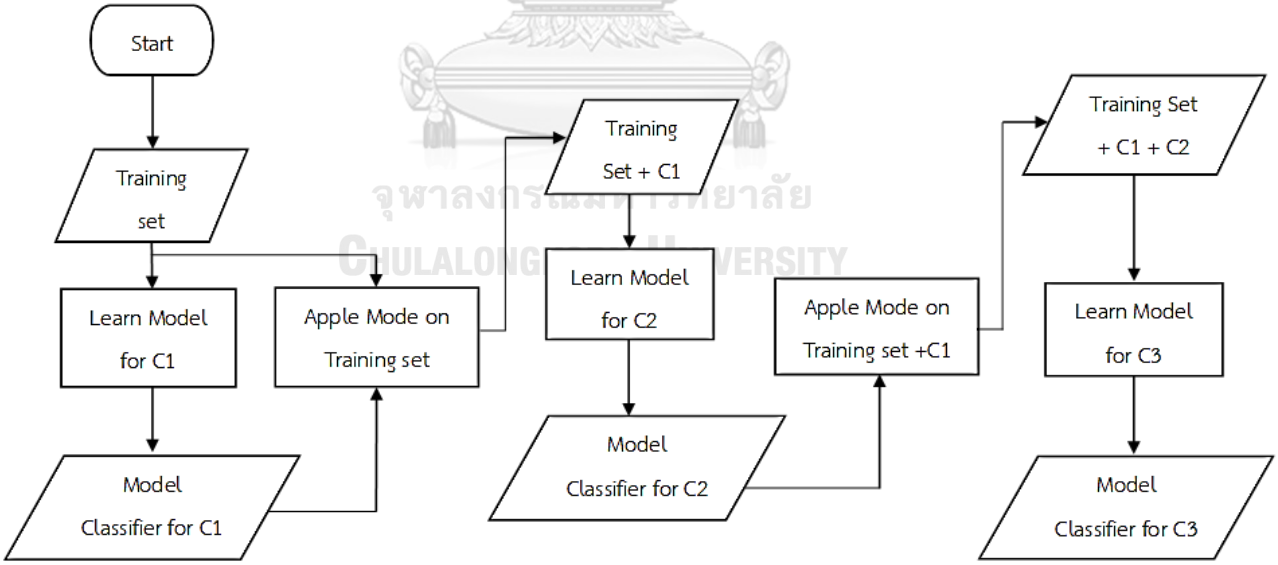
X	Class2
X1	0
X2	0
X3	0

X	Class3
X1	1
X2	0
X3	1

รูปที่ 3 Binary Relevance

2.4.2 Classifier Chains

Classifier Chains เกี่ยวข้องกับการจำแนกแบบไบนารี q เช่นเดียวกับ Binary Relevance โดยจะมาแก้ไขข้อจำกัดของ Binary Relevance ซึ่งคำนึกถึงฉลากที่มีความสัมพันธ์เชื่อมโยงกันเป็นสายโซ่ (Chains) แต่ละลิงค์ในสายโซ่จะแสดงด้วยการเชื่อมโยงฉลากของลิงก์ก่อนหน้าทั้งหมด [4] ตามตัวอย่างในรูปที่ 4



รูปที่ 4 Classifier Chains

2.4.3 Label Power-set

Label Power-set จะพิจารณาการเกิดขึ้นที่ไม่ซ้ำกันของฉลากในชุดข้อมูลการฝึกอบรมแบบหลายฉลากแปลงเป็นหนึ่งคลาสสำหรับชุดข้อมูลที่แปลงใหม่ ตามตัวอย่างในรูปที่ 4 ในตัวอย่าง X1 เชื่อมโยงกับ 3 ฉลาก คือ 0, 0, 1 จะแปลงเป็นฉลากเดียวอันใหม่คือ 1 ดังนั้นชุดข้อมูลที่เปลี่ยนรูปแบบใหม่จึงสามารถใช้แบบจำลองการจำแนกประเภทแบบฉลากเดียวได้ [4] ตามตัวอย่างในรูปที่ 5

X	Class1	Class2	Class3
X1	0	0	1
X2	0	1	0
X3	0	0	1
X4	1	0	1

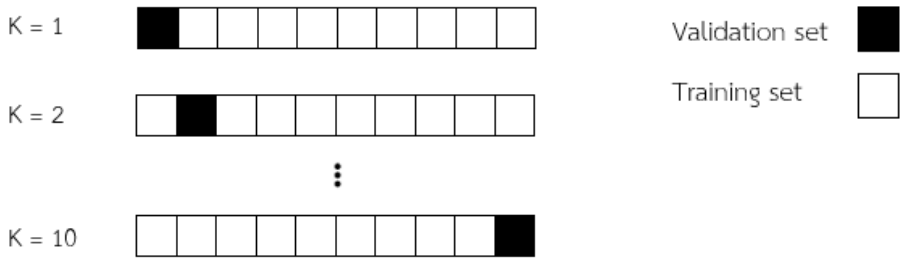
X	Class
X1	1
X2	2
X3	1
X4	3

รูปที่ 5 Label Power-set

2.5 ขั้นตอนวิธีการตรวจสอบไขว้ (K-fold Cross Validation)

ขั้นตอนวิธีการตรวจสอบไขว้เป็นหนึ่งในวิธีการสุ่มตัวอย่างที่ใช้กันอย่างแพร่หลายมากที่สุด เพื่อประเมินข้อผิดพลาดในการทำนายที่แท้จริงของแบบจำลอง โดยวิธีการ K-fold Cross Validation ชุดข้อมูลจะถูกแบ่งเป็นส่วนย่อย K ส่วนที่ไร้สมาชิกร่วม (disjoint) และมีขนาดเท่ากันโดยประมาณ ซึ่งแบบจำลองได้รับการฝึกอบรมโดยใช้ชุดย่อย K-1 ชุด เป็นตัวแทนของชุดการฝึก (Training Set) จากนั้นแบบจำลองจะถูกนำไปใช้กับเซตย่อยที่เหลือ 1 ชุด ซึ่งแสดงเป็นชุดการตรวจสอบ (Validation Set) และวัดประสิทธิภาพ ขั้นตอนนี้ถูกทำซ้ำจนกระทั่งแต่ละเซตย่อย K ทำหน้าที่เป็นชุดการตรวจสอบทั้งหมด แสดงในรูปที่ 6 [5]

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY



รูปที่ 6 ขั้นตอนวิธีการตรวจสอบไขว้

2.6 การวัดประสิทธิภาพของแบบจำลอง

2.6.1 ชุดข้อมูล

การวัดจำนวนนวนฉลากในชุดข้อมูล มีลักษณะสำคัญอยู่ 2 ประการ สำหรับการจำแนกประเภทแบบหลายฉลาก คือ Label Cardinality และ Label Density ซึ่งเป็นอิทธิพลสำหรับประสิทธิภาพของวิธีการจำแนกประเภทหลายฉลาก โดยสามารถคำนวณได้ดังนี้

- Label Cardinality คือ ค่าเฉลี่ยของจำนวนฉลากในชุดข้อมูล [14]

$$Card = \frac{1}{N}\sum_{i=1}^{N}|Y_i|$$

- Label Density คือ ค่าเฉลี่ยของจำนวนฉลากในชุดข้อมูลหารด้วยขนาดของเซตฉลาก [14]

$$Dens = \frac{1}{N}\sum_{i=1}^{N}\frac{|Y_i|}{|L|}$$

N = *จำนวนของชุดข้อมูล*

Y_i = เซตของฉลาก

L = ขนาดของเซตฉลาก

การจำแนกประเภทหลายฉลากจะไม่สามารถประเมินในลักษณะเดียวกับการจำแนกประเภทฉลากเดียว อย่างไรก็ตาม สามารถประเมินได้ 2 วิธีหลัก [10][11]

2.6.2 ตัวชี้วัดประสิทธิภาพการเลือกตอบตามตัวอย่าง (Example-based Metrics)

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

Example-based Metrics คือ การประเมินแต่ละตัวอย่างการทดสอบก่อนหาค่าเฉลี่ยในชุดข้อมูลการทดสอบทั้งหมด

- Hamming loss คือ การคำนวณเพื่อวัดผลการทำนายฉลากที่ไม่ถูกต้องในตัวอย่าง จากนั้นจึงหาค่าเฉลี่ยตัวอย่างทั้งหมดในชุดข้อมูลทดสอบ มีสูตรดังนี้ [10]

$$\text{Hamming loss} = \frac{1}{m}\sum_{i=1}^{m}|Z_i \neq Y_i|$$

- ML-accuracy คือ การคำนวณโดยฉลากที่ทำนายไว้อย่างถูกต้องของกลุ่มตัวอย่าง จากนั้นจึงหาค่าเฉลี่ยตัวอย่างทั้งหมดในชุดข้อมูล มีสูตรดังนี้ [10]

$$\text{ML-accuracy} = \frac{1}{m}\sum_{i=1}^{m}\frac{|Z_i \cap Y_i|}{|Z_i \cup Y_i|}$$

- Subset accuracy คือ การคำนวณเพื่อทำนายฉลากที่เหมือนกันทุกประการกับป้ายกำกับจริง โดยที่ I (True) = 1 และ I (False) = 0 มีสูตรดังนี้ [10]

$$\text{Subset accuracy} = \frac{1}{m}\sum_{i=1}^{m} I|Z_i = Y_i|$$

- ค่าแม่นยำ (Precision) คือ อัตราส่วนของฉลากที่ทำนายไว้อย่างถูกต้องกับจำนวนฉลากที่คาดการณ์ไว้ทั้งหมด จากนั้นจึงหาค่าเฉลี่ยตัวอย่างทั้งหมดในชุดข้อมูลทดสอบ มีสูตรดังนี้ [10]

$$\text{ค่าแม่นยำ (Precision)} = \frac{1}{m}\sum_{i=1}^{m} \frac{|Z_i \cap Y_i|}{|Z_i|}$$

- ค่าเรียกคืน (Recall) คือ อัตราส่วนของฉลากที่ทำนายไว้อย่างถูกต้องกับจำนวนฉลากจริงทั้งหมด จากนั้นจึงหาค่าเฉลี่ยตัวอย่างทั้งหมดในชุดข้อมูลทดสอบ มีสูตรดังนี้ [10]

$$\text{ค่าเรียกคืน (Recall)} = \frac{1}{m}\sum_{i=1}^{m} \frac{|Z_i \cap Y_i|}{|Y_i|}$$

- ตัววัด F1 คือ ค่าเฉลี่ยระหว่าง Precision และ Recall มีสูตรดังนี้ [10]

$$\text{ตัววัด F1 (F-Measure)} = \frac{1}{m}\sum_{i=1}^{m} \frac{2|Z_i \cap Y_i|}{|Z_i| + |Y_i|}$$

2.6.3 ตัวชี้วัดประสิทธิภาพการจำแนกประเภทหลายฉลาก (Label-based Metrics)

Label-based Metrics คือ การคำนวณสำหรับแต่ละฉลากในชุดข้อมูลทดสอบ โดยใช้แนวทางค่าเฉลี่ยระดับไมโคร ประกอบไปด้วย ค่าแม่นยำ (Precision), ค่าเรียกคืน (Recall) และ ตัววัด F1 (F-Measure) โดย B จะเท่ากับ ค่าแม่นยำ (Precision), ค่าเรียกคืน (Recall) และ ตัววัด F1 (F-Measure) [10]

$$B_{micro} = B(\sum_{i=1}^{n} tp_i, \sum_{i=1}^{n} fp_i, \sum_{i=1}^{n} tn_i, \sum_{i=1}^{n} fn_i)$$

m = จำนวนทั้งหมดของตัวอย่างในชุดทดสอบ (test dataset)

n = จำนวนฉลากที่เป็นไปได้ทั้งหมด

Z_i = ผลทำนายฉลาก

Y_i = ผลเฉลยฉลาก

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

2.7 Multi-label confusion matrix (MLCM)

การประเมินขั้นตอนวิธีการเรียนรู้ของเครื่องอย่างรัดกุมและชัดเจนเป็นกุญแจสำคัญในการออกแบบการแยกประเภทและการปรับปรุงประสิทธิภาพ ในการจำแนกประเภทแบบหลายคลาส ซึ่งแต่ละตัวอย่างสามารถถูกระบุว่าเป็นฉลากเดียวเท่านั้น อย่างไรก็ตาม ในการจัดประเภทแบบหลายฉลาก ซึ่งแต่ละตัวอย่างสามารถติดฉลากได้มากกว่าหนึ่งฉลาก ซึ่งในวิธีที่มีอยู่เป็นของ ไพทอนไลบรารี่ sklearn ฟังก์ชั่น multilabel_confusion_matrix คำนวณ True Positive (TP), True Negative (TN), False Positive (FP) และ False Negative (FN) ในแต่ละฉลาก วิธีการนี้ตรวจสอบฉลากครั้งละหนึ่งฉลากเท่านั้นและไม่ได้สนใจฉลากจริงและฉลากที่ทำนายอื่น ๆ ในตัวอย่างส่งผลให้ประสิทธิภาพไม่สมบูรณ์และคลุมเครือ

ในวิธีการ multi-label confusion matrix (MLCM) จากงานวิจัย [16] จะอิงจากปัญหาหลายฉาก 3 ประเภท ดังที่จะอธิบายต่อไปนี้ กำหนดให้ T_i คือ เซตของฉลากจริงที่กำหนดให้กับตัวอย่าง i โดยแบ่งเซตของ T_i เป็น 2 เซตย่อย ประกอบด้วย T_{i1} คือ เซตของชุดฉลากที่ทำนาย และ T_{i2} คือ เซตของชุดฉลากที่ไม่ได้ทำนาย และ P_i คือ เซตของฉลากที่ทำนายสำหรับตัวอย่าง โดยแบ่งเซตของ P_i เป็น 2 เซตย่อย ประกอบด้วย P_{i1} คือ เซตของฉลากที่ทำนายถูกต้อง และ P_{i2} คือ เซตของฉลากที่ทำนายไม่ถูกต้อง จากนิยามนี้เห็นได้ว่า $T_{i1} = P_{i1}$

นอกจากนี้ยังเพิ่มคอลัมน์ No Predicted Label (NPL) หมายถึง สถานการณ์ที่ฉลากจริงหนึ่งรายการหรือมากกว่าไม่ถูกทำนายในขณะที่ไม่มีการทำนายฉลากที่ไม่ถูกต้อง และยังเพิ่มแถว No True Label (NTL) หมายถึง สถานการณ์ในการรวมฉลากจริงและการทำนายฉลากโดยที่ไม่มีฉลากจริงที่กำหนดให้กับตัวอย่าง i ของชุดข้อมูล

2.7.1 $P_i \subseteq T_i$ หมายถึง สถานการณ์ที่ฉลากจริงทั้งหมดหรือบางส่วนได้รับการทำนายอย่างถูกต้องและไม่มีการทำนายที่ไม่ถูกต้อง

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ตัวอย่าง	ฉลากจริง			ฉลากที่ทำนาย		
	C0	C1	C2	C0	C1	C2
1	1	1	0	1	1	0
2	1	1	1	1	0	1
3	0	0	0	0	0	0

สำหรับตัวอย่างที่ 1 ฉลากทั้ง C0 และ C1 ทำนายได้ถูกต้อง ($P_i = T_i$) ดังนั้นค่าใน MLCM คอลัมน์ C0 แถว C0 และ คอลัมน์ C1 แถว C1 ต้องเพิ่มขึ้นหนึ่งเพื่อนับ TP [16]

สำหรับตัวอย่างที่ 2 ฉลาก C0 และ C2 ทำนายได้ถูกต้อง ดังนั้นค่าใน MLCM คอลัมน์ C0 แถว C0 และ คอลัมน์ C2 แถว C2 ต้องเพิ่มขึ้นหนึ่งเพื่อนับ TP แต่ในฉลาก C1 เป็นฉลากที่ไม่ได้ถูกทำนาย ($T_{i2} = \{C1\}$ และ $P_{i2} = \emptyset$} สถานการณ์นี้คือ FN สำหรับ C1 ดังนั้นค่าใน MLCM คอลัมน์ NPL แถว C1 เพิ่มขึ้นหนึ่ง

สำหรับตัวอย่างที่ 3 ไม่มีฉลากใดถูกกำหนดให้กับตัวอย่างนี้และไม่มีการทำนายฉลากสำหรับตัวอย่างนี้ ดังนั้นค่าใน MLCM คอลัมน์ NPL แถว NTL จะเพิ่มขึ้นหนึ่ง

	C0	C1	C2	NPL
C0	1	0	0	0
C1	0	1	0	0
C2	0	0	0	0
NTL	0	0	0	0

	C0	C1	C2	NPL
C0	1	0	0	0
C1	0	0	0	1
C2	0	0	1	0
NTL	0	0	0	0

	C0	C1	C2	NPL
C0	0	0	0	0
C1	0	0	0	0
C2	0	0	0	0
NTL	0	0	0	1

รูปที่ 7 MLCM ตัวอย่างที่ 1,2 และ 3

2.7.2 $T_i \subset P_i$ หมายถึง สถานการณ์ที่ฉลากจริงได้รับการทำนายอย่างถูกต้อง แต่ก็มีการทำนายที่ไม่ถูกต้องเช่นกัน ($T_{i2} = \emptyset$ และ $P_{i2} \neq \emptyset$)

ตัวอย่าง	ฉลากจริง			ฉลากที่ทำนาย		
	C0	C1	C2	C0	C1	C2
4	1	0	0	1	1	1
5	1	1	0	1	1	1
6	0	0	0	0	1	1

สำหรับตัวอย่างที่ 4 ฉลาก C1 และ C2 ถูกทำนายฉลากไม่ถูกต้องแต่จะมีฉลาก C0 ที่ทำนายได้อย่างถูกต้อง ในกรณีนี้ค่าใน MLCM คอลัมน์ C0 แถว C0 ต้องเพิ่มขึ้นหนึ่งเพื่อนับ TP แต่ในแถว C0 จำเป็นต้องได้รับการอัปเดตสำหรับฉลากที่ทำนายไม่ถูกต้องเพิ่มเติมทั้งหมด โดยเพิ่มค่าใน MLCM คอลัมน์ C1 และ C2 ของแถว C0 เพิ่มขึ้นหนึ่งเพื่อนับฉลากที่ทำนายไม่ถูกต้อง

สำหรับตัวอย่างที่ 5 ฉลาก C0 และ C1 ทั้งสองฉลากทำนายได้ถูกต้อง แต่มีฉลากที่ทำนายหนึ่งฉลาก คือ C2 ทำนายไม่ถูกต้อง ในกรณีนี้ค่าใน MLCM คอลัมน์ C0 แถว

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

C0 และ คอลัมน์ C1 แถว C1 ต้องเพิ่มขึ้นหนึ่งเพื่อนับ TP จากนั้นค่าใน MLCM คอลัมน์ C2 จะต้องเพิ่มขึ้นสำหรับทั้งแถว C0 และ C1 เพื่อแสดงความเป็นไปได้ที่ฉลากทำนายไม่ถูกต้องกับฉลากจริงทั้ง C0 และ C1

สำหรับตัวอย่างที่ 6 ไม่มีการกำหนดฉลากสำหรับตัวอย่างที่ 6 แต่ถูกทำนายฉลากเป็น C1 และ C2 ซึ่งไม่ถูกต้อง สำหรับสถานการณ์นี้ ค่าใน MLCM คอลัมน์ C1 และ C2 แถว NTL เพิ่มขึ้นหนึ่งเพื่อแสดงฉลากที่ทำนายไม่ถูกต้องของการไม่มีฉลากให้กับคลาส C1 และ C2

	C0	C1	C2	NPL
C0	1	1	1	0
C1	0	0	0	0
C2	0	0	0	0
NTL	0	0	0	0

	C0	C1	C2	NPL
C0	1	0	1	0
C1	0	1	1	0
C2	0	0	0	0
NTL	0	0	0	0

	C0	C1	C2	NPL
C0	0	0	0	0
C1	0	0	0	0
C2	0	0	0	0
NTL	0	1	1	0

รูปที่ 8 MLCM ตัวอย่างที่ 4,5 และ 6

2.7.3 $T_{i2} \neq \emptyset$ และ $P_{i2} \neq \emptyset$ หมายถึง ฉลากจริงบางส่วนที่ไม่ได้ถูกทำนาย และมีการทำนายฉลากที่ไม่ถูกต้อง

ตัวอย่าง	ฉลากจริง			ฉลากที่ทำนาย		
	C0	C1	C2	C0	C1	C2
7	1	0	0	0	1	1
8	1	1	0	1	0	1
9	1	1	0	0	0	1

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

สำหรับตัวอย่างที่ 7 ฉลากจริงคือ C0 แต่ทำนายฉลาก C1 และ C2 ซึ่งเป็นการทำนายไม่ถูกต้อง ($T_{i1} = \emptyset$ และ $T_{i2} = T_i$) ค่าใน MLCM คอลัมน์ C1 และ C2 ของแถว C0 เพิ่มขึ้นหนึ่งเพื่อแสดงฉลากที่ทำนายไม่ถูกต้อง

สำหรับตัวอย่างที่ 8 ฉลาก C0 ทำนายได้อย่างถูกต้อง ดังนั้นค่าใน MLCM ของคอลัมน์ C0 แถว C0 จึงเพิ่มขึ้นหนึ่งเป็น TP แต่มีฉลาก C1 ที่ไม่ถูกทำนาย (T_{i2} = C1) และ C2 ทำนายผิดพลาด (P_{i2} = C2) ในขั้นตอนวิธีการของ MLCM ประเภทที่ 3 จะเพิ่มค่าที่สอดคล้องกันขององค์ประกอบคอลัมน์ที่เกี่ยวข้องกับฉลากใน P_{i2} สำหรับทุกแถวที่สอดคล้องกับฉลากใน T_{i2} จึงต้องมีค่าเพิ่มขึ้นสำหรับ คอลัมน์ C2 ของแถว C1 ดังนั้นค่าใน MLCM ของคอลัมน์ C2 แถว C1 จะเพิ่มขึ้นหนึ่ง เป็น FN ของฉลาก C1 รวมถึง FP ของฉลาก C2

สำหรับตัวอย่างที่ 9 ไม่มีฉลากใดที่ทำนายได้ถูกต้อง (Ti1 = ø และ Ti2 = Ti) และฉลาก C2 คาดการณ์ไม่ถูกต้อง ดังนั้นค่าใน MLCM คอลัมน์ C2 แถว C0 และ C1 จะเพิ่มขึ้นหนึ่ง

	C0	C1	C2	NPL
C0	0	1	1	0
C1	0	0	0	0
C2	0	0	0	0
NTL	0	0	0	0

	C0	C1	C2	NPL
C0	1	0	0	0
C1	0	0	1	0
C2	0	0	0	0
NTL	0	0	0	0

	C0	C1	C2	NPL
C0	0	0	1	0
C1	0	0	1	0
C2	0	0	0	0
NTL	0	0	0	0

รูปที่ 9 MLCM ตัวอย่างที่ 7,8 และ 9

วิธีการ multi-label confusion matrix ที่อิงจากปัญหาหลายฉลาก 3 ประเภทข้างต้นในตัวอย่างที่ 1-9 ผลของ MLCM ในแต่ละตัวอย่างจะถูกรวมเป็น multi-label confusion matrix ที่ประกอบไปด้วยทุกฉลากผลลัพธ์ตามรูปที่ 11

	C0	C1	C2	NPL
C0	5	2	4	0
C1	0	2	3	0
C2	0	0	1	0
NTL	0	0	0	1

รูปที่ 10 Total MLCM

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

2.8 งานวิจัยที่เกี่ยวข้อง

2.8.1 การตัดคำภาษาไทย

งานวิจัยการตัดคำภาษาไทยมีหลายวิธีการในการตัดคำ และมีหลายเครื่องมือให้เลือกใช้ ในงานวิจัยนี้เลือกใช้วิธีการตัดคำแบบ Dictionary based อัลกอริทึม Maximal Matching โดยใช้เครื่องมือในภาษาไพทอน คือ pythainlp และ วิธีการตัดคำแบบ โครงข่ายประสาทแบบคอนโวลูชัน (CNN) โดยใช้เครื่องมือในภาษาไพทอน คือ Deepcut งานวิจัยที่เกี่ยวข้องมีดังต่อไปนี้

งานวิจัยของ Choochart Haruechaiyasak [2] งานวิจัยนี้จะวิเคราะห์และเปรียบเทียบวิธีการตัดคำในภาษาไทย ซึ่งแบ่งออกเป็น 2 ประเภทที่แตกต่างกัน คือ Dictionary-based และ Machine learning-based โดยวิธีการตัดคำโดยใช้ Dictionary-based จะใช้ชุดข้อมูลจากพจนานุกรมสำหรับการวิเคราะห์และตัดคำ วิธีการนี้จะเข้าไปค้นหาลำดับของอักขระในพจนานุกรมเพื่อจับคู่คำที่ถูกต้องตามอัลกอริทึมคือ Maximal matching, Longest matching แต่การตัดคำโดยใช้พจนานุกรมจะมีปัญหาคำที่ไม่มีในพจนานุกรมจะไม่สามารถตัดคำได้ โดยประสิทธิภาพของวิธีการตัดคำโดยใช้ Dictionary-based ขึ้นอยู่กับคุณภาพและขนาดของพจนานุกรม และสามารถปรับปรุงประสิทธิภาพโดยการเพิ่มคำใหม่หรือคำเฉพาะที่ไม่มีในพจนานุกรมลงในพจนานุกรมที่ใช้สำหรับกระบวนการตัดคำได้ ส่วนของการตัดคำโดยใช้วิธี Machine learning-based จะอาศัยแบบจำลองทางสถิติจากเทคนิคการเรียนรู้ด้วยเครื่อง คือ ต้นไม้ตัดสินใจ (Decision Tree), นาอีฟเบย์ (Naïve Bayes), ซัพพอร์ตเวกเตอร์แมชชีน, แบบจำลองคอนดิชันนอลแรนดอมฟิลด์ส (conditional random field) โดยเทคนิคนี้ใช้การติดแท็กคำโดยมีการระบุอักขระที่เริ่มต้นของคำ และอักขระภายในคำนั้น ๆ ข้อดีของ Machine learning-based ไม่ต้องมีพจนานุกรมรองรับในการตัดคำ แต่จะขึ้นอยู่คลังข้อมูลที่นำมาฝึกฝนแบบจำลอง และปัญหาคำที่ไม่รู้จักและคำที่กำกวมจะได้รับการจัดการโดยการเตรียมชุดตัวอย่างการฝึกฝนของแบบจำลองมากเพียงพอเพื่อให้การตัดคำได้แม่นยำ

งานวิจัยของ Pattarawat Chormai [6] จะอธิบายเกี่ยวกับการปรับปรุงประสิทธิภาพของความเร็วและความแม่นยำของการตัดคำภาษาไทยที่เรียกว่า AttaCut เปรียบเทียบกับระบบการตัดคำที่มีอยู่เช่น pythainlp, DeepCut และ Sertis เป็นต้น การตัดคำวิธีการ AttaCut จะวิเคราะห์และปรับปรุงวิธีจากสถาปัตยกรรมของ DeepCut ซึ่งเป็นวิธีโครงข่ายประสาทแบบคอนโวลูชัน (CNN) โดยจะลดชั้นของคอนโวลูชันเหลือ 3 ชั้น และทำคอนโวลูชั่นแบบขยาย (Dilated Convolution) ทำให้การประมวลผลเร็วขึ้น และมีการใช้

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ลักษณะของอักขระกับพยางค์มาช่วยในการระบุการตัดคำให้มีความถูกต้อง ผลประสิทธิภาพความแม่นยำการตัดคำของวิธีการ AttaCut จะมีความใกล้เคียงกับวิธีการ DeepCut บนชุดข้อมูล BEST-2010 แต่ประสิทธิภาพความเร็วในการประมวลผลของวิธีการ AttaCut เร็วกว่าวิธีการ DeepCut อย่างน้อย 5.6 เท่า แต่เมื่อมาเทียบประสิทธิภาพกับชุดข้อมูล TNHC เป็นชุดข้อมูลเกี่ยวกับวรรณกรรมที่เขียนด้วยบทกวีกลับกลายเป็นวิธีการของ pythainlp สามารถตัดคำได้มีความแม่นยำที่สุด

จากการศึกษาวิธีการตัดคำทั้งแบบ Dictionary-Based ซึ่งเป็นวิธีการ maximal Matching , Machine learning-based และ วิธีการตัดคำแบบโครงข่ายประสาทแบบลึกที่เป็นเครื่องมือ DeepCut และ Attacut โดยถูกเปิดเป็นโอเพนซอร์ส (Opensource) ให้ทุกคนสามารถเข้ามาใช้งานได้ พบว่าในบางชุดข้อมูลวิธีการ Dictionary-Based จะสามารถตัดได้ดีกว่าแบบวิธีการตัดคำแบบโครงข่ายประสาท และการเลือกใช้เครื่องมือต่าง ๆ อาจต้องคำนึกถึงความเร็วในการประมวลผล เช่น การเลือกใช้เครื่องมือ DeepCut กับชุดข้อมูลที่มีขนาดใหญ่อาจไม่เหมาะสมเพราะจะทำให้เวลาในการประมวลนานเกินไป เมื่อเทียบกับเครื่องมือของ AttaCut เป็นต้น หรือ อาจต้องคำนึงถึงชุดข้อมูลที่ถูกนำมาฝึกฝนในการตัดคำทั้งวิธีการ Machine learning-based และ วิธีการตัดคำแบบโครงข่ายประสาทแบบลึก

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ตารางที่ 1 เปรียบเทียบงานวิจัยการตัดคำภาษาไทย

งานวิจัยที่เกี่ยวข้อง	เนื้อหางานวิจัย	สรุปผล
A comparative study on Thai word segmentation approaches	อธิบายวิธีการตัดคำภาษาไทยและข้อดีข้อเสียของการใช้พจนานุกรมและการเรียนรู้ด้วยเครื่อง	• การตัดคำใช้พจนานุกรมไม่สามารถตัดคำที่กำกวมหรือที่ไม่มีในพจนานุกรมได้แต่สามารถปรับปรุงประสิทธิภาพโดยการเพิ่มคำลงในพจนานุกรม • การตัดคำด้วยวิธีการเรียนรู้ด้วยเครื่องไม่ต้องมีพจนานุกรมแต่จะขึ้นอยู่กับชุดข้อมูลในการเรียนรู้และแก้ไขปัญหาของคำที่กำกวมโดยการเตรียมชุดตัวอย่างการฝึกฝนของแบบจำลองมากเพียงพอเพื่อให้การตัดคำได้แม่นยำ
AttaCut: A Fast and Accurate Neural Thai Word Segmenter	การตัดคำด้วยวิธีโครงข่ายประสาทแบบคอนโวลูชันปรับปรุงประสิทธิภาพของสถาปัตยกรรมของ DeepCut และมีการใช้ลักษณะของอักขระกับพยางค์มาช่วยในการระบุการตัดคำให้มีความถูกต้องมากขึ้น	• ความเร็วในการตัดคำของวิธีการ AttaCut เร็วกว่าวิธีการ DeepCut อย่างน้อย 5.6 เท่า บนชุดข้อมูล BEST-2010 แต่มีความแม่นยำใกล้เคียงกัน

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

2.8.2 การสกัดคำสำคัญ

งานวิจัยของ Shahzad Qaiser [7] งานวิจัยนี้จะอธิบายอัลกอริทึม TF-IDF ในการตรวจสอบคำสำคัญจากเอกสารที่มาจากเว็บไซต์ต่างที่แตกต่างกัน 5 Domain มาอย่างละ 5 เว็บไซต์ ซึ่งอัลกอริทึม Term frequency-inverse document frequency (TF-IDF) คือ การสกัดคำสำคัญที่เกี่ยวข้องกับเอกสารต่าง ๆ เป็นการรวมกันของ term frequency กับ inverse document frequency

- Term frequency คือ การนับคำศัพท์ที่เกิดขึ้นในเอกสารนั้น ยกตัวอย่างในคำภาษาไทย เช่น เอกสารหนึ่งมีคำทั้งหมด 1000 คำ โดยมีคำว่า ‘สวนสัตว์’ อยู่ 10 คำ ในเอกสารนั้น วิธีการคิดคือ จำนวนคำที่เกิดขึ้นที่เอกสารใด ๆ จะถูกหารด้วยจำนวนคำทั้งหมดที่มีอยู่ในเอกสาร ในกรณีนี้ จะคำนวณค่า TF ได้ดังนี้
    - TF = 10/1000 = 0.01
- Inverse document frequency คือ การคำนวณของ term frequency จะสังเกตได้ว่าขั้นตอนวิธีนี้ถือว่าทุกคำเป็น คำสำคัญเท่ากันหมด ซึ่งไม่ถูกต้อง เพราะทุกคำสำคัญมีความสำคัญที่ต่างกัน ซึ่ง inverse document frequency กำหนดน้ำหนักน้อยให้คำที่เกิดขึ้นบ่อยครั้งในเอกสารทั้งหมด และ กำหนดน้ำหนักของคำนั้นมากขึ้นสำหรับคำที่ถูกพบไม่บ่อยในเอกสารทั้งหมด เช่น
    - idf = log(จำนวนเอกสารทั้งหมด/จำนวนเอกสารที่มีคำนั้นปรากฏอยู่)

TF-IDF คือ การนำ Term frequency คูณกับ Inverse document frequency จะอธิบายได้ว่า ถ้าคำที่อยู่ในเอกสารนั้นมีความถี่สูง หมายถึง คำที่ปรากฏในเอกสารทั้งหมดมีไม่มาก และ ถ้าคำที่อยู่ในเอกสารนั้นมีความถี่น้อย หมายถึง คำที่ปรากฏบ่อยครั้งในเอกสารนั้น ๆ หรือ ปรากฏบ่อยครั้งในเอกสารทั้งหมด ซึ่งคำที่มีความถี่ที่สูงสามารถใช้เป็น คำสำคัญในเอกสารนั้นได้ ข้อจำกัดที่สำคัญของขั้นตอนวิธี TF-IDF คือ TF-IDF ไม่สามารถระบุคำที่เปลี่ยนเปลงเพียงเล็กน้อยแต่ยังมีความหมายเดิม เช่น go และ goes จะถูกนับเป็นคนละคำกันและ TD-IDF ไม่สามารถระบุความหมายของข้อความในเอกสารได้ โดย ขั้นตอนวิธี TF-IDF มีประโยชน์ถึงระดับคำศัพท์เท่านั้นจึงต้องมีการเตรียมข้อมูลให้ดีก่อนนำมาใช้กับ ขั้นตอนวิธี TF-IDF

งานวิจัยของ Tanatorn Tanantong [8] นำเสนอเกี่ยวกับการสกัด คำสำคัญข้อความภาษาไทย จาก Social Media Twitter ใช้เทคนิค A N-gram-based word-

combination เพื่อช่วยอำนวยความสะดวกให้ประชาชนทั่วไปอัพเดทข่าวสาร และยังช่วยลดเวลาในการระบุเนื้อหาหลักจากข้อมูลมหาศาล โดยข้อมูลที่นำมาวิเคราะห์ ใช้ Twitter API ดึงข้อมูล เกี่ยวกับ 40 มหาลัยในประเทศไทย ผ่านการเตรียมข้อมูลโดยการตัดคำใช้วิธีการ Maximal Matching จากคลังข้อมูลไทย LEXiTron ที่ถูกพัฒนาโดย NECTEC และทำการลบคำที่ไม่สำคัญออกเป็นอีกวิธีที่สำคัญในการเตรียมข้อมูลก่อนนำมาสกัด คำสำคัญในส่วนของการสกัด คำสำคัญจะใช้วิธีการรวมคำที่เกิดจากการเตรียมข้อมูลมาเปรียบเทียบกับข้อความทวิตเตอร์เดิมถ้าคำที่เกิดจากการเตรียมข้อมูลอยู่ในตำแหน่งที่ติดกันจะถูกพิจารณาเป็นคำ ๆ เดียวกันและจะถูกพิจารณาความถี่ของคำนั้นที่เกิดขึ้นในแต่ละข้อความทวิตเตอร์ ซึ่งการวัดประสิทธิภาพของการสกัด คำสำคัญจะเปรียบเทียบกับ คำสำคัญที่ถูกกำหนดโดยผู้ประเมินที่เป็นมนุษย์ ถ้าคำสำคัญมีความคล้ายคลึงกันมากกว่าหรือเท่ากับ 75 เปอร์เซ็นต์จะกำหนดให้คำสำคัญที่สกัดมาถูกต้องและใช้การวัดความแม่นยำเป็นตัวหลักในการวัดประสิทธิภาพของการสกัดคำสำคัญ สรุปผลประสิทธิภาพของการวัดความแม่นยำการสกัด คำสำคัญมากกว่า 70 เปอร์เซ็นต์ บนข้อมูลทวิตเตอร์ที่เกี่ยวกับมหาวิทยาลัยในประเทศไทย

จากการศึกษางานวิจัยเกี่ยวกับการสกัดคำสำคัญ การสกัดคำสำคัญจากเอกสารเพื่อให้รู้ความหมายสำคัญของเอกสารนั้นหรือนำไปจำแนกประเภทของเอกสารได้ ก่อนที่จะสกัดคำสำคัญจะต้องทำการเตรียมข้อมูลก่อน เช่น การตัดคำ การลบคำหยุด การลบเครื่องหมายต่าง ๆ เป็นต้น เพื่อให้การสกัดคำสำคัญมีประสิทธิภาพมากขึ้น โดยงานวิจัยนี้ใช้วิธีการ Term frequency-inverse document frequency (TF-IDF)

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ตารางที่ 2 เปรียบเทียบงานวิจัยการสกัดคำสำคัญ

งานวิจัยที่เกี่ยวข้อง	เนื้อหางานวิจัย	สรุปงานวิจัย
Text Mining: Use of TF-IDF to Examine the Relevance of Words to Documents	อธิบายวิธีการ TF-IDF ในการสกัดคำสำคัญจากเอกสาร โดยจะมีการคำนวณอยู่ 2 ส่วน คือ (1) Term frequency คือ การนับคำศัพท์ที่เกิดขึ้นในเอกสารนั้น (2) Inverse document frequency คือ การกำหนดน้ำหนักน้อยให้คำที่เกิดขึ้นบ่อยครั้งในเอกสารทั้งหมด และ กำหนดน้ำหนักของคำนั้นมากขึ้นสำหรับคำที่ถูกพบไม่บ่อยในเอกสารทั้งหมด	• การนำวิธีการ TF-IDF มาสกัดคำสำคัญ คือการนำ Term frequency คูณกับ Inverse document frequency ถ้าคำที่มีความถี่สูงจะสามารถใช้เป็นคีย์เวิร์ดในเอกสารนั้นได้ • ข้อจำกัด TF-IDF ไม่สามารถระบุคำที่มีความหมายเหมือนกันแต่ลักษณะต่างรูปกันได้ จะต้องเตรียมข้อมูลก่อนมาใช้งานวิธีการ TF-IDF
Extraction of Trend Keywords from Thai Twitters using N-Gram Word Combination	การสกัดคำสำคัญในงานวิจัยนี้ใช้วิธีการเปรียบเทียบข้อความที่ตัดคำด้วยวิธีการ Maximal Matching กับข้อความเดิม ถ้าข้อความที่ถูกตัดมามีตำแหน่งติดกันจะถูกพิจารณาเป็นคำเดียวกันและจะถูกพิจารณาความถี่ของคำนั้นที่เกิดขึ้นในแต่ละข้อความม	• การวัดประสิทธิภาพของการสกัด คำสำคัญต้องมีความคล้ายคลึงกันกับผลเฉลยมากกว่าหรือเท่ากับ 75 เปอร์เซ็นต์ ซึ่งประสิทธิภาพของการวัดความแม่นยำการสกัดคำสำคัญได้มากกว่า 70 เปอร์เซ็นต์ บนข้อมูลทวิตเตอร์ที่เกี่ยวกับมหาวิทยาลัยในประเทศไทยเท่านั้น

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

2.8.3 การจำแนกประเภทข้อความ

2.8.3.1 การจำแนกประเภทบทความแบบฉลากเดียว

งานวิจัยของ Tran Thanh Dien [3] อธิบายเกี่ยวกับการจำแนกประเภทของบทความจากวารสารหรือนิตยสาร ที่จะช่วยลดเวลาในการคัดแยกทั้งผู้เขียนและกองบรรณาธิการ วิธีการจำแนกข้อความที่แตกต่างกันมีมากมายยกตัวอย่าง เช่น เพื่อนบ้านใกล้สุดเคตัว (K Nearest Neighbors), ซัพพอร์ตเวกเตอร์แมชชีน, นาอีฟเบย์ และ โครงข่ายประสาทเทียม (neural network) เป็นต้น โดยงานวิจัยนี้เสนอแนวทางการจัดประเภทบทความอัตโนมัติที่ส่งผ่านระบบออนไลน์ ระบบนี้จะดึงข้อมูลของชื่อเรื่องและบทคัดย่อของผู้เขียน ใช้วิธีการสกัดข้อมูล TF-IDF และจัดหมวดหมู่หัวข้อให้เหมาะสมบนตัวอย่าง 2 data sets ของบทความ คือ บทความทางวิทยาศาสตร์โดยจะแยกประเภทของหัวข้อทั้งหมด 10 หัวข้อ คือ เทคโนโลยี, เศรษฐศาสตร์, การศึกษา เป็นต้น และหัวข้อข่าว VNEXPRESS ประกอบด้วย 10 หัวข้อ คือ ธุรกิจ, สุขภาพ, ไอที, กฎหมาย, กีฬา เป็นต้น แบบจำลองที่ถูกใช้เปรียบเทียบการจำแนกประเภทของข้อความมีอยู่ 3 แบบจำลอง คือ ซัพพอร์ตเวกเตอร์แมชชีน, เพื่อนบ้านใกล้สุดเคตัว และ นาอีฟเบย์ ประสิทธิภาพของแบบจำลองที่สามารถจำแนกประเภทของข้อความได้ดีที่สุด คือ เพื่อนบ้านใกล้สุดเคตัว มีค่าความถูกต้องอยู่ที่ 91 เปอร์เซ็นต์

งานวิจัยของ Muhammad Azam [9] อธิบายถึงปัญหาของการจำแนกประเภทของเอกสารเพราะการตีพิมพ์ทางวิทยาศาสตร์เพิ่มขึ้นอย่างมากและการเพิ่มขึ้นของการตีพิมพ์ทางวิทยาศาสตร์ต้องมีประสิทธิภาพในการจำแนกบทความของเอกสารต่าง ๆ งานวิจัยนี้วิเคราะห์ประสิทธิภาพของแบบจำลองการจำแนกประเภท 2 อย่างคือ เพื่อนบ้านใกล้สุดเคตัว และ นาอีฟเบย์ ซึ่งนำวิธีการ Bagging กับ Boosting มาช่วยเพิ่มประสิทธิภาพการจำแนกประเภท โดยใช้ดาต้าเซตจาก Scopus 10,000 เอกสาร ชุดข้อมูลประกอบด้วย 5 ประเภท คือ แพทย์ศาสตร์, คณิตศาสตร์, การเงิน, เกษตรกรรมและวิทยาศาสตร์ชีวภาพ และวิศวกรรมศาสตร์ โดยใช้ข้อความจากบทคัดย่อมาจำแนกประเภทของบทความซึ่งก่อนที่จะนำมาจำแนกประเภทของข้อความจะต้องนำข้อความจากบทคัดย่อมาเตรียมข้อมูล เช่น การตัดคำ, การลบคำที่ไม่สำคัญออก เป็นต้น แล้วจึงเปลี่ยนให้เป็นเวกเตอร์ของคำด้วยวิธีการ TF-IDF โดยจะกำหนดคำที่เกิดขึ้นให้เป็นความถี่แล้วจึงมาจำแนกประเภทของข้อความ แบบจำลองจะถูกประเมินจากค่าสำคัญ 3 ค่าคือ ค่าความถูกต้อง (Accuracy), ค่าแม่นยำ (Precision) และ ค่าเรียกคืน (Recall) สรุปผลเปรียบเทียบประสิทธิภาพของแบบจำลองการจำแนกประเภทแบบเพื่อนบ้านใกล้สุดเคตัว ด้วยวิธีการ Bagging และ Boosting

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ดีกว่าประสิทธิภาพของแบบจำลองการจำแนกประเภทแบบนาอีฟเบย์ ด้วยวิธีการ Bagging และ Boosting

งานวิจัยของ Tipajin Thaipisutikul [18] มุ้งเน้นที่จะพัฒนาการจำแนกประเภทและการสร้างภาพนามธรรม (Visualization) ของอาชญากรรมและความรุนแรงโดยอัตโนมัติจากบทความข่าวออนไลน์ภาษาไทย ประกอบด้วย 5 หมวดหมู่อาชญากรรม คือ การลักทรัพย์ ยาเสพติด ฆาตกรรม อุบัติเหตุ และ การทุจริต โดยเปรียบเทียบวิธีการจำแนกประเภท 6 แบบจำลอง คือ นาอีฟเบย์แบบอเนกนาม (Multinomial Naive Bayes), แบบจำลองต้นไม้การตัดสินใจส่งเสริมการไล่ระดับสี (Gradient Boosting Machine), ป่าไม้สุ่ม (Random Forest), เพื่อนบ้านใกล้สุดเคตัว, การวิเคราะห์ถดถอยโลจิสติกพหุวิภาค (Multinomial Logistic Regression) และ ซัพพอร์ตเวกเตอร์แมชชีน นอกจากนี้ยังดึงข้อมูลทางภูมิศาสตร์จากเหตุการณ์ความรุนแรงแต่ละเหตุการณ์เพื่อระบุตำแหน่งที่แน่นอนจากนั้นป้อนข้อมูลไปยังการสร้างภาพนามธรรมแสดงเหตุการณ์ที่เกิดขึ้นเชิงพื้นที่ของเหตุการณ์ความรุนแรงแต่ละประเภท เพื่อวิเคราะห์รูปแบบอาชญากรรมในการป้องกันชุมชน เนื่องจากปัญหาอาชญากรรมและความรุนแรงที่เพิ่มขึ้นเป็นจำนวนมากส่งผลกระทบต่อสังคมและเศรษฐกิจ โดยงานวิจัยนี้รวบรวมบทความข่าวออนไลน์ภาษาไทยจากเว็บไซต์ Sanook จำนวน 7,879 บทความ การเตรียมข้อมูลใช้วิธีการ NLTK และ Lexto สำหรับการตัดคำและใช้วิธี TF-IDF สำหรับการสกัดข้อมูล ผลการทดลองพบว่าประสิทธิภาพของแบบจำลองที่สามารถจำแนกประเภทของบทความข่าวอาชญากรรมและความรุนแรงได้ดีที่สุด คือ ซัพพอร์ตเวกเตอร์แมชชีน มีค่าความถูกต้องอยู่ที่ 79.41 เปอร์เซ็นต์

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ตารางที่ 3 เปรียบเทียบงานวิจัยการจำแนกประเภทข้อความแบบฉลากเดียว

งานวิจัยที่เกี่ยวข้อง	เนื้อหางานวิจัย	สรุปงานวิจัย
Article Classification using Natural Language Processing and Machine Learning	อธิบายประสิทธิภาพของแบบจำลองในการจำแนกประเภทข้อความและสกัดคำสำคัญด้วยวิธีการ TF-IDF	• ประสิทธิภาพของแบบจำลองที่สามารถจำแนกประเภทของข้อความได้ดีที่สุด คือ เพื่อนบ้านใกล้สุดเคตัว มีค่าความถูกต้องอยู่ที่ 91 เปอร์เซ็นต์ ซึ่งดีกว่า แบบจำลองซัพพอร์ตเวกเตอร์แมชชีน และ นาอีฟเบย์
Feature Extraction based Text Classification using K-Nearest Neighbor Algorithm	เปรียบเทียบประสิทธิภาพของแบบจำลองจำแนกประเภทซึ่งใช้ ซึ่งนำวิธีการ Bagging กับ Boosting มาช่วยเพิ่มประสิทธิภาพการจำแนกประเภท และสกัดคำสำคัญด้วยวิธีการ TF-IDF	• ประสิทธิภาพของแบบจำลองเพื่อนบ้านใกล้สุดเคตัว ทั้งวิธีการ Bagging กับ Boosting ดีกว่าแบบจำลองนาอีฟเบย์
Automated classification of criminal and violent activities in Thailand from online news articles	เปรียบเทียบประสิทธิภาพของแบบจำลองจำแนกประเภท และ สร้างภาพนามธรรม สำหรับบทความข่าวภาษาไทยอาชญากรรมและความรุนแรง	• ประสิทธิภาพของแบบจำลองที่สามารถจำแนกประเภทของบทความข่าวอาชญากรรมและความรุนแรงได้ดีที่สุด คือ ซัพพอร์ตเวกเตอร์แมชชีน มีค่าความถูกต้องอยู่ที่ 79.41 เปอร์เซ็นต์

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

2.8.3.2 การจำแนกประเภทบทความแบบหลายฉลาก

งานวิจัยของ Jadon Mayurisingh Nareshpalsingh [4] อธิบายและแนะนำวิธีการต่าง ๆ ของการจำแนกประเภทหลายฉลาก (multi-label classification) ยกตัวอย่างการจำแนกประเภทหลายฉลาก เช่น การวินิจฉัยทางการแพทย์ผู้ป่วยอาจป่วยได้มากกว่า 1 โรคพร้อมกัน เป็นต้น โดยขั้นตอนของการจำแนกประเภทแบบหลายฉลากจะมีด้วยกันอยู่ 2 ขั้นตอน คือ

1. ขั้นตอนการเปลี่ยนรูปของการจำแนกประเภทแบบหลายฉลากให้เป็นการจำแนกประเภทฉลากเดียวแล้วจากนั้นจะดำเนินการจำแนกประเภทในลักษณะเดียวกับการจำแนกประเภทฉลากเดียว
2. การปรับขั้นตอนวิธีซึ่งวิธีการของการจำแนกประเภทฉลากเดียวจะได้รับการปรับปรุงแก้ไขแล้วนำไปใช้โดยตรงกับการจำแนกประเภทแบบหลายฉลาก

โดยขั้นตอนการเปลี่ยนรูปของการจำแนกประเภทหลายฉลากจะวิธีอยู่หลายวิธี และแต่ละวิธีความแม่นยำก็จะต่างกันออกไป เช่น Binary Relevance เป็นแนวคิดที่ค่อนข้างง่ายและรวดเร็วแต่เมื่อคลาสไม่สมดุลกันจะได้รับผลกระทบจากฉลากที่เป็นอิสระต่อกันด้วย Classifier Chains จะมาแก้ไขข้อจำกัดวิธีการ Binary Relevance โดยจะคำนึกความสัมพันธ์ของฉลากที่เชื่อมโยงกัน ซึ่งช่วยให้ประสิทธิภาพทำนายผลลัพธ์เพิ่มขึ้นได้ แต่จะไม่สามารถใช้ข้อมูลที่ไม่มีฉลากได้ เป็นต้น

งานวิจัยของ Nawal Aljedani [10] มุ่งเน้นที่จะนำเสนอการทบทวนวิธีการและเทคนิคการจำแนกประเภทหลายฉลากที่มีอยู่ ซึ่งสามารถจัดการกับปัญหาหลายฉลากได้ และมุ่งเน้นที่ภาษาอาหรับที่เกี่ยวข้องของการจำแนกหลายฉลาก บทความนี้ยังนำเสนอการเปรียบเทียบเชิงทดลองของวิธีการจำแนกประเภทหลายฉลากต่างๆ ที่ใช้กับภาษาอาหรับโดยการเติบโตอย่างมากของเอกสารข้อความบนเว็บ ส่งผลให้มีความต้องการที่จะจัดระเบียบและจัดประเภทเอกสารอิเล็กทรอนิกส์โดยอัตโนมัติ ซึ่งในภาษาอาหรับมีการศึกษาวิจัยน้อยและไม่เพียงพอที่ตรวจสอบปัญหาการจำแนกข้อความหลายฉลาก งานวิจัยนี้ได้ทดลองกับชุดข้อมูล RTAnews โดยมีตัวอย่างทั้งหมด 23,837 จำนวนฉลากมีทั้งหมด 40 ฉลาก เพื่อให้การประเมินมีความน่าเชื่อถือมากขึ้น งานวิจัยนี้จะใช้ขั้นตอนวิธีการตรวจสอบไขว้โดย K = 10 ซึ่งทำการทดลอง 2 การทดลอง คือ

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

1. การทดลองที่ 1 ตรวจสอบประสิทธิภาพของวิธีการจำแนกแบบหลายฉลาก คือ Binary Relevance (BR), Classifier Chains (CC), Label Power-set (LP), Pruned Set (PS), ML-KNN และ Hierarchy of Multilabel classifiers (HOMER) ซึ่งวิธีการ Problem Transformation (PT) คือ BR, CC, LP, PS ประเมินโดยใช้นาอีฟเบย์ ในการจำแนกประเภท , ขั้นตอนวิธี HOMER ทำงานโดยใช้ตัวจำแนกประเภท คือ BR และ นาอีฟเบย์ ในขณะที่ ML-KNN ได้รับการประเมินโดยกำหนด K = 10 สรุปผลการทดลองวิธีการ ML-KNN โดย K=10 จะมีผลลัพธ์ที่ดีโดยการวัดค่า Hamming loss, subset accuracy, ค่าเฉลี่ยไมโครสำหรับค่าแม่นยำ ค่าเฉลี่ยไมโครสำหรับตัววัด F1 ส่วนค่า ML-Accuracy นั้น LP มีค่าดีที่สุดคือ 0.6219 ในขณะเดียวกัน micro-averaged recall = 0.8552 เป็นผลลัพธ์ที่ดีที่สุด ซึ่งได้จากวิธี BR
2. การทดลองที่ 2 มีจุดมุ่งหมายเพื่อวัดผลกระทบของขั้นตอนวิธีการจำแนกประเภท 3 ขั้นตอนวิธี คือ นาอีฟเบย์, ซัพพอร์ตเวกเตอร์แมชชีน และ ต้นไม้ตัดสินใจ ที่เกี่ยวกับประสิทธิภาพของวิธีการ PT คือ BR, CC, LP และ PS ซึ่งประสิทธิภาพการจำแนกประเภทของวิธีการ PT ขั้นตอนวิธีซัพพอร์ตเวกเตอร์แมชชีน จะให้ผลดีที่สุดด้วยวิธีการ PT ทั้งหมด และมีผลลัพธ์ที่ดีในตัวชี้วัดการประเมินคือ Hamming loss, ML-accuracy, Subset accuracy, ค่าเฉลี่ยไมโครสำหรับค่าแม่นยำ และ ค่าเฉลี่ยไมโครสำหรับตัววัด F1 ในขณะที่ ค่าเฉลี่ยไมโครสำหรับค่าเรียกคืน การจำแนกประเภทนาอีฟเบย์ ด้วยวิธี BR และ CC ได้รับผลลัพธ์ที่ดีที่สุด

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

งานวิจัยของ Edward K. Y. Yapp [11] งานวิจัยนี้จะเปรียบเทียบวิธีการจำแนกแบบหลายฉลาก โดยใช้แบบจำลองในการจำแนกประเภท 4 วิธีการ คือ ซัพพอร์ตเวกเตอร์แมชชีน, เพื่อนบ้านใกล้สุดเคตัว, นาอีฟเบย์ และ ต้นไม้ตัดสินใจ ในชุดข้อมูล 11 ชุดข้อมูล ซึ่งชุดข้อมูล 5 ชุดข้อมูลแรก จะใช้ขั้นตอนวิธีการตรวจสอบไขว้ โดย K = 10 และ 6 ชุดข้อมูลหลัง ใช้วิธีการ train-test split ในการประเมินใช้วิธีการประเมิน 16 วิธีการ และการจำแนกประเภทแบบหลายฉลากในงานวิจัยนี้ใช้ 7 วิธีการมาเปรียบเทียบกัน แบ่งออกเป็น 2 ประเภท คือ

1.) Problem Transformation ประกอบด้วย 5 วิธีการ

1.1 Binary Relevance (BR)

1.2 Classifier Chain (CC)

1.3 Calibrated Label Ranking (CLR)

1.4 Quick Weighted Multi-Label Learning (QWML)

1.5 Hierarchy of Multilabel Classifiers (HOMER)

2.) Ensemble Method ประกอบด้วย 2 วิธีการ

2.1 Random k-labelsets (RAkEL)

2.2 Ensemble Classifier Chain (ECC)

โดยการทดลองเปรียบเทียบความสัมพันธ์ระหว่างแบบจำลองการจำแนกประเภท และ ขนาดของชุดข้อมูล พบว่า การจำแนกประเภทต้นไม้ตัดสินใจ ทำงานได้ดีสำหรับชุดข้อมูลขนาดใหญ่ ในขณะที่ซัพพอร์ตเวกเตอร์แมชชีน ทำงานได้ดีสำหรับชุดข้อมูลขนาดเล็ก และ วิธีเพื่อนบ้านใกล้สุดเคตัว และ นาอีฟเบย์ ไม่ได้แสดงการพึ่งพาขนาดของชุดข้อมูลอย่างชัดเจน และ การทดลองเปรียบเทียบความสัมพันธ์ระหว่างแบบจำลองการจำแนกประเภท และ การจำแนกประเภทหลายฉลาก พบว่าซัพพอร์ตเวกเตอร์แมชชีน เป็นแบบจำลองการจำแนกประเภทที่ดีที่สุดสำหรับ BR, CC, CLR, QWML และ RAkEL, อย่างไรก็ตาม เพื่อนบ้านใกล้สุดเคตัว และ นาอีฟเบย์ ดีสำหรับ HOMER และ DT ดีสำหรับ ECC

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ตารางที่ 4 เปรียบเทียบงานวิจัยการจำแนกประเภทข้อความแบบหลายฉลาก

งานวิจัยที่เกี่ยวข้อง	เนื้อหางานวิจัย	สรุปงานวิจัย
Multi-label Classification Methods: A Comparative Study	อธิบายวิธีการต่าง ๆของการจำแนกประเภทแบบหลายฉลาก โดยจะต้องเปลี่ยนรูปให้กลายเป็นการจำแนกประเภทเดียวก่อนจึงจะนำมาจำแนกประเภทได้	• การจำแนกประเภทแบบหลายฉลากในแต่ละคลาสจะต้องเป็นไบนารีเท่านั้น และมีวิธีการหลักอยู่ 2 ประเภท คือ (1) การเปลี่ยนรูปให้กลายเป็นฉลากเดียว (2) การปรับขั้นตอนวิธีการจำแนกประเภท
Multi-Label Arabic Text Classification: An Overview	งานวิจัยนำเสนอการทบทวนวิธีการและเทคนิคการจำแนกประเภทหลายฉลากที่มีอยู่ และเปรียบเทียบเชิงทดลองของวิธีการจำแนกประเภทหลายฉลากต่าง ๆ ที่ใช้กับภาษาอาหรับ	• ขั้นตอนวิธี ซัพพอร์ตเวกเตอร์แมชชีน จะให้ผลลัพธ์ดีที่สุดด้วยวิธีการ PT ทั้งหมด คือ BR, CC, LP, PS
Comparison of base classifiers for multi-label learning	งานวิจัยนี้จะเปรียบเทียบวิธีการจำแนกแบบหลายฉลาก โดยใช้แบบจำลองในการจำแนกประเภท 4 วิธีการ คือ ซัพพอร์ตเวกเตอร์แมชชีน, เพื่อนบ้านใกล้สุดเคตัว , นาอีฟเบย์ และ ต้นไม้ตัดสินใจ ในชุดข้อมูล 11 ชุดข้อมูล	• แบบจำลองการจำแนกประเภท ต้นไม้ตัดสินใจทำงานได้ดีกับชุดข้อมูลที่มีขนาดใหญ่ ในขณะที่ ซัพพอร์ตเวกเตอร์แมชชีนทำงานได้ดีกับชุดข้อมูลที่มีขนาดเล็ก • ซัพพอร์ตเวกเตอร์แมชชีนเป็นแบบจำลองการจำแนกประเภทหลายฉลากที่ดีที่สุด

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

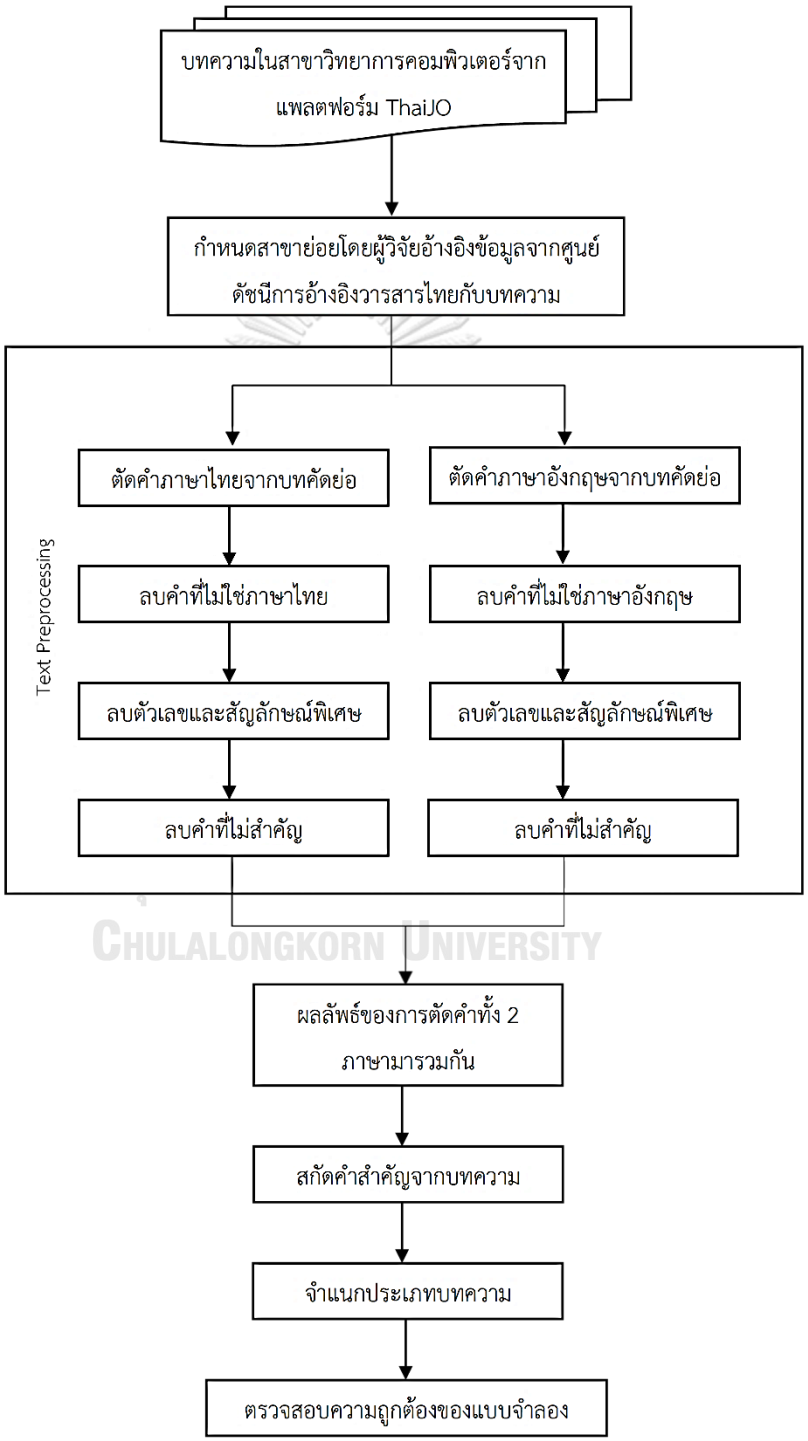
จากการศึกษางานวิจัยที่เกี่ยวข้องการจำแนกประเภทของบทความ ซึ่งในงานวิจัยที่ได้ศึกษามาที่เกี่ยวข้องกับการจำแนกประเภทแบบฉลากเดียวในการจำแนกประเภทของบทความ มีแบบจำลอง 3 แบบจำลอง คือ (1) ซัพพอร์ตเวกเตอร์แมชชีน, (2) เพื่อนบ้านใกล้สุดเคตัว และ (3) นาอีฟเบย์ ซึ่งประสิทธิภาพแบบจำลองเพื่อนบ้านใกล้สุดเคตัว จะดีที่สุดในการจำแนกประเภทบทความแบบฉลากป้ายกำกับเดียว

อย่างไรก็ตาม ในการใช้งานจริงในการจำแนกประเภทของบทความ บทความหนึ่งอาจจะมีได้หลายประเภทหรือหลายหมวดหมู่ ซึ่งในงานวิจัยที่ได้ศึกษามาที่เกี่ยวข้องกับการจำแนกประเภทแบบหลายฉลาก ซึ่งผลลัพธ์วิธีการที่ใช้ขั้นตอนวิธีซัพพอร์ตเวกเตอร์แมชชีน จะให้ผลลัพธ์ที่ดีที่สุดในการจำแนกประเภทหลายฉลากด้วยวิธีการ Problem Transformation ได้แก่ Binary Relevance (BR), Classifier Chains (CC), Label Power-set (LP) เป็นต้น โดยงานวิจัยนี้จะทำการจำแนกประเภทของบทความแบบหลายฉลากโดยใช้ขั้นตอนวิธีซัพพอร์ตเวกเตอร์แมชชีน ร่วมกับ วิธีการ Problem Transformation คือ Binary Relevance , Classifier Chains และ Label Power-set

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

บทที่ 3
วิธีการดำเนินงานวิจัย

3.1 ขั้นตอนการดำเนินงาน



รูปที่ 11 แสดงผังงานขั้นตอนการดำเนินการ

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

จากรูปที่ 7 แสดงขั้นตอนวิธีการดำเนินงาน คือ ขั้นตอนแรก ผู้วิจัยต้องกำหนดสาขาย่อยให้กับบทความโดย อ้างอิงสาขาย่อยของวิทยาการคอมพิวเตอร์จากศูนย์ดัชนีการอ้างอิงวารสารไทย ต่อจากนั้น เมื่อกำหนดสาขาย่อยเสร็จแล้วจะมาเข้าสู่ขั้นตอนการเตรียมข้อมูล (Text Preprocessing) ซึ่งวิธีนี้จะเป็นการตัดคำและทำความสะอาดข้อความที่ได้รับมา เมื่อผ่านกระบวนการเตรียมข้อมูลแล้วจะต้องทำการแปลงข้อความให้อยู่ในรูปที่คอมพิวเตอร์สามารถเข้าใจได้ ซึ่งในงานวิจัยนี้ใช้วิธี TF-IDF จากนั้นจึงนำข้อมูลมาจำแนกประเภทของบทความได้

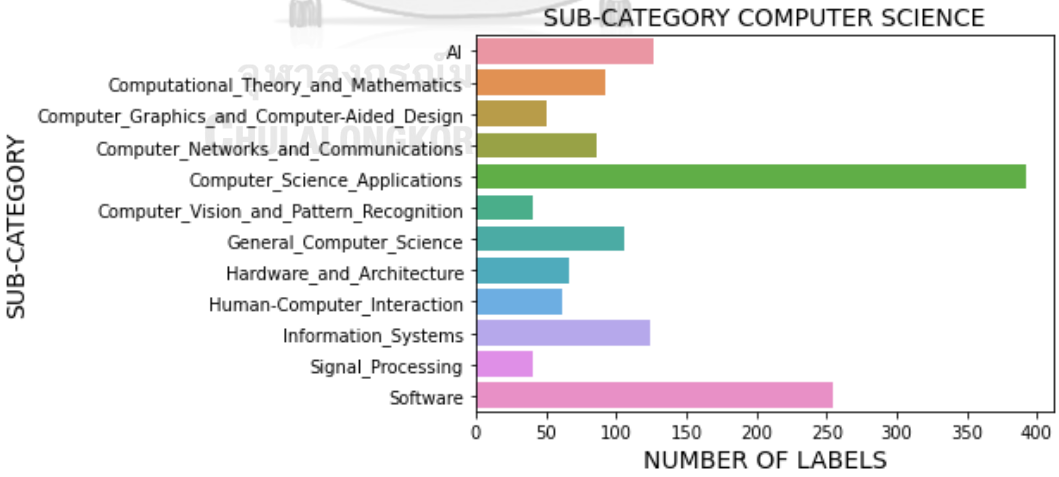
3.2 การกำหนดสาขาย่อยของบทความโดยผู้วิจัย

การกำหนดสาขาย่อยให้กับบทความโดยผู้วิจัย จะดำเนินการดังนี้

3.2.1 เลือกบทความที่เกี่ยวข้องกับสาขาวิทยาการคอมพิวเตอร์จากบทความบนแพลตฟอร์ม ThaiJO จำนวนไม่น้อยกว่า 500 บทความ และมีจำนวนบทความที่เกี่ยวข้องกับสาขาย่อยมากกว่าหนึ่งสาขาย่อยจำนวนไม่ต่ำกว่าร้อยละ 50 จากบทความทั้งหมดที่เลือก

3.2.2 ผู้วิจัยกำหนดสาขาย่อยของบทความตามสาขาย่อยด้านวิทยาการคอมพิวเตอร์ อ้างอิงจากข้อมูลศูนย์ดัชนีการอ้างอิงวารสารไทย (TCI) แสดงตามรูปที่ 12 จำนวนของแต่ละฉลากที่ถูกกำหนดให้กับบทความวิทยาการคอมพิวเตอร์ โดยใน 1 บทความสามารถกำหนดได้มากกว่า 1 ฉลาก

3.2.3 ตรวจสอบความถูกต้องของการระบุสาขาย่อยของบทความ โดยผู้เชี่ยวชาญด้านคอมพิวเตอร์ ประสบการณ์อย่างน้อย 10 ปี จำนวนอย่างน้อย 1 คน



รูปที่ 12 จำนวนสาขาย่อยในแต่ละฉลาก

3.3 Text Pre-processing

บทคัดย่อของบทความวิทยาการคอมพิวเตอร์ของไทย มีคำศัพท์เทคนิคเฉพาะเป็นภาษาอังกฤษ ซึ่งเป็นคำศัพท์ที่เกี่ยวกับคอมพิวเตอร์ จึงจำต้องใช้การตัดคำภาษาอังกฤษและภาษาไทย จากนั้นนำผลลัพธ์ของการตัดคำทั้ง 2 ภาษามารวมกัน

3.3.1 การตัดคำภาษาอังกฤษ

บทคัดย่อของบทความวิทยาการคอมพิวเตอร์ของไทยมีคำศัพท์เทคนิคเฉพาะเป็นภาษาอังกฤษที่เกี่ยวของกับคอมพิวเตอร์ใน เช่น Network, HTMLTags, และ Database เป็นต้น ในงานวิจัยนี้จึงต้องใช้การตัดคำภาษาอังกฤษ โดยใช้ ไลบรารี่บนภาษาไพทอน คือ NLTK

3.3.2 การตัดคำภาษาไทย

งานวิจัยนี้ใช้เครื่องมือในการตัดคำภาษาไทยอยู่ 2 ไลบรารี่บนภาษาไพทอน คือ pythainlp และ DeepCut โดยการใช้ไลบรารี่ pythainlp ในการตัดคำเลือกขั้นตอนวิธี Maximal Matching และใช้พจนานุกรมแบบเดิมกับเพิ่มคำสำคัญจากบทคัดย่อของบทความลงในพจนานุกรม

3.3.3 ทำความสะอาดข้อความและลบคำที่ไม่สำคัญ

ข้อความเมื่อผ่านการตัดคำภาษาไทยและภาษาอังกฤษ จะมีคำที่ไม่สำคัญ, ภาษาอื่น ๆที่ไม่ใช้ภาษาไทยหรือภาษาอังกฤษ และสัญลักษณ์ตัวเลขต่าง ๆ ที่ไม่นำมาคิดเวลาสกัดคำสำคัญลบออกจากเอกสารเพราะจะทำให้ความถี่ของคำที่เกิดขึ้นมีความผิดพลาดได้ ขั้นตอนมีดังนี้

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

3.3.3.1 การลบภาษาอื่น ๆที่ไม่ใช่ภาษาไทยและภาษาอังกฤษ

การตัดคำของภาษาไทยและภาษาอังกฤษจะประมวลผลแยกออกจากกัน ซึ่งการใช้ไลบราลี่ตัดคำภาษาไทย pythainlp และ DeepCut ไม่สามารถที่จะตัดคำที่อยู่ในบทคัดย่อที่เป็นภาษาอื่นได้ เช่น ภาษาจีน, ภาษาอังกฤษ เป็นต้น ทำให้ข้อความที่เป็นภาษาอื่น ๆ ถูกตัดคำออกมาผิดพลาดจึงต้องทำการลบโดยใช้ Regular Expression ในการเลือกแต่คำภาษาไทยเท่านั้น อย่างไรก็ตามการใช้ไลบราลี่ตัดคำภาษาอังกฤษ NLTK ทำการลบภาษาอื่น ๆ โดยใช้ Regular Expression ในการเลือกแต่คำภาษาอังกฤษเท่านั้น

3.3.3.2 การลบตัวเลขและสัญลักษณ์พิเศษจากข้อความ

เมื่อผ่านการตัดคำภาษาไทยและการตัดคำภาษาอังกฤษมาแล้วจะมีข้อมูลตัวเลขและสัญลักษณ์พิเศษอยู่ในข้อมูลของเอกสารนั้น ยกตัวอย่างเช่น วงเล็บ, ตัวเลข, ไม้ยมก เป็นต้น ซึ่งไม่ได้มีความหมายและความสำคัญจากเอกสาร ต้องลบออกโดยใช้วิธีการ Regular Expression ลบตัวเลขเหล่านั้นและกำหนดรายการสัญลักษณ์พิเศษเพื่อลบสัญลักษณ์พิเศษออกจากเอกสารนั้น ๆ

3.3.3.3 การลบคำที่ไม่สำคัญ

วิธีการกำจัดคำที่ไม่สำคัญเป็นอีกส่วนหนึ่งที่ต้องลบออกเพราะไม่มีความหมายและความสำคัญที่อยู่ในเอกสารนั้น โดยจะใช้วิธีการลบคำหยุดจากไลบรารี่ pythainlp โดยเลือกใช้โมดูล thai_stopwords สำหรับภาษาไทย เพื่อเช็คคำที่ไม่สำคัญที่อยู่ในโมดูล thai_stopwords และ ใช้โมดูล stopwords ของ NLTK สำหรับภาษาอังกฤษ เพื่อเช็คคำที่ไม่สำคัญที่อยู่ในโมดูล stopwords

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ตารางที่ 5 ตัวอย่างข้อความที่ถูกตัดคำจากไลบรารี่ pythainlp โดยใช้พจนานุกรมเดิม

บทคัดย่อ	**ข้อความที่ถูกตัดโดยไลบารี่ pythainlp ใช้พจนานุกรมแบบเดิม และทำความสะอาดข้อความ**
การวิจัยครั้งนี้มีวัตถุประสงค์เพื่อ1) เพื่อพัฒนาระบบเฝ้าระวังและแจ้งเตือนอัคคีภัยโดยใช้ Lightweight Protocol Microcontroller และระบบแสดงผล และ 2) เพื่อประเมินระบบเฝ้าระวังและแจ้งเตือนอัคคีภัย และประเมินความพึงพอใจของผู้ใช้งาน กลุ่มตัวอย่างที่ใช้ในการวิจัยได้แก่นักเรียนนายเรือ จำนวน 40 คน ที่ฝึกอยู่เรือหลวงบางปะกง เครื่องมือที่ใช้ในการวิจัย คือ ฮาร์ดแวร์ และโปรแกรม สถิติที่ใช้ในการวิจัย ได้แก่ ค่าเฉลี่ย และค่าเบี่ยงเบนมาตรฐาน ผลการวิจัยพบว่า 1) ระบบ Lightweight Protocol และ Microcontroller แบบ esp8266 (NodeMCU) สามารถนำมาประยุกต์ใช้ในการพัฒนาระบบเฝ้าระวังและแจ้งเตือนอัคคีภัยได้ เมื่อใช้ร่วมกับ เครือข่ายไร้สาย นอกจากนั้นยังสามารถส่งข้อมูลระหว่าง Sensor Microcontroller มายังส่วนแสดงผลได้อย่างมีประสิทธิภาพ ทำให้ ผู้ปฏิบัติงานสามารถเห็นภาพรวมซึ่งนำไปสู่การวิเคราะห์สถานการณ์ได้ง่ายยิ่งขึ้น 2) ผลการประเมินระบบเฝ้าระวัง และแจ้งเตือนอัคคีภัย แบ่งเป็นการประเมินระบบ และการประเมินความพึงพอใจจากผู้ใช้งาน พบว่าผลการประเมินอยู่ในระดับดี ทั้งสองประเภท [12]	การวิจัย|วัตถุประสงค์|พัฒนา|ระบบ|เฝ้า|ระวัง|แจ้ง|เตือน|อัคคีภัย|ระบบ|แสดงผล|ประเมิน|ระบบ|เฝ้า|ระวัง|แจ้ง|เตือน|อัคคีภัย|ประเมิน|ความพึงพอใจ|ผู้ใช้งาน|กลุ่มตัวอย่าง|การวิจัย|นักเรียน|เรือ|จำนวน|คน|ฝึก|เรือหลวง|บางปะกง|เครื่องมือ|การวิจัย|ฮาร์ดแวร์|โปรแกรม|สถิติ|การวิจัย|ค่าเฉลี่ย|ค่า|เบี่ยงเบน|มาตรฐาน|ผลการวิจัย|ระบบ|ประยุกต์ใช้|การพัฒนา|ระบบ|เฝ้า|ระวัง|แจ้ง|เตือน|อัคคีภัย|ร่วมกับ|เครือข่าย|ไร้สาย|ข้อมูล|มายัง|ผลได้|มีประสิทธิภาพ|ผู้ปฏิบัติงาน|ภาพรวม|นำไปสู่|วิเคราะห์|สถานการณ์|การประเมิน|ระบบ|เฝ้า|ระวัง|แจ้ง|เตือน|อัคคีภัย|แบ่ง|การประเมิน|ระบบ|การประเมิน|ความพึงพอใจ|ผู้ใช้งาน|การประเมิน|ระดับ|ดี|ทั้งสอง|ประเภท |lightweight|protocol|microcontroller |lightweight|protocol|microcontroller |esp|nodemcu|sensor|microcontroller|

ตารางที่ 6 ตัวอย่างข้อความที่ถูกตัดคำจากไลบรารี่ pythainlp โดยใช้พจนานุกรมเพิ่มคำสำคัญ

บทคัดย่อ	**ข้อความที่ถูกตัดโดยไลบารี่ pythainlp ใช้พจนานุกรมที่เพิ่มคำสำคัญ และทำความสะอาดข้อความ**
การวิจัยครั้งนี้มีวัตถุประสงค์เพื่อ1) เพื่อพัฒนาระบบเฝ้าระวังและแจ้งเตือนอัคคีภัยโดยใช้ Lightweight Protocol Microcontroller และระบบแสดงผล และ 2) เพื่อประเมินระบบเฝ้าระวังและแจ้งเตือนอัคคีภัย และประเมินความพึงพอใจของผู้ใช้งาน กลุ่มตัวอย่างที่ใช้ในการวิจัยได้แก่นักเรียนนายเรือ จำนวน 40 คน ที่ฝึกอยู่เรือหลวงบางปะกง เครื่องมือที่ใช้ในการวิจัย คือ ฮาร์ดแวร์ และโปรแกรม สถิติที่ใช้ในการวิจัย ได้แก่ ค่าเฉลี่ย และค่าเบี่ยงเบนมาตรฐาน ผลการวิจัยพบว่า 1) ระบบ Lightweight Protocol และ Microcontroller แบบ esp8266 (NodeMCU) สามารถนำมาประยุกต์ใช้ในการพัฒนาระบบเฝ้าระวังและแจ้งเตือนอัคคีภัยได้ เมื่อใช้ร่วมกับ เครือข่ายไร้สาย นอกจากนั้นยังสามารถส่งข้อมูลระหว่าง Sensor Microcontroller มายังส่วนแสดงผลได้อย่างมีประสิทธิภาพ ทำให้ ผู้ปฏิบัติงานสามารถเห็นภาพรวมซึ่งนำไปสู่การวิเคราะห์สถานการณ์ได้ง่ายยิ่งขึ้น 2) ผลการประเมินระบบเฝ้าระวัง และแจ้งเตือนอัคคีภัย แบ่งเป็นการประเมินระบบ และการประเมินความพึงพอใจจากผู้ใช้งาน พบว่าผลการประเมินอยู่ในระดับดี ทั้งสองประเภท [12]	การวิจัย|วัตถุประสงค์|พัฒนา|ระบบ|เฝ้า|ระวัง|แจ้งเตือนอัคคีภัย|ระบบ|แสดงผล|ประเมิน|ระบบ|เฝ้า|ระวัง|แจ้งเตือนอัคคีภัย|ประเมิน|ความพึงพอใจของผู้ใช้|งาน|กลุ่มตัวอย่าง|การวิจัย|นักเรียน|เรือ|จำนวน|คน|ฝึก|เรือหลวง|บางปะกง|เครื่องมือ|การวิจัย|ฮาร์ดแวร์|โปรแกรม|สถิติ|การวิจัย|ค่าเฉลี่ย|ค่า|เบี่ยงเบน|มาตรฐาน|ผลการวิจัย|ระบบ|ประยุกต์ใช้|การพัฒนาระบบ|เฝ้า|ระวัง|แจ้งเตือนอัคคีภัย|ร่วมกับ|เครือข่าย|ไร้สาย|ข้อมูล|มายัง|ผลได้|มีประสิทธิภาพ|ผู้ปฏิบัติงาน|ภาพรวม|นำไปสู่|การวิเคราะห์|สถานการณ์|การประเมิน|ระบบ|เฝ้า|ระวัง|แจ้งเตือนอัคคีภัย|แบ่ง|การประเมิน|ระบบ|การประเมิน|ความพึงพอใจ|ผู้ใช้งาน|การประเมิน|ระดับ|ดี|ทั้งสอง|ประเภท| |lightweight|protocol|microcontroller |lightweight|protocol|microcontroller |esp|nodemcu|sensor|microcontroller|

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ตารางที่ 7 ตัวอย่างข้อความที่ถูกตัดคำจากไลบรารี่ Deepcut

บทคัดย่อ	**ข้อความที่ถูกตัดโดยไลบารี่ Deepcut และทำความสะอาดข้อความ**
การวิจัยครั้งนี้มีวัตถุประสงค์เพื่อ1) เพื่อพัฒนาระบบเฝ้าระวังและแจ้งเตือนอัคคีภัยโดยใช้ Lightweight Protocol Microcontroller และระบบแสดงผล และ 2) เพื่อประเมินระบบเฝ้าระวังและแจ้งเตือนอัคคีภัย และประเมินความพึงพอใจของผู้ใช้งาน กลุ่มตัวอย่างที่ใช้ในการวิจัยได้แก่นักเรียนนายเรือ จำนวน 40 คน ที่ฝึกอยู่เรือหลวงบางปะกง เครื่องมือที่ใช้ในการวิจัย คือ ฮาร์ดแวร์ และโปรแกรม สถิติที่ใช้ในการวิจัย ได้แก่ ค่าเฉลี่ย และค่าเบี่ยงเบนมาตรฐาน ผลการวิจัยพบว่า 1) ระบบ Lightweight Protocol และ Microcontroller แบบ esp8266 (NodeMCU) สามารถนำมาประยุกต์ใช้ในการพัฒนาระบบเฝ้าระวังและแจ้งเตือนอัคคีภัยได้ เมื่อใช้ร่วมกับ เครือข่ายไร้สาย นอกจากนั้นยังสามารถส่งข้อมูลระหว่าง Sensor Microcontroller มายังส่วนแสดงผลได้อย่างมีประสิทธิภาพ ทำให้ ผู้ปฏิบัติงานสามารถเห็นภาพรวมซึ่งนำไปสู่การวิเคราะห์สถานการณ์ได้ง่ายยิ่งขึ้น 2) ผลการประเมินระบบเฝ้าระวัง และแจ้งเตือนอัคคีภัย แบ่งเป็นการประเมินระบบ และการประเมินความพึงพอใจจากผู้ใช้งาน พบว่าผลการประเมินอยู่ในระดับดี ทั้งสองประเภท [12]	วิจัย|วัตถุประสงค์|พัฒนา|ระบบ|เฝ้า|ระวัง|แจ้ง|เตือน|อัคคีภัย|ระบบ|ประเมิน|ระบบ|เฝ้า|ระวัง|แจ้ง|เตือน|อัคคีภัย|ประเมิน|พึงพอใจ|งาน|ตัวอย่าง|วิจัย|เรียน|เรือ|จำนวน|คน|ฝึก|เรือ|หลวง|บางปะกง|เครื่องมือ|วิจัย|ฮาร์ดแวร์|โปรแกรม|สถิติ|วิจัย|ค่า|เฉลี่ย|ค่า|เบี่ยงเบน|มาตรฐาน|วิจัย|ระบบ|ประยุกต์|พัฒนา|ระบบ|เฝ้า|ระวัง|แจ้ง|เตือน|อัคคีภัย|เครือข่าย|ไร้|สาย|ข้อมูล|ประสิทธิภาพ|ทำ|งาน|ภาพ|วิเคราะห์|สถานการณ์|ประเมิน|ระบบ|เฝ้า|ระวัง|แจ้ง|เตือน|อัคคีภัย|แบ่ง|ประเมิน|ระบบ|ประเมิน|พึงพอใจ|งาน|ประเมิน|ระดับ|ดี|สอง|ประเภท| |lightweight|protocol|microcontroller |lightweight|protocol|microcontroller |esp|nodemcu|sensor|microcontroller|

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

จากตารางที่ 1-3 แสดงตัวอย่างผลลัพธ์ของ Text-Preprocessing โดยเปรียบเทียบวิธีการตัดคำโดย (1) การใช้ไลบรารี่ pythainlp ในการตัดคำภาษาไทย (2) การเพิ่มคำสำคัญลงในพจนานุกรมก่อนใช้ไลบรารี่ pythainlp ในการตัดคำภาษาไทย และ (3) การใช้ไลบรารี่ Deepcut ในการตัดคำภาษาไทย และทุกขั้นตอนเมื่อตัดคำเสร็จแล้วต้องมีการทำขั้นตอนการทำความสะอาดข้อความและลบคำที่ไม่สำคัญออกจากข้อความก่อนนำไปวิเคราะห์ต่อไป

3.4 การสกัดคำสำคัญจากบทความ

การสกัดคำสำคัญจะทำหลังจากการเตรียมข้อมูลเสร็จเรียบร้อยแล้ว นั่นคือหลังจาก การตัดคำ การลบคำที่ไม่สำคัญออกจากข้อความ และลบสัญลักษณ์พิเศษต่าง ๆ ออกจากข้อความ เมื่อเตรียมข้อมูลเสร็จเรียบร้อยจะสามารถใช้วิธีการสกัดข้อมูลโดยใช้ขั้นตอนวิธี TF-IDF ได้ ในงานวิจัยนี้จะใช้ไลบรารี่ sklearn เพื่อคำนวณความถี่ของคำทั้งหมดที่เกิดขึ้นในบทความต่าง ๆ

ตารางที่ 8 ตัวอย่างคำสำคัญที่เกิดขึ้นโดยขั้นตอนวิธี TF-IDF 10 อันดับแรก ตามวิธีการตัดคำ

วิธีการตัดคำ	**คำสำคัญ โดยใช้ TF-IDF ความถี่สูงสุด 10 อันดับ**
pythainlp	เครื่องใช้ไฟฟ้า,แอพพลิเคชั่น, Arduino, Ionic,การควบคุม, UNO,แอนดรอยด์, Framework, Relay, ภายในบ้าน
Add keyword pythainlp	แอพพลิเคชั่น, เครื่องใช้ไฟฟ้า, Arduino, Ionic, การควบคุม, ระบบปฏิบัติการแอนดรอยด์, UNO, Framework, Relay, ภายในบ้าน
Deepcut	แอพพลิเคชั่น, Arduino, ไฟฟ้า, Ionic, เครื่อง, UNO, แอนดรอยด์, Framework, Relay, ปฏิบัติการ

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

จากตารางที่ 8 แสดงตัวอย่างผลลัพธ์การสกัดคำสำคัญ โดยใช้วิธีการ TF-IDF ยกตัวอย่างความถี่สูงสุด 10 อันดับแรกมาแสดง โดยใช้วิธีการตัดคำ คือ (1) pythainlp , (2) ปรับปรุงพจนานุกรมก่อนใช้ pythainlp และ (3) Deepcut โดยในตารางที่ 8 แสดงตัวอย่างผลลัพธ์คำสำคัญที่ถูกสกัดออกมาได้

3.5 การจำแนกประเภทของบทความ

เมื่อผ่านขั้นตอนของการเตรียมข้อมูลและสกัดคำสำคัญด้วยวิธีการ TF-IDF แล้ว จะนำบทความสาขา วิทยาการคอมพิวเตอร์ที่เลือกไว้มาทำการจำแนกประเภทของบทความแบบหลายฉลาก ซึ่งบทความถูกกำหนดเป็นสาขาย่อยทั้งหมด 12 สาขาย่อยตามข้อมูลจากศูนย์ดัชนีการอ้างอิงวารสารไทย ดังแสดงในตารางที่ 9 โดยบทความหนึ่งสามารถอยู่ได้หลายสาขาย่อย

สาขาย่อยวิทยาการคอมพิวเตอร์	
General Computer Science (GCS)	Artificial Intelligence (AI)
Computer Vision and Pattern – Recognition (CV)	Hardware and Architecture (HA)
Computational Theory and Mathematics (CTM)	Human-Computer Interaction (HCI)
Computer Graphics and Computer-Aided Design (CG)	Information Systems (IS)
Computer Networks and -Communications (CNC)	Signal Processing (SP)
Computer Science Applications (CSA)	Software (SW)

ตารางที่ 9 สาขาย่อยของวิทยาการคอมพิวเตอร์

การจำแนกบทความแบบหลายฉลากสาขาย่อยจะใช้วิธีการ Problem Transformation เพราะใน 1 บทความสามารถมีสาขาย่อยได้มากกว่า 1 สาขาย่อย ด้วยวิธีการ Binary Relevance, Classifier Chain และ Label-Power set จากนั้นนำแบบจำลองซัพพอร์ตเวกเตอร์แมชชีนด้วย kernel 3 ประเภทคือ linear kernel, RBF kernel และ polynomial kernel มาจำแนกบทความตามแต่ละวิธีการตัดคำ สำหรับ RBF kernel และ polynomial kernel ใช้ Grid Search ในการหาค่าพารามิเตอร์ที่ดีที่สุด เพื่อประสิทธิภาพดีที่สุดของการจำแนกประเภท

ผู้วิจัยจะนำข้อมูลของบทความมาแบ่งโดยออกเป็นเซตย่อยโดยใช้ วิธีการตรวจสอบไขว้ (K-Fold Cross Validation) ประกอบด้วยชุดการฝึก (Training Set) และ ชุดการตรวจสอบ (Validation Set) จากนั้นแบบจำลองจะถูกนำไปฝึกฝนในเซตย่อยของชุดฝึกฝน และ และตรวจสอบวัดประสิทธิภาพในชุดการตรวจสอบต่อไป ซึ่งจะทำซ้ำจนกระทั่งแต่ละเซตย่อยได้ทำหน้าที่เป็นชุดการตรวจสอบทั้งหมด

3.6 การวัดประสิทธิภาพของแบบจำลอง

การวัดประสิทธิภาพแบบจำลองจะนำค่าที่ได้จากการทดลอง ซึ่งแบ่งข้อมูลออกเป็นทั้งหมด 10 ส่วน ด้วยวิธีวิธีการตรวจสอบไขว้ (10-fold Cross Validation) โดยข้อมูล 1 ส่วน เป็นชุดข้อมูลสำหรับการทดสอบ (Test Set) และ อีก 9 ส่วนเป็นชุดข้อมูลสำหรับการฝึกฝน (Training Set) ซึ่งทำการแบ่งข้อมูล เพื่อทดสอบประสิทธิภาพการจำแนกของแบบจำลอง 10 รอบ มาหาค่าเฉลี่ยของการจำแนกประเภทหลายฉลาก ด้วยวิธีการวัดประสิทธิภาพแบบ ตัวชี้วัดประสิทธิภาพการเลือกตอบตามตัวอย่าง และ ตัวชี้วัดประสิทธิภาพการจำแนกประเภทหลายฉลาก จากนั้น นำผลลัพธ์การตรวจสอบความถูกต้องของแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน สำหรับการจำแนกประเภทหลายฉลากของบทความโดยการตัดคำ ที่ใช้ไลบราลี่ pythainlp, ไลบราลี่ pythainlp + พจนานุกรมเพิ่มคำสำคัญ และ ไลบราลี่ Deepcut ด้วยวิธีการ Binary Relevance, Classifier Chain และ Label Power-set มาเปรียบเทียบกัน

การจำแนกประเภทแบบหลายฉลากสามารถวัดประสิทธิภาพได้ 2 วิธีหลัก โดยจะยกตัวอย่างการวัดประสิทธิภาพทั้ง ตัวชี้วัดประสิทธิภาพการเลือกตอบตามตัวอย่าง และ ตัวชี้วัดประสิทธิภาพการจำแนกประเภทหลายฉลาก ดังนี้

กำหนดให้ Y_i = ผลเฉลยของเซตฉลาก, Z_i = ผลทำนายของเซตฉลาก และ m = จำนวนตัอย่างทั้งหมดในชุดตัวอย่างทดสอบบ

ตัวอย่างที่ 1

Y1 = [0 0 0 0 1 0 0 1 0 0 0 0]

Z1 = [0 0 0 0 1 0 0 0 0 0 0 0]

ตัวอย่างที่ 2

Y2 = [0 0 0 0 0 0 1 0 0 1 0 0]

Z2 = [0 0 0 0 1 0 1 0 0 0 0 0]

ตัวอย่างที่ 3

Y3 = [0 0 0 1 1 0 0 1 0 0 0 1]

Z3 = [0 0 0 1 1 0 0 1 0 0 0 1]

3.6.1 ตัวชี้วัดประสิทธิภาพการเลือกตอบตามตัวอย่าง (Example-based Metrics)

- Hamming loss สามารถคำนวนณตามสูตรดังนี้ โดย L = ขนาดของเซตฉลาก

$$\text{Hamming loss} = \frac{1}{mL}\sum_{i=1}^{m} |Z_i \neq Y_i|$$

$$\text{Hamming loss} = \frac{1}{mL}\sum_{i=1}^{m} |Z_i \neq Y_i| = \frac{3}{3 \times 12} = 0.083$$

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

- ML-accuracy สามารถคำนวนณตามสูตรดังนี้

$$\text{ML-accuracy} = \frac{1}{m}\sum_{i=1}^{m}\frac{|Z_i \cap Y_i|}{|Z_i \cup Y_i|}$$

$|Z_i \cap Y_i|$ = [1,1,4] และ $|Z_i \cup Y_i|$ = [2,3,4]

$$\text{ML- accuracy} = \frac{1}{m}\sum_{i=1}^{m}\frac{|Z_i \cap Y_i|}{|Z_i \cup Y_i|} = [0.5,0.3,1] = 1.8/3 = 0.6$$

- Subset accuracy สามารถคำนวนณตามสูตรดังนี้

$$\text{Subset accuracy} = \frac{1}{m}\sum_{i=1}^{m} I|Z_i = Y_i|$$

$I|Z_i = Y_i|$ = [0,0,1]

$$\text{Subset accuracy} = \frac{1}{m}\sum_{i=1}^{m} I|Z_i = Y_i| = 1/3 = 0.33$$

- ค่าแม่นยำ (Precision) คือ สามารถคำนวนณตามสูตรดังนี้

$$\text{ค่าแม่นยำ (Precision)} = \frac{1}{m}\sum_{i=1}^{m}\frac{|Z_i \cap Y_i|}{|Z_i|}$$

$|Z_i \cap Y_i|$ = [1,1,4] และ $|Z_i|$ = [1,2,4]

$$\text{ค่าแม่นยำ (Precision)} = \frac{1}{m}\sum_{i=1}^{m}\frac{|Z_i \cap Y_i|}{|Z_i|} = [1,0.5,1] = 2.5/3 = 0.83$$

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

- ค่าเรียกคืน (Recall) สามารถคำนวนณตามสูตรดังนี้

$$\text{ค่าเรียกคืน (Recall)} = \frac{1}{m}\sum_{i=1}^{m}\frac{|Z_i \cap Y_i|}{|Y_i|}$$

$|Z_i \cap Y_i|$ = [1,1,4] และ $|Y_i|$ = [2,2,4]

$$\text{ค่าเรียกคืน (Recall)} = \frac{1}{m}\sum_{i=1}^{m}\frac{|Z_i \cap Y_i|}{|Y_i|} = [0.5,0.5,1] = 2/3 = 0.67$$

- ตัววัด F1 (F-Measure) คือ ค่าเฉลี่ยระหว่าง ค่าแม่นยำ และ ค่าเรียกคืน มีสูตรดังนี้

$$\text{ตัววัด F1 (F-Measure)} = \frac{1}{m}\sum_{i=1}^{m}\frac{2|Z_i \cap Y_i|}{|Z_i|+|Y_i|}$$

$2|Z_i \cap Y_i|$ = [2,2,8], $|Z_i|$ = [1,2,4] และ $|Y_i|$ = [2,2,4]

$$\text{ตัววัด F1 (F-Measure)} = \frac{1}{m}\sum_{i=1}^{m}\frac{2|Z_i \cap Y_i|}{|Z_i|+|Y_i|} = [0.67,0.5,1] = 2.17/3 = 0.72$$

3.6.2 ตัวชี้วัดประสิทธิภาพการจำแนกประเภทหลายฉลาก (Label-based Metrics)

Label-based Metrics คือการวัดประสิทธิภาพในการทำนายแต่สำหรับแต่ละฉลากในชุดข้อมูลทดสอบในงานวิจัยนี้ใช้แนวทางค่าเฉลี่ยไมโคร โดยสามารถคำนวณได้ตามสูตรดังนี้

$$B_{micro} = B(\sum_{i=1}^{n} tp_i, \sum_{i=1}^{n} fp_i, \sum_{i=1}^{n} tn_i, \sum_{i=1}^{n} fn_i)$$

กำหนดให้ B = ค่าแม่นยำ (Precision), ค่าเรียกคืน (Recall) และ ตัววัด F1 (F-Measure) โดย n = จำนวนฉลากที่เป็นไปได้ทั้งหมด

- *ค่าเฉลี่ยไมโครสำหรับค่าแม่นยำ* (Micro-average precision) สามารถคำนวณได้ดังนี้

$$\text{Micro-average precision} = \frac{\sum_{i=1}^{n} tp_i}{\sum_{i=1}^{n} tp_i + \sum_{i=1}^{n} fp_i} = \frac{6}{6+1} = 0.85$$

- *ค่าเฉลี่ยไมโครสำหรับค่าเรียกคืน* (Micro-average recall) สามารถคำนวณได้ดังนี้

$$\text{Micro-average recall} = \frac{\sum_{i=1}^{n} tp_i}{\sum_{i=1}^{n} tp_i + \sum_{i=1}^{n} fn_i} = \frac{6}{6+2} = 0.75$$

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

- *ค่าเฉลี่ยไมโครสำหรับตัววัด* F1 (Micro-average F-Measure) สามารถคำนวณได้ดังนี้

$$\text{Micro-average F-Measure} = 2 \cdot \frac{micro-average\ precisioin * micro-aveage\ recall}{micro-average\ precisioin + micro-aveage\ recall}$$

$$\text{Micro-average F-Measure} = 2 \cdot \frac{0.85 * 0.75}{0.85 + 0.75} = 0.79$$

บทที่ 4

ผลการทดลอง

แบบจำลองการจำแนกประเภทหลายฉลากที่ทำการทดลองจะเปรียบเทียบ 3 วิธีการ คือBinary Relevance, Classifier Chain และ Label-Power Set ร่วมกับการจำแนกประเภทซัพพอร์ตเวกเตอร์แมชชีน และ 3 วิธีการตัดคำภาษาไทย คือ pythainlp, pythainlp Custom Dictionary และ Deepcut ซึ่งใช้บทความภาษาไทย 590 บทความในสาขาวิชาวิทยาการคอมพิวเตอร์ และ กำหนด 12 หมวดหมู่ย่อยให้กับบทความ โดยทดสอบประสิทธิภาพด้วยขั้นตอนวิธีการตรวจสอบไขว้ (10-Fold Cross Validation) และเปรียบเทียบด้วยค่าตัวชี้วัดประสิทธิภาพการเลือกตอบตามตัวอย่างและ ตัวชี้วัดประสิทธิภาพการจำแนกประเภทหลายฉลาก

4.1 ชุดข้อมูล

ชุดข้อมูลในงานวิจัยนี้ใช้บทความวิทยาการคอมพิวเตอร์มีทั้งหมด 590 บทความ มีฉลากในชุดข้อมูลทั้งหมด 1441 ฉลาก ซึ่งค่า Label Cardinality = 2.442 และ Label Density = 0.20แสดงตามตารางที่ 10

ตารางที่ 10 ตารางชุดข้อมูล

บทความ	จำนวนบทความ	จำนวนฉลาก	Label Cardinality	Label Density
วิทยาการคอมพิวเตอร์	590	1441	2.44	0.20

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ยกตัวอย่างการวัดลักษณะของชุดข้อมูลทั้ง Label Cardinality และ Label Density สามารถคำนวณได้ดังนี้

กำหนดให้ Y_i = เซตของฉลาก, N = *จำนวนของชุดข้อมูล และ* L = ขนาดของเซตฉลาก

ตัวอย่างที่ 1

Y_1 = [0 0 0 0 1 0 0 1 0 0 0 0]

ตัวอย่างที่ 2

Y_2 = [0 0 0 1 0 0 1 0 0 1 0 0]

- Label Cardinality สามารถคำนวณได้ดังนี้

$$Card = \frac{1}{N}\sum_{i=1}^{N}|Y_i|$$

$$\text{Label Cardinality} = \frac{(2+3)}{2} = 2.5$$

- Label Density สามารถคำนวณได้ดังนี้

$$Dens = \frac{1}{N}\sum_{i=1}^{N}\frac{|Y_i|}{|L|}$$

$$\text{Label Density} = \frac{(2+3)}{2 \cdot 12} = 0.25$$

4.2 ผลการทดลองของการจำแนกประเภทหลายฉลาก

4.2.1 แบบจำลองซัพพอร์ตเวกเตอร์แมชชีนด้วย linear kernel

4.2.1.1 การวัดประสิทธิภาพ ตัวชี้วัดประสิทธิภาพการเลือกตอบตามตัวอย่าง

ผลการทดลองสำหรับการวัดประสิทธิภาพสำหรับ ตัวชี้วัดประสิทธิภาพการเลือกตอบตามตัวอย่าง พบว่า การวิธีการจำแนกประเภทหลายฉลากด้วยวิธีการ Classifier Chain ร่วมกับ ซัพพอร์ตเวกเตอร์แมชชีน และ วิธีการตัดคำโดยใช้ Deepcut มีประสิทธิภาพดีที่สุดเมื่อวัดประสิทธิภาพด้วย ML-accuracy (0.572), Subset accuracy (0.286) และตัววัด F1 (F-measure) (0.666)

อย่างไรก็ตาม สำหรับจำแนกประเภทหลายฉลากด้วยวิธีการ Binary Relevance ร่วมกับ ซัพพอร์ตเวกเตอร์แมชชีน และ วิธีการตัดคำโดยใช้ pythainlp มีประสิทธิภาพดีที่สุดเมื่อวัดประสิทธิภาพด้วย Hamming loss (0.108) และ ค่าแม่นยำ (Precision) (0.737) สำหรับจำแนกประเภทหลายฉลากด้วยวิธีการ Label Power-set ร่วมกับ ซัพพอร์ตเวกเตอร์แมชชีน และ วิธีการตัดคำโดยใช้ pythainlp มีประสิทธิภาพดีที่สุดสำหรับ ค่าเรียกคืน (0.679) ตามตารางที่ 11 สรุปผลการทดลองการจำแนกประเภทสำหรับการวัดประสิทธิภาพแบบ ตัวชี้วัดประสิทธิภาพการเลือกตอบตามตัวอย่าง

ตารางที่ 11 ตารางผลการทดลองการจำแนกประเภทหลายฉลากสำหรับ ตัวชี้วัดประสิทธิภาพการเลือกตอบตามตัวอย่าง สำหรับแบบจำลองซัพพอร์ตเวกเตอร์แมชชีนด้วย linear kernel

PT and word segmentation method	Hamming loss	ML-accuracy	Subset accuracy	Precision	Recall	F-Measure
BR-SVM-pythainlp	**0.108**	0.556	0.271	**0.737**	0.623	0.640
BR-SVM-Custom-Dictionary (pythainlp)	0.114	0.530	0.250	0.710	0.595	0.611
BR-SVM-Deepcut	0.112	0.550	0.257	0.729	0.630	0.640
CC-SVM-pythainlp	0.110	0.568	0.277	0.726	0.645	0.651
CC-SVM-Custom-Dictionary (pythainlp)	0.117	0553	0.264	0.704	0.624	0.629
CC-SVM-Deepcut	0.112	**0.572**	**0.286**	0.727	0.669	**0.666**
LP-SVM-pythainlp	0.133	0.552	0.261	0.694	**0.679**	0.656
LP-SVM-Custom-Dictionary (pythainlp)	0.130	0.566	0.274	0.689	0.675	0.652
LP-SVM-Deepcut	0.132	0.557	0.261	0.665	0.661	0.634

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

4.2.1.2 การวัดประสิทธิภาพ ตัวชี้วัดประสิทธิภาพการจำแนกประเภทหลายฉลาก

ผลการทดลองสำหรับการวัดประสิทธิภาพด้วย ตัวชี้วัดประสิทธิภาพการจำแนกประเภทหลายฉลาก พบว่า การวิธีการจำแนกประเภทหลายฉลากด้วยวิธีการ Classifier Chain ร่วมกับ ซัพพอร์ตเวกเตอร์แมชชีน และ วิธีการตัดคำโดยใช้ pythainlp และ Deepcut มีประสิทธิภาพดีที่สุดทั้ง ค่าเฉลี่ยไมโครสำหรับค่าแม่นยำ (0.57) และ ค่าเฉลี่ยไมโครสำหรับตัววัด F1 (0.70)

สำหรับจำแนกประเภทหลายฉลากด้วยวิธีการ Binary Relevance ร่วมกับซัพพอร์ตเวกเตอร์แมชชีน และ วิธีการตัดคำโดยใช้ pythainlp มีประสิทธิภาพดีที่สุดเมื่อวัดประสิทธิภาพด้วย ค่าเฉลี่ยไมโครสำหรับค่าแม่นยำ (0.57) และ วิธีการ Label Power-set ร่วมกับ ซัพพอร์ตเวกเตอร์แมชชีน และ วิธีการตัดคำโดยใช้ pythainlp ที่ปรับปรุงพจนานุกรม มีประสิทธิภาพดีที่สุดเมื่อวัดประสิทธิภาพด้วย ค่าเฉลี่ยไมโครสำหรับค่าเรียกคืน (0.678) ตามลำดับ ตามตารางที่ 12 สรุปผลการทดลองการจำแนกประเภทสำหรับการวัดประสิทธิภาพแบบ ตัวชี้วัดประสิทธิภาพการจำแนกประเภทหลายฉลาก

ตารางที่ 12 ตารางผลการทดลองการจำแนกประเภทหลายฉลากสำหรับ ตัวชี้วัดประสิทธิภาพการจำแนกประเภทหลายฉลาก สำหรับแบบจำลองซัพพอร์ตเวกเตอร์แมชชีนด้วย linear kernel

PT and word segmentation method	Micro-average precision	Micro-average recall	Micro-average F-Measure
BR-SVM-pythainlp	**0.570**	0.623	0.699
BR-SVM-Custom-Dictionary (pythainlp)	0.553	0.601	0.680
BR-SVM-Deepcut	0.564	0.627	0.694
CC-SVM-pythainlp	**0.570**	0.645	**0.700**
CC-SVM-Custom-Dictionary (pythainlp)	0.552	0.639	0.689
CC-SVM-Deepcut	**0.570**	0.659	**0.703**
LP-SVM-pythainlp	0.516	0.662	0.667
LP-SVM-Custom-Dictionary (pythainlp)	0.527	**0.678**	0.678
LP-SVM-Deepcut	0.519	0.666	0.671

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

4.2.2 แบบจำลองซัพพอร์ตเวกเตอร์แมชชีนด้วย RBF kernel

สำหรับ RBF Kernel ใช้ Grid Search ในการหาค่าพารามิเตอร์ที่ดีที่สุดสำหรับแต่ละวิธีการจำแนกประเภทหลายฉลากและวิธีการตัดคำได้ผลลัพธ์ ตามตารางที่ 13

ตารางที่ 13 พารามิเตอร์สำหรับแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน ด้วย RBF kernel

วิธีการจำแนกประเภทและวิธีการตัดคำ	γ	C
BR-pythainlp	0.01	100
BR-Custom-Dictionary (pythainlp)	0.01	1000
BR-Deepcut	0.01	100
CC-pythainlp	0.001	1000
CC-Custom-Dictionary (pythainlp)	0.001	1000
CC-Deepcut	0.1	10
LP-pythainlp	0.01	100
LP-Custom-Dictionary (pythainlp)	0.01	1000
LP-Deepcut	0.001	1000

4.2.2.1 การวัดประสิทธิภาพ ตัวชี้วัดประสิทธิภาพการเลือกตอบตามตัวอย่าง

ผลการทดลองสำหรับการวัดประสิทธิภาพสำหรับ ตัวชี้วัดประสิทธิภาพการเลือกตอบตามตัวอย่าง พบว่า วิธีการจำแนกประเภทหลายฉลากด้วยวิธีการ Classifier Chain ร่วมกับ ซัพพอร์ตเวกเตอร์แมชชีน และ วิธีการตัดคำโดยใช้ pythainlp มีประสิทธิภาพดีที่สุดเมื่อวัดประสิทธิภาพด้วย ML-accuracy (0.578), Subset accuracy (0.300) และ ค่าเรียกคืน (Recall) (0.670)

อย่างไรก็ตาม สำหรับจำแนกประเภทหลายฉลากด้วยวิธีการ Binary Relevance ร่วมกับ ซัพพอร์ตเวกเตอร์แมชชีน และ วิธีการตัดคำโดยใช้ pythainlp มีประสิทธิภาพดีที่สุดเมื่อวัดประสิทธิภาพด้วย Hamming loss (0.106), ค่าแม่นยำ (Precision) (0.735) และ ตัววัด F1 (F-measure) (0.665) ตามตารางที่ 14 สรุปผลการทดลองการจำแนกประเภทสำหรับการวัดประสิทธิภาพแบบ ตัวชี้วัดประสิทธิภาพการเลือกตอบตามตัวอย่าง

ตารางที่ 14 ตารางผลการทดลองการจำแนกประเภทหลายฉลากสำหรับ ตัวชี้วัดประสิทธิภาพการเลือกตอบตามตัวอย่าง สำหรับแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน ด้วย RBF kernel

PT and word segmentation method	Hamming loss	ML-accuracy	Subset accuracy	Precision	Recall	F-Measure
BR-SVM-pythainlp	**0.106**	0.577	0.279	**0.735**	0.666	**0.665**
BR-SVM-Custom-Dictionary (pythainlp)	0.113	0.553	0.255	0.710	0.641	0.642
BR-SVM-Deepcut	0.114	0.557	0.250	0.705	0.658	0.647
CC-SVM-pythainlp	0.111	**0.578**	**0.300**	0.723	**0.670**	0.664
CC-SVM-Custom-Dictionary (pythainlp)	0.115	0565	0.272	0.719	0.658	0.654
CC-SVM-Deepcut	0.116	0.573	0.283	0.712	0.667	0.659
LP-SVM-pythainlp	0.131	0.559	0.264	0.691	0.663	0.649
LP-SVM-Custom-Dictionary (pythainlp)	0.139	0.544	0.261	0.672	0.648	0.634
LP-SVM-Deepcut	0.131	0.560	0.284	0.686	0.662	0.648

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

4.2.2.2 การวัดประสิทธิภาพ ตัวชี้วัดประสิทธิภาพการจำแนกประเภทหลายฉลาก

ผลการทดลองสำหรับการวัดประสิทธิภาพด้วย ตัวชี้วัดประสิทธิภาพการจำแนกประเภทหลายฉลาก พบว่า วิธีการจำแนกประเภทหลายฉลากด้วยวิธีการ Binary Relevance ร่วมกับ ซัพพอร์ตเวกเตอร์แมชชีน และ วิธีการตัดคำโดยใช้ pythainlp มีประสิทธิภาพดีที่สุดทั้ง ค่าเฉลี่ยไมโครสำหรับค่าแม่นยำ (0.586) และ ค่าเฉลี่ยไมโครสำหรับตัววัด F1 (0.715)

สำหรับการจำแนกประเภทหลายฉลากด้วยวิธีการ Classifier Chain ร่วมกับ ซัพพอร์ตเวกเตอร์แมชชีน และ วิธีการตัดคำโดยใช้ pythainlp มีประสิทธิภาพดีที่สุดเมื่อวัดประสิทธิภาพด้วย ค่าเฉลี่ยไมโครสำหรับค่าเรียกคืน (0.670) ตามตารางที่ 15 สรุปผลการทดลองการจำแนกประเภทสำหรับการวัดประสิทธิภาพแบบ ตัวชี้วัดประสิทธิภาพการจำแนกประเภทหลายฉลาก

ตารางที่ 15 ตารางผลการทดลองการจำแนกประเภทหลายฉลากสำหรับ ตัวชี้วัดประสิทธิภาพการจำแนกประเภทหลายฉลาก สำหรับแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน ด้วย RBF kernel

PT and word segmentation method	Micro-average precision	Micro-average recall	Micro-average F-Measure
BR-SVM-pythainlp	**0.586**	0.657	**0.715**
BR-SVM-Custom-Dictionary (pythainlp)	0.564	0.642	0.697
BR-SVM-Deepcut	0.561	0.652	0.697
CC-SVM-pythainlp	0.573	**0.670**	0.708
CC-SVM-Custom-Dictionary (pythainlp)	0.562	0.663	0.699
CC-SVM-Deepcut	0.558	0.664	0.697
LP-SVM-pythainlp	0.520	0.654	0.669
LP-SVM-Custom-Dictionary (pythainlp)	0.500	0.641	0.652
LP-SVM-Deepcut	0.519	0.655	0.669

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

4.2.3 แบบจำลองซัพพอร์ตเวกเตอร์แมชชีนด้วย polynomial kernel

สำหรับ polynomial kernel ใช้ Grid Search ในการหาค่าพารามิเตอร์ที่ดีที่สุดสำหรับแต่ละวิธีการจำแนกประเภทหลายฉลากและวิธีการตัดคำได้ผลลัพธ์ ตามตารางที่ 16

ตารางที่ 16 พารามิเตอร์สำหรับแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน ด้วย polynomial kernel

วิธีการจำแนกประเภทและวิธีการตัดคำ	Degree (d)	C
BR-pythainlp	2	10
BR-Custom-Dictionary (pythainlp)	2	10
BR-Deepcut	2	10
CC-pythainlp	2	10
CC-Custom-Dictionary (pythainlp)	3	100
CC-Deepcut	2	10
LP-pythainlp	2	10
LP-Custom-Dictionary (pythainlp)	2	10
LP-Deepcut	2	10

4.2.3.1 การวัดประสิทธิภาพ ตัวชี้วัดประสิทธิภาพการเลือกตอบตามตัวอย่าง

ผลการทดลองสำหรับการวัดประสิทธิภาพสำหรับ ตัวชี้วัดประสิทธิภาพการเลือกตอบตามตัวอย่าง พบว่า วิธีการจำแนกประเภทหลายฉลากร่วมกับซัพพอร์ตเวกเตอร์แมชชีนด้วย polynomial kernel มีผลการทดลองที่แย่ที่สุดเมื่อเปรียบเทียบกับ linear kernel และ RBF kernel สำหรับทุกวิธีการตัดคำตามตารางที่ 17 สรุปผลการทดลองการจำแนกประเภทสำหรับการวัดประสิทธิภาพแบบ ตัวชี้วัดประสิทธิภาพการเลือกตอบตามตัวอย่าง

4.2.3.2 การวัดประสิทธิภาพ ตัวชี้วัดประสิทธิภาพการจำแนกประเภทหลายฉลาก

ผลการทดลองสำหรับการวัดประสิทธิภาพสำหรับ ตัวชี้วัดประสิทธิภาพการจำแนกประเภทหลายฉลาก ด้วยวิธีการจำแนกประเภทหลายฉลากร่วมกับซัพพอร์ตเวกเตอร์แมชชีนด้วย polynomial kernel มีผลการทดลองที่แย่ที่สุดเมื่อเปรียบเทียบกับ linear kernel และ RBF kernel สำหรับทุกวิธีการตัดคำ เช่นเดียวกับการวัดประสิทธิภาพสำหรับตัวชี้วัดประสิทธิภาพการเลือกตอบตามตัวอย่าง ตามตารางที่ 18 สรุปผลการทดลองการจำแนกประเภทสำหรับการวัดประสิทธิภาพแบบ ตัวชี้วัดประสิทธิภาพการจำแนกประเภทหลายฉลาก

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

ตารางที่ 17 ตารางผลการทดลองการจำแนกประเภทหลายฉลากสำหรับ ตัวชี้วัดประสิทธิภาพการเลือกตอบตามตัวอย่าง สำหรับแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน ด้วย polynomial kernels

PT and word segmentation method	Hamming loss	ML-accuracy	Subset accuracy	Precision	Recall	F-Measure
BR-SVM-pythainlp	0.158	0.343	0.094	**0.683**	0.354	0.441
BR-SVM-Custom-Dictionary (pythainlp)	0.162	0.319	0.072	0.674	0.326	0.418
BR-SVM-Deepcut	**0.152**	0.362	0.106	0.672	0.381	**0.647**
CC-SVM-pythainlp	0.168	0.437	0.171	0.582	0.521	0.530
CC-SVM-Custom-Dictionary (pythainlp)	0.171	0.429	0.164	0.569	0.516	0.522
CC-SVM-Deepcut	0.167	**0.440**	**0.174**	0.578	**0.530**	0.532
LP-SVM-pythainlp	0.226	0.378	0.132	0.460	0.520	0.473
LP-SVM-Custom-Dictionary (pythainlp)	0.228	0.373	0.129	0.454	0.515	0.467
LP-SVM-Deepcut	0.219	0.391	0.146	0.475	**0.530**	0.485

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ตารางที่ 18 ตารางผลการทดลองการจำแนกประเภทหลายฉลากสำหรับ ตัวชี้วัดประสิทธิภาพการจำแนกประเภทหลายฉลาก สำหรับแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน ด้วย polynomial kernel

PT and word segmentation method	Micro-average precision	Micro-average recall	Micro-average F-Measure
BR-SVM-pythainlp	0.391	0.358	0.479
BR-SVM-Custom-Dictionary (pythainlp)	0.375	0.336	0.455
BR-SVM-Deepcut	0.412	0.394	0.511
CC-SVM-pythainlp	0.418	0.546	0.568
CC-SVM-Custom-Dictionary (pythainlp)	0.412	0.543	0.563
CC-SVM-Deepcut	**0.421**	**0.558**	**0.574**
LP-SVM-pythainlp	0.343	0.551	0.498
LP-SVM-Custom-Dictionary (pythainlp)	0.340	0.549	0.494
LP-SVM-Deepcut	0.352	0.561	0.509

สำหรับการเปรียบเทียบผลการทดลองวิธีการจำแนกประเภทหลายฉลากร่วมกับแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน ด้วย linear kernel, RBF kernel และ polynomial kernel ในแต่ละวิธีตัดคำ จากการวัดประสิทธิภาพแบบ ตัวชี้วัดประสิทธิภาพการเลือกตอบตามตัวอย่าง และ ตัวชี้วัดประสิทธิภาพการจำแนกประเภทหลายฉลาก พบว่า วิธีการจำแนกประเภทหลายฉลากร่วมกับแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน ด้วย RBF kernel โดยรวมมีประสิทธิภาพดีที่สุด ซึ่งวิธีการจำแนกประเภทหลายฉลาก Classifier Chain ร่วมกับแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน ด้วย RBF kernel และวิธีการตัดคำ pythainlp ให้ผลที่ดีที่สุดในการวัดด้วย ML-accuracy เท่ากับ 0.578, Subset accuracy เท่ากับ 0.300, ค่าเรียกคืน (Recall) เท่ากับ 0.670 และ ค่าเฉลี่ยไมโครสำหรับค่าเรียกคืน เท่ากับ 0.670 อย่างไรก็ตาม วิธีการจำแนกประเภทหลายฉลาก Binary Relevance ร่วมกับแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน ด้วย RBF kernel และวิธีการตัดคำ pythainlp ให้ผลที่ดีที่สุดในการวัดด้วย Hamming loss เท่ากับ 0.106, ค่าแม่นยำ (Precision) เท่ากับ 0.735, ตัววัด F1

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

(F-measure) เท่ากับ 0.665, ค่าเฉลี่ยไมโครสำหรับค่าแม่นยำ เท่ากับ 0.586 และ ค่าเฉลี่ยไมโครสำหรับตัววัด F1 เท่ากับ 0.715

การวัดประสิทธิภาพของตัวชี้วัดประสิทธิภาพการเลือกตอบตามตัวอย่าง ด้วย Subset accuracy ในงานวิจัยนี้จะให้ค่าค่อนข้างน้อยเมื่อเทียบกับงานวิจัยการจำแนกประเภทหลายฉลากในภาษาอื่น ๆ [10] ซึ่ง การวัดด้วย Subset accuracy จะเข้มงวดมากเกี่ยวกับการทำนายฉลากที่ไม่ถูกต้องโดยไม่อนุญาตให้มีการทำนายที่ผิดพลาดในชุดฉลากที่คาดการณ์เลย ดังนั้น เมื่อขนาดของ Label Cardinality และ ขนาดของฉลากมีค่าที่สูงขึ้น การทำนายฉลากให้ถูกต้องทั้งหมดจะยิ่งเป็นไปได้ยากขึ้น ทำให้ Subset accuracy มักเข้าใกล้ 0 ดังนั้นการวัดประสิทธิภาพ Example-based Metrics ดังนั้น งานวิจัย [14][15] จึงแนะนำให้ใช้การวัด ตัววัด F1 (F-measure) มาพิจารณาแทน Subset accuracy ได้

ตารางที่ 19 MLCM ของ Classifier Chain ร่วมกับ ซัพพอร์ตเวกเตอร์แมชชีน *RBF kernel*

		Predicted Labels												
		AI	CTM	CG	CNC	CSA	CV	GCS	HA	HCI	IS	SP	SW	NPL
True Labels	AI	**5**	2	0	0	2	0	0	0	0	3	0	3	2
	CTM	3	**6**	0	0	4	0	1	0	0	2	0	3	1
	CG	0	0	**4**	0	2	0	0	0	0	0	0	0	2
	CNC	0	0	0	**4**	3	0	0	1	0	1	0	1	1
	CSA	1	0	0	0	**29**	0	0	0	0	2	0	3	3
	CV	0	0	0	0	1	**4**	0	0	0	0	0	1	0
	GCS	0	0	0	0	5	0	**3**	0	0	1	0	2	3
	HA	0	0	0	0	1	0	0	**3**	0	0	0	0	2
	HCI	0	0	0	0	2	0	0	0	**3**	1	0	0	1
	IS	0	0	0	0	1	0	0	0	0	**7**	0	1	2
	SP	0	0	0	0	0	0	0	1	0	0	**3**	0	2
	SW	0	0	0	0	2	0	1	1	0	2	0	**20**	2
	NLT	0	0	0	0	0	0	0	0	0	0	0	0	0

ผลลัพธ์ของการจำแนกประเภทหลายฉลากด้วยวิธี Classifier Chain ร่วมกับแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน ด้วย RBF kernel เมื่อวิเคราะห์เพิ่มเติมใน MLCM ตามตารางที่ 19 ฉลากที่ทำนายได้ถูกเยอะที่สุด 3 อันดับแรกคือ 1. Computer Science Applications ทำนายถูก 29 ครั้ง 2. Software ทำนายถูก 20 ครั้ง และ 3. Information Systems ทำนายถูก 7 ครั้ง แต่ในทางกลับกันฉลาก Computer Science Applications, Information Systems และ Software พบว่าถูกจำแนก

ประเภทอย่างไม่ถูกต้อง เนื่องจากจำนวนสาขาย่อยจากรูปที่ 12 คือ Computer Science Computer Science Applications, Information Systems และ Software มีจำนวนมากเป็นอันดับต้น ๆ ทำให้แบบจำลองจำแนกหลายฉลากทำนายถูกต้องส่วนใหญ่เป็น Computer Science Applications, Information Systems และ Software

ตารางที่ 20 MLCM ของ Binary Relevance ร่วมกับ ซัพพอร์ตเวกเตอร์แมชชีน RBF kernel

		Predicted Labels												
		AI	CTM	CG	CNC	CSA	CV	GCS	HA	HCI	IS	SP	SW	NPL
True Labels	AI	**9**	1	0	0	2	1	1	0	0	1	0	1	2
	CTM	1	**7**	0	0	3	0	0	0	0	0	0	0	4
	CG	0	0	**1**	0	0	0	0	0	0	0	0	0	1
	CNC	0	0	0	**4**	0	0	1	0	0	0	0	0	1
	CSA	1	0	0	0	**34**	0	2	0	0	2	0	1	3
	CV	1	0	0	0	1	**2**	0	0	0	0	0	0	0
	GCS	0	0	0	0	3	0	**4**	0	0	0	0	0	2
	HA	0	0	0	0	1	0	0	**6**	0	0	1	0	3
	HCI	0	0	0	0	0	0	1	0	**2**	0	0	0	2
	IS	1	0	0	0	3	0	0	0	0	**9**	0	0	6
	SP	1	0	0	0	0	0	0	0	0	0	**1**	0	1
	SW	1	0	0	0	4	0	0	0	0	3	0	**23**	6
	NLT	0	0	0	0	0	0	0	0	0	0	0	0	0

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

เช่นเดียวกับผลลัพธ์ของการจำแนกประเภทหลายฉลากด้วยวิธี Binary Relevance ร่วมกับแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน ด้วย RBF kernel เมื่อวิเคราะห์เพิ่มเติมใน MLCM ตามตารางที่ 20 ฉลากที่ทำนายได้ถูกเยอะที่สุดเหมือนกับวิธี Classifier Chain ร่วมกับแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน ด้วย RBF kernel แต่พบว่าถูกจำแนกอย่างไม่ถูกต้องในฉลาก Computer Science Applications ซึ่งมีจำนวนสาขาเยอะที่สุดในรูปที่ 12 และวิธี Binary Relevance ยังมีค่า NPL สูงในทุกฉลากเกิดจากการที่ฉลากนั้น ๆ ไม่ได้ถูกทำนาย

บทที่ 5
สรุปผลและอภิปรายผลการทดลอง

5.1 สรุปผลการวิจัย

บทความทางวิทยาการคอมพิวเตอร์ สามารถมีสาขาย่อยในบทความนั้น ๆ ได้ แต่การตีพิมพ์บทความทางวิทยาการคอมพิวเตอร์ของไทยเพิ่มมากขึ้นเรื่อย ๆ ส่งผลให้การจำแนกสาขาย่อยของบทความทำให้มีความยากขึ้น โดยงานวิจัยนี้มีวัตถุประสงค์เพื่อจำแนกประเภทสาขาย่อยของบทความทางวิทยาการคอมพิวเตอร์ ช่วยให้การจัดประเภทสาขาย่อยของบทความวิจัยง่ายขึ้น สามารถลดปริมาณงานของบรรณาธิการและนักวิจัย และสามารถนำไปใช้ในแอปพลิเคชันต่างๆ ในการจัดประเภทข้อความได้

ในงานวิจัยนี้ ได้ทดลองวิธีการจำแนกประเภทหลายหลายฉลาก คือ (1) Binary Relevance, (2) Classifier Chain และ Label Power-set สำหรับบทความวิทยาการคอมพิวเตอร์แบบภาษาไทย ร่วมกับ วิธีการตัดคำภาษาไทย การสกัดคำสำคัญ (TF-IDF) และแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน ด้วย linear kernel, RBF kernel และ polynomial kernel เพื่อเปรียบเทียบประสิทธิภาพของการจำแนกประเภทหลายฉลากในบทความภาษาไทย

ผลการทดลองของการจำแนกประเภทหลายฉลาก แสดงให้เห็นว่า วิธีการ Classifier Chain ร่วมกับ วิธีการตัดคำภาษาไทย pythainlp และแบบจำลองซัพพอร์ตเวกเตอร์แมชชีน ด้วย RBF kernel มีประสิทธิภาพดีสำหรับการวัดประสิทธิภาพ ตัวชี้วัดประสิทธิภาพการเลือกตอบตามตัวอย่าง และ ตัวชี้วัดประสิทธิภาพการจำแนกประเภทหลายฉลาก คือ ML-accuracy (0.578), Subset accuracy (0.300), Recall (0.670) และ ค่าเฉลี่ยไมโครสำหรับค่าเรียกคืน (0.670) โดยยังมีวิธี Binary Relevance ร่วมกับวิธีการตัดคำภาษาไทย pythainlp และ แบบจำลองซัพพอร์ตเวกเตอร์แมชชีน ด้วย RBF kernel มีประสิทธิภาพดีสำหรับการวัดประสิทธิภาพ ตัวชี้วัดประสิทธิภาพการเลือกตอบตามตัวอย่าง และ ตัวชี้วัดประสิทธิภาพการจำแนกประเภทหลายฉลาก คือ Hamming loss (0.106), ค่าแม่นยำ (Precision) (0.735), ตัววัด F1 (F-measure) (0.665), ค่าเฉลี่ยไมโครสำหรับค่าเรียกคืน (0.586) และ ค่าเฉลี่ยไมโครสำหรับตัววัด F1 (0.715) แต่ผลลัพธ์ยังทำนายได้อย่างถูกต้องเยอะที่สุดเฉพาะ 4 ฉลากเท่านั้นคือ (1) Artificial Intelligence (2) Computer Science Applications (3) Information Systems และ (4) Software เนื่องจากเป็นฉลากที่มีจำนวนมากในชุดข้อมูล

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

5.2 ข้อเสนอแนะ

งานวิจัยนี้การจำแนกประเภทหลายฉลากสำหรับบทความวิทยาการคอมพิวเตอร์ยังจำเป็นต้องมีการวิจัยเพิ่มเติมเพื่อปรับปรุงประสิทธิภาพการตัดคำภาษาไทย และ การสกัดคำสำคัญ ผลลัพธ์อาจจะยังไม่ได้เป็นคำที่สมบูรณ์ เช่น คำศัพท์ภาษาไทยที่ทับศัพท์คำศัพท์ภาษาอังกฤษ มายเอสคิวแอล นาอีฟเบย์ ถูกตัดคำออกมาได้เป็น |มาย|เอส|คิว|แอล|, |นา|อีฟ|เบย์| ซึ่งจำเป็นต้องปรับปรุงแก้ไขต่อไปและรวบรวมค้นคว้าบทความวิทยาการคอมพิวเตอร์เพิ่มเติมในฉลากที่ยังมีจำนวนน้อยเพื่อปรับปรุงประสิทธิภาพให้ทำนายในฉลากอื่น ๆ ได้ถูกต้องมากขึ้น ซึ่งจากการทบทวนงานวิจัยอื่น ๆ ของการจำแนกประเภทหลายฉลากต้องคำนึงถึงค่า Label Cardinality และ Label Density ซึ่งส่งผลต่อประสิทธิภาพของการจำแนกประเภทหลายฉลาก

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

บรรณานุกรม

1. Thai Journals Online (ThaiJO). ประวัติความเป็นมา *Thai Journals Online*. 2010 October 10, 2022]; Available from: https://www.tci-thaijo.org.
2. Haruechaiyasak, C., S. Kongyoung, and M. Dailey. *A comparative study on Thai word segmentation approaches*. in *2008 5th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology*. 2008. IEEE.
3. Dien, T.T., B.H. Loc, and N. Thai-Nghe. *Article classification using natural language processing and machine learning*. in *2019 International Conference on Advanced Computing and Applications (ACOMP)*. 2019. IEEE.
4. Nareshpalsingh, J.M. and H.N. Modi, *Multi-label classification methods: a comparative study.* International Research Journal of Engineering and Technology (IRJET), 2017. **4**(12): p. 263-270.
5. Berrar, D., *Cross-Validation.* Encyclopedia of Bioinformatics and Computational Biology, 2019. **1**: p. 542–545.
6. Chormai, P., P. Prasertsom, and A. Rutherford, *AttaCut: A Fast and Accurate Neural Thai Word Segmenter.* arXiv preprint arXiv:1911.07056, 2019.
7. Qaiser, S. and R. Ali, *Text mining: use of TF-IDF to examine the relevance of words to documents.* International Journal of Computer Applications, 2018. **181**(1): p. 25-29.
8. Tanantong, T., S. Kreangkriwanich, and N. Laosen. *Extraction of Trend Keywords from Thai Twitters using N-Gram Word Combination*. in *2020 17th International Conference on Electrical Engineering/Electronics, Computer, Telecommunications and Information Technology (ECTI-CON)*. 2020. IEEE.
9. Azam, M., et al., *Feature extraction based text classification using k-nearest neighbor algorithm.* IJCSNS Int. J. Comput. Sci. Netw. Secur, 2018. **18**(12): p. 95-101.

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

10. Aljedani, N., R. Alotaibi, and M. Taileb, *Multi-Label Arabic Text Classification: An Overview.* (IJACSA) International Journal of Advanced Computer Science and Applications, 2020. **11**(10): p. 694-706.

11. Yapp, E.K., et al., *Comparison of base classifiers for multi-label learning.* Neurocomputing, 2020. **394**: p. 51-60.

12. สัมภัตตะกุล, จ., การพัฒนาระบบเฝ้าระวังและแจ้งเตือนอัคคีภัยกองทัพเรือ. วารสารวิชาการ "การประยุกต์ใช้เทคโนโลยีสารสนเทศ", 2021. 7(1): p. 96-108.

13. Kariuki, C. Multi-Label Classification with Scikit-MultiLearn. 2021 October 20, 2021]; Available from: https://www.section.io/engineering-education/multi-label-classification-with-scikit-multilearn/.

14. Bernardini, F.C., et al., Cardinality and density measures and their influence to multi-label learning methods. Learning and Nonlinear Models - Learning and Nonlinear Models - Journal of the Brazilian Society on Computational Intelligence (SBIC), 2014. 12: p. 53-71.

15. Nam, J., et al., Maximizing subset accuracy with recurrent neural networks in multi-label classification, in 31st Conference on neural information processing systems (NIPS 2017). 2017.

16. Heydarian, M., T.E. Doyle, and R. Samavi, *MLCM: multi-label confusion matrix.* IEEE Access, 2022. **10**: p. 19083-19095.

17. Patle, A. and D.S. Chouhan. *SVM kernel functions for classification*. in *2013 International Conference on Advances in Technology and Engineering (ICATE)*. 2013. IEEE.

18. Thaipisutikul, T., et al. Automated classification of criminal and violent activities in Thailand from online news articles. in 2021 13th International Conference on Knowledge and Smart Technology (KST). 2021. IEEE.

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY

จุฬาลงกรณ์มหาวิทยาลัย
Chulalongkorn University

ประวัติผู้เขียน

ชื่อ-สกุล	จินตรัย พุทธิพรชัย
วัน เดือน ปี เกิด	21 มีนาคม 2539
สถานที่เกิด	กรุงเทพมหานคร
วุฒิการศึกษา	ปริญญาตรี มหาวิทยาลัยศิลปากร 2557-2561
ที่อยู่ปัจจุบัน	21 ซอยบางพรม50 ถนนบางพรม แขวงบางพรม เขตตลิ่งชัน กรุงเทพมหานคร

จุฬาลงกรณ์มหาวิทยาลัย
CHULALONGKORN UNIVERSITY